# STATA BASE REFERENCE MANUAL RELEASE 18 



A Stata Press Publication
StataCorp LLC
College Station, Texas

Copyright (c) 1985-2023 StataCorp LLC
All rights reserved
Version 18

Published by Stata Press, 4905 Lakeway Drive, College Station, Texas 77845
ISBN-10: 1-59718-395-4
ISBN-13: 978-1-59718-395-6

This manual is protected by copyright. All rights are reserved. No part of this manual may be reproduced, stored in a retrieval system, or transcribed, in any form or by any means-electronic, mechanical, photocopy, recording, or otherwise-without the prior written permission of StataCorp LLC unless permitted subject to the terms and conditions of a license granted to you by StataCorp LLC to use the software and documentation. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted by this document.

StataCorp provides this manual "as is" without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. StataCorp may make improvements and/or changes in the product(s) and the program(s) described in this manual at any time and without notice.

The software described in this manual is furnished under a license agreement or nondisclosure agreement. The software may be copied only in accordance with the terms of the agreement. It is against the law to copy the software onto DVD, CD, disk, diskette, tape, or any other medium for any purpose other than backup or archival purposes.

The automobile dataset appearing on the accompanying media is Copyright (c) 1979 by Consumers Union of U.S., Inc., Yonkers, NY 10703-1057 and is reproduced by permission from CONSUMER REPORTS, April 1979.

Stata, STPTE Stata Press, Mata, məTE and NetCourse are registered trademarks of StataCorp LLC.
Stata and Stata Press are registered trademarks with the World Intellectual Property Organization of the United Nations. StataNow and NetCourseNow are trademarks of StataCorp LLC.

Other brand and product names are registered trademarks or trademarks of their respective companies.
For copyright information about the software, type help copyright within Stata.

The suggested citation for this software is
StataCorp. 2023. Stata 18. Statistical software. StataCorp LLC.
The suggested citation for this manual is
StataCorp. 2023. Stata 18 Base Reference Manual. College Station, TX: Stata Press.

## Contents

${ }^{+}$This manual includes features that are part of StataNow.
Intro Introduction to base reference manual ..... 1
about Display information about your Stata ..... 8
ado update Update community-contributed packages ..... 9
ameans Arithmetic, geometric, and harmonic means ..... 13
anova Analysis of variance and covariance ..... 17
anova postestimation Postestimation tools for anova ..... 59
areg Linear regression with many indicator variables ${ }^{+}$ ..... 76
areg postestimation Postestimation tools for areg ..... 86
betareg Beta regression ..... 91
betareg postestimation Postestimation tools for betareg ..... 101
binreg Generalized linear models: Extensions to the binomial family ..... 105
binreg postestimation Postestimation tools for binreg ..... 118
biprobit Bivariate probit regression ..... 122
biprobit postestimation Postestimation tools for biprobit ..... 130
bitest Binomial probability test ..... 134
bootstrap Bootstrap sampling and estimation ..... 139
bootstrap postestimation Postestimation tools for bootstrap ..... 162
boxcox Box-Cox regression models ..... 166
boxcox postestimation Postestimation tools for boxcox ..... 177
brier Brier score decomposition ..... 182
bsample Sampling with replacement ..... 189
bstat Report bootstrap results ..... 198
centile Report centile and confidence interval ..... 206
churdle Cragg hurdle regression ..... 213
churdle postestimation Postestimation tools for churdle ..... 224
ci Confidence intervals for means, proportions, and variances ..... 230
clogit Conditional (fixed-effects) logistic regression ..... 249
clogit postestimation Postestimation tools for clogit ..... 266
cloglog Complementary log-log regression ..... 272
cloglog postestimation Postestimation tools for cloglog ..... 282
cls Clear Results window ..... 286
cnsreg Constrained linear regression ..... 287
cnsreg postestimation Postestimation tools for cnsreg ..... 293
constraint Define and list constraints ..... 297
contrast Contrasts and linear hypothesis tests after estimation ..... 301
contrast postestimation Postestimation tools for contrast ..... 368
copyright Display copyright information ..... 370
Copyright Apache Apache copyright notification ..... 371
Copyright autolink autolink copyright notification ..... 375
Copyright Boost Boost copyright notification ..... 376
Copyright flexmark flexmark copyright notification ..... 377
Copyright Hamcrest Hamcrest copyright notification ..... 378
Copyright H2O H2O copyright notification ..... 379
Copyright ICD-10 ICD-10 copyright notification ..... 383
Copyright ICU ICU copyright notification ..... 384
Copyright JAXB JAXB copyright notification ..... 385
Copyright JGoodies Common JGoodies Common copyright notification ..... 391
Copyright JGoodies Forms JGoodies Forms copyright notification ..... 392
Copyright JSON JSON for Modern C++ copyright notification ..... 393
Copyright jsoup jsoup copyright notification ..... 394
Copyright LAPACK LAPACK copyright notification ..... 395
Copyright libHaru HARU copyright notification ..... 396
Copyright libpng libpng copyright notification ..... 397
Copyright Mersenne Twister Mersenne Twister copyright notification ..... 399
Copyright MiG Layout MiG Layout copyright notification ..... 400
Copyright Parsington Parsington copyright notification ..... 401
Copyright PolyHook PolyHook copyright notification ..... 402
Copyright ReadStat ReadStat copyright notification ..... 403
Copyright Scintilla Scintilla copyright notification ..... 404
Copyright slf4j slf4j copyright notification ..... 405
Copyright ttf2pt1 ttf 2 pt 1 copyright notification ..... 406
Copyright Win32 Dark Mode Win32 Dark Mode copyright notification ..... 408
Copyright zlib zlib copyright notification ..... 409
correlate Correlations of variables ..... 410
cpoisson Censored Poisson regression ..... 420
cpoisson postestimation Postestimation tools for cpoisson ..... 427
cumul Cumulative distribution ..... 433
cusum Cusum plots and tests for binary variables ..... 438
dbLaunch dialog443
demandsys Estimation of flexible demand systems ..... 445
demandsys postestimation Postestimation tools for demandsys ..... 489
Diagnostic plots Distributional diagnostic plots ..... 505
display Substitute for a hand calculator ..... 520
do Execute commands from a file ..... 522
doedit Edit do-files and other text files ..... 524
dotplot Comparative distribution dotplots ..... 526
dstdize Direct and indirect standardization ..... 534
dtable Create a table of descriptive statistics ..... 554
dydx Calculate numeric derivatives and integrals ..... 585
eform_option Displaying exponentiated coefficients ..... 592
eivreg Errors-in-variables regression ..... 594
eivreg postestimation Postestimation tools for eivreg ..... 601
Epitab Tables for epidemiologists ..... 604
Error messages Error messages and return codes ..... 669
esize Effect size based on mean comparison ..... 670
estat Postestimation statistics ..... 683
estat classification Classification statistics and table ..... 684
estat gof Pearson or Hosmer-Lemeshow goodness-of-fit test ..... 688
estat ic Display information criteria ..... 698
estat summarize Summarize estimation sample ..... 706
estat vce Display covariance matrix estimates ..... 710
estimates Save and manipulate estimation results ..... 714
estimates describe Describe estimation results ..... 718
estimates for Repeat postestimation command across models ..... 721
estimates notes Add notes to estimation results ..... 723
estimates replay Redisplay estimation results ..... 726
estimates save Save and use estimation results ..... 729
estimates selected Show selected coefficients ..... 733
estimates stats Model-selection statistics ..... 737
estimates store Store and restore estimation results ..... 741
estimates table Compare estimation results ..... 744
estimates title Set title for estimation results ..... 752
Estimation options Estimation options ..... 753
etable Create a table of estimation results ..... 757
exit Exit Stata ..... 780
exlogistic Exact logistic regression ..... 781
exlogistic postestimation Postestimation tools for exlogistic ..... 799
expoisson Exact Poisson regression ..... 804
expoisson postestimation Postestimation tools for expoisson ..... 818
fp Fractional polynomial regression ..... 820
fp postestimation Postestimation tools for fp ..... 844
fracreg Fractional response regression ..... 853
fracreg postestimation Postestimation tools for fracreg ..... 863
frontier Stochastic frontier models ..... 871
frontier postestimation Postestimation tools for frontier ..... 887
fvrevar Factor-variables operator programming command ..... 892
fvset Declare factor-variable settings ..... 896
gllamm Generalized linear and latent mixed models ..... 902
glm Generalized linear models ..... 904
glm postestimation Postestimation tools for glm ..... 939
gmm Generalized method of moments estimation ..... 946
gmm postestimation Postestimation tools for gmm 1017
grmeanbyGraph means and medians by categorical variables 1023
hausman Hausman specification test 1026
heckman Heckman selection model 1036
heckman postestimation Postestimation tools for heckman 1053
heckoprobit Ordered probit model with sample selection 1060
heckoprobit postestimation Postestimation tools for heckoprobit 1069
heckpoisson Poisson regression with sample selection 1075
heckpoisson postestimation Postestimation tools for heckpoisson 1085
heckprobit Probit model with sample selection 1091
heckprobit postestimation Postestimation tools for heckprobit 1100
help
Display help in Stata 1107
hetoprobit Heteroskedastic ordered probit regression 1109
hetoprobit postestimation Postestimation tools for hetoprobit 1121
hetprobit Heteroskedastic probit model 1126
hetprobit postestimation Postestimation tools for hetprobit 1134
hetregress Heteroskedastic linear regression 1138
hetregress postestimation Postestimation tools for hetregress 1152

IC note . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . Calculating and interpreting information criteria 1168

Inequality . . . . . . . . . . . . . . . . . . . . . Inequality, poverty, and other distributional summaries 1197
intreg ................................. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . Interval regression 1201
intreg postestimation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . Postestimation tools for intreg 1211
ivfprobit ................. . Fractional probit model with continuous endogenous covariates 1217
ivfprobit postestimation . . . . . . . . . . . . . . . . . . . . . . . . . . . . Postestimation tools for ivfprobit 1224
ivpoisson ......................... . Poisson model with continuous endogenous covariates 1231
ivpoisson postestimation . . . . . . . . . . . . . . . . . . . . . . . . . . Postestimation tools for ivpoisson 1247
ivprobit . . . . . . . . . . . . . . . . . . . . . . . . . Probit model with continuous endogenous covariates 1253
ivprobit postestimation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . Postestimation tools for ivprobit 1267
ivqregress ........................................ . . . Instrumental-variables quantile regression 1275
ivqregress postestimation . . . . . . . . . . . . . . . . . . . . . . . . . Postestimation tools for ivqregress 1304
ivregress . . . . . . . . . . . . . . . . . . . . . . . . . . . Single-equation instrumental-variables regression 1315
ivregress postestimation . . . . . . . . . . . . . . . . . . . . . . . . . . Postestimation tools for ivregress ${ }^{+} 1333$
ivtobit ................................ . . Tobit model with continuous endogenous covariates 1362
ivtobit postestimation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . Postestimation tools for ivtobit 1373
jackknife ................................................................ . Jackknife estimation 1382
jackknife postestimation . . . . . . . . . . . . . . . . . . . . . . . . . . Postestimation tools for jackknife 1396
kappa . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . Interrater agreement 1398
kdensity . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . Univariate kernel density estimation 1414
ksmirnov ............................. . . . . Kolmogorov-Smirnov equality-of-distributions test 1424

ladder . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . Ladder of powers 1432
level ......................... . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . Set default confidence level 1440

lincom .................................................. . . Linear combinations of parameters 1447
linktest . . . . . . . . . . . . . . . . . . . . . . . . . . . . . Specification link test for single-equation models 1456
lnskew0 ...................................... . . Find zero-skewness log or Box-Cox transform 1463

logistic . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . Logistic regression, reporting odds ratios 1473
logistic postestimation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . Postestimation tools for logistic 1484
logit . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . Logistic regression, reporting coefficients 1496
logit postestimation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . Postestimation tools for logit 1510
loneway ......................... Large one-way ANOVA, random effects, and reliability 1518
lowess ............................. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . Lowess smoothing 1525
lpoly . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . Kernel-weighted local polynomial smoothing 1532
lroc ................................. . . . Compute area under ROC curve and graph the curve 1543
lrtest ............................ . . . . . . . . . . . . . . . . . . . Likelihood-ratio test after estimation 1550
1sens ............................. . Graph sensitivity and specificity versus probability cutoff 1561
lv .................................. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . Letter-value displays 1567
makespline
Spline generation 1574
margins ......................... . . Marginal means, predictive margins, and marginal effects 1586
margins postestimation .................................... . . Postestimation tools for margins 1644
margins, contrast Contrasts of margins 1646
margins, pwcompare Pairwise comparisons of margins 1664
marginsplot Graph results from margins (profile plots, etc.) 1669
Maximize Details of iterative maximization 1704
mean Estimate means 1711
mean postestimation Postestimation tools for mean 1726
mfp Multivariable fractional polynomial models 1728
mfp postestimation Postestimation tools for mfp 1741
misstable Tabulate missing values 1747
ml Maximum likelihood estimation 1757
mlexp Maximum likelihood estimation of user-specified expressions ..... 1785
mlexp postestimation Postestimation tools for mlexp 1800
mlogit Multinomial (polytomous) logistic regression 1803
mlogit postestimation Postestimation tools for mlogit 1818
more The -more- message 1829
mprobit Multinomial probit regression 1831
mprobit postestimationPostestimation tools for mprobit 1839
nbregNegative binomial regression 1843
nbreg postestimation Postestimation tools for nbreg and gnbreg 1857Nested model statistics1863
net Install and manage community-contributed additions from the Internet 1870
net search Search the Internet for installable packages ..... 1888
netio Control Internet connections 1892
nl Nonlinear least-squares estimation 1894
nl postestimation Postestimation tools for nl 1915
nlcom Nonlinear combinations of parameters ..... 1920
nlsur Estimation of nonlinear systems of equations ..... 1933
nlsur postestimation Postestimation tools for nlsur 1955
npregress intro Introduction to nonparametric regression 1959
npregress kernel Nonparametric kernel regression 1974
npregress kernel postestimation Postestimation tools for npregress kernel ..... 1998
npregress series Nonparametric series regression 2003
npregress series postestimation Postestimation tools for npregress series 2020
nptrend
Tests for trend across ordered groups 2025ologitOrdered logistic regression 2047
ologit postestimation Postestimation tools for ologit 2057
oneway ..... One-way analysis of variance 2062
oprobit Ordered probit regression 2074
oprobit postestimation Postestimation tools for oprobit 2080
orthogpcorrPartial and semipartial correlation coefficients 2092
permute Permutation tests ..... 2096
pk Pharmacokinetic (biopharmaceutical) data 2112
pkcollapse Generate pharmacokinetic measurement dataset 2120
pkcross Analyze crossover experiments ..... 2124
pkequiv Perform bioequivalence tests 2133
pkexamine Calculate pharmacokinetic measures 2140
pkshape Reshape (pharmacokinetic) Latin-square data ..... 2147
pksumm Summarize pharmacokinetic data 2155
poisson Poisson regression 2161
poisson postestimation Postestimation tools for poisson 2172
postest Postestimation Selector 2180
predict Obtain predictions, residuals, etc., after estimation 2183
predictnl Obtain nonlinear predictions, standard errors, etc., after estimation 2195
probit Probit regression 2208
probit postestimation Postestimation tools for probit 2221
proportion Estimate proportions 2226
proportion postestimation Postestimation tools for proportion 2235prtestTests of proportions 2237
pwcompare Pairwise comparisons 2247
pwcompare postestimation Postestimation tools for pwcompare 2281
pwmean Pairwise comparisons of means 2283
pwmean postestimation Postestimation tools for pwmean 2296
QCQuality control charts 2298
qregQuantile regression 2314
qreg postestimation Postestimation tools for qreg, iqreg, sqreg, and bsqreg 2347
queryDisplay system parameters 2355
ranksumEquality tests on unmatched data 2366
ratio Estimate ratios 2374
ratio postestimation Postestimation tools for ratio 2384
reg3 Three-stage estimation for systems of simultaneous equations 2385
reg3 postestimation Postestimation tools for reg3 2407
regress Linear regression 2413
regress postestimation Postestimation tools for regress 2443
regress postestimation diagnostic plots Postestimation plots for regress 2478
regress postestimation time series Postestimation tools for regress with time series 2496
reri Relative excess risk due to interaction 2508
\#review Review previous commands 2524
roc Receiver operating characteristic (ROC) analysis 2526
roccomp Tests of equality of ROC areas 2528
rocfitParametric ROC models 2540
rocfit postestimation Postestimation tools for rocfit 2547
rocreg Receiver operating characteristic (ROC) regression 2551rocreg postestimationPostestimation tools for rocreg 2607
rocregplot Plot marginal and covariate-specific ROC curves after rocreg 2623
roctab Nonparametric ROC analysis 2644
rreg Robust regression 2655
rreg postestimation Postestimation tools for rreg 2663
runtestTest for random order 2666
scobitSkewed logistic regression 2672
scobit postestimation Postestimation tools for scobit 2681
sdtest Variance-comparison tests 2685
search Search Stata documentation and other resources ..... 2692
serrbar Graph standard error bar chart 2698
set Overview of system parameters 2702set cformatFormat settings for coefficient tables 2720
set_defaults Reset system parameters to original Stata defaults 2723
set emptycells Set what to do with empty cells in interactions ..... 2724
set iter Control iteration settings 2725
set rng Set which random-number generator (RNG) to use 2729
set rngstream Specify the stream for the stream random-number generator 2731
set seed
Specify random-number seed and state ..... 2739
set showbaselevels Display settings for coefficient tables 2744
signrankEquality tests on matched data 2753
simulate Monte Carlo simulations 2760
sj Stata Journal installation instructions 2767
sktest Skewness and kurtosis tests for normality 2769
slogit Stereotype logistic regression 2775
slogit postestimation Postestimation tools for slogit 2789
smoothRobust nonlinear smoother 2794
spearman Spearman's and Kendall's correlations 2803spikeplotSpike plots and rootograms 2818
ssc Install and uninstall packages from SSC 2823
stem ..... Stem-and-leaf displays 2831
stepwise Stepwise estimation 2836
Stored results ..... Stored results 2847
suest Seemingly unrelated estimation ..... 2852
summarize Summary statistics 2871
sunflower Density-distribution sunflower plots ..... 2882
sureg Zellner's seemingly unrelated regression 2888
sureg postestimation Postestimation tools for sureg 2897
swilk Shapiro-Wilk and Shapiro-Francia tests for normality 2901
symmetry Symmetry and marginal homogeneity tests 2907
table intro Introduction to tables of frequencies, summaries, and command results 2916
table oneway One-way tabulation 2921
table twoway Two-way tabulation 2929
table multiway Multiway tables 2939
table summary Table of summary statistics 2953
table hypothesis tests Table of hypothesis tests ..... 2966
table regression Table of regression results ..... 2976
table Table of frequencies, summaries, and command results ..... 2988
tabstat Compact table of summary statistics 3005
tabulate oneway One-way table of frequencies 3011
tabulate twoway Two-way table of frequencies 3020
tabulate, summarize() One- and two-way tables of summary statistics 3038
test Test linear hypotheses after estimation 3043
testnl Test nonlinear hypotheses after estimation 3065
tetrachoric ..... Tetrachoric correlations for binary variables 3075
tnbreg Truncated negative binomial regression 3086
tnbreg postestimation Postestimation tools for tnbreg 3095
tobitTobit regression 3100
tobit postestimation Postestimation tools for tobit 3109total
total postestimationEstimate totals3114
tpoisson Truncated Poisson regression ..... 3122
tpoisson postestimation Postestimation tools for tpoisson 3130
translate Print and translate logs 3136
truncreg Truncated regression 3148
truncreg postestimation Postestimation tools for truncreg 3155
ttest $t$ tests (mean-comparison tests) 3159
update Check for official updates 3171
vce_option Variance estimators 3174
view View files and logs 3179
vwls Variance-weighted least squares 3182
vwls postestimation Postestimation tools for vwls 3189
which Display location of an ado-file 3192
wildbootstrap Wild cluster bootstrap inference 3194
xi Interaction expansion 3208
zinb Zero-inflated negative binomial regression 3219
zinb postestimation Postestimation tools for zinb 3228
ziologit Zero-inflated ordered logit regression 3233
ziologit postestimation Postestimation tools for ziologit 3242
zioprobitZero-inflated ordered probit regression 3248
zioprobit postestimation Postestimation tools for zioprobit 3257
zip Zero-inflated Poisson regression 3264
zip postestimation Postestimation tools for zip 3272
ztest $z$ tests (mean-comparison tests, known variance) ..... 3278
Subject and author index ..... 3293

## Cross-referencing the documentation

When reading this manual, you will find references to other Stata manuals, for example, [U] 27 Overview of Stata estimation commands; [R] regress; and [D] reshape. The first example is a reference to chapter 27, Overview of Stata estimation commands, in the User's Guide; the second is a reference to the regress entry in the Base Reference Manual; and the third is a reference to the reshape entry in the Data Management Reference Manual.

All the manuals in the Stata Documentation have a shorthand notation:

| [GSM] | Getting Started with Stata for Mac |
| :--- | :--- |
| [GSU] | Getting Started with Stata for Unix |
| [GSW] | Getting Started with Stata for Windows |
| [U] | Stata User's Guide |
| [R] | Stata Base Reference Manual |
| [ADAPT] | Stata Adaptive Designs: Group Sequential Trials Reference Manual |
| [BAYES] | Stata Bayesian Analysis Reference Manual |
| [BMA] | Stata Bayesian Model Averaging Reference Manual |
| [CAUSAL] | Stata Causal Inference and Treatment-Effects Estimation Reference Manual |
| [CM] | Stata Choice Models Reference Manual |
| [D] | Stata Data Management Reference Manual |
| [DSGE] | Stata Dynamic Stochastic General Equilibrium Models Reference Manual |
| [ERM] | Stata Extended Regression Models Reference Manual |
| [FMM] | Stata Finite Mixture Models Reference Manual |
| [FN] | Stata Functions Reference Manual |
| [G] | Stata Graphics Reference Manual |
| [IRT] | Stata Item Response Theory Reference Manual |
| [LASSO] | Stata Lasso Reference Manual |
| [XT] | Stata Longitudinal-Data/Panel-Data Reference Manual |
| [META] | Stata Meta-Analysis Reference Manual |
| [ME] | Stata Multilevel Mixed-Effects Reference Manual |
| [MI] | Stata Multiple-Imputation Reference Manual |
| [MV] | Stata Multivariate Statistics Reference Manual |
| [PSS] | Stata Power, Precision, and Sample-Size Reference Manual |
| [P] | Stata Programming Reference Manual |
| [RPT] | Stata Reporting Reference Manual |
| [SP] | Stata Spatial Autoregressive Models Reference Manual |
| [SEM] | Stata Structural Equation Modeling Reference Manual |
| [SVY] | Stata Survey Data Reference Manual |
| [ST] | Stata Survival Analysis Reference Manual |
| [TABLES] | Stata Customizable Tables and Collected Results Reference Manual |
| [TS] | Stata Time-Series Reference Manual |
| [I] | Stata Index |
| [M] | Mata Reference Manual |

## Title

Intro - Introduction to base reference manual

Description Remarks and examples Also see

## Description

This entry describes the organization of the reference manuals.

## Remarks and examples

The complete list of reference manuals is as follows:
[R] Stata Base Reference Manual
[ADAPT] Stata Adaptive Designs: Group Sequential Trials Reference Manual
[BAYES] Stata Bayesian Analysis Reference Manual
[BMA] Stata Bayesian Model Averaging Reference Manual
[CAUSAL] Stata Causal Inference and Treatment-Effects Estimation Reference Manual
[CM] Stata Choice Models Reference Manual
[D] Stata Data Management Reference Manual
[DSGE] Stata Dynamic Stochastic General Equilibrium Models Reference Manual
[ERM] Stata Extended Regression Models Reference Manual
[FMM] Stata Finite Mixture Models Reference Manual
[FN] Stata Functions Reference Manual
[G] Stata Graphics Reference Manual
[IRT] Stata Item Response Theory Reference Manual
[LASSO] Stata Lasso Reference Manual
[XT] Stata Longitudinal-Data/Panel-Data Reference Manual
[META] Stata Meta-Analysis Reference Manual
[ME] Stata Multilevel Mixed-Effects Reference Manual
[MI] Stata Multiple-Imputation Reference Manual
[MV] Stata Multivariate Statistics Reference Manual
[PSS] Stata Power, Precision, and Sample-Size Reference Manual
[P] Stata Programming Reference Manual
[RPT] Stata Reporting Reference Manual
[SP] Stata Spatial Autoregressive Models Reference Manual
[SEM] Stata Structural Equation Modeling Reference Manual
[SVY] Stata Survey Data Reference Manual
[ST] Stata Survival Analysis Reference Manual
[TABLES] Stata Customizable Tables and Collected Results Reference Manual
[TS] Stata Time-Series Reference Manual
[M] Mata Reference Manual

When we refer to "reference manuals", we mean all manuals listed above.
When we refer to the specialty manuals, we mean all the manuals listed above except [R].

## Arrangement of the reference manuals

Each manual contains the following sections:

- Contents.

A table of contents can be found at the beginning of each manual.

- Cross-referencing the documentation.

This entry lists all the manuals and explains how they are cross-referenced.

- Introduction.

This entry-usually called intro-provides an overview of the manual. In the specialty manuals, this introduction suggests entries that you might want to read first and provides information about new features.

Each specialty manual contains an overview of the commands described in it.

- Entries.

Entries are arranged in alphabetical order. Most entries describe Stata commands, but some entries discuss concepts, and others provide overviews.
Entries that describe estimation commands are followed by an entry discussing postestimation commands that are available for use after the estimation command. For example, the xtlogit entry in the [XT] manual is followed by the xtlogit postestimation entry.

- Glossary.

A glossary is contained in all the specialty manuals.
To find information and commands quickly, use Stata's search command; see $[R]$ search (see the entry search in the [R] manual).

Each reference manual does not contain its own author or subject index. A combined author index and a combined subject index for all reference manuals can be found in the Stata Index, [I]. This manual also contains a combined subject table of contents for all reference manuals and the User's Guide, an acronym glossary, and a vignette index.

## Arrangement of each entry

Entries in most of the Stata reference manuals contain the following sections, which are explained below:

Description<br>Quick start<br>Menu<br>Syntax<br>Options<br>Remarks and examples<br>Stored results<br>Methods and formulas<br>Acknowledgments<br>References<br>Also see

## Description

The purpose of the command is briefly described here.

## Quick start

A quick start lists common uses of the command and the corresponding syntax for each.
For details on the syntax elements shown in the Quick start examples and to further customize the examples shown, see the Syntax and Options sections of the entry.

If you prefer to use the GUI, see the Menu section of the entry.
For applied examples of the syntax, see Remarks and examples.

## Menu

A menu indicates how the dialog box for the command may be accessed using the menu system.

## Syntax

A command's syntax diagram shows how to type the command, indicates all possible options, and gives the minimal allowed abbreviations for all the items in the command. For instance, the syntax diagram for the summarize command is
summarize $[$ varlist $][$ if $][$ in $][$ weight $][$, options $]$
options
Main
detail
meanonly
format
separator(\#)
display_options

Description
display additional statistics
suppress the display; calculate only the mean; programmer's option use variable's display format draw separator line after every \# variables; default is separator (5) control spacing and base and empty cells
varlist may contain factor variables; see [U] 11.4.3 Factor variables.
varlist may contain time-series operators; see [U] 11.4.4 Time-series varlists.
by, collect, rolling, and statsby are allowed; see [U] 11.1.10 Prefix commands.
aweights, fweights, and iweights are allowed. However, iweights may not be used with the detail option; see
[U] 11.1.6 weight.
Items in the typewriter-style font should be typed exactly as they appear in the diagram, although they may be abbreviated. Underlining indicates the shortest abbreviations where abbreviations are allowed. For instance, summarize may be abbreviated su, sum, summ, etc., or it may be spelled out completely. Items in the typewriter font that are not underlined may not be abbreviated.

Square brackets denote optional items. In the syntax diagram above, varlist, if, in, weight, and the options are optional.

The options are listed in a table immediately following the diagram, along with a brief description of each.

Items typed in italics represent arguments for which you are to substitute variable names, observation numbers, and the like.

The diagrams use the following symbols:
\# Indicates a literal number, for example, 5; see [U] 12.2 Numbers.
[ ] Anything enclosed in brackets is optional.
$\} \quad$ At least one of the items enclosed in braces must appear.
| The vertical bar separates alternatives.
$\%$ fmt Any Stata format, for example, \%8.2f; see [U] 12.5 Formats: Controlling how data are displayed.
depvar The dependent variable in an estimation command; see [U] 20 Estimation and postestimation commands.
$\exp \quad$ Any algebraic expression, for example, (5+myvar)/2; see [U] 13 Functions and expressions.
filename Any filename; see [U] 11.6 Filenaming conventions.
indepvars The independent variables in an estimation command; see [U] 20 Estimation and postestimation commands.
newvar A variable that will be created by the current command; see [U] 11.4.2 Lists of new variables.
numlist A list of numbers; see [U] 11.1.8 numlist.
oldvar A previously created variable; see [U] 11.4.1 Lists of existing variables.
options A list of options; see [U] 11.1.7 options.
range An observation range, for example, 5/20; see [U] 11.1.4 in range.
"string" Any string of characters enclosed in double quotes; see [U] 12.4 Strings.
varlist A list of variable names; see [U] 11.4 varname and varlists. If varlist allows factor variables, a note to that effect will be shown below the syntax diagram; see [U] 11.4.3 Factor variables. If varlist allows time-series operators, a note to that effect will be shown below the syntax diagram; see [U] 11.4.4 Time-series varlists.
varname A variable name; see [U] 11.3 Naming conventions.
weight A [wgttype=exp] modifier; see [U] 11.1.6 weight and [U] 20.24 Weighted estimation.
xvar The variable to be displayed on the horizontal axis.
$y v a r \quad$ The variable to be displayed on the vertical axis.
The Syntax section will indicate whether factor variables or time-series operators may be used with a command. summarize allows factor variables and time-series operators.

If a command allows prefix commands, this will be indicated immediately following the table of options. summarize allows by.

If a command allows weights, the types of weights allowed will be specified, with the default weight listed first. summarize allows aweights, fweights, and iweights, and if the type of weight is not specified, the default is aweights.

## Options

If the command allows any options, they are explained here, and for dialog users the location of the options in the dialog is indicated. For instance, in the logistic entry in this manual, the Options section looks like this:
$\qquad$ Model


Maximization

## Remarks and examples

The explanations under Description and Options are exceedingly brief and technical; they are designed to provide a quick summary. The remarks explain in English what the preceding technical jargon means. Examples are used to illustrate the command.

Links to video examples posted on Stata's YouTube channel are provided at the end of this section.

## Stored results

Commands are classified as e-class, r-class, s-class, or n-class, according to whether they store calculated results in $e(), r(), s()$, or not at all. These results can then be used in subroutines by other programs (ado-files). Such stored results are documented here; see [U] 18.8 Accessing results calculated by other programs and [U] 18.9 Accessing results calculated by estimation commands.

## Methods and formulas

The techniques and formulas used in obtaining the results are described here as tersely and technically as possible.

## Acknowledgments

Some Stata commands began as community-contributed commands, or they were enhanced after suggestions by a Stata user. Here we acknowledge these contributions.

## References

Published sources are listed that either were directly referenced in the preceding text or might be of interest.

## Also see <br> Other manual entries relating to this entry are listed that might also interest you.

Elizabeth L. ("Betty") Scott (1917-1988) was an astronomer and mathematician trained at the University of California at Berkeley. She published her first paper when she was just 22 years old, and her work was focused on comets for much of her early academic career.

During World War II, Scott began working at the statistical laboratory at Berkeley, which had recently been established by Jerzy Neyman, sparking what would be a long and fruitful collaboration with him. After the war, she shifted her focus toward mathematics and statistics, partly because of limited career opportunities as an astronomer, though she still applied her research to astronomical topics. For example, in 1949 she published a paper using statistical techniques to analyze the distribution of binary star systems. She also published papers examining the distribution of galaxies, and she is the name behind the "Scott effect", which helps determine the distances to galaxies. Later in her career, Scott applied her statistical knowledge to problems associated with ozone depletion and its effects on the incidence of skin cancer as well as weather modification. She was also a champion of equality for women graduate students and faculty.
Among Scott's many awards and accomplishments, she was elected an honorary fellow of the Royal Statistical Society and was a fellow of the American Association for the Advancement of Science. In 1992, the Committee of Presidents of Statistical Societies established the Elizabeth L. Scott Award, a biannual award to recognize those who have strived to enhance the status of women within the statistics profession.

Janet Lippe Norwood (1923-2015) was born in Newark, New Jersey. She obtained her PhD from Tufts University and taught political science at Wellesley College. Norwood made significant contributions while she was the first female commissioner for the Bureau of Labor Statistics (BLS). She accomplished the goal of conducting the Consumer Expenditure Survey annually, a long-time goal of the BLS, and saved the National Longitudinal Survey from termination. As commissioner, she would present data on national unemployment before the Joint Economic Committee on a monthly basis, a duty that she performed with unwavering impartiality. Under her direction, the statistical quality of reported indicators improved, as did the cooperation of the BLS with the Census Bureau and the National Center for Health Statistics.

After her role as commissioner, she was the Chair of the Advisory Council on Unemployment Compensation, appointed by presidents George H. W. Bush and Bill Clinton. She also held a variety of other leadership positions, including president of the American Statistical Association, board member of the American Economic Association, and chair of a statistical committee for the OECD. Norwood was a trailblazer. In her honor, the University of Alabama at Birmingham created the Janet L. Norwood Award to recognize women in statistics.

## Also see

[U] 1.1 Getting Started with Stata

## Title

about - Display information about your Stata

Description Menu Syntax Remarks and examples Also see

## Description

about displays information about your version of Stata.

## Menu

Help $>$ About Stata

## Syntax

about

## Remarks and examples

If you are running Stata for Windows, information about memory is also displayed:

```
. about
StataNow/MP 18.5 for Windows (64-bit x86-64)
Revision date
Copyright 1985-2023 StataCorp LLC
Total usable memory: 8388608 KB
Stata license: 10-user 32-core network perpetual
Serial number: 18.5
    Licensed to: Stata Developer
                            StataCorp LLC
```


## Also see

[R] which - Display location of an ado-file
[U] 3 Resources for learning and using Stata
[U] 5 Editions of Stata

## Title

ado update - Update community-contributed packages

| Description | Quick start | Syntax | Options |
| :--- | :--- | :--- | :--- |
| Remarks and examples | Stored results | Also see |  |

## Description

ado update checks for available updates to community-contributed packages. To update packages, use ado update, update. By default, only packages in the PLUS directory are checked.

## Quick start

List available updates for community-contributed packages
ado update
Install updates for community-contributed packages
ado update, update
Install updates from Statistical Software Components (SSC) Archive only ado update, update ssconly

## Syntax

ado update $[$ pkglist $][$, options $]$

| options | Description |
| :--- | :--- |
| update | perform update; default is to list packages that have updates, but not to <br> update them |
| all | include packages that might have updates; default is to list or update <br> only packages that are known to have updates <br> check only packages obtained from SSC Archive; default is to check all installed <br> packages |
| ssconly | dir $($ dir $)$ <br> verbose packages installed in dir; default is to check those installed in PLUS <br> provide output to assist in debugging network problems |

## Options

update specifies that packages with updates be updated. The default is simply to list the packages that could be updated without actually performing the update.

The first time you ado update, do not specify this option. Once you see ado update work, you will be more comfortable with it. Then type

```
. ado update, update
```

The packages that can be updated will be listed and updated.
all is rarely specified. Sometimes, ado update cannot determine whether a package you previously installed has been updated. ado update can determine that the package is still available over the web but is unsure whether the package has changed. Usually, the package has not changed, but if you want to be certain that you are using the latest version, reinstall from the source.
Specifying all does this. Typing
. ado update, all
adds such packages to the displayed list as needing updating but does not update them. Typing
. ado update, update all
lists such packages and updates them.
ssconly specifies that ado update check only packages obtained from the Statistical Software Components (SSC) Archive at Boston College, which is provided at http://repec.org. See [R] ssc for more information on the SSC Archive.
$\operatorname{dir}(d i r)$ specifies which installed packages be checked. The default is $\operatorname{dir}$ (PLUS), and that is probably correct. If you are responsible for maintaining a large system, however, you may have previously installed packages in dir (SITE), where they are shared across users. See [P] sysdir for an explanation of these directory codewords. You may also specify an actual directory name, such as C: \mydir.
verbose is specified when you suspect network problems. It provides more detailed output that may help you diagnose the problem.

## Remarks and examples

Community-contributed additions to Stata are called packages and can add remarkable abilities to Stata. Community-contributed packages are updated by their developers, just as official Stata software is updated by StataCorp.

Do not confuse ado update with update. Use ado update to update community-contributed files. Use update to update the components (including ado-files) of the official Stata software. To use either command, you must be connected to the Internet.

Although Stata checks for updates automatically and can even be set to update automatically in Stata for Windows and Stata for Mac, you must remember to type ado update. Doing this regularly can help prevent errors that occur when accidentally running older versions of community-contributed packages.

Remarks are presented under the following headings:
Using ado update
Notes for developers

## Using ado update

The first time you try ado update, type

- ado update
ado update without the update option produces a report but does not update any files. The first time you run ado update, you may see messages such as

```
. ado update
note: ado update updates community-contributed files; type update to check for
    updates to official Stata.
Checking status of installed packages...
    [1] sjlatex at http://svn.stata.com/svn/press/press/production:
            installed package is up to date
        (output omitted)
```

Having the same packages installed multiple times can lead to confusion; ado update cleans that up.

To update all of your community-contributed packages that need updating, type

```
. ado update, update
```

You can also update a subset of your packages. You can specify one or many packages after the ado update command. You can even use wildcards such as st* to mean all packages that start with st or st*8 to mean all packages that start with st and end with 8 . For example, if the report indicated package st0008 had an update available, type the following to update that one package:

```
. ado update st0008, update
```


## Notes for developers

ado update reports whether an installed package is up to date by comparing its distribution date with that of the package available over the web.

If you are distributing software, include the line
d Distribution-Date: date
somewhere in your .pkg file. The capitalization of Distribution-Date does not matter, but include the hyphen and the colon as shown. Code the date in either of two formats:
all numeric: yyyymmdd, for example, 20200701
Stata standard: ddMONyyyy, for example, 01jul2020

## Stored results

ado update stores the following in $r()$ :
Macros
$r$ (pkglist) a space-separated list of package names that need updating (update not specified) or that were updated (update specified)

## Also see

[R] net - Install and manage community-contributed additions from the Internet
$[R]$ search - Search Stata documentation and other resources
[R] ssc - Install and uninstall packages from SSC
$[\mathrm{R}]$ update - Check for official updates

## Title

ameans - Arithmetic, geometric, and harmonic means

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| Acknowledgments | References | Also see |  |

## Description

ameans computes the arithmetic, geometric, and harmonic means, with their corresponding confidence intervals, for each variable in varlist or for all the variables in the data if varlist is not specified. gmeans and hmeans are synonyms for ameans.

## Quick start

Arithmetic, geometric, and harmonic means of variable v 1 ameans v1

Same as above, but for variables v1, v2, and v3
ameans v1 v2 v3
Means for all variables in the dataset
ameans
Add $n$ to each observation before calculating means
ameans v1, add(n)
Add $n$ to each observation only for variables with at least 1 nonpositive value ameans v1 v2 v3, add(n) only

Request $99 \%$ confidence intervals
ameans v1, level(99)

## Menu

Statistics $>$ Summaries, tables, and tests $>$ Summary and descriptive statistics $>$ Arith./geometric/harmonic means

## Syntax

```
ameans [varlist] [if] [in] [weight] [, options]
```

options

## Description

Main
add (\#)
only
level(\#)
add \# to each variable in varlist
add \# only to variables with nonpositive values
set confidence level; default is level (95)
by and collect are allowed; see [D] by.
aweights and fweights are allowed; see [U] 11.1.6 weight.

## Options

$\qquad$ Main
add(\#) adds the value \# to each variable in varlist before computing the means and confidence intervals. This option is useful when analyzing variables with nonpositive values.
only modifies the action of the add(\#) option so that it adds \# only to variables with at least one nonpositive value.
level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals.

## Remarks and examples

## $>$ Example 1

We have a dataset containing 8 observations on a variable named x . The eight values are 5, 4, $-4,-5,0,0$, missing, and 7 .

| ameans x <br> Variable | Type | Obs | Mean | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| x | Arithmetic | 7 | 1 | -3.204405 | 5.204405 |
|  | Geometric | 3 | 5.192494 | 2.57899 | 10.45448 |
|  | Harmonic | 3 | 5.060241 | 3.023008 | 15.5179 |


| . ameans $\mathrm{x}, \operatorname{add}(5)$ |  |  |  |  |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| Variable | Type | Obs | Mean | [95\% conf. interval] |  |
| x | Arithmetic | 7 | 6 | 1.795595 | $10.2044 *$ |
|  | Geometric | 6 | 5.477226 | 2.1096 | $14.22071^{*}$ |
|  | Harmonic | 6 | 3.540984 | . | .$*$ |

[^0]The number of observations displayed for the arithmetic mean is the number of nonmissing observations. The number of observations displayed for the geometric and harmonic means is the number of nonmissing, positive observations. Specifying the add (5) option produces 3 more positive observations. The confidence interval for the harmonic mean is not reported; see Methods and formulas below.

## Video example

Descriptive statistics in Stata

## Stored results

ameans stores the following in r() :
Scalars

| r (N) | number of nonmissing observations; used for arithmetic mean |
| :---: | :---: |
| r(N_pos) | number of nonmissing positive observations; used for geometric and harmonic means |
| r (mean) | arithmetic mean |
| r(lb) | lower bound of confidence interval for arithmetic mean |
| $\mathrm{r}(\mathrm{ub})$ | upper bound of confidence interval for arithmetic mean |
| r(Var) | variance of untransformed data |
| r (mean_g) | geometric mean |
| r(lb_g) | lower bound of confidence interval for geometric mean |
| r (ub_g) | upper bound of confidence interval for geometric mean |
| r (Var_g) | variance of $\ln x_{i}$ |
| r (mean_h) | harmonic mean |
| r(lb_h) | lower bound of confidence interval for harmonic mean |
| r (ub_h) | upper bound of confidence interval for harmonic mean |
| r (Var_h) | variance of $1 / x_{i}$ |
| r(level) | confidence level of confidence interval |

## Methods and formulas

See Armitage, Berry, and Matthews (2002) or Snedecor and Cochran (1989). For a history of the concept of the mean, see Plackett (1958).

When restricted to the same set of values (that is, to positive values), the arithmetic mean $(\bar{x})$ is greater than or equal to the geometric mean, which in turn is greater than or equal to the harmonic mean. Equality holds only if all values within a sample are equal to a positive constant.

The arithmetic mean and its confidence interval are identical to those provided by ci; see $[\mathrm{R}]$ ci.
To compute the geometric mean, ameans first creates $u_{j}=\ln x_{j}$ for all positive $x_{j}$. The arithmetic mean of the $u_{j}$ and its confidence interval are then computed as in ci. Let $\bar{u}$ be the resulting mean, and let $[L, U]$ be the corresponding confidence interval. The geometric mean is then $\exp (\bar{u})$, and its confidence interval is $[\exp (L), \exp (U)]$.

The same procedure is followed for the harmonic mean, except that then $u_{j}=1 / x_{j}$. The harmonic mean is then $1 / \bar{u}$, and its confidence interval is $[1 / U, 1 / L]$ if $L$ is greater than zero. If $L$ is not greater than zero, this confidence interval is not defined, and missing values are reported.

When weights are specified, ameans applies the weights to the transformed values, $u_{j}=\ln x_{j}$ and $u_{j}=1 / x_{j}$, respectively, when computing the geometric and harmonic means. For details on how the weights are used to compute the mean and variance of the $u_{j}$, see [R] summarize. Without weights, the formula for the geometric mean reduces to

$$
\exp \left\{\frac{1}{n} \sum_{j} \ln \left(x_{j}\right)\right\}
$$

Without weights, the formula for the harmonic mean is

$$
\frac{n}{\sum_{j} \frac{1}{x_{j}}}
$$

## Acknowledgments

This improved version of ameans is based on the gmci command (Carlin, Vidmar, and Ramalheira 1998) and was written by John Carlin of the Murdoch Children's Research Institute and the University of Melbourne; Suzanna Vidmar of the University of Melbourne; and Carlos Ramalheira of Coimbra University Hospital, Portugal.

## References

Armitage, P., G. Berry, and J. N. S. Matthews. 2002. Statistical Methods in Medical Research. 4th ed. Oxford: Blackwell.

Carlin, J. B., S. Vidmar, and C. Ramalheira. 1998. sg75: Geometric means and confidence intervals. Stata Technical Bulletin 41: 23-25. Reprinted in Stata Technical Bulletin Reprints, vol. 7, pp. 197-199. College Station, TX: Stata Press.

Keynes, J. M. 1911. The principal averages and the laws of error which lead to them. Journal of the Royal Statistical Society 74: 322-331. https://doi.org/10.2307/2340444.
Plackett, R. L. 1958. Studies in the history of probability and statistics: VII. The principle of the arithmetic mean. Biometrika 45: 130-135. https://doi.org/10.2307/2333051.
Snedecor, G. W., and W. G. Cochran. 1989. Statistical Methods. 8th ed. Ames, IA: Iowa State University Press.
Stigler, S. M. 1985. Arithmetric means. In Vol. 1 of Encyclopedia of Statistical Sciences, ed. S. Kotz and N. L. Johnson, 126-129. New York: Wiley.
Vogel, R. M. 2022. The geometric mean? Communications in Statistics—Theory and Methods 51: 82-94. https://doi.org/10.1080/03610926.2020.1743313.

[^1]
## Title

anova - Analysis of variance and covariance

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | References |
| Also see |  |  |  |

## Description

The anova command fits analysis-of-variance (ANOVA) and analysis-of-covariance (ANCOVA) models for balanced and unbalanced designs, including designs with missing cells; for repeated-measures ANOVA; and for factorial, nested, or mixed designs.

## Quick start

One-way ANOVA model of $y$ for factor a anova y a

Two-way full-factorial ANOVA for factors $a$ and $b$ anova y a b a\#b

Same as above
anova y a\#\#b
ANCOVA model including continuous variable x
anova y a\#\#b c.x
Factor b nested within a
anova y a / bla /
Repeated-measures ANOVA with repeated variable rvar
anova y a rvar, repeated(rvar)
Repeated-measures ANOVA with subjects, idvar, observed at each level of rvar anova y a / idvarla rvar rvar\#a, repeated(rvar)

## Menu

Statistics $>$ Linear models and related $>$ ANOVA/MANOVA $>$ Analysis of variance and covariance

## Syntax

$$
\text { anova varname }[\text { termlist }][\text { if }][\text { in }][\text { weight }][\text {, options }]
$$

where termlist is a factor-variable list (see [U] 11.4.3 Factor variables) with the following additional features:

- Variables are assumed to be categorical; use the c. factor-variable operator to override this.
- The $\mid$ symbol (indicating nesting) may be used in place of the \# symbol (indicating interaction).
- The / symbol is allowed after a term and indicates that the following term is the error term for the preceding terms.
options
Model
repeated (varlist)
partial
sequential
noconstant
dropemptycells
Adv. model
bse (term)
bseunit (varname)
grouping (varname)

Description
variables in terms that are repeated-measures variables
use partial (or marginal) sums of squares
use sequential sums of squares
suppress constant term
drop empty cells from the design matrix
between-subjects error term in repeated-measures ANOVA variable representing lowest unit in the between-subjects error term grouping variable for computing pooled covariance matrix
bootstrap, by, collect, fp, jackknife, and statsby are allowed; see [U] 11.1.10 Prefix commands.
Weights are not allowed with the bootstrap prefix; see $[R]$ bootstrap.
aweights are not allowed with the jackknife prefix; see [R] jackknife.
aweights and fweights are allowed; see [U] 11.1.6 weight.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

$\qquad$ Model
repeated (varlist) indicates the names of the categorical variables in the terms that are to be treated as repeated-measures variables in a repeated-measures ANOVA or ANCOVA.
partial presents the ANOVA table using partial (or marginal) sums of squares. This setting is the default. Also see the sequential option.
sequential presents the ANOVA table using sequential sums of squares.
noconstant suppresses the constant term (intercept) from the ANOVA or regression model.
dropemptycells drops empty cells from the design matrix. If c (emptycells) is set to keep (see $[\mathrm{R}]$ set emptycells), this option temporarily resets it to drop before running the ANOVA model. If c (emptycells) is already set to drop, this option does nothing.
bse (term) indicates the between-subjects error term in a repeated-measures ANOVA. This option is needed only in the rare case when the anova command cannot automatically determine the between-subjects error term.
bseunit (varname) indicates the variable representing the lowest unit in the between-subjects error term in a repeated-measures ANOVA. This option is rarely needed because the anova command automatically selects the first variable listed in the between-subjects error term as the default for this option.
grouping (varname) indicates a variable that determines which observations are grouped together in computing the covariance matrices that will be pooled and used in a repeated-measures ANOVA. This option is rarely needed because the anova command automatically selects the combination of all variables except the first (or as specified in the bseunit() option) in the between-subjects error term as the default for grouping observations.

## Remarks and examples

Remarks are presented under the following headings:

Introduction<br>One-way ANOVA<br>Two-way ANOVA<br>$N$-way ANOVA<br>Weighted data<br>ANCOVA<br>Nested designs<br>Mixed designs<br>Latin-square designs<br>Repeated-measures ANOVA<br>Video examples

## Introduction

anova uses least squares to fit the linear models known as ANOVA or ANCOVA (henceforth referred to simply as ANOVA models).

If you want to fit one-way ANOVA models, you may find the oneway or loneway command more convenient; see $[R]$ oneway and $[R]$ loneway. If you are interested in MANOVA or MANCOVA, see [MV] manova.

Structural equation modeling provides a more general framework for fitting ANOVA models; see the Stata Structural Equation Modeling Reference Manual.

ANOVA was pioneered by Fisher. It features prominently in his texts on statistical methods and his design of experiments $(1925,1935)$. Many books discuss ANOVA; see, for instance, Altman (1991); van Belle et al. (2004); Cobb (1998); Snedecor and Cochran (1989); or Winer, Brown, and Michels (1991). For a classic source, see Scheffé (1959). Kennedy and Gentle (1980) discuss ANOVA's computing problems. Edwards (1985) is concerned primarily with the relationship between multiple regression and ANOVA. Acock (2023, chap. 9) and Baldwin (2019, chap. 5 and 6) illustrate their discussion with Stata output. Repeated-measures ANOVA is discussed in Winer, Brown, and Michels (1991) and Milliken and Johnson (2009). Pioneering work in repeated-measures ANOVA can be found in Box (1954); Geisser and Greenhouse (1958); Huynh and Feldt (1976); and Huynh (1978). For a Stata-specific discussion of ANOVA contrasts, see Mitchell (2021, chap. 7-9; 2015, chap. 4-9).

## One-way ANOVA

anova, entered without options, performs and reports standard ANOVA. For instance, to perform a one-way layout of a variable called endog on exog, you would type anova endog exog.

## > Example 1: One-way ANOVA

We run an experiment varying the amount of fertilizer used in growing apple trees. We test four concentrations, using each concentration in three groves of 12 trees each. Later in the year, we measure the average weight of the fruit.

If all had gone well, we would have had 3 observations on the average weight for each of the four concentrations. Instead, two of the groves were mistakenly leveled by a confused man on a large bulldozer. We are left with the following data:

```
. use https://www.stata-press.com/data/r18/apple
(Apple trees)
. list, abbrev(10) sepby(treatment)
```

| treatment | weight |
| ---: | ---: |
| 1 | 117.5 |
| 1 | 113.8 |
| 1 | 104.4 |
| 2 | 48.9 |
| 2 | 50.4 |
| 2 | 58.9 |
| 3 | 70.4 |
| 3 | 86.9 |
| 4 | 87.7 |
| 4 | 67.3 |

To obtain one-way ANOVA results, we type . anova weight treatment

|  | $\begin{array}{ll} \text { Number of obs } & = \\ \text { Root MSE } & = \end{array}$ | $\begin{array}{r} 10 \\ 9.07002 \end{array}$ | R-squared |  | $\begin{aligned} & 0.9147 \\ & 0.8721 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | 5295.5443 | 3 | 1765.1814 | 21.46 | 0.0013 |
| treatment | 5295.5443 | 3 | 1765.1814 | 21.46 | 0.0013 |
| Residual | 493.59167 | 68 | 82.265278 |  |  |
| Total | 5789.136 | 96 | 643.23733 |  |  |

We find significant (at better than the $1 \%$ level) differences among the four concentrations.
Although the output is a usual ANOVA table, let's run through it anyway. Above the table is a summary of the underlying regression. The model was fit on 10 observations, and the root mean squared error (Root MSE) is 9.07. The $R^{2}$ for the model is 0.9147 , and the adjusted $R^{2}$ is 0.8721 .

The first line of the table summarizes the model. The sum of squares (Partial SS) for the model is 5295.5 with 3 degrees of freedom (df). This line results in a mean square (MS) of 5295.5/3 $\approx 1765.2$. The corresponding $F$ statistic is 21.46 and has a significance level of 0.0013 . Thus, the model appears to be significant at the $0.13 \%$ level.

The next line summarizes the first (and only) term in the model, treatment. Because there is only one term, the line is identical to that for the overall model.

The third line summarizes the residual. The residual sum of squares is 493.59 with 6 degrees of freedom, resulting in a mean squared error of 82.27 . The square root of this latter number is reported as the Root MSE.

The model plus the residual sum of squares equals the total sum of squares, which is reported as 5789.1 in the last line of the table. This is the total sum of squares of weight after removal of the mean. Similarly, the model plus the residual degrees of freedom sum to the total degrees of freedom, 9. Remember that there are 10 observations. Subtracting 1 for the mean, we are left with 9 total degrees of freedom.

## - Technical note

Rather than using the anova command, we could have performed this analysis by using the oneway command. Example 1 in $[R]$ oneway repeats this same analysis. You may wish to compare the output.

The regress command (see [R] regress) is used to fit the underlying regression model corresponding to an ANOVA model fit using the anova command. Type regress after anova to see the coefficients, standard errors, etc., of the regression model for the last run of anova.
> Example 2: Regression table from a one-way ANOVA
Returning to the apple tree experiment, we found that the fertilizer concentration appears to significantly affect the average weight of the fruit. Although that finding is interesting, we next want to know which concentration appears to grow the heaviest fruit. One way to find out is by examining the underlying regression coefficients.


See $[R]$ regress for an explanation of how to read this table. The baselevels option of regress displays a row indicating the base category for our categorical variable, treatment. In summary, we find that concentration 1, the base (omitted) group, produces significantly heavier fruits than concentration 2 , 3 , and 4 ; concentration 2 produces the lightest fruits; and concentrations 3 and 4 appear to be roughly equivalent.

## > Example 3: ANOVA replay

We previously typed anova weight treatment to produce and display the ANOVA table for our apple tree experiment. Typing regress displays the regression coefficients. We can redisplay the ANOVA table by typing anova without arguments:
. anova

|  | $\begin{array}{ll} \text { Number of obs } & = \\ \text { Root MSE } & = \end{array}$ | $\begin{array}{r} 10 \\ 9.07002 \end{array}$ | 2 R-squ | R -squared | $\begin{aligned} & 0.9147 \\ & 0.8721 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | 5295.5443 | 3 | 1765.1814 | 21.46 | 0.0013 |
| treatment | 5295.5443 | 3 | 1765.1814 | 21.46 | 0.0013 |
| Residual | 493.59167 | 6 | 82.265278 |  |  |
| Total | 5789.136 | 9 | 643.23733 |  |  |

## Two-way ANOVA

You can include multiple explanatory variables with the anova command, and you can specify interactions by placing '\#' between the variable names. For instance, typing anova y a b performs a two-way layout of y on a and b . Typing anova y a b a\#b performs a full two-way factorial layout. The shorthand anova y a\#\#b does the same.

With the default partial sums of squares, when you specify interacted terms, the order of the terms does not matter. Typing anova y a b a\#b is the same as typing anova y bab\#a.
> Example 4: Two-way factorial ANOVA
The classic two-way factorial ANOVA problem, at least as far as computer manuals are concerned, is a two-way ANOVA design from Afifi and Azen (1979).

Fifty-eight patients, each suffering from one of three different diseases, were randomly assigned to one of four different drug treatments, and the change in their systolic blood pressure was recorded. Here are the data:

|  | Disease 1 | Disease 2 | Disease 3 |
| :--- | :--- | :--- | :--- |
| Drug 1 | $42,44,36$ | $33,26,33$ | $31,-3,25$ |
|  | $13,19,22$ | 21 | 25,24 |
| Drug 2 | $28,23,34$ | $34,33,31$ | $3,26,28$ |
|  | 42,13 | 36 | $32,4,16$ |
| Drug 3 | $1,29,19$ | $11,9,7$ | $21,1,9$ |
|  |  | $1,-6$ | 3 |
| Drug 4 | $24,9,22$ | $27,12,12$ | $22,7,25$ |
|  | $-2,15$ | $-5,16,15$ | 5,12 |

Let's assume that we have entered these data into Stata and stored the data as systolic.dta. Below we use the data, list the first 10 observations, summarize the variables, and tabulate the control variables:

```
. use https://www.stata-press.com/data/r18/systolic
(Systolic blood pressure data)
. list in 1/10
```

|  | drug | disease | systolic |
| :---: | ---: | ---: | ---: |
| 1. | 1 | 1 | 42 |
| 2. | 1 | 1 | 44 |
| 3. | 1 | 1 | 36 |
| 4. | 1 | 1 | 13 |
| 5. | 1 | 1 | 19 |
| 6. | 1 | 1 | 22 |
| 7. | 1 | 2 | 33 |
| 8. | 1 | 2 | 26 |
| 9. | 1 | 2 | 33 |
| 10. | 1 | 2 | 21 |

. summarize

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| drug | 58 | 2.5 | 1.158493 | 1 | 4 |
| disease | 58 | 2.017241 | .8269873 | 1 | 3 |
| systolic | 58 | 18.87931 | 12.80087 | -6 | 44 |

. tabulate drug disease

|  | Patient's disease |  |  |  |
| ---: | :---: | :---: | :---: | ---: |
| Drug used | 1 | 2 | 3 | Total |
| 1 | 6 | 4 | 5 | 15 |
| 2 | 5 | 4 | 6 | 15 |
| 3 | 3 | 5 | 4 | 12 |
| 4 | 5 | 6 | 5 | 16 |
| Total | 19 | 19 | 20 | 58 |

Each observation in our data corresponds to one patient, and for each patient we record drug, disease, and the increase in the systolic blood pressure, systolic. The tabulation reveals that the data are not balanced-there are not equal numbers of patients in each drug-disease cell. Stata does not require that the data be balanced. We can perform a two-way factorial ANOVA by typing
. anova systolic drug disease drug\#disease

|  | $\begin{array}{ll} \text { Number of obs } & = \\ \text { Root MSE } & = \end{array}$ | $\begin{array}{r} 58 \\ 10.5096 \end{array}$ | R-squared = <br> Adj R-squared $=$ |  | $\begin{aligned} & 0.4560 \\ & 0.3259 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | 4259.3385 | 113 | 387.21259 | 3.51 | 0.0013 |
| drug | 2997.4719 | 39 | 999.15729 | 9.05 | 0.0001 |
| disease | 415.87305 | 2 | 207.93652 | 1.88 | 0.1637 |
| drug\#disease | 707.26626 | 61 | 117.87771 | 1.07 | 0.3958 |
| Residual | 5080.8167 | $46 \quad 1$ | 110.45254 |  |  |
| Total | 9340.1552 | $57 \quad 1$ | 163.86237 |  |  |

Although Stata's table command does not perform ANOVA, it can produce useful summary tables of your data (see [R] table):

```
. table drug disease, statistic(mean systolic) nformat(%8.2f)
```

|  | Patient's disease |  |  |  |
| :---: | :---: | :---: | :---: | :---: |
|  | 1 | 2 | 3 | Total |
| Drug used |  |  |  |  |
| 1 | 29.33 | 28.25 | 20.40 | 26.07 |
| 2 | 28.00 | 33.50 | 18.17 | 25.53 |
| 3 | 16.33 | 4.40 | 8.50 | 8.75 |
| 4 | 13.60 | 12.83 | 14.20 | 13.50 |
| Total | 22.79 | 18.21 | 15.80 | 18.88 |

These are simple means and are not influenced by our anova model. More useful is the margins command (see [R] margins) that provides marginal means and adjusted predictions. Because drug is the only significant factor in our ANOVA, we now examine the adjusted marginal means for drug.

```
. margins drug, asbalanced
Adjusted predictions Number of obs = 58
Expression: Linear prediction, predict()
At: drug (asbalanced)
        disease (asbalanced)
```

|  | Delta-method |  |  |  |  | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| drug |  |  |  |  |  |  |
| 1 | 25.99444 | 2.751008 | 9.45 | 0.000 | 20.45695 | 31.53194 |
| 2 | 26.55556 | 2.751008 | 9.65 | 0.000 | 21.01806 | 32.09305 |
| 3 | 9.744444 | 3.100558 | 3.14 | 0.003 | 3.503344 | 15.98554 |
| 4 | 13.54444 | 2.637123 | 5.14 | 0.000 | 8.236191 | 18.8527 |

These adjusted marginal predictions are not equal to the simple drug means (see the totals from the table command); they are based upon predictions from our ANOVA model. The asbalanced option of margins corresponds with the interpretation of the $F$ statistic produced by ANOVA-each cell is given equal weight regardless of its sample size (see the following three technical notes). You can omit the asbalanced option and obtain predictive margins that take into account the unequal sample sizes of the cells.
. margins drug
Predictive margins $\quad$ Number of obs $=58$
Expression: Linear prediction, predict()

|  | Delta-method <br> std. err. |  |  |  | t | $\mathrm{P}>\|\mathrm{t}\|$ |
| :---: | ---: | ---: | ---: | ---: | ---: | ---: |
| Margin | [95\%conf. interval] |  |  |  |  |  |
| drug |  |  |  |  |  |  |
| 1 | 25.89799 | 2.750533 | 9.42 | 0.000 | 20.36145 | 31.43452 |
| 2 | 26.41092 | 2.742762 | 9.63 | 0.000 | 20.89003 | 31.93181 |
| 3 | 9.722989 | 3.099185 | 3.14 | 0.003 | 3.484652 | 15.96132 |
| 4 | 13.55575 | 2.640602 | 5.13 | 0.000 | 8.24049 | 18.871 |

## Technical note

How do you interpret the significance of terms like drug and disease in unbalanced data? If you are familiar with SAS, the sums of squares and the $F$ statistic reported by Stata correspond to SAS type III sums of squares. (Stata can also calculate sequential sums of squares, but we will postpone that topic for now.)

Let's think in terms of the following table:

|  | Disease 1 | Disease 2 | Disease 3 |  |
| :--- | :---: | :---: | :---: | :---: |
| Drug 1 | $\mu_{11}$ | $\mu_{12}$ | $\mu_{13}$ | $\mu_{1}$. |
| Drug 2 | $\mu_{21}$ | $\mu_{22}$ | $\mu_{23}$ | $\mu_{2}$ |
| Drug 3 | $\mu_{31}$ | $\mu_{32}$ | $\mu_{33}$ | $\mu_{3}$ |
| Drug 4 | $\mu_{41}$ | $\mu_{42}$ | $\mu_{43}$ | $\mu_{4}$ |
|  | $\mu \cdot 1$ | $\mu_{\cdot 2}$ | $\mu \cdot 3$ | $\mu \cdot$. |

In this table, $\mu_{i j}$ is the mean increase in systolic blood pressure associated with drug $i$ and disease $j$, while $\mu_{i}$. is the mean for drug $i, \mu_{\cdot j}$ is the mean for disease $j$, and $\mu$.. is the overall mean.

If the data are balanced, meaning that there are equal numbers of observations going into the calculation of each mean $\mu_{i j}$, the row means, $\mu_{i}$, are given by

$$
\mu_{i}=\frac{\mu_{i 1}+\mu_{i 2}+\mu_{i 3}}{3}
$$

In our case, the data are not balanced, but we define the $\mu_{i}$. according to that formula anyway. The test for the main effect of drug is the test that

$$
\mu_{1 .}=\mu_{2}=\mu_{3 .}=\mu_{4}
$$

To be absolutely clear, the $F$ test of the term drug, called the main effect of drug, is formally equivalent to the test of the three constraints:

$$
\begin{aligned}
& \frac{\mu_{11}+\mu_{12}+\mu_{13}}{3}=\frac{\mu_{21}+\mu_{22}+\mu_{23}}{3} \\
& \frac{\mu_{11}+\mu_{12}+\mu_{13}}{3}=\frac{\mu_{31}+\mu_{32}+\mu_{33}}{3} \\
& \frac{\mu_{11}+\mu_{12}+\mu_{13}}{3}=\frac{\mu_{41}+\mu_{42}+\mu_{43}}{3}
\end{aligned}
$$

In our data, we obtain a significant $F$ statistic of 9.05 and thus reject those constraints.

## - Technical note

Stata can display the symbolic form underlying the test statistics it presents, as well as display other test statistics and their symbolic forms; see Obtaining symbolic forms in [R] anova postestimation. Here is the result of requesting the symbolic form for the main effect of drug in our data:

```
. test drug, symbolic
drug
    1 -(r2+r3+r4)
    2 r2
    r3
    r4
disease
    10
    2 0
    3 0
drug#disease
    1 1 -1/3 (r2+r3+r4)
        1 2 -1/3 (r2+r3+r4)
        1 3 -1/3 (r2+r3+r4)
        2 1 1/3 r2
        2 2 1/3 r2
        2 3 1/3 r2
        3 1 1/3 r3
        3 2 1/3 r3
        3 3 1/3 r3
        4 1 1/3 r4
        4 2 1/3 r4
    4 3 1/3 r4
_cons
```

This says exactly what we said in the previous technical note.

## - Technical note

Saying that there is no main effect of a variable is not the same as saying that it has no effect at all. Stata's ability to perform ANOVA on unbalanced data can easily be put to ill use.

For example, consider the following table of the probability of surviving a bout with one of two diseases according to the drug administered to you:

|  | Disease 1 | Disease 2 |
| :--- | :---: | :---: |
| Drug 1 | 1 | 0 |
| Drug 2 | 0 | 1 |

If you have disease 1 and are administered drug 1, you live. If you have disease 2 and are administered drug 2, you live. In all other cases, you die.

This table has no main effects of either drug or disease, although there is a large interaction effect. You might now be tempted to reason that because there is only an interaction effect, you would be indifferent between the two drugs in the absence of knowledge about which disease infects you. Given an equal chance of having either disease, you reason that it does not matter which drug is administered to you-either way, your chances of surviving are 0.5 .

You may not, however, have an equal chance of having either disease. If you knew that disease 1 was 100 times more likely to occur in the population, and if you knew that you had one of the two diseases, you would express a strong preference for receiving drug 1.

When you calculate the significance of main effects on unbalanced data, you must ask yourself why the data are unbalanced. If the data are unbalanced for random reasons and you are making predictions for a balanced population, the test of the main effect makes perfect sense. If, however, the data are unbalanced because the underlying populations are unbalanced and you are making predictions for such unbalanced populations, the test of the main effect may be practically-if not statistically - meaningless.

## > Example 5: ANOVA with missing cells

Stata can perform ANOVA not only on unbalanced populations, but also on populations that are so unbalanced that entire cells are missing. For instance, using our systolic blood pressure data, let's refit the model eliminating the drug 1 -disease 1 cell. Because anova follows the same syntax as all other Stata commands, we can explicitly specify the data to be used by typing the if qualifier at the end of the anova command. Here we want to use the data that are not for drug 1 and disease 1 :

|  | $\begin{aligned} \text { Number of obs } & = \\ \text { Root MSE } & = \end{aligned}$ | $\begin{array}{r} 52 \\ 10.1615 \end{array}$ | R-squared = <br> Adj R-squared = |  | $\begin{aligned} & 0.4545 \\ & 0.3215 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | 3527.959 | 10 | 352.7959 | 3.42 | 0.0025 |
| drug | 2686.5783 | 38 | 895.52611 | 8.67 | 0.0001 |
| disease | 327.7926 | 2 | 163.8963 | 1.59 | 0.2168 |
| drug\#disease | 703.0076 | 51 | 140.60152 | 1.36 | 0.2586 |
| Residual | 4233.4833 | 41 1 | 103.25569 |  |  |
| Total | 7761.4423 | 51 | 152.18514 |  |  |

Here we used drug\#\#disease as a shorthand for drug disease drug\#disease.

## - Technical note

The test of the main effect of drug in the presence of missing cells is more complicated than that for unbalanced data. Our underlying tableau now has the following form:

|  | Disease 1 | Disease 2 | Disease 3 |  |
| :--- | :---: | :---: | :---: | :---: |
| Drug 1 |  | $\mu_{12}$ | $\mu_{13}$ |  |
| Drug 2 | $\mu_{21}$ | $\mu_{22}$ | $\mu_{23}$ | $\mu_{2}$. |
| Drug 3 | $\mu_{31}$ | $\mu_{32}$ | $\mu_{33}$ | $\mu_{3}$. |
| Drug 4 | $\mu_{41}$ | $\mu_{42}$ | $\mu_{43}$ | $\mu_{4}$. |
|  |  | $\mu_{\cdot 2}$ | $\mu_{\cdot 3}$ |  |

The hole in the drug 1-disease 1 cell indicates that the mean is unobserved. Considering the main effect of drug, the test is unchanged for the rows in which all the cells are defined:

$$
\mu_{2}=\mu_{3 .}=\mu_{4}
$$

The first row, however, requires special attention. Here we want the average outcome for drug 1, which is averaged only over diseases 2 and 3 , to be equal to the average values of all other drugs averaged over those same two diseases:

$$
\frac{\mu_{12}+\mu_{13}}{2}=\frac{\left(\mu_{22}+\mu_{23}\right) / 2+\left(\mu_{32}+\mu_{33}\right) / 2+\left(\mu_{42}+\mu_{43}\right) / 2}{3}
$$

Thus, the test contains three constraints:

$$
\begin{array}{rcc}
\frac{\mu_{21}+\mu_{22}+\mu_{23}}{3} & = & \frac{\mu_{31}+\mu_{32}+\mu_{33}}{3} \\
\frac{\mu_{21}+\mu_{22}+\mu_{23}}{3} & = & \frac{\mu_{41}+\mu_{42}+\mu_{43}}{3} \\
\frac{\mu_{12}+\mu_{13}}{2} & = & \frac{\mu_{22}+\mu_{23}+\mu_{32}+\mu_{33}+\mu_{42}+\mu_{43}}{6}
\end{array}
$$

Stata can calculate two types of sums of squares, partial and sequential. If you do not specify which sums of squares to calculate, Stata calculates partial sums of squares. The technical notes above have gone into great detail about the definition and use of partial sums of squares. Use the sequential option to obtain sequential sums of squares.

## - Technical note

Before we illustrate sequential sums of squares, consider one more feature of the partial sums. If you know how such things are calculated, you may worry that the terms must be specified in some particular order, that Stata would balk or, even worse, produce different results if you typed, say, anova systolic drug\#disease drug disease rather than anova systolic drug disease drug\#disease. We assure you that is not the case.

When you type a model, Stata internally reorganizes the terms, forms the cross-product matrix, inverts it, converts the result to an upper-Hermite form, and then performs the hypothesis tests. As a final touch, Stata reports the results in the same order that you typed the terms.

## > Example 6: Sequential sums of squares

We wish to estimate the effects on systolic blood pressure of drug and disease by using sequential sums of squares. We want to introduce disease first, then drug, and finally, the interaction of drug and disease:

|  | $\begin{array}{ll} \text { Number of obs } & = \\ \text { Root MSE } & = \end{array}$ | $\begin{array}{r} 58 \\ 10.5096 \end{array}$ | R -squared $=$ <br> Adj R -squared $=$ |  | $\begin{aligned} & 0.4560 \\ & 0.3259 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Seq. SS | df | MS | F | Prob>F |
| Model | 4259.3385 | 113 | 387.21259 | 3.51 | 0.0013 |
| disease | 488.63938 | 2 | 244.31969 | 2.21 | 0.1210 |
| drug | 3063.4329 | 31 | 1021.1443 | 9.25 | 0.0001 |
| disease\#drug | 707.26626 | 61 | 117.87771 | 1.07 | 0.3958 |
| Residual | 5080.8167 | $46 \quad 1$ | 110.45254 |  |  |
| Total | 9340.1552 | $57 \quad 1$ | 163.86237 |  |  |

The $F$ statistic on disease is now 2.21 . When we fit this same model by using partial sums of squares, the statistic was 1.88 .

## N-way ANOVA

You may include high-order interaction terms, such as a third-order interaction between the variables $\mathrm{A}, \mathrm{B}$, and C, by typing A\#B\#C.

## > Example 7: Three-way factorial ANOVA

We wish to determine the operating conditions that maximize yield for a manufacturing process. There are three temperature settings, two chemical supply companies, and two mixing methods under investigation. Three observations are obtained for each combination of these three factors.


Sorted by:

We wish to perform a three-way factorial ANOVA. We could type

```
. anova yield temp chem temp#chem meth temp#meth chem#meth temp#chem#meth
```

but prefer to use the \#\# factor-variable operator for brevity.

|  | $\begin{aligned} \text { Number of obs } & = \\ \text { Root MSE } & = \end{aligned}$ | $\begin{array}{r} 36 \\ 2.62996 \end{array}$ | R-squared =Adj R-squared = |  | $\begin{aligned} & 0.5474 \\ & 0.3399 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | 200.75 | 11 | 18.25 | 2.64 | 0.0227 |
| temperature | 30.5 | 2 | 15.25 | 2.20 | 0.1321 |
| chemical | 12.25 | 1 | 12.25 | 1.77 | 0.1958 |
| temperature\#chemical | 24.5 | 2 | 12.25 | 1.77 | 0.1917 |
| method | 42.25 | 1 | 42.25 | 6.11 | 0.0209 |
| temperature\#method | 87.5 | 2 | 43.75 | 6.33 | 0.0062 |
| chemical\#method | . 25 | 1 | . 25 | 0.04 | 0.8508 |
| temperature\#chemical\# method | 3.5 | 2 | 1.75 | 0.25 | 0.7785 |
| Residual | 166 | 24 | 6.9166667 |  |  |
| Total | 366.75 | 35 | 10.478571 |  |  |

The interaction between temperature and method appears to be the important story in these data. A table of means for this interaction is given below.

| table method temp, statistic(mean yield) nformat ( |  |  |  |  |
| :--- | :---: | :---: | :---: | ---: |
|  | Machine temperature setting <br> Low |  |  |  |
|  | Medium | High | Total |  |
| Mixing method |  |  |  |  |
| Stir | 7.50 | 6.00 | 6.00 | 6.50 |
| Fold | 5.50 | 9.00 | 11.50 | 8.67 |
| Total | 6.50 | 7.50 | 8.75 | 7.58 |

Here our ANOVA is balanced (each cell has the same number of observations), and we obtain the same values as in the table above (but with additional information such as confidence intervals) by using the margins command. Because our ANOVA is balanced, using the asbalanced option with margins would not produce different results. We request the predictive margins for the two terms that appear significant in our ANOVA: temperature\#method and method.

| ```Predictive margins Number of obs = 36 Expression: Linear prediction, predict()``` |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  |  |  |  |
|  | Margin | lta-method std. err. | t | $P>\|t\|$ | [95\% conf | interval] |
| temperature\# method |  |  |  |  |  |  |
| Low\#Stir | 7.5 | 1.073675 | 6.99 | 0.000 | 5.284044 | 9.715956 |
| Low\#Fold | 5.5 | 1.073675 | 5.12 | 0.000 | 3.284044 | 7.715956 |
| Medium\#Stir | 6 | 1.073675 | 5.59 | 0.000 | 3.784044 | 8.215956 |
| Medium\#Fold | 9 | 1.073675 | 8.38 | 0.000 | 6.784044 | 11.21596 |
| High\#Stir | 6 | 1.073675 | 5.59 | 0.000 | 3.784044 | 8.215956 |
| High\#Fold | 11.5 | 1.073675 | 10.71 | 0.000 | 9.284044 | 13.71596 |
| method |  |  |  |  |  |  |
| Stir | 6.5 | . 6198865 | 10.49 | 0.000 | 5.220617 | 7.779383 |
| Fold | 8.666667 | . 6198865 | 13.98 | 0.000 | 7.387284 | 9.946049 |

We decide to use the folding method of mixing and a high temperature in our manufacturing process.

## Weighted data

Like all estimation commands, anova can produce estimates on weighted data. See [U] 11.1.6 weight for details on specifying the weight.

## > Example 8: Three-way factorial ANOVA on grouped data

We wish to investigate the prevalence of byssinosis, a form of pneumoconiosis that can afflict workers exposed to cotton dust. We have data on 5,419 workers in a large cotton mill. We know whether each worker smokes, his or her race, and the dustiness of the work area. The variables are

| smokes | smoker or nonsmoker in the last five years |
| :--- | :--- |
| race | white or other |
| workplace | 1 (most dusty), 2 (less dusty), 3 (least dusty) |

We wish to fit an ANOVA model explaining the prevalence of byssinosis according to a full factorial model of smokes, race, and workplace.

The data are unbalanced. Moreover, although we have data on 5,419 workers, the data are grouped according to the explanatory variables, along with some other variables, resulting in 72 observations. For each observation, we know the number of workers in the group (pop), the prevalence of byssinosis (prob), and the values of the three explanatory variables. Thus, we wish to fit a three-way factorial model on grouped data.

We begin by showing a bit of the data, which are from Higgins and Koch (1977).


Sorted by:
. list in 1/5, abbrev(10) divider

| smokes | race | workplace | pop | prob |
| :---: | :---: | ---: | ---: | ---: |
| Yes | White | Most | 40 | .075 |
| Yes | White | Less | 74 | 0 |
| Yes | White | Least | 260 | .0076923 |
| Yes | Other | Most | 164 | .152439 |
| Yes | Other | Less | 88 | 0 |

The first observation in the data represents a group of 40 white workers who smoke and work in a "most" dusty work area. Of those 40 workers, $7.5 \%$ have byssinosis. The second observation represents a group of 74 white workers who also smoke but who work in a "less" dusty environment. None of those workers has byssinosis.

Almost every Stata command allows weights. Here we want to weight the data by pop. We can, for instance, make a table of the number of workers by their smoking status and race:

| Smokes | Other | White | Total |
| :---: | :---: | :---: | :---: |
| No | 799 | 1,431 | 2,230 |
| Yes | 1,104 | 2,085 | 3,189 |
| Total | 1,903 | 3,516 | 5,419 |

The [ $f \mathrm{w}=\mathrm{pop}$ ] at the end of the tabulate command tells Stata to count each observation as representing pop persons. When making the tally, tabulate treats the first observation as representing 40 workers, the second as representing 74 workers, and so on.

Similarly, we can make a table of the dustiness of the workplace:
. tabulate workplace [fw=pop]

| Dustiness <br> of <br> workplace | Freq. | Percent | Cum. |
| ---: | ---: | ---: | ---: |
| Least | 3,450 | 63.66 | 63.66 |
| Less | 1,300 | 23.99 | 87.65 |
| Most | 669 | 12.35 | 100.00 |
| Total | 5,419 | 100.00 |  |

We can discover the average incidence of byssinosis among these workers by typing . summarize prob [fw=pop]

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| prob | 5,419 | .0304484 | .0567373 | 0 | .287037 |

We discover that $3.04 \%$ of these workers have byssinosis. Across all cells, the byssinosis rates vary from 0 to $28.7 \%$. Just to prove that there might be something here, let's obtain the average incidence rates according to the dustiness of the workplace:
. table (workplace) (race smokes) [fw=pop], statistic(mean prob) nototals

|  | Race |  |  |  |
| :---: | :---: | :---: | :---: | :---: |
|  | Other <br> Smokes |  | White Smokes |  |
|  | No | Yes | No | Yes |
| Dustiness of workplace |  |  |  |  |
| Least | . 0107527 | . 0101523 | . 0081549 | . 0162774 |
| Less | . 02 | . 0081633 | . 0136612 | . 0143149 |
| Most | . 0820896 | . 1679105 | . 0833333 | . 2295082 |

Let's now fit the ANOVA model.
. anova prob workplace smokes race workplace\#smokes workplace\#race smokes\#race
( workplace\#smokes\#race [aweight=pop]
(sum of wgt is 5,419 )

|  | $\begin{array}{ll} \text { Number of obs } & = \\ \text { Root MSE } & = \end{array}$ | $\begin{array}{r} 65 \\ .025902 \end{array}$ | R-squared |  | $\begin{aligned} & 0.8300 \\ & 0.7948 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | . 17364654 | 11 | . 01578605 | 23.53 | 0.0000 |
| workplace | . 09762518 | 2 | . 04881259 | 72.76 | 0.0000 |
| smokes | . 01303081 | 1. | . 01303081 | 19.42 | 0.0001 |
| race | . 00109472 | 1 | . 00109472 | 1.63 | 0.2070 |
| workplace\#smokes | . 01969034 | 2 | . 00984517 | 14.67 | 0.0000 |
| workplace\#race | . 00135252 | 2 | . 00067626 | 1.01 | 0.3718 |
| smokes\#race | . 00166287 | 1 | . 00166287 | 2.48 | 0.1214 |
| workplace\#smokes\#race | . 00095084 | 2 | . 00047542 | 0.71 | 0.4969 |
| Residual | . 03555777 | 53 | . 0006709 |  |  |
| Total | . 2092043 | 64 . | . 00326882 |  |  |

Of course, if we want to see the underlying regression, we could type regress.

Above, we examined simple means of the cells of workplace\#smokes\#race. Our ANOVA shows workplace, smokes, and their interaction as being the only significant factors in our model. We now examine the predictive marginal mean byssinosis rates for these terms.

```
. margins workplace#smokes workplace smokes
Predictive margins Number of obs = 65
Expression: Linear prediction, predict()
```

|  | Delta-method <br> std. err. |  |  |  |  | t |
| :---: | ---: | ---: | ---: | ---: | ---: | ---: |
|  | Margin | $\mathrm{P}>\|\mathrm{tl}\|$ | [95\% conf. interval] |  |  |  |
| workplace\# <br> smokes |  |  |  |  |  |  |
| Least\#No | .0090672 | .0062319 | 1.45 | 0.152 | -.0034323 | .0215667 |
| Least\#Yes | .0141264 | .0053231 | 2.65 | 0.010 | .0034497 | .0248032 |
| Less\#No | .0158872 | .009941 | 1.60 | 0.116 | -.0040518 | .0358263 |
| Less\#Yes | .0121546 | .0087353 | 1.39 | 0.170 | -.0053662 | .0296755 |
| Most\#No | .0828966 | .0182151 | 4.55 | 0.000 | .0463617 | .1194314 |
| Most\#Yes | .2078768 | .012426 | 16.73 | 0.000 | .1829533 | .2328003 |
|  |  |  |  |  |  |  |
| workplace |  |  |  |  |  |  |
| Least | .0120701 | .0040471 | 2.98 | 0.004 | .0039526 | .0201875 |
| Less | .0137273 | .0065685 | 2.09 | 0.041 | .0005526 | .0269019 |
| Most | .1566225 | .0104602 | 14.97 | 0.000 | .1356419 | .177603 |
| smokes |  |  |  |  |  |  |
| No | .0196915 | .0050298 | 3.91 | 0.000 | .0096029 | .02978 |
| Yes | .0358626 | .0041949 | 8.55 | 0.000 | .0274488 | .0442765 |

Smoking combined with the most dusty workplace produces the highest byssinosis rates.

Ronald Aylmer Fisher (1890-1962) (Sir Ronald from 1952) studied mathematics at Cambridge. Even before he finished his studies, he had published on statistics. He worked as a statistician at Rothamsted Experimental Station (1919-1933), as professor of eugenics at University College London (1933-1943), as professor of genetics at Cambridge (1943-1957), and in retirement at the CSIRO Division of Mathematical Statistics in Adelaide. His many fundamental and applied contributions to statistics and genetics mark him as one of the greatest statisticians of all time, including original work on tests of significance, distribution theory, theory of estimation, fiducial inference, and design of experiments.

[^2]```
. use https://www.stata-press.com/data/r18/census2
(1980 Census data by state)
. summarize drate age region
```

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | :---: | :---: | :---: | ---: | ---: |
| drate | 50 | 84.3 | 13.07318 | 40 | 107 |
| age | 50 | 29.5 | 1.752549 | 24 | 35 |
| region | 50 | 2.66 | 1.061574 | 1 | 4 |

age is coded in integral years from 24 to 35 , and region is coded from 1 to 4 , with 1 standing for the Northeast, 2 for the North Central, 3 for the South, and 4 for the West.

When we examine the data more closely, we discover large differences in the deathrate across regions of the country:

| . tabulate region, summarize(drate) |  |  |  |
| ---: | ---: | :--- | ---: |
| Census <br> region | Summary of Deathrate <br> Mean <br> Std. dev. |  | Freq. |
| NE | 93.444444 | 7.0553368 | 9 |
| N Cntrl | 88.916667 | 5.5833899 | 12 |
| South | 88.3125 | 8.5457104 | 16 |
| West | 68.769231 | 13.342625 | 13 |
| Total | 84.3 | 13.073185 | 50 |

Naturally, we wonder if these differences might not be explained by differences in the median ages of the populations. To find out, we fit a regression model (via anova) of drate on region and age. In the anova example below, we treat age as a categorical variable.
. anova drate region age

|  | $\begin{aligned} \text { Number of obs } & = \\ \text { Root MSE } & = \end{aligned}$ | $\begin{array}{r} 50 \\ 6.7583 \end{array}$ | R -squared $=$ <br> Adj R -squared $=$ |  | $\begin{aligned} & 0.7927 \\ & 0.7328 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | 6638.8653 | 11 | 603.53321 | 13.21 | 0.0000 |
| region | 1320.0097 | 3 | 440.00324 | 9.63 | 0.0001 |
| age | 2237.2494 | 8 | 279.65617 | 6.12 | 0.0000 |
| Residual | 1735.6347 | 38 | 45.674598 |  |  |
| Total | 8374.5 | 49 | 170.90816 |  |  |

We have the answer to our question: differences in median ages do not eliminate the differences in deathrates across the four regions. The ANOVA table summarizes the two terms in the model, region and age. The region term contains 3 degrees of freedom, and the age term contains 8 degrees of freedom. Both are significant at better than the $1 \%$ level.

The age term contains 8 degrees of freedom. Because we did not explicitly indicate that age was to be treated as a continuous variable, it was treated as categorical, meaning that unique coefficients were estimated for each level of age. The only clue of this labeling is that the number of degrees of freedom associated with the age term exceeds 1 . The labeling becomes more obvious if we review the regression coefficients:
. regress, baselevels

| Source | SS | df | MS |
| ---: | ---: | :---: | :---: |
| Model | 6638.86529 | 11 | 603.533208 |
| Residual | 1735.63471 | 38 | 45.6745977 |
| Total | 8374.5 | 49 | 170.908163 |


| Number of obs | $=$ | 50 |
| :--- | :--- | ---: |
| $\mathrm{~F}(11,38)$ | $=$ | 13.21 |
| Prob $>$ F | $=$ | 0.0000 |
| R-squared | $=$ | 0.7927 |
| Adj R-squared | $=$ | 0.7328 |
| Root MSE | $=$ | 6.7583 |


| drate | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| :---: | ---: | :--- | ---: | :--- | ---: | ---: |
| region |  |  |  |  |  |  |
| NE | 0 | (base) |  |  |  |  |
| N Cntrl | .4428387 | 3.983664 | 0.11 | 0.912 | -7.621668 | 8.507345 |
| South | -.2964637 | 3.934766 | -0.08 | 0.940 | -8.261981 | 7.669054 |
| West | -13.37147 | 4.195344 | -3.19 | 0.003 | -21.8645 | -4.878439 |
|  |  |  |  |  |  |  |
| age | 0 | (base) |  |  |  |  |
| 24 | -15 | 9.557677 | -1.57 | 0.125 | -34.34851 | 4.348506 |
| 26 | 14.30833 | 7.857378 | 1.82 | 0.076 | -1.598099 | 30.21476 |
| 27 | 12.66011 | 7.495513 | 1.69 | 0.099 | -2.51376 | 27.83399 |
| 28 | 18.861 | 7.28918 | 2.59 | 0.014 | 4.104825 | 33.61717 |
| 29 | 20.87003 | 7.210148 | 2.89 | 0.006 | 6.273847 | 35.46621 |
| 30 | 29.91307 | 8.242741 | 3.63 | 0.001 | 13.22652 | 46.59963 |
| 31 | 27.02853 | 8.509432 | 3.18 | 0.003 | 9.802089 | 44.25498 |
| 32 | 38.925 | 9.944825 | 3.91 | 0.000 | 18.79275 | 59.05724 |
| 35 |  |  |  |  |  |  |
|  | 68.37147 | 7.95459 | 8.60 | 0.000 | 52.26824 | 84.47469 |
| _cons |  |  |  |  |  |  |

The regress command displayed the anova model as a regression table. We used the baselevels option to display the dropped level (or base) for each term.

If we want to treat age as a continuous variable, we must prepend c. to age in our anova.
. anova drate region c.age

|  | $\begin{array}{ll} \text { Number of obs } & = \\ \text { Root MSE } & = \end{array}$ | $\begin{array}{r} 50 \\ 7.21483 \end{array}$ | R-squared |  | $\begin{aligned} & 0.7203 \\ & 0.6954 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | 6032.0825 | 4 | 1508.0206 | 28.97 | 0.0000 |
| region | 1645.6623 | 3 | 548.55409 | 10.54 | 0.0000 |
| age | 1630.4666 | 1 | 1630.4666 | 31.32 | 0.0000 |
| Residual | 2342.4175 | 45 | 52.053721 |  |  |
| Total | 8374.5 | 49 | 170.90816 |  |  |

The age term now has 1 degree of freedom. The regression coefficients are

| Source | SS | df | MS | Number of obs $F(4,45)$ <br> Prob > F <br> R-squared <br> Adj R-squared <br> Root MSE |  | 50 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Model | 6032.08254 | 4 | 1508.02064 |  |  | 0.0000 |
| Residual | 2342.41746 | 45 | 52.0537213 |  |  | 0.7203 |
|  |  |  |  |  |  | 0.6954 |
| Total | 8374.5 | 49 | 170.908163 |  |  | 7.2148 |
| drate | Coefficient | Std. err. | t | $P>\|t\|$ | [95\% conf. interval] |  |
| region | 0 (base) |  |  |  |  |  |
| NE |  |  |  |  |  |  |  |  |
| N Cntrl | 1.792526 | 3.375925 | 0.53 | 0.598 | -5.006935 | 8.591988 |
| South | . 6979912 | 3.18154 | 0.22 | 0.827 | -5.70996 | 7.105942 |
| West | -13.37578 | 3.723447 | -3.59 | 0.001 | -20.87519 | -5.876377 |
| age | 3.922947 | . 7009425 | 5.60 | 0.000 | 2.511177 | 5.334718 |
| _cons | -28.60281 | 21.93931 | -1.30 | 0.199 | -72.79085 | 15.58524 |

Although we started analyzing these data to explain the regional differences in deathrate, let's focus on the effect of age for a moment. In our first model, each level of age had a unique deathrate associated with it. For instance, the predicted deathrate in a north central state with a median age of 28 was

$$
0.44+12.66+68.37 \approx 81.47
$$

whereas the predicted deathrate from our current model is

$$
1.79+3.92 \times 28-28.60 \approx 82.95
$$

Our previous model had an $R^{2}$ of 0.7927 , whereas our current model has an $R^{2}$ of 0.7203 . This "small" loss of predictive power accompanies a gain of 7 degrees of freedom, so we suspect that the continuous-age model is as good as the discrete-age model.

## - Technical note

There is enough information in the two ANOVA tables to attach a statistical significance to our suspicion that the loss of predictive power is offset by the savings in degrees of freedom. Because the continuous-age model is nested within the discrete-age model, we can perform a standard Chow test. For those of us who know such formulas off the top of our heads, the $F$ statistic is

$$
\frac{(2342.41746-1735.63471) / 7}{45.6745977}=1.90
$$

There is, however, a better way.
We can find out whether our continuous model is as good as our discrete model by putting age in the model twice: once as a continuous variable and once as a categorical variable. The categorical variable will then measure deviations around the straight line implied by the continuous variable, and the $F$ test for the significance of the categorical variable will test whether those deviations are jointly zero.
. anova drate region c.age age

|  | $\begin{array}{ll} \text { Number of obs } & = \\ \text { Root MSE } & = \end{array}$ | $\begin{array}{r} 50 \\ 6.7583 \end{array}$ | R -squared $=$ <br> Adj R-squared $=$ |  | $\begin{aligned} & 0.7927 \\ & 0.7328 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | 6638.8653 | 11 | 603.53321 | 13.21 | 0.0000 |
| region | 1320.0097 | 3 | 440.00324 | 9.63 | 0.0001 |
| age | 699.74137 | 1 | 699.74137 | 15.32 | 0.0004 |
| age | 606.78275 | 7 | 86.68325 | 1.90 | 0.0970 |
| Residual | 1735.6347 | 38 | 45.674598 |  |  |
| Total | 8374.5 | 49 | 170.90816 |  |  |

We find that the $F$ test for the significance of the (categorical) age variable is 1.90 , just as we calculated above. It is significant at the $9.7 \%$ level. If we hold to a $5 \%$ significance level, we cannot reject the null hypothesis that the effect of age is linear.
> Example 10: Interaction of continuous and categorical variables
In our census data, we still find significant differences across the regions after controlling for the median age of the population. We might now wonder whether the regional differences are differences in level-independent of age - or are instead differences in the regional effects of age. Just as we can interact categorical variables with other categorical variables, we can interact categorical variables with continuous variables.

| $\begin{array}{ll} \text { Number of obs } & = \\ \text { Root MSE } & = \end{array}$ |  | $\begin{array}{r} 50 \\ 7.24852 \end{array}$ | $\begin{array}{ll} \text { R-squared } & = \\ \text { Adj R-squared }= \end{array}$ |  | $\begin{aligned} & 0.7365 \\ & 0.6926 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | 6167.7737 | 78 | 881.11053 | 16.77 | 0.0000 |
| region | 188.7136 | 36 | 62.904534 | 1.20 | 0.3225 |
| age | 873.4256 | 1 | 873.4256 | 16.62 | 0.0002 |
| region\#age | 135.69116 | 3 | 45.230387 | 0.86 | 0.4689 |
| Residual | 2206.7263 | 425 | 52.541102 |  |  |
| Total | 8374.5 | $49 \quad 17$ | 170.90816 |  |  |

The region\#c. age term in our model measures the differences in slopes across the regions. We cannot reject the null hypothesis that there are no such differences. The region effect is now "insignificant". This status does not mean that there are no regional differences in deathrates because each test is a marginal or partial test. Here, with region\#c.age included in the model, region is being tested at the point where age is zero. Apart from this value not existing in the dataset, it is also a long way from the mean value of age, so the test of region at this point is meaningless (although it is valid if you acknowledge what is being tested).

To obtain a more sensible test of region, we can subtract the mean from the age variable and use this in the model.

```
. quietly summarize age
```

. generate mage $=$ age $-r$ (mean)
. anova drate region c.mage region\#c.mage

|  | Number of obs $=$ <br> Root MSE = | $\begin{array}{r} 50 \\ 7.24852 \end{array}$ | R -squared |  | $\begin{aligned} & 0.7365 \\ & 0.6926 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | 6167.7737 | 7 | 881.11053 | 16.77 | 0.0000 |
| region | 1166.1473 | 3 | 388.71578 | 7.40 | 0.0004 |
| mage | 873.4256 | 1 | 873.4256 | 16.62 | 0.0002 |
| region\#mage | 135.69116 | 3 | 45.230387 | 0.86 | 0.4689 |
| Residual | 2206.7263 | 42 | 52.541102 |  |  |
| Total | 8374.5 | 49 | 170.90816 |  |  |

region is significant when tested at the mean of the age variable.

Remember that we can specify interactions by typing varname\#varname. We have seen examples of interacting categorical variables with categorical variables and, in the examples above, a categorical variable (region) with a continuous variable (age or mage).

We can also interact continuous variables with continuous variables. To include an age ${ }^{2}$ term in our model, we could type c.age\#c.age. If we also wanted to interact the categorical variable region with the age ${ }^{2}$ term, we could type region\#c.age\#c.age (or even c.age\#region\#c.age).

## Nested designs

In addition to specifying interaction terms, nested terms can also be specified in an ANOVA. A vertical bar is used to indicate nesting: A|B is read as A nested within B. A|B|C is read as A nested within $B$, which is nested within $C . A \mid B \# C$ is read as $A$ is nested within the interaction of $B$ and $C$. $A \# B \mid C$ is read as the interaction of $A$ and $B$, which is nested within $C$.

Different error terms can be specified for different parts of the model. The forward slash is used to indicate that the next term in the model is the error term for what precedes it. For instance, anova y $\mathrm{A} / \mathrm{B} \mid \mathrm{A}$ indicates that the $F$ test for A is to be tested by using the mean square from $\mathrm{B} \mid \mathrm{A}$ in the denominator. Error terms (terms following the slash) are generally not tested unless they are themselves followed by a slash. Residual error is the default error term.

For example, consider $A / B / C$, where $A, B$, and $C$ may be arbitrarily complex terms. Then, anova will report $A$ tested by $B$ and $B$ tested by $C$. If we add one more slash on the end to form $A / B / C /$, then anova will also report $C$ tested by the residual error.
> Example 11: Simple nested ANOVA
We have collected data from a manufacturer that is evaluating which of five different brands of machinery to buy to perform a particular function in an assembly line. Twenty assembly-line employees were selected at random for training on these machines, with four employees assigned to learn a particular machine. The output from each employee (operator) on the brand of machine for which he trained was measured during four trial periods. In this example, the operator is nested
within machine. Because of sickness and employee resignations, the final data are not balanced. The following table gives the mean output and sample size for each machine and operator combination.

```
. use https://www.stata-press.com/data/r18/machine, clear
(Machine data)
```

. table machine operator, statistic(mean output) statistic(freq)
> totals(machine) nformat (\%8.2f mean)

|  | Operator nested in machine |  |  |  |  |
| :--- | ---: | ---: | ---: | ---: | ---: |
|  |  | 1 | 2 | 3 | 4 | Total

Assuming that operator is random (that is, we wish to infer to the larger population of possible operators) and machine is fixed (that is, only these five machines are of interest), the typical test for machine uses operator nested within machine as the error term. operator nested within machine can be tested by residual error. Our earlier warning concerning designs with either unplanned missing cells or unbalanced cell sizes, or both, also applies to interpreting the ANOVA results from this unbalanced nested example.

|  | $\begin{aligned} \text { Number of obs } & = \\ \text { Root MSE } & = \end{aligned}$ | $\begin{array}{r} 57 \\ 1.47089 \end{array}$ | R-squared = <br> Adj R -squared $=$ |  | $\begin{aligned} & 0.8661 \\ & 0.8077 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | 545.82229 | 17 | 32.107193 | 14.84 | 0.0000 |
| machine | 430.98079 | 4 | 107.7452 | 13.82 | 0.0001 |
| operator\|machine | 101.3538 | 13 | 7.7964465 |  |  |
| operator\|machine | 101.3538 | 13 | 7.7964465 | 3.60 | 0.0009 |
| Residual | 84.376658 | 39 | 2.1635041 |  |  |
| Total | 630.19895 | 56 | 11.253553 |  |  |

operator|machine is preceded by a slash, indicating that it is the error term for the terms before it (here machine). operator|machine is also followed by a slash that indicates it should be tested with residual error. The output lists the operator|machine term twice, once as the error term for machine and again as a term tested by residual error. A line is placed in the ANOVA table to separate the two. In general, a dividing line is placed in the output to separate the terms into groups that are tested with the same error term. The overall model is tested by residual error and is separated from the rest of the table by a blank line at the top of the table.

The results indicate that the machines are not all equal and that there are significant differences between operators.

## > Example 12: ANOVA with multiple levels of nesting

Your company builds and operates sewage treatment facilities. You want to compare two particulate solutions during the particulate reduction step of the sewage treatment process. For each solution, two area managers are randomly selected to implement and oversee the change to the new treatment process in two of their randomly chosen facilities. Two workers at each of these facilities are trained to operate the new process. A measure of particulate reduction is recorded at various times during the month at each facility for each worker. The data are described below.

| ```. use https://www.stata-press.com/data/r18/sewage (Sewage treatment) . describe``` |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: |
| Contains data from https://www.stata-press.com/data/r18/sewage.dta |  |  |  |  |
| Observation |  | 64 |  | Sewage treatment |
| Variable |  | 5 |  | 9 May 2022 12:43 |
| Variable name | Storage type | $\begin{aligned} & \text { Display } \\ & \text { format } \end{aligned}$ | Value <br> label | Variable label |
| particulate | byte | $\% 9.0 \mathrm{~g}$ |  | Particulate reduction |
| solution | byte | \%9.0g |  | 2 particulate solutions |
| manager | byte | \%9.0g |  | 2 managers per solution |
| facility | byte | \%9.0g |  | 2 facilities per manager |
| worker | byte | \%9.0g |  | 2 workers per facility |

You want to determine if the two particulate solutions provide significantly different particulate reduction. You would also like to know if manager, facility, and worker are significant effects. solution is a fixed factor, whereas manager, facility, and worker are random factors.

In the following anova command, we use abbreviations for the variable names, which can sometimes make long ANOVA model statements easier to read.
. anova particulate $s / m|s / f| m|s / w| f|m| s /, d r o p e m p t y c e l l s$

|  | $\begin{array}{ll} \text { Number of obs } & = \\ \text { Root MSE } & = \end{array}$ | $12.74$ | R-squared = <br> Adj R-squared = |  | $\begin{aligned} & 0.6338 \\ & 0.5194 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | 13493.609 | 15 | 899.57396 | 5.54 | 0.0000 |
| solution | 7203.7656 | 1 | 7203.7656 | 17.19 | 0.0536 |
| manager\|solution | 838.28125 | 2 | 419.14063 |  |  |
| manager\|solution | 838.28125 | 2 | 419.14063 | 0.55 | 0.6166 |
| facility\|manager| solution | 3064.9375 | 4 | 766.23438 |  |  |
| facility\|manager| |  |  |  |  |  |
| solution | 3064.9375 | 4 | 766.23438 | 2.57 | 0.1193 |
| worker\|facilityl |  |  |  |  |  |
| manager\|solution | 2386.625 | 8 | 298.32813 |  |  |
| worker\|facilityl |  |  |  |  |  |
| manager\|solution | 2386.625 | 8 | 298.32813 | 1.84 | 0.0931 |
| Residual | 7796.25 | 48 | 162.42188 |  |  |
| Total | 21289.859 | 63 | 337.93428 |  |  |

While solution is not declared significant at the $5 \%$ significance level, it is near enough to that threshold to warrant further investigation (see example 3 in $[\mathrm{R}]$ anova postestimation for a continuation of the analysis of these data).

## - Technical note

Why did we use the dropemptycells option with the previous anova? By default, Stata retains empty cells when building the design matrix and currently treats $\mid$ and \# the same in how it determines the possible number of cells. Retaining empty cells in an ANOVA with nested terms can cause your design matrix to become too large. In example 12, there are $1024=2 \times 4 \times 8 \times 16$ cells that are considered possible for the worker|facility|manager|solution term because the worker, facility, and manager variables are uniquely numbered. With the dropemptycells option, the worker|facility|manager|solution term requires just 16 columns in the design matrix (corresponding to the 16 unique workers).

Why did we not use the dropemptycells option in example 11, where operator is nested in machine? If you look at the table presented at the beginning of that example, you will see that operator is compactly instead of uniquely numbered (you need both operator number and machine number to determine the operator). Here the dropemptycells option would have only reduced our design matrix from 26 columns down to 24 columns (because there were only 3 operators instead of 4 for machines 3 and 5).

We suggest that you specify dropemptycells when there are nested terms in your ANOVA. You could also use the set emptycells drop command to accomplish the same thing; see [R] set.

## Mixed designs

An ANOVA can consist of both nested and crossed terms. A split-plot ANOVA design provides an example.

## > Example 13: Split-plot ANOVA

Two reading programs and three skill-enhancement techniques are under investigation. Ten classes of first-grade students were randomly assigned so that five classes were taught with one reading program and another five classes were taught with the other. The 30 students in each class were divided into six groups with 5 students each. Within each class, the six groups were divided randomly so that each of the three skill-enhancement techniques was taught to two of the groups within each class. At the end of the school year, a reading assessment test was administered to all the students. In this split-plot ANOVA, the whole-plot treatment is the two reading programs, and the split-plot treatment is the three skill-enhancement techniques.


Sorted by:
In this split-plot ANOVA, the error term for program is class nested within program. The error term for skill and the program by skill interaction is the class by skill interaction nested within program. Other terms are also involved in the model and can be seen below.

Our anova command is too long to fit on one line of this manual. Where we have chosen to break the command into multiple lines is arbitrary. If we were typing this command into Stata, we would just type along and let Stata automatically wrap across lines, as necessary.

|  | $\begin{array}{ll} \text { Number of obs } & = \\ \text { Root MSE } & = \end{array}$ | $\begin{array}{r} 300 \\ 14.6268 \end{array}$ | R-squared = <br> Adj R-squared $=$ |  | $\begin{aligned} & 0.3738 \\ & 0.2199 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | 30656.517 | 59 | 519.60198 | 2.43 | 0.0000 |
| program | 4493.07 | 1 | 4493.07 | 8.73 | 0.0183 |
| class\|program | 4116.6133 | 8 | 514.57667 |  |  |
| skill | 1122.6467 | 2 | 561.32333 | 1.54 | 0.2450 |
| program\#skill | 5694.62 | 2 | 2847.31 | 7.80 | 0.0043 |
| class\#skill\|program | 5841.4667 | 16 | 365.09167 |  |  |
| class\#skill\|program group|class\#skill| | 5841.4667 | 16 | 365.09167 | 1.17 | 0.3463 |
| program | 9388.1 | 30 | 312.93667 |  |  |
| group\|class\#skill| program | 9388.1 | 30 | 312.93667 | 1.46 | 0.0636 |
| Residual | 51346.4 | 240 | 213.94333 |  |  |
| Total | 82002.917 | 299 | 274.25725 |  |  |

The program\#skill term is significant, as is the program term. Let's look at the predictive margins for these two terms and at a marginsplot for the first term.

```
. margins, within(program skill)
Predictive margins Number of obs = 300
Expression: Linear prediction, predict()
Within: program skill
Empty cells: reweight
```

|  | Margin | lta-metho std. err. | t | $P>\|t\|$ | [95\% con | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| program\#skill |  |  |  |  |  |  |
| 11 | 68.16 | 2.068542 | 32.95 | 0.000 | 64.08518 | 72.23482 |
| 12 | 52.86 | 2.068542 | 25.55 | 0.000 | 48.78518 | 56.93482 |
| 13 | 61.54 | 2.068542 | 29.75 | 0.000 | 57.46518 | 65.61482 |
| 21 | 50.7 | 2.068542 | 24.51 | 0.000 | 46.62518 | 54.77482 |
| 22 | 56.54 | 2.068542 | 27.33 | 0.000 | 52.46518 | 60.61482 |
| 23 | 52.1 | 2.068542 | 25.19 | 0.000 | 48.02518 | 56.17482 |

```
. marginsplot, plot2opts(m(D)) plot3opts(m(T))
Variables that uniquely identify margins: program skill
```



| Predictive margins |  |  |  |  | Number of obs $=300$ |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Expression: Linear prediction, predict() <br> Within: program <br> Empty cells: reweight | ```Linear prediction, predict() program reweight``` |  |  |  |  |  |
|  | Margin | lta-method <br> std. err | t | $P>\|t\|$ | [95\% con | interval] |
| program |  |  |  |  |  |  |
| 1 | 60.85333 | 1.194273 | 50.95 | 0.000 | 58.50074 | 63.20593 |
| 2 | 53.11333 | 1.194273 | 44.47 | 0.000 | 50.76074 | 55.46593 |

Because our ANOVA involves nested terms, we used the within() option of margins; see [R] margins.
skill 2 produces a low score when combined with program 1 and a high score when combined with program 2, demonstrating the interaction between the reading program and the skill-enhancement technique. You might conclude that the first reading program and the first skill-enhancement technique perform best when combined. However, notice the overlapping confidence interval for the first reading program and the third skill-enhancement technique.

## - Technical note

There are several valid ways to write complicated anova terms. In the reading experiment example (example 13), we had a term grouplclass\#skill|program. This term can be read as group nested within both class and skill and further nested within program. You can also write this term as grouplclass\#skill\#program or grouplprogram\#class\#skill or grouplskill\#class|program, etc. All variations will produce the same result. Some people prefer having only one ' $\mid$ ' in a term and would use grouplclass\#skill\#program, which is read as group nested within class, skill, and program.

Gertrude Mary Cox (1900-1978) was born on a farm near Dayton, Iowa. Initially intending to become superintendent of an orphanage, she enrolled at Iowa State College. There she majored in mathematics and attained the college's first Master's degree in statistics. After working on her PhD in psychological statistics for two years at the University of California-Berkeley, she decided to go back to Iowa State to work with George W. Snedecor. There she pursued her interest in and taught a course in design of experiments. That work led to her collaboration with W. G. Cochran, which produced a classic text. In 1940, when Snedecor shared with her his list of men he was nominating to head the statistics department at North Carolina State College, she wanted to know why she had not been included. He added her name, she won the position, and she built an outstanding department at North Carolina State. Cox retired early so she could work at the Research Triangle Institute in North Carolina. She consulted widely, served as editor of Biometrics, and was elected to the National Academy of Sciences.

## Latin-square designs

You can use anova to analyze a Latin-square design. Consider the following example, published in Snedecor and Cochran (1989).

## > Example 14: Latin-square ANOVA

Data from a Latin-square design are as follows:

| Row | Column 1 | Column 2 | Column 3 | Column 4 | Column 5 |
| ---: | :---: | :---: | :---: | :---: | :---: |
| 1 | $257(\mathrm{~B})$ | $230(\mathrm{E})$ | $279(\mathrm{~A})$ | $287(\mathrm{C})$ | $202(\mathrm{D})$ |
| 2 | $245(\mathrm{D})$ | $283(\mathrm{~A})$ | $245(\mathrm{E})$ | $280(\mathrm{~B})$ | $260(\mathrm{C})$ |
| 3 | $182(\mathrm{E})$ | $252(\mathrm{~B})$ | $280(\mathrm{C})$ | $246(\mathrm{D})$ | $250(\mathrm{~A})$ |
| 4 | $203(\mathrm{~A})$ | $204(\mathrm{C})$ | $227(\mathrm{D})$ | $193(\mathrm{E})$ | $259(\mathrm{~B})$ |
| 5 | $231(\mathrm{C})$ | $271(\mathrm{D})$ | $266(\mathrm{~B})$ | $334(\mathrm{~A})$ | $338(\mathrm{E})$ |

In Stata, the data might appear as follows:

```
. use https://www.stata-press.com/data/r18/latinsq
```

. list

| row | $c 1$ | $c 2$ | $c 3$ | $c 4$ | $c 5$ |
| ---: | ---: | ---: | ---: | ---: | ---: |
| 1 | 257 | 230 | 279 | 287 | 202 |
| 2 | 245 | 283 | 245 | 280 | 260 |
| 3 | 182 | 252 | 280 | 246 | 250 |
| 4 | 203 | 204 | 227 | 193 | 259 |
| 5 | 231 | 271 | 266 | 334 | 338 |

Before anova can be used on these data, the data must be organized so that the outcome measurement is in one column. reshape is inadequate for this task because there is information about the treatments in the sequence of these observations. pkshape is designed to reshape this type of data; see [R] pkshape.
. pkshape row row c1-c5, order(beacd daebc ebcda acdeb cdbae)
. list

|  | sequence | outcome | treat | carry | period |
| :---: | :---: | :---: | :---: | :---: | :---: |
| 1. | beacd | 257 | b | 0 | 1 |
| 2. | daebc | 245 | d | 0 | 1 |
| 3. | ebcda | 182 | e | 0 | 1 |
| 4. | acdeb | 203 | a | 0 | 1 |
| 5. | cdbae | 231 | c | 0 | 1 |
| 6. | beacd | 230 | e | b | 2 |
| 7. | daebc | 283 | a | d | 2 |
| 8. | ebcda | 252 | b | e | 2 |
| 9. | acdeb | 204 | c | a | 2 |
| 10. | cdbae | 271 | d | c | 2 |
| 11. | beacd | 279 | a | e | 3 |
| 12. | daebc | 245 | e | a | 3 |
| 13. | ebcda | 280 | c | b | 3 |
| 14. | acdeb | 227 | d | c | 3 |
| 15. | cdbae | 266 | b | d | 3 |
| 16. | beacd | 287 | c | a | 4 |
| 17. | daebc | 280 | b | e | 4 |
| 18. | ebcda | 246 | d | c | 4 |
| 19. | acdeb | 193 | e | d | 4 |
| 20. | cdbae | 334 | a | b | 4 |
| 21. | beacd | 202 | d | c | 5 |
| 22. | daebc | 260 | c | b | 5 |
| 23. | ebcda | 250 | a | d | 5 |
| 24. | acdeb | 259 | b | e | 5 |
| 25. | cdbae | 338 | e | a | 5 |

anova outcome sequence period treat

|  | $\begin{array}{ll} \text { Number of obs } & = \\ \text { Root MSE } & = \end{array}$ | $\begin{array}{r} 25 \\ 32.4901 \end{array}$ | R-squared = <br> Adj R-squared = |  | $\begin{aligned} & 0.6536 \\ & 0.3073 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | 23904.08 | 12 | 1992.0067 | 1.89 | 0.1426 |
| sequence | 13601.36 | 4 | 3400.34 | 3.22 | 0.0516 |
| period | 6146.16 | 4 | 1536.54 | 1.46 | 0.2758 |
| treat | 4156.56 | 4 | 1039.14 | 0.98 | 0.4523 |
| Residual | 12667.28 | 12 | 1055.6067 |  |  |
| Total | 36571.36 | 24 | 1523.8067 |  |  |

These methods will work with any type of Latin-square design, including those with replicated measurements. For more information, see $[R] \mathbf{p k},[R]$ pkcross, and $[R]$ pkshape.

## Repeated-measures ANOVA

One approach for analyzing repeated-measures data is to use multivariate ANOVA (MANOVA); see [MV] manova. In this approach, the data are placed in wide form (see [D] reshape), and the repeated measures enter the MANOVA as dependent variables.

A second approach for analyzing repeated measures is to use anova. However, one of the underlying assumptions for the $F$ tests in ANOVA is independence of observations. In a repeated-measures design, this assumption is almost certainly violated. In a repeated-measures ANOVA, the subjects (or whatever the experimental units are called) are observed for each level of one or more of the other categorical variables in the model. These variables are called the repeated-measure variables. Observations from the same subject are likely to be correlated, though this is only a problem if the observations violate compound symmetry or the sphericity condition.

The approach used in repeated-measures ANOVA to correct for violation of compound symmetry or sphericity is to apply correction to the degrees of freedom of the $F$ test for terms in the model that involve repeated measures. This correction factor, $\epsilon$, lies between the reciprocal of the degrees of freedom for the repeated term and 1. Box (1954) provided the pioneering work in this area. Milliken and Johnson (2009) refer to the lower bound of this correction factor as Box's conservative correction factor. Winer, Brown, and Michels (1991) call it simply the conservative correction factor.

Geisser and Greenhouse (1958) provide an estimate for the correction factor called the GreenhouseGeisser $\epsilon$. This value is estimated from the data. Huynh and Feldt (1976) show that the GreenhouseGeisser $\epsilon$ tends to be conservatively biased. They provide a revised correction factor called the Huynh-Feldt $\epsilon$. When the Huynh-Feldt $\epsilon$ exceeds 1 , it is set to 1 . Thus, there is a natural ordering for these correction factors:

$$
\text { Box's conservative } \epsilon \leq \text { Greenhouse-Geisser } \epsilon \leq \text { Huynh-Feldt } \epsilon \leq 1
$$

A correction factor of 1 is the same as no correction.
anova with the repeated() option computes these correction factors and displays the revised test results in a table that follows the standard ANOVA table. In the resulting table, H-F stands for Huynh-Feldt, G-G stands for Greenhouse-Geisser, and Box stands for Box's conservative $\epsilon$.
$>$ Example 15: Repeated-measures ANOVA
This example is taken from table 4.3 of Winer, Brown, and Michels (1991). The reaction time for five subjects each tested with four drugs was recorded in the variable score. Here is a table of the data (see [P] tabdisp if you are unfamiliar with tabdisp):

```
. use https://www.stata-press.com/data/r18/t43, clear
(T4.3 -- Winer, Brown, Michels)
. tabdisp person drug, cellvar(score)
```

|  | Drug <br> Person |  |  |  |
| ---: | ---: | ---: | ---: | ---: |
|  | 1 | 2 | 3 | 4 |
| 1 | 30 | 28 | 16 | 34 |
| 2 | 14 | 18 | 10 | 22 |
| 3 | 24 | 20 | 18 | 30 |
| 4 | 38 | 34 | 20 | 44 |
| 5 | 26 | 28 | 14 | 30 |

drug is the repeated variable in this simple repeated-measures ANOVA example. The ANOVA is specified as follows:
. anova score person drug, repeated(drug)


Here the Huynh-Feldt $\epsilon$ is 1.0789 , which is larger than 1 . It is reset to 1 , which is the same as making no adjustment to the standard test computed in the main ANOVA table. The Greenhouse-Geisser $\epsilon$ is 0.6049 , and its associated $p$-value is computed from an $F$ ratio of 24.76 using $1.8147(=3 \epsilon)$ and $7.2588(=12 \epsilon)$ degrees of freedom. Box's conservative $\epsilon$ is set equal to the reciprocal of the degrees of freedom for the repeated term. Here it is $1 / 3$, so Box's conservative test is computed using 1 and 4 degrees of freedom for the observed $F$ ratio of 24.76.

Even for Box's conservative $\epsilon$, drug is significant with a $p$-value of 0.0076 . The following table gives the predictive marginal mean score (that is, response time) for each of the four drugs:

| . margins drug | Number of obs $=20$ |
| :--- | :--- |
| Predictive margins |  |
| Expression: Linear prediction, predict() |  |


|  | MarginDelta-method <br> std. err. | t | $\mathrm{P}>\mid \mathrm{tI}$ | [95\% conf. interval] |  |  |
| :---: | ---: | ---: | ---: | ---: | ---: | ---: |
| drug |  |  |  |  |  |  |
| 1 | 26.4 | 1.371131 | 19.25 | 0.000 | 23.41256 | 29.38744 |
| 2 | 25.6 | 1.371131 | 18.67 | 0.000 | 22.61256 | 28.58744 |
| 3 | 15.6 | 1.371131 | 11.38 | 0.000 | 12.61256 | 18.58744 |
| 4 | 32 | 1.371131 | 23.34 | 0.000 | 29.01256 | 34.98744 |

The ANOVA table for this example provides an $F$ test for person, but you should ignore it. An appropriate test for person would require replication (that is, multiple measurements for person and drug combinations). Also, without replication there is no test available for investigating the interaction between person and drug.

## > Example 16: Repeated-measures ANOVA with nesting

Table 7.7 of Winer, Brown, and Michels (1991) provides another repeated-measures ANOVA example. There are four dial shapes and two methods for calibrating dials. Subjects are nested within calibration method, and an accuracy score is obtained. The data are shown below.

```
. use https://www.stata-press.com/data/r18/t77
(T7.7 -- Winer, Brown, Michels)
. tabdisp shape subject calib, cell(score)
```

| 4 dial shapes | 2 methods for calibrating dials and Subject nested in calib |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  |  |  |  |
|  | 1 | 2 | 3 | 1 | 2 | 3 |
| 1 | 0 | 3 | 4 | 4 | 5 | 7 |
| 2 | 0 | 1 | 3 | 2 | 4 | 5 |
| 3 | 5 | 5 | 6 | 7 | 6 | 8 |
| 4 | 3 | 4 | 2 | 8 | 6 | 9 |

The calibration method and dial shapes are fixed factors, whereas subjects are random. The appropriate test for calibration method uses the nested subject term as the error term. Both the dial shape and the interaction between dial shape and calibration method are tested with the dial shape by subject interaction nested within calibration method. Here we drop this term from the anova command, and it becomes residual error. The dial shape is the repeated variable because each subject is tested with all four dial shapes. Here is the anova command that produces the desired results:


The repeated-measure $\epsilon$ corrections are applied to any terms that are tested in the main ANOVA table and have the repeated variable in the term. These $\epsilon$ corrections are given in a table below the main ANOVA table. Here the repeated-measures tests for shape and calib\#shape are presented.

Calibration method is significant, as is dial shape. The interaction between calibration method and dial shape is not significant. The repeated-measure $\epsilon$ corrections do not change these conclusions, but they do change the significance level for the tests on shape and calib\#shape. Here, though, unlike in the example 15, the Huynh-Feldt $\epsilon$ is less than 1 .

Here are the predictive marginal mean scores for calibration method and dial shapes. Because the interaction was not significant, we request only the calib and shape predictive margins.

| Predictive margins |  |  |  |  | Number of obs $=24$ |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Expression: Linear prediction, predict() <br> Within: calib <br> Empty cells: reweight |  |  |  |  |  |  |
|  | Margin | lta-method std. err | t | $P>\|t\|$ | [95\% conf | interval] |
| calib |  |  |  |  |  |  |
| 1 | 3 | . 3209506 | 9.35 | 0.000 | 2.300709 | 3.699291 |
| 2 | 5.916667 | . 3209506 | 18.43 | 0.000 | 5.217375 | 6.615958 |
| . margins, within(shape) |  |  |  |  |  |  |
| Predictive margins Number of obs $=24$ |  |  |  |  |  |  |
| Expression: Linear prediction, predict() Within: shape |  |  |  |  |  |  |
| Empty cells: reweight |  |  |  |  |  |  |


|  | Delta-method <br> std. err. |  |  |  | t | $\mathrm{P}>\|\mathrm{t}\|$ |
| :---: | ---: | ---: | ---: | ---: | ---: | ---: |
|  | Margin | [95\% conf. interval] |  |  |  |  |
| shape |  |  |  |  |  |  |
| 1 | 3.833333 | .4538926 | 8.45 | 0.000 | 2.844386 | 4.82228 |
| 2 | 2.5 | .4538926 | 5.51 | 0.000 | 1.511053 | 3.488947 |
| 3 | 6.166667 | .4538926 | 13.59 | 0.000 | 5.17772 | 7.155614 |
| 4 | 5.333333 | .4538926 | 11.75 | 0.000 | 4.344386 | 6.32228 |

## Technical note

The computation of the Greenhouse-Geisser and Huynh-Feldt epsilons in a repeated-measures ANOVA requires the number of levels and degrees of freedom for the between-subjects error term, as well as a value computed from a pooled covariance matrix. The observations are grouped based on all but the lowest-level variable in the between-subjects error term. The covariance over the repeated variables is computed for each resulting group, and then these covariance matrices are pooled. The dimension of the pooled covariance matrix is the number of levels of the repeated variable (or combination of levels for multiple repeated variables). In example 16, there are four levels of the repeated variable (shape), so the resulting covariance matrix is $4 \times 4$.

The anova command automatically attempts to determine the between-subjects error term and the lowest-level variable in the between-subjects error term to group the observations for computation of the pooled covariance matrix. anova issues an error message indicating that the bse() or bseunit () option is required when anova cannot determine them. You may override the default selections of anova by specifying the bse(), bseunit(), or grouping() option. The term specified in the bse() option must be a term in the ANOVA model.

The default selection for the between-subjects error term (the bse() option) is the interaction of the nonrepeated categorical variables in the ANOVA model. The first variable listed in the between-subjects error term is automatically selected as the lowest-level variable in the between-subjects error term but can be overridden with the bseunit (varname) option. varname is often a term, such as subject or subsample within subject, and is most often listed first in the term because of the nesting notation of ANOVA. This term makes sense in most repeated-measures ANOVA designs when the terms of the model are written in standard form. For instance, in example 16, there were three categorical variables (subject, calib, and shape), with shape being the repeated variable. Here anova looked for a term involving only subject and calib to determine the between-subjects error term. It found subject|calib as the term with six levels and 4 degrees of freedom. anova then picked subject as the default for the bseunit () option (the lowest variable in the between-subjects error term) because it was listed first in the term.

The grouping of observations proceeds, based on the different combinations of values of the variables in the between-subjects error term, excluding the lowest level variable (as found by default or as specified with the bseunit() option). You may specify the grouping() option to change the default grouping used in computing the pooled covariance matrix.

The between-subjects error term, number of levels, degrees of freedom, lowest variable in the term, and grouping information are presented after the main ANOVA table and before the rest of the repeated-measures output.
> Example 17: Repeated-measures ANOVA with two repeated variables
Data with two repeated variables are given in table 7.13 of Winer, Brown, and Michels (1991). The accuracy scores of subjects making adjustments to three dials during three different periods are recorded. Three subjects are exposed to a certain noise background level, whereas a different set of three subjects is exposed to a different noise background level. Here is a table of accuracy scores for the noise, subject, period, and dial variables:

```
. use https://www.stata-press.com/data/r18/t713
(T7.13 -- Winer, Brown, Michels)
. tabdisp subject dial period, by(noise) cell(score) stubwidth(11)
```

| Noise background and Subject nested in noise | 1 | $\begin{gathered} 10 \\ 1 \\ 2 \end{gathered}$ | nute <br> 3 |  | $\begin{gathered} \mathrm{ods} \\ 2 \\ 2 \end{gathered}$ | T 3 | $\frac{\mathrm{d}}{1}$ | $\begin{gathered} 3 \\ 2 \end{gathered}$ | 3 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 1 |  |  |  |  |  |  |  |  |  |
| 1 | 45 | 53 | 60 | 40 | 52 | 57 | 28 | 37 | 46 |
| 2 | 35 | 41 | 50 | 30 | 37 | 47 | 25 | 32 | 41 |
| 3 | 60 | 65 | 75 | 58 | 54 | 70 | 40 | 47 | 50 |
| 2 |  |  |  |  |  |  |  |  |  |
| 1 | 50 | 48 | 61 | 25 | 34 | 51 | 16 | 23 | 35 |
| 2 | 42 | 45 | 55 | 30 | 37 | 43 | 22 | 27 | 37 |
| 3 | 56 | 60 | 77 | 40 | 39 | 57 | 31 | 29 | 46 |

noise, period, and dial are fixed, whereas subject is random. Both period and dial are repeated variables. The ANOVA for this example is specified next.
. anova score noise / subject|noise period noise\#period /
> period\#subject|noise dial noise\#dial /
> dial\#subject|noise period\#dial noise\#period\#dial, repeated(period dial)

|  | $\begin{array}{ll} \text { Number of obs } & = \\ \text { Root MSE } & = \end{array}$ | $\begin{array}{r} 54 \\ 2.81859 \end{array}$ | R-squared = <br> Adj R-squared $=$ |  | $\begin{aligned} & 0.9872 \\ & 0.9576 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | 9797.7222 | 37 | 264.8033 | 33.33 | 0.0000 |
| noise | 468.16667 | 1 | 468.16667 | 0.75 | 0.4348 |
| subject\|noise | 2491.1111 | 4 | 622.77778 |  |  |
| period | 3722.3333 | 2 | 1861.1667 | 63.39 | 0.0000 |
| noise\#period | 333 | 2 | 166.5 | 5.67 | 0.0293 |
| period\#subject\|noise | 234.88889 | 8 | 29.361111 |  |  |
| dial | 2370.3333 | 2 | 1185.1667 | 89.82 | 0.0000 |
| noise\#dial | 50.333333 | 2 | 25.166667 | 1.91 | 0.2102 |
| dial\#subject\|noise | 105.55556 | 8 | 13.194444 |  |  |
| period\#dial | 10.666667 | 4 | 2.6666667 | 0.34 | 0.8499 |
| noise\#period\#dial | 11.333333 | 4 | 2.8333333 | 0.36 | 0.8357 |
| Residual | 127.11111 | 16 | 7.9444444 |  |  |
| Total | 9924.8333 | 53 | 187.26101 |  |  |

Between-subjects error term: subject|noise
Levels: 6 (4 df)
Lowest b.s.e. variable: subject
Covariance pooled over: noise (for repeated variables)
Repeated variable: period
Huynh-Feldt epsilon $=1.0668$
*Huynh-Feldt epsilon reset to 1.0000
Greenhouse-Geisser epsilon $=0.6476$
Box's conservative epsilon $=0.5000$

| Source | df | F | - Prob > F |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  | Regular | H-F | G-G | Box |
| period | 2 | 63.39 | 0.0000 | 0.0000 | 0.0003 | 0.0013 |
| noise\#period | 2 | 5.67 | 0.0293 | 0.0293 | 0.0569 | 0.0759 |
| period\#subject\|noise | 8 |  |  |  |  |  |

Repeated variable: dial
Huynh-Feldt epsilon $=2.0788$
*Huynh-Feldt epsilon reset to 1.0000
Greenhouse-Geisser epsilon $=0.9171$
Box's conservative epsilon $=0.5000$


Repeated variables: period\#dial
Huynh-Feldt epsilon $=1.3258$
*Huynh-Feldt epsilon reset to 1.0000
Greenhouse-Geisser epsilon $=0.5134$
Box's conservative epsilon $=0.2500$

|  |  |  | Prob $>\mathrm{F}$ |  |  |  |  |
| ---: | ---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | Source | df | F | Regular | $\mathrm{H}-\mathrm{F}$ | $\mathrm{G}-\mathrm{G}$ |  |
| period\#dial | 4 | 0.34 | 0.8499 | 0.8499 | 0.7295 | 0.5934 |  |
| noise\#period\#dial | 4 | 0.36 | 0.8357 | 0.8357 | 0.7156 | 0.5825 |  |
| Residual | 16 |  |  |  |  |  |  |

For each repeated variable and for each combination of interactions of repeated variables, there are different $\epsilon$ correction values. The anova command produces tables for each applicable combination.

The two most significant factors in this model appear to be dial and period. The noise by period interaction may also be significant, depending on the correction factor you use. Below is a table of predictive margins for the accuracy score for dial, period, and noise by period.

```
. margins, within(dial)
Predictive margins Number of obs = 54
Expression: Linear prediction, predict()
Within: dial
Empty cells: reweight
```

|  | Delta-method |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | Margin | std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| dial |  |  |  |  |  |  |
| 1 | 37.38889 | .6643478 | 56.28 | 0.000 | 35.98053 | 38.79724 |
| 2 | 42.22222 | .6643478 | 63.55 | 0.000 | 40.81387 | 43.63058 |
| 3 | 53.22222 | .6643478 | 80.11 | 0.000 | 51.81387 | 54.63058 |

```
. margins, within(period)
Predictive margins Number of obs = 54
Expression: Linear prediction, predict()
Within: period
Empty cells: reweight
```

|  | Delta-method |  |  |  |  |  |
| ---: | ---: | :---: | :---: | :---: | :---: | :---: |
|  | Margin | std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| period |  |  |  |  |  |  |
| 1 | 54.33333 | .6643478 | 81.78 | 0.000 | 52.92498 | 55.74169 |
| 2 | 44.5 | .6643478 | 66.98 | 0.000 | 43.09165 | 45.90835 |
| 3 | 34 | .6643478 | 51.18 | 0.000 | 32.59165 | 35.40835 |


| . margins, within(noise period) |
| :--- |
| Predictive margins |
| Expression: <br> Within: <br> Empty cells:Linear prediction, predict() <br> noise period <br> reweight |

Dial shape 3 produces the highest score, and scores decrease over the periods.

Example 17 had two repeated-measurement variables. Up to four repeated-measurement variables may be specified in the anova command.

## Video examples

Analysis of covariance in Stata
Two-way ANOVA in Stata

## Stored results

anova stores the following in e() :
Scalars
e(N)
e(mss)
e(df_m)
e(rss)
e(df_r)
e(r2)
e(r2_a)
e(F)
e(rmse)
e(11)
e(ll_0)
e(ss_\#)
e(df_\#)
e(ssdenom_\#)
e(dfdenom_\#)
e(F_\#)
e(N_bse)
e(df_bse)
e(box\#)
e(gg\#)
e(hf\#)
e(rank)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(varnames)
e(term_\#)
e(errorterm_\#)
e(sstype)
e(repvars)
e(repvar\#)
e(model)
e(wtype)
e(wexp)
e(properties)
e(estat_cmd)
$e$ (predict)
e(asbalanced)
e(asobserved)
Matrices

```
e(b)
e(V)
```

e(Srep)
Functions
e(sample)
number of observations
model sum of squares
model degrees of freedom
residual sum of squares
residual degrees of freedom
$R^{2}$
adjusted $R^{2}$
$F$ statistic
root mean squared error
log likelihood
log likelihood, constant-only model
sum of squares for term \#
numerator degrees of freedom for term \#
denominator sum of squares for term \# (when using nonresidual error)
denominator degrees of freedom for term \# (when using nonresidual error)
$F$ statistic for term \# (if computed)
number of levels of the between-subjects error term
degrees of freedom for the between-subjects error term
Box's conservative epsilon for a particular combination of repeated variables (repeated () only)
Greenhouse-Geisser epsilon for a particular combination of repeated variables (repeated() only)
Huynh-Feldt epsilon for a particular combination of repeated variables (repeated() only)
rank of $e(V)$
anova
command as typed
name of dependent variable
names of the right-hand-side variables
term \#
error term for term \# (when using nonresidual error)
type of sum of squares; sequential or partial
names of repeated variables (repeated() only)
names of repeated variables for a particular combination (repeated() only)
ols
weight type
weight expression
b V
program used to implement estat
program used to implement predict
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
variance-covariance matrix of the estimators
covariance matrix based on repeated measures (repeated() only)
marks estimation sample

In addition to the above, the following is stored in r() :

```
Matrices
    r(table)
```

matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in r() are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## References

Acock, A. C. 2023. A Gentle Introduction to Stata. Rev. 6th ed. College Station, TX: Stata Press.
Afifi, A. A., and S. P. Azen. 1979. Statistical Analysis: A Computer Oriented Approach. 2nd ed. New York: Academic Press.
Altman, D. G. 1991. Practical Statistics for Medical Research. London: Chapman and Hall/CRC.
Anderson, R. L. 1990. Gertrude Mary Cox 1900-1978. Biographical Memoirs, National Academy of Sciences 59: 116-132.

Baldwin, S. 2019. Psychological Statistics and Psychometrics Using Stata. College Station, TX: Stata Press.
Box, G. E. P. 1954. Some theorems on quadratic forms applied in the study of analysis of variance problems, I. Effect of inequality of variance in the one-way classification. Annals of Mathematical Statistics 25: 290-302. https://doi.org/10.1214/aoms/1177728786.

Box, J. F. 1978. R. A. Fisher: The Life of a Scientist. New York: Wiley.
Chatfield, M. D., and A. P. Mander. 2009. The Skillings-Mack test (Friedman test when there are missing data). Stata Journal 9: 299-305.

Cobb, G. W. 1998. Introduction to Design and Analysis of Experiments. New York: Springer.
Dietz, T., and L. Kalof. 2009. Introduction to Social Statistics: The Logic of Statistical Reasoning. Chichester, UK: Wiley.
Edwards, A. L. 1985. Multiple Regression and the Analysis of Variance and Covariance. 2nd ed. New York: Freeman.
Fisher, R. A. 1925. Statistical Methods for Research Workers. Edinburgh: Oliver and Boyd.
—_. 1935. The Design of Experiments. Edinburgh: Oliver and Boyd.
_-. 1990. Statistical Methods, Experimental Design, and Scientific Inference. Oxford: Oxford University Press.
Geisser, S., and S. W. Greenhouse. 1958. An extension of Box's results on the use of the $F$ distribution in multivariate analysis. Annals of Mathematical Statistics 29: 885-891. https://doi.org/10.1214/aoms/1177706545.
Goodman, M. S. 2018. Biostatistics for Clinical and Public Health Research. New York: Routledge.
Hall, N. S. 2010. Ronald Fisher and Gertrude Cox: Two statistical pioneers sometimes cooperate and sometimes collide. American Statistician 64: 212-220. https://doi.org/10.1198/tast.2010.10043.
Higgins, J. E., and G. G. Koch. 1977. Variable selection and generalized chi-square analysis of categorical data applied to a large cross-sectional occupational health survey. International Statistical Review 45: 51-62. https://doi.org/10.2307/1403003.

Huber, C. 2013. Measures of effect size in Stata 13. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2013/09/05/measures-of-effect-size-in-stata-13/.
Huynh, H. 1978. Some approximate tests for repeated measurement designs. Psychometrika 43: $161-175$. https://doi.org/10.1007/BF02293860.

Huynh, H., and L. S. Feldt. 1976. Estimation of the Box correction for degrees of freedom from sample data in randomized block and split-plot designs. Journal of Educational Statistics 1: 69-82. https://doi.org/10.2307/1164736.
Kennedy, W. J., Jr., and J. E. Gentle. 1980. Statistical Computing. New York: Dekker.
Lalanne, C., and M. Mesbah. 2016. Biostatistics and Computer-based Analysis of Health Data Using Stata. London: ISTE Press.

Marchenko, Y. V. 2006. Estimating variance components in Stata. Stata Journal 6: 1-21.
Mehmetoglu, M., and T. G. Jakobsen. 2022. Applied Statistics Using Stata: A Guide for the Social Sciences. 2nd ed. Thousand Oaks, CA: Sage.

Milliken, G. A., and D. E. Johnson. 2009. Analysis of Messy Data, Volume 1: Designed Experiments. 2nd ed. Boca Raton, FL: CRC Press.
Mitchell, M. N. 2015. Stata for the Behavioral Sciences. College Station, TX: Stata Press.
——. 2021. Interpreting and Visualizing Regression Models Using Stata. 2nd ed. College Station, TX: Stata Press.
Mooi, E., M. Sarstedt, and I. Mooi-Reci. 2018. Market Research: The Process, Data, and Methods Using Stata. Singapore: Springer.
Scheffé, H. 1959. The Analysis of Variance. New York: Wiley.

Snedecor, G. W., and W. G. Cochran. 1989. Statistical Methods. 8th ed. Ames, IA: Iowa State University Press.
van Belle, G., L. D. Fisher, P. J. Heagerty, and T. S. Lumley. 2004. Biostatistics: A Methodology for the Health Sciences. 2nd ed. New York: Wiley.
Winer, B. J., D. R. Brown, and K. M. Michels. 1991. Statistical Principles in Experimental Design. 3rd ed. New York: McGraw-Hill.

## Also see

[R] anova postestimation - Postestimation tools for anova
$[R]$ contrast - Contrasts and linear hypothesis tests after estimation
$[\mathrm{R}]$ icc - Intraclass correlation coefficients
[R] loneway - Large one-way ANOVA, random effects, and reliability
[ $R$ ] oneway - One-way analysis of variance
$[R]$ regress - Linear regression
[MV] manova - Multivariate analysis of variance and covariance
[PSS-2] power oneway - Power analysis for one-way analysis of variance
[PSS-2] power repeated - Power analysis for repeated-measures analysis of variance
[PSS-2] power twoway - Power analysis for two-way analysis of variance
Stata Structural Equation Modeling Reference Manual
[U] 20 Estimation and postestimation commands

## Title

anova postestimation - Postestimation tools for anova

| Postestimation commands | predict | margins | test |
| :--- | :--- | :--- | :--- |
| Remarks and examples | References | Also see |  |

## Postestimation commands

The following postestimation commands are of special interest after anova:

| Command | Description |
| :--- | :--- |
| dfbeta | DFBETA influence statistics |
| estat hettest | tests for heteroskedasticity |
| estat imtest | information matrix test |
| estat ovtest | Ramsey regression specification-error test for omitted variables |
| estat szroeter | Szroeter's rank test for heteroskedasticity |
| estat vif | variance inflation factors for the independent variables |
| estat esize | $\eta^{2}$ and $\omega^{2}$ effect sizes |
| rvfplot | residual-versus-fitted plot |
| avplot | added-variable plot |
| avplots | all added-variables plots in one image |
| cprplot | component-plus-residual plot |
| acprplot | augmented component-plus-residual plot |
| rvpplot | residual-versus-predictor plot |
| lvr2plot | leverage-versus-squared-residual plot |

The following standard postestimation commands are also available:

| Command | Description |
| :--- | :--- |
| contrast | contrasts and ANOVA-style joint tests of estimates <br> estat ic |
| Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian in- |  |
| formation criteria (AIC, CAIC, AICc, and BIC) |  |
| estat vce | variance-covariance matrix of the estimators (VCE) <br> estimates <br> etable |
| cataloging estimation results |  |
| table of estimation results |  |
| hausman | dynamic forecasts and simulations |
| lincom | Hausman's specification test |
| point estimates, standard errors, testing, and inference for linear combinations of |  |
| linktest | link test for model specification |
| lrtest | likelihood-ratio test |
| margins | marginal means, predictive margins, marginal effects, and average marginal effects <br> graph the results from margins (profile plots, interaction plots, etc.) |
| marginsplot | point estimates, standard errors, testing, and inference for nonlinear combinations <br> of coefficients |
| nlcom | predictions and their SEs, leverage statistics, distance statistics, etc. <br> point estimates, standard errors, testing, and inference for generalized predictions |
| predict | pairwise comparisons of estimates |
| predictnl | seemingly unrelated estimation |
| pwcompare | Wald tests of simple and composite linear hypotheses |
| suest | Wald tests of nonlinear hypotheses |

## predict

predict after anova follows the same syntax as predict after regress and can provide predictions, residuals, standardized residuals, Studentized residuals, the standard error of the residuals, the standard error of the prediction, the diagonal elements of the projection (hat) matrix, and Cook's $D$. See $[R]$ regress postestimation for details.

## margins

margins after anova follows the same syntax as margins after regress. See [R] regress postestimation for details.

## test

## Description for test

In addition to the standard syntax of test (see [R] test), test after anova has three additionally allowed syntaxes; see below. test performs Wald tests of expressions involving the coefficients of the underlying regression model. Simple and composite linear hypotheses are possible.

## Menu for test

Statistics $>$ Linear models and related $>$ ANOVA/MANOVA $>$ Test linear hypotheses after anova

## Syntax for test

| test, test(matname) $[\underline{\text { mtest }[(o p t)}]$ matvlc(matname) $]$ | syntax a |
| :--- | :--- |
| test, showorder | syntax b |
| $\underline{\text { test }[\text { term }[\text { term } . . .]][/ \operatorname{term}[\text { term } . . .]][\text {, symbolic }]}$ | syntax c |

syntax a test expression involving the coefficients of the underlying regression model; you provide information as a matrix
syntax b show underlying order of design matrix, which is useful when constructing matname argument of the test () option
syntax c test effects and show symbolic forms

## Options for test

test (matname) is required with syntax a of test. The rows of matname specify linear combinations of the underlying design matrix of the ANOVA that are to be jointly tested. The columns correspond to the underlying design matrix (including the constant if it has not been suppressed). The column and row names of matname are ignored.

A listing of the constraints imposed by the test () option is presented before the table containing the tests. You should examine this table to verify that you have applied the linear combinations you desired. Typing test, showorder allows you to examine the ordering of the columns for the design matrix from the ANOVA.
mtest $[(o p t)]$ specifies that tests are performed for each condition separately. opt specifies the method for adjusting $p$-values for multiple testing. Valid values for opt are

| bonferroni | Bonferroni's method |
| :--- | :--- |
| holm | Holm's method |
| šidak | Šidák's method |
| noadjust | no adjustment is to be made |

Specifying mtest with no argument is equivalent to mtest (noadjust).
matvlc(matname), a programmer's option, saves the variance-covariance matrix of the linear combinations involved in the suite of tests. For the test $\mathbf{L b}=\mathbf{c}$, what is returned in matname is $\mathbf{L} V \mathbf{L}^{\prime}$, where $V$ is the estimated variance-covariance matrix of $\mathbf{b}$.
showorder causes test to list the definition of each column in the design matrix. showorder is not allowed with any other option.
symbolic requests the symbolic form of the test rather than the test statistic. When this option is specified with no terms (test, symbolic), the symbolic form of the estimable functions is displayed.

## Remarks and examples

Remarks are presented under the following headings:

```
Testing effects
Obtaining symbolic forms
Testing coefficients and contrasts of margins
Video example
```

See examples $4,7,8,13,15,16$, and 17 in $[R]$ anova for examples that use the margins command.

## Testing effects

After fitting a model using anova, you can test for the significance of effects in the ANOVA table, as well as for effects that are not reported in the ANOVA table, by using the test or contrast command. You follow test or contrast by the list of effects that you wish to test. By default, these commands use the residual mean squared error in the denominator of the $F$ ratio. You can specify other error terms by using the slash notation, just as you would with anova. See [R] contrast for details on this command.

## > Example 1: Testing effects

Recall our byssinosis example (example 8) in [R] anova:

|  | $\begin{array}{ll} \text { Number of obs } & = \\ \text { Root MSE } & = \end{array}$ | $\begin{array}{r} 65 \\ .025902 \end{array}$ | $\begin{array}{ll} \text { R-squared } & = \\ \text { Adj R-squared }= \end{array}$ |  | $\begin{aligned} & 0.8300 \\ & 0.7948 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | . 17364654 | 11 | . 01578605 | 23.53 | 0.0000 |
| workplace | . 09762518 | 2. | . 04881259 | 72.76 | 0.0000 |
| smokes | . 01303081 | 1. | . 01303081 | 19.42 | 0.0001 |
| race | . 00109472 | 1 | . 00109472 | 1.63 | 0.2070 |
| workplace\#smokes | . 01969034 | 2 | . 00984517 | 14.67 | 0.0000 |
| workplace\#race | . 00135252 | 2 | . 00067626 | 1.01 | 0.3718 |
| smokes\#race | . 00166287 | 1 | . 00166287 | 2.48 | 0.1214 |
| workplace\#smokes\#race | . 00095084 | 2 | . 00047542 | 0.71 | 0.4969 |
| Residual | . 03555777 | 53 | . 0006709 |  |  |
| Total | . 2092043 | 64. | . 00326882 |  |  |

We can easily obtain a test on a particular term from the ANOVA table. Here are two examples:

```
. test smokes
```

| Source | Partial SS | df | MS | F | Prob>F |
| ---: | ---: | ---: | ---: | ---: | ---: |
| smokes | .01303081 | 1 | .01303081 | 19.42 | 0.0001 |
| Residual | .03555777 | 53 | .0006709 |  |  |

. test smokes\#race

| Source | Partial SS | df | MS | F | Prob>F |
| ---: | ---: | ---: | :---: | :---: | :---: |
| smokes\#race | .00166287 | 1 | .00166287 | 2.48 | 0.1214 |
| Residual | .03555777 | 53 | .0006709 |  |  |

Both of these tests use residual error by default and agree with the ANOVA table produced earlier.
We could have performed these same tests with contrast:

```
. contrast smokes
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | P>F |
| ---: | ---: | ---: | ---: |
| smokes | 1 | 19.42 | 0.0001 |
| Denominator | 53 |  |  |

. contrast smokes\#race
Contrasts of marginal linear predictions
Margins: asbalanced

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| :--- | :---: | :---: | ---: |
| smokes\#race | 1 | 2.48 | 0.1214 |
| Denominator | 53 |  |  |

## - Technical note

After anova, you can use the '/' syntax in test or contrast to perform tests with a variety of non $-\sigma^{2} \mathbf{I}$ error structures. However, in most unbalanced models, the mean squares are not independent and do not have equal expectations under the null hypothesis. Also, be warned that you assume responsibility for the validity of the test statistic.

## Example 2: Testing effects with different error terms

We return to the nested ANOVA example (example 11) in [R] anova, where five brands of machinery were compared in an assembly line. We can obtain appropriate tests for the nested terms using test, even if we had run the anova command without initially indicating the proper error terms.

```
. use https://www.stata-press.com/data/r18/machine
(Machine data)
. anova output machine / operator|machine /
\begin{tabular}{|c|c|c|c|c|c|}
\hline & \[
\begin{array}{ll}
\text { Number of obs } & = \\
\text { Root MSE } & =
\end{array}
\] & \[
\begin{array}{r}
57 \\
1.47089
\end{array}
\] & \multicolumn{2}{|l|}{\begin{tabular}{l}
R -squared \(=\) \\
Adj R-squared \(=\)
\end{tabular}} & \[
\begin{aligned}
& 0.8661 \\
& 0.8077
\end{aligned}
\] \\
\hline Source & Partial SS & df & MS & F & Prob>F \\
\hline Model & 545.82229 & 17 & 32.107193 & 14.84 & 0.0000 \\
\hline machine & 430.98079 & 4 & 107.7452 & 13.82 & 0.0001 \\
\hline operator|machine & 101.3538 & 13 & 7.7964465 & & \\
\hline operator|machine & 101.3538 & 13 & 7.7964465 & 3.60 & 0.0009 \\
\hline Residual & 84.376658 & 39 & 2.1635041 & & \\
\hline Total & 630.19895 & 56 & 11.253553 & & \\
\hline
\end{tabular}
```

In this ANOVA table, machine is tested with residual error. With this particular nested design, the appropriate error term for testing machine is operator nested within machine, which is easily obtained from test.

```
. test machine / operator|machine
\begin{tabular}{r|rrrrr} 
Source & Partial SS & df & MS & F & Prob>F \\
\hline machine & 430.98079 & 4 & 107.7452 & 13.82 & 0.0001 \\
operator machine & 101.3538 & 13 & 7.7964465 & &
\end{tabular}
```

This result from test matches what we obtained from our anova command.

## Example 3: Pooling terms when testing effects

The other nested ANOVA example (example 12) in [R] anova was based on the sewage data. The ANOVA table is presented here again. As before, we will use abbreviations of variable names in typing the commands.


In practice, it is often beneficial to pool nonsignificant nested terms to increase the power of tests on remaining terms. One rule of thumb is to allow the pooling of a term whose $p$-value is larger than 0.25 . In this sewage example, the $p$-value for the test of manager is 0.6166 . This value indicates that the manager effect is negligible and might be ignored. Currently, solution is tested by manager|solution, which has only 2 degrees of freedom. If we pool the manager and facility terms and use this pooled estimate as the error term for solution, we would have a term with 6 degrees of freedom.

Below are two tests: a test of solution with the pooled manager and facility terms and a test of this pooled term by worker.

| Source | Partial SS | df | MS | F | Prob>F |
| :---: | :---: | :---: | :---: | :---: | :---: |
| solution manager\|solution facility|manager| solution | 7203.7656 3903.2188 | 1 6 | 7203.7656 650.53646 | 11.07 | 0.0159 |
| . test $\mathrm{m}\|\mathrm{s} \mathrm{f}\| \mathrm{m}\|\mathrm{s} / \mathrm{w}\| \mathrm{f} \mid \mathrm{m}$ <br> Source | Partial SS | df | MS | F | Prob>F |
| manager\|solution facility|manager| solution worker|facility|manager| solution | 3903.2188 2386.625 | 6 8 | $\begin{aligned} & 650.53646 \\ & 298.32813 \end{aligned}$ | 2.18 | 0.1520 |

In the first test, we included two terms after the forward slash ( $\mathrm{m} \mid \mathrm{s}$ and $\mathrm{f}|\mathrm{m}| \mathrm{s}$ ). test after anova allows multiple terms both before and after the slash. The terms before the slash are combined and are then tested by the combined terms that follow the slash (or residual error if no slash is present).

The $p$-value for solution using the pooled term is 0.0159 . Originally, it was 0.0536 . The increase in the power of the test is due to the increase in degrees of freedom for the pooled error term.

We can get identical results if we drop manager from the anova model. (This dataset has unique numbers for each facility so that there is no confusion of facilities when manager is dropped.)


This output agrees with our earlier test results.

In the following example, two terms from the anova are jointly tested (pooled).

## > Example 4: Obtaining overall significance of a term using contrast

In example 10 of $[R]$ anova, we fit the model anova drate region c.mage region\#c.mage. Now, we use the contrast command to test for the overall significance of region.

```
. contrast region region#c.mage, overall
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | $d f$ | $F$ | $P>F$ |
| ---: | :---: | :---: | ---: |
| region | 3 | 7.40 | 0.0004 |
| region\#c.mage | 3 | 0.86 | 0.4689 |
| Overall | 6 | 5.65 | 0.0002 |
| Denominator | 42 |  |  |

The overall $F$ statistic associated with the region and region\#c.mage terms is 5.65 , and it is significant at the $0.02 \%$ level.

In the ANOVA output, the region term, by itself, had a sum of squares of 1166.15 , which, based on 3 degrees of freedom, yielded an $F$ statistic of 7.40 and a significance level of 0.0004 . This is the same test that is reported by contrast in the row labeled region. Likewise, the test from the ANOVA output for the region\#c.mage term is reproduced in the second row of the contrast output.

## Obtaining symbolic forms

test can produce the symbolic form of the estimable functions and symbolic forms for particular tests.

## > Example 5: Symbolic form of the estimable functions

After fitting an ANOVA model, we type test, symbolic to obtain the symbolic form of the estimable functions. For instance, returning to our blood pressure data introduced in example 4 of $[R]$ anova, let's begin by reestimating systolic on drug, disease, and drug\#disease:


To obtain the symbolic form of the estimable functions, type

```
. test, symbolic
drug
    -(r2+r3+r4-r0)
    r2
    3 r3
disease
    -(r6+r7-r0)
        r6
        r7
drug#disease
    1 1 -(r2+r3+r4+r6+r7-r12-r13-r15-r16-r18-r19-r0)
    1 2 r6 - (r12+r15+r18)
    1 3 r7 - (r13+r16+r19)
    2 1 r2 - (r12+r13)
    2 2 r12
    2 3 r13
    3 1 r3 - (r15+r16)
    3 2 r15
    3 3 r16
    4 1 r4 - (r18+r19)
    4 r r18
    4 3 r19
_cons r0
```

> Example 6: Symbolic form for a particular test
To obtain the symbolic form for a particular test, we type test term [term ...], symbolic. For instance, the symbolic form for the test of the main effect of drug is

| $\begin{aligned} & \text {. tes } \\ & \text { drug } \end{aligned}$ | 1 | $-(r 2+r 3+r 4)$ |
| :---: | :---: | :---: |
|  | 2 | r2 |
|  | 3 | r3 |
|  | 4 | r4 |
| disease |  |  |
|  | 1 | 0 |
|  | 2 | 0 |
|  | 3 | 0 |
| drug\#disease |  |  |
| 1 | 1 | $-1 / 3$ (r2+r3+r4) |
| 1 | 2 | $-1 / 3$ (r2+r3+r4) |
| 1 | 3 | $-1 / 3$ (r2+r3+r4) |
| 2 | 1 | 1/3 r2 |
| 2 | 2 | 1/3 r2 |
| 2 | 3 | 1/3 r2 |
| 3 | 1 | 1/3 r3 |
| 3 | 2 | 1/3 r3 |
| 3 | 3 | 1/3 r3 |
| 4 | 1 | 1/3 r4 |
| 4 | 2 | 1/3 r4 |
| 4 | 3 | 1/3 r4 |
| _cons |  | 0 |

If we omit the symbolic option, we instead see the result of the test:

```
. test drug
```

| Source | Partial SS | df | MS | F | Prob>F |
| ---: | ---: | ---: | :---: | :---: | :---: |
| drug | 2997.4719 | 3 | 999.15729 | 9.05 | 0.0001 |
| Residual | 5080.8167 | 46 | 110.45254 |  |  |

## Testing coefficients and contrasts of margins

The test command allows you to perform tests directly on the coefficients of the underlying regression model. For instance, the coefficient on the third drug and the second disease is referred to as 3.drug\#2.disease. This could also be written as i3.drug\#i2.disease, or _b[3.drug\#2.disease], or even _coef[i3.drug\#i2.disease]; see [U] 13.5 Accessing coefficients and standard errors.
> Example 7: Testing linear combinations of coefficients
Let's begin by testing whether the coefficient on the third drug is equal to the coefficient on the fourth drug in our blood pressure data. We have already fit the model anova systolic drug\#\#disease (equivalent to anova systolic drug disease drug\#disease), and you can see the results of that estimation in example 5. Even though we have performed many tasks since we fit the model, Stata still remembers, and we can perform tests at any time.

```
. test 3.drug = 4.drug
( 1) 3.drug - 4.drug = 0
    F( 1, 46) = 0.13
        Prob > F = 0.7234
```

We find that the two coefficients are not significantly different, at least at any significance level smaller than $73 \%$.

For more complex tests, the contrast command often provides a more concise way to specify the test we are interested in and prevents us from having to write the tests in terms of the regression coefficients. With contrast, we instead specify our tests in terms of differences in the marginal means for the levels of a particular factor. For example, if we want to compare the third and fourth drugs, we can test the difference in the mean impact on systolic blood pressure separately for each disease using the @ operator. We also use the reverse adjacent operator, ar., to compare the fourth level of drug with the previous level.

```
. contrast ar4.drug@disease
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| :---: | :---: | :---: | :---: |
| drug@disease |  |  |  |
| (4 vs 3) 1 | 1 | 0.13 | 0.7234 |
| (4 vs 3) 2 | 1 | 1.76 | 0.1917 |
| (4 vs 3) 3 | 1 | 0.65 | 0.4230 |
| Joint | 3 | 0.85 | 0.4761 |
| Denominator | 46 |  |  |


|  | Contrast | Std. err. | [95\% conf. interval] |  |
| ---: | ---: | :--- | :--- | :--- |
| drug@disease |  |  |  |  |
| $(4$ vs 3) 1 | -2.733333 | 7.675156 | -18.18262 | 12.71595 |
| $(4$ vs 3) 2 | 8.433333 | 6.363903 | -4.376539 | 21.24321 |
| $(4$ vs 3) 3 | 5.7 | 7.050081 | -8.491077 | 19.89108 |

None of the individual contrasts shows significant differences between the third drug and the fourth drug. Likewise, the overall $F$ statistic is 0.85 , which is hardly significant. We cannot reject the hypothesis that the third drug has the same effect as the fourth drug.

## - Technical note

Alternatively, we could have specified these tests based on the coefficients of the underlying regression model using the test command. We would have needed to perform tests on the coefficients for drug and for the coefficients on drug interacted with disease to test for differences in the means mentioned above. To do this, we start with our previous test command:

```
. test 3.drug = 4.drug
```

Notice that the $F$ statistic for this test is equivalent to the test labeled ( 4 vs 3 ) 1 in the contrast output. Let's now add the constraint that the coefficient on the third drug interacted with the third disease is equal to the coefficient on the fourth drug, again interacted with the third disease. We do that by typing the new constraint and adding the accumulate option:

```
. test 3.drug#3.disease = 4.drug#3.disease, accumulate
( 1) 3.drug - 4.drug = 0
( 2) 3.drug#3.disease - 4.drug#3.disease = 0
    F( 2, 46) = 0.39
        Prob > F = 0.6791
```

So far, our test includes the equality of the two drug coefficients, along with the equality of the two drug coefficients when interacted with the third disease. Now, we add two more equations, one for each of the remaining two diseases:

```
. test 3.drug#2.disease = 4.drug#2.disease, accumulate
( 1) 3.drug - 4.drug = 0
( 2) 3.drug#3.disease - 4.drug#3.disease = 0
( 3) 3.drug#2.disease - 4.drug#2.disease = 0
    F( 3, 46) = 0.85
    Prob > F = 0.4761
. test 3.drug#1.disease = 4.drug#1.disease, accumulate
( 1) 3.drug - 4.drug = 0
( 2) 3.drug#3.disease - 4.drug#3.disease = 0
( 3) 3.drug#2.disease - 4.drug#2.disease = 0
(4) 3o.drug#1b.disease - 4o.drug#1b.disease = 0
    Constraint 4 dropped
    F( 3, 46) = 0.85
        Prob > F = 0.4761
```

The overall $F$ statistic reproduces the one from the joint test in the contrast output.
You may notice that we also got the message "Constraint 4 dropped". For the technically inclined, this constraint was unnecessary, given the normalization of the model. If we specify all the constraints involved in our test or use contrast, we need not worry about the normalization because Stata handles this automatically.

The test () option of test provides another alternative for testing coefficients. Instead of spelling out each coefficient involved in the test, a matrix representing the test provides the needed information. test, showorder shows the order of the terms in the ANOVA corresponding to the order of the columns for the matrix argument of test ().

## > Example 8: Another way to test linear combinations of coefficients

We repeat the last test of example 7 above with the test () option. First, we view the definition and order of the columns underlying the ANOVA performed on the systolic data.

```
. test, showorder
Order of columns in the design matrix
    (drug==1)
    (drug==2)
    (drug==3)
    (drug==4)
    (disease==1)
    (disease==2)
    (disease==3)
    (drug==1)*(disease==1)
    (drug==1)*(disease==2)
    (drug==1)*(disease==3)
    (drug==2)*(disease==1)
    (drug==2)*(disease==2)
    (drug==2)*(disease==3)
    (drug==3)*(disease==1)
    (drug==3)*(disease==2)
    (drug==3)*(disease==3)
    (drug==4)*(disease==1)
    (drug==4)*(disease==2)
    (drug==4)*(disease==3)
    _cons
```

Columns 1-4 correspond to the four levels of drug. Columns 5-7 correspond to the three levels of disease. Columns $8-19$ correspond to the interaction of drug and disease. The last column corresponds to _cons, the constant in the model.

We construct the matrix $\operatorname{dr} 3 \mathrm{vs} 4$ with the same four constraints as the last test shown in example 7 and then use the test (dr3vs4) option to perform the test.

```
. matrix dr3vs4 = (0,0,1,-1, 0,0,0, 0,0,0,0,0,0,0,0,0, 0, 0, 0, 0 \
> 0,0,0,0,0,0,0, 0,0,0,0,0,0,0,0,1,0,0,-1,0\
    0,0,0, 0, 0,0,0, 0,0,0,0,0,0,0,1,0, 0,-1, 0, 0\
    0,0,0, 0, 0,0,0, 0,0,0,0,0,0,1,0,0,-1, 0, 0, 0)
. test, test(dr3vs4)
( 1) 3.drug - 4.drug = 0
( 2) 3.drug#3.disease - 4.drug#3.disease = 0
( 3) 3.drug#2.disease - 4.drug#2.disease = 0
( 4) 3o.drug#1b.disease - 4o.drug#1b.disease = 0
    Constraint 4 dropped
    F( 3, 46) = 0.85
        Prob > F = 0.4761
```

Here the effort involved with spelling out the coefficients is similar to that of constructing a matrix and using it in the test () option. When the test involving coefficients is more complicated, the test () option may be more convenient than specifying the coefficients directly in test. However, as previously demonstrated, contrast may provide an even simpler method for testing the same hypothesis.

After fitting an ANOVA model, various contrasts (1-degree-of-freedom tests comparing different levels of a categorical variable) are often of interest. contrast can perform each 1-degree-of-freedom test in addition to the combined test, even in cases in which the contrasts do not correspond to one of the contrast operators.

## > Example 9: Testing particular contrasts of interest

Rencher and Schaalje (2008) illustrate 1-degree-of-freedom contrasts for an ANOVA comparing the net weight of cans filled by five machines (labeled A-E). The data were originally obtained from Ostle and Mensing (1975). Rencher and Schaalje use a cell-means ANOVA model approach for this problem. We could do the same by using the noconstant option of anova; see [R] anova. Instead, we obtain the same results by using the standard overparameterized ANOVA approach (that is, we keep the constant in the model).

```
. use https://www.stata-press.com/data/r18/canfill
(Can fill data)
. list, sepby(machine)
```

|  | machine | weight |
| :---: | :---: | :---: |
| 1. | A | 11.95 |
| 2. | A | 12.00 |
| 3. | A | 12.25 |
| 4. | A | 12.10 |
| 5. | B | 12.18 |
| 6. | B | 12.11 |
| 7. | C | 12.16 |
| 8. | C | 12.15 |
| 9. | C | 12.08 |
| 10. | D | 12.25 |
| 11. | D | 12.30 |
| 12. | D | 12.10 |
| 13. | E | 12.10 |
| 14. | E | 12.04 |
| 15. | E | 12.02 |
| 16. | E | 12.02 |

. anova weight machine

|  | $\begin{aligned} \text { Number of obs } & = \\ \text { Root MSE } & = \end{aligned}$ | $\begin{array}{r} 16 \\ .087758 \end{array}$ | R -squared = <br> Adj R-squared $=$ |  | $\begin{aligned} & 0.4123 \\ & 0.1986 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | . 05942699 | 4 | . 01485675 | 1.93 | 0.1757 |
| machine | . 05942699 | 4 | . 01485675 | 1.93 | 0.1757 |
| Residual | . 0847167 | 11 | . 00770152 |  |  |
| Total | . 14414369 | 15 | . 00960958 |  |  |

The four 1-degree-of-freedom tests of interest among the five machines are $A$ and $D$ versus $B, C$, and E ; B and E versus C ; A versus D ; and B versus E . We can specify these tests as user-defined contrasts by placing the corresponding contrast coefficients into positions related to the five levels of machine as described in User-defined contrasts of [R] contrast.

```
. contrast {machine 3-2 -2 3-2}
```



```
> {machine 1 0 0 0 -1 0}
> {machine 0 1 0 0 -1}, noeffects
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | :---: | :---: | :---: |
| machine |  |  |  |
| $(1)$ | 1 | 0.75 | 0.4055 |
| $(2)$ | 1 | 0.31 | 0.5916 |
| $(3)$ | 1 | 4.47 | 0.0582 |
| (4) | 1 | 1.73 | 0.2150 |
| Joint | 4 | 1.93 | 0.1757 |
| Denominator | 11 |  |  |

contrast produces a 1-degree-of-freedom test for each of the specified contrasts as well as a joint test. We included the noeffects option so that the table displaying the values of the individual contrasts with their confidence intervals was suppressed.

The significance values above are not adjusted for multiple comparisons. We could have produced the Bonferroni-adjusted significance values by using the mcompare(bonferroni) option.

```
. contrast {machine 3-2 -2 3-2}
> {machine 0 1 -2 0 1}
> {machine 1 0 0 -1 0}
> {machine 0 1 0 0 -1}, mcompare(bonferroni) noeffects
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | Bonferroni |  |
| :---: | :---: | :---: | :---: | :---: |
|  |  |  | P>F | $\mathrm{P}>\mathrm{F}$ |
| machine |  |  |  |  |
| (1) | 1 | 0.75 | 0.4055 | 1.0000 |
| (2) | 1 | 0.31 | 0.5916 | 1.0000 |
| (3) | 1 | 4.47 | 0.0582 | 0.2329 |
| (4) | 1 | 1.73 | 0.2150 | 0.8601 |
| Joint | 4 | 1.93 | 0.1757 |  |
| Denominator | 11 |  |  |  |

Note: Bonferroni-adjusted p-values are reported for tests
on individual contrasts only.

## $>$ Example 10: Linear and quadratic contrasts

Here there are two factors, A and B, each with three levels. The levels are quantitative so that linear and quadratic contrasts are of interest.
. use https://www.stata-press.com/data/r18/twowaytrend
. anova Y A B A\#B

|  | $\begin{array}{ll} \text { Number of obs } & = \\ \text { Root MSE } & = \end{array}$ | $\begin{array}{r} 36 \\ 2.6736 \end{array}$ | R-squared = <br> Adj R-squared = |  | $\begin{aligned} & 0.9304 \\ & 0.9097 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | 2578.5556 | 8 | 322.31944 | 45.09 | 0.0000 |
| A | 2026.7222 | 2 | 1013.3611 | 141.77 | 0.0000 |
| B | 383.72222 | 2 | 191.86111 | 26.84 | 0.0000 |
| A\#B | 168.11111 | 4 | 42.027778 | 5.88 | 0.0015 |
| Residual | 193 | 27 | 7.1481481 |  |  |
| Total | 2771.5556 | 35 | 79.187302 |  |  |

We can use the p. contrast operator to obtain the 1-degree-of-freedom tests for the linear and quadratic effects of $A$ and $B$.

```
. contrast p.A p.B, noeffects
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| :---: | :---: | ---: | ---: |
| A |  |  |  |
| (linear) | 1 | 212.65 | 0.0000 |
| (quadratic) | 1 | 70.88 | 0.0000 |
| Joint | 2 | 141.77 | 0.0000 |
| B |  |  |  |
| (linear) | 1 | 26.17 | 0.0000 |
| (quadratic) | 1 | 27.51 | 0.0000 |
| Joint | 2 | 26.84 | 0.0000 |
| Denominator | 27 |  |  |

All the above tests appear to be significant. In addition to presenting the 1-degree-of-freedom tests, the combined tests for A and B are produced and agree with the original ANOVA results.

Now, we explore the interaction between A and B.

```
. contrast p.A#p1.B, noeffects
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | $d f$ | $F$ | $P>F$ |
| ---: | ---: | ---: | ---: |
| A\#B |  |  |  |
| (linear) (linear) | 1 | 17.71 | 0.0003 |
| (quadratic) (linear) | 1 | 0.07 | 0.7893 |
| Joint | 2 | 8.89 | 0.0011 |
| Denominator | 27 |  |  |

The 2-degrees-of-freedom test of the interaction of A with the linear components of B is significant at the 0.0011 level. But, when we examine the two 1-degree-of-freedom tests that compose this result,
the significance is due to the linear A by linear B contrast (significance level of 0.0003). A significance value of 0.7893 for the quadratic A by linear B indicates that this factor is not significant for these data.

```
. contrast p.A#p2.B, noeffects
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | :---: | :---: | :---: |
| (linear) (quadratic) |  |  |  |
| (quadratic) (quadratic) | 1 | 2.80 | 0.1058 |
|  | Joint | 1 | 2.94 |
|  | 2 | 2.87 | 0.0979 |
|  | Denominator | 27 |  |

The test of A with the quadratic components of B does not fall below the 0.05 significance level.

## Video example

Introduction to contrasts in Stata: One-way ANOVA

## References

Mitchell, M. N. 2021. Interpreting and Visualizing Regression Models Using Stata. 2nd ed. College Station, TX: Stata Press.

Ostle, B., and R. W. Mensing. 1975. Statistics in Research. 3rd ed. Ames, IA: Iowa State University Press.
Rencher, A. C., and G. B. Schaalje. 2008. Linear Models in Statistics. 2nd ed. New York: Wiley.

## Also see

[ R ] anova - Analysis of variance and covariance
[R] regress postestimation - Postestimation tools for regress
[R] regress postestimation diagnostic plots - Postestimation plots for regress
[U] 20 Estimation and postestimation commands

## Title

areg - Linear regression with many indicator variables ${ }^{+}$
${ }^{+}$This command includes features that are part of StataNow.

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| Acknowledgment | References | Also see |  |

## Description

areg fits linear regression absorbing categorical factors. areg is designed for datasets with categorical variables that have many groups, but the number of groups for each variable does not increase with the sample size. See the xtreg, fe command in [XT] xtreg for an estimator that handles the case in which one categorical variable (often a panel identifier) has a number of groups that increases with the sample size; additional categorical variables with fixed group sizes may also be included.

## Quick start

Linear regression of y on x , absorbing indicator variables for the levels of cvar
areg $\mathrm{y} x, \operatorname{absorb(cvar)}$
Same as above, but add categorical variable a
areg y x i.a, absorb(cvar)
Same as above, but with cluster-robust standard errors
areg $y \mathrm{x}$ i.a, absorb(cvar) vce(cluster cvar2)
Linear regression of y on x , absorbing indicator variables for the levels of cvar1 and cvar2 (StataNow)
$\operatorname{areg} \mathrm{y} x, \operatorname{absorb}(c v a r 1 \mathrm{cvar} 2)$

## Menu

Statistics $>$ Linear models and related $>$ Other $>$ Linear regression absorbing cat. variables

## Syntax

areg depvar [indepvars][if][in][weight], absorb(...) [options]
options
Description
Model

* absorb (varname) categorical variable to be absorbed
${ }^{+*}$ absorb (varlist $[$, method $]$ ) specify categorical variables to be absorbed
${ }^{+}$noabsorbtest suppress the $F$ test of absorbed indicators
SE/Robust
vce (vcetype) vcetype may be ols, robust, cluster clustvarlist, bootstrap, jackknife, or hc2 $[$ clustvar $]$

Reporting
level (\#)
clustertable
display_options
set confidence level; default is level (95)
display table of multiway cluster combinations
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

Optimization
$+\dagger$ tolerance (\#)
$+\dagger_{\text {iterate }}(\#)$
$+\dagger$ nolog
coeflegend
convergence tolerance for maximum absolute difference; default is tolerance ( $1 \mathrm{e}-8$ )
maximum number of iterations for alternating projection method (APM); default is iterate(50)
suppress the APM iteration $\log$
display legend instead of statistics
${ }^{+}$These features are part of StataNow.
*absorb() is required.
$\dagger_{\text {Ignored if only one absorbed variable is specified. }}$
indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
depvar and indepvars may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bootstrap, by, collect, fp, jackknife, mi estimate, rolling, and statsby are allowed; see [U] 11.1.10 Prefix commands.
vce (bootstrap) and vce (jackknife) are not allowed with the mi estimate prefix; see [MI] mi estimate.
Weights are not allowed with the bootstrap prefix; see $[R]$ bootstrap.
aweights are not allowed with the jackknife prefix; see $[R]$ jackknife.
aweights, fweights, and pweights are allowed; see [U] 11.1.6 weight.
coeflegend does not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options


absorb (varlist $[$, method $]$ ) is part of StataNow. It specifies the categorical variables to be absorbed. The results are adjusted as if indicator variables for each level of each variable in varlist were included in the regression. absorb() is required.
When more than one categorical variable is absorbed, an APM iterative algorithm is used to project the depvar and indepvars to absorb these variables. method specifies the APM and is one of halperin or cimmino.
halperin, the default, uses the product of the projection matrices.
cimmino uses the mean of the projection matrices.
The two methods typically perform similarly. See Stammann (2018) for details.
method is ignored if only one absorbed variable is specified.
noabsorbtest is part of StataNow. It prevents computation of the $F$ test that all coefficients on indicators for absorbed variables are jointly zero. Computation of this test requires an iterative search. If the $F$ test is of no interest, you may specify noabsorbtest and save computational time.

SE/Robust
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (ols), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvarlist), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
vce(ols), the default, uses the standard variance estimator for ordinary least-squares regression.
vce(cluster clustvarlist) specifies that standard errors allow for intragroup correlation within groups defined by one or more variables in clustvarlist, relaxing the usual requirement that the observations be independent. For example, vce(cluster clustvar1) produces clusterrobust standard errors that allow for observations that are independent across groups defined by clustvar1 but not necessarily independent within groups. You could also type vce (cluster clustvar1 clustvar2 ... clusterp) to account for correlation within groups formed by $p$ variables (multiway clustering).
areg also allows the following:
vce (hc2 [clustvar] [, dfadjust]) specifies an alternative bias correction for the robust variance calculation. In the unclustered case, vce(robust) uses $\widehat{\sigma}_{j}^{2}=\{n /(n-k)\} u_{j}^{2}$ as an estimate of the variance of the $j$ th observation, where $n$ is the number of observations, $k$ is the number of regressors, $u_{j}$ is the calculated residual, and $n /(n-k)$ is included to improve the overall estimate's small-sample properties.
vce (hc2) instead uses $u_{j}^{2} /\left(1-h_{j j}\right)$ as the observation's variance estimate, where $h_{j j}$ is the diagonal element of the hat (projection) matrix. This estimate is unbiased if the model really is homoskedastic. vce (hc2) tends to produce slightly more conservative confidence intervals. vce (hc2 clustvar) produces estimates that allow for intragroup correlation within groups defined by clustvar. dfadjust computes the Bell and McCaffrey (2002) adjusted degrees of freedom based on clustvar. Note that dfadjust does not affect multiple-imputation results when the command is used with mi estimate. See Methods and formulas in [R] regress for a description of the computation when clustvar is specified.
vce(hc2 clustvar) is not allowed when multiple variables are absorbed.
level (\#); see [R] Estimation options.
clustertable displays a table reporting cluster combinations and the number of clusters per combination. This option is available only when vce (cluster clustvarlist) is specified with more than one variable in clustvarlist to compute multiway cluster-robust standard errors.
display_options: noci, nopvalues, dfci, dfpvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat(\% firt), pformat (\% fint), sformat (\%fmt), and nolstretch; see [R] Estimation options.
dfci specifies that parameter degrees of freedom and confidence intervals be reported in the coefficient table.
dfpvalues specifies that parameter degrees of freedom and $p$-values be reported in the coefficient table.

Optimization
tolerance (\#) is part of StataNow. It specifies the limit for the maximum absolute difference between iterations for the projected depvar and indepvars. The default is tolerance(1e-8).
iterate (\#) is part of StataNow. It specifies the maximum number of iterations for the APM. The default is iterate (50).
nolog is part of StataNow. It specifies that no APM iterative log be displayed.
The following option is available with areg but is not shown in the dialog box: coeflegend; see [R] Estimation options.

## Remarks and examples

Suppose that we have a linear regression model that includes among predictor variables indicators for the levels of a high-dimensional categorical variable $C$. For instance, the levels of $C$ could represent counties, neighborhoods, or streets. We could write the model as

$$
\mathbf{y}=\mathbf{X} \boldsymbol{\beta}+\mathbf{d}_{1} \gamma_{1}+\mathbf{d}_{2} \gamma_{2}+\cdots+\mathbf{d}_{m} \gamma_{m}+\boldsymbol{\epsilon}
$$

Alternatively, if there are $N$ observations and $m$ categories of variable $C$, we could write this as

$$
\mathbf{y}=\mathbf{X} \boldsymbol{\beta}+\mathbf{D} \gamma+\epsilon
$$

where $\mathbf{D}$ is an $N \times m$ indicator matrix for the categories of $C$ and

$$
\mathbf{D}=\left(\begin{array}{llll}
\mathbf{d}_{1} & \mathbf{d}_{2} & \ldots & \mathbf{d}_{m}
\end{array}\right) \gamma=\left(\begin{array}{c}
\gamma_{1} \\
\gamma_{2} \\
\vdots \\
\gamma_{m}
\end{array}\right)
$$

Each $\mathbf{d}_{a}$ is a column vector, the indicator variable for category $a$ of variable $C$.
One option would be to fit the model with regress,
but this may be computationally expensive when the dimension, $m$, is very large. Moreover, you may not be interested in the estimates of $\gamma$, but you want the remaining results to be adjusted for the inclusion of $C$ in the model.
areg provides a faster way of obtaining estimates of $\boldsymbol{\beta}$ —but not the $\gamma_{i}$ 's—in these cases. The effects of the $C$ variable are said to be absorbed.

## Example 1

So that we can compare the results produced by areg with Stata's other regression commands, we will fit a model in which $m$ is small. areg's real use, however, is when $m$ is large.

In our automobile data, we have a variable called rep78 that is coded $1,2,3,4$, and 5 , where 1 means poor and 5 means excellent. Let's assume that we wish to fit a regression of mpg on weight, gear_ratio, and rep78 (parameterized as a set of indicators).

```
. use https://www.stata-press.com/data/r18/auto2
(1978 automobile data)
. regress mpg weight gear_ratio b5.rep78
```

| Source | SS | df | MS |
| ---: | :---: | ---: | :---: |
| Model | 1575.97621 | 6 | 262.662702 |
| Residual | 764.226686 | 62 | 12.3262369 |
| Total | 2340.2029 | 68 | 34.4147485 |


| Number of obs | $=$ | 69 |
| :--- | :--- | ---: |
| $\mathrm{~F}(6,62)$ | $=$ | 21.31 |
| Prob $>$ F | $=$ | 0.0000 |
| R-squared | $=$ | 0.6734 |
| Adj R-squared | $=$ | 0.6418 |
| Root MSE | $=$ | 3.5109 |


| mpg | Coefficient | Std. err. | t | $\mathrm{P}>\mid \mathrm{tl}$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| weight | -.0051031 | .0009206 | -5.54 | 0.000 | -.0069433 | -.003263 |
| gear_ratio | .901478 | 1.565552 | 0.58 | 0.567 | -2.228015 | 4.030971 |
| rep78 |  |  |  |  |  |  |
| Poor | -2.036937 | 2.740728 | -0.74 | 0.460 | -7.515574 | 3.4417 |
| Fair | -2.419822 | 1.764338 | -1.37 | 0.175 | -5.946682 | 1.107039 |
| Average | -2.557432 | 1.370912 | -1.87 | 0.067 | -5.297846 | .1829814 |
| Good | -2.788389 | 1.395259 | -2.00 | 0.050 | -5.577473 | .0006939 |
| _cons | 36.23782 | 7.01057 | 5.17 | 0.000 | 22.22389 | 50.25175 |

To fit the areg equivalent, we type
. areg mpg weight gear_ratio, absorb(rep78)
Linear regression, absorbing indicators Number of obs $=69$ Absorbed variable: rep78

| No. of categories | $=$ | 5 |
| :--- | :--- | ---: |
| F $(2,62)$ | 41.64 |  |
| Prob $>\mathrm{F}$ | $=0.0000$ |  |
| R-squared | $=0.6734$ |  |
| Adj R-squared | $=0.6418$ |  |
| Root MSE | $=3.5109$ |  |


| mpg | Coefficient | Std. err. | t | $\mathrm{P}>\mid \mathrm{tI}$ | [95\% conf. interval] |  |
| ---: | :---: | :---: | :---: | :---: | ---: | ---: |
| weight | -.0051031 | .0009206 | -5.54 | 0.000 | -.0069433 | -.003263 |
| gear_ratio | .901478 | 1.565552 | 0.58 | 0.567 | -2.228015 | 4.030971 |
| _cons | 34.05889 | 7.056383 | 4.83 | 0.000 | 19.95338 | 48.1644 |
| F test of absorbed indicators: $F(4,62)=1.117$ |  | Prob $>\mathrm{F}=0.356$ |  |  |  |  |

Both regress and areg display the same $R^{2}$ values, root mean squared error, and-for weight and gear_ratio-the same parameter estimates, standard errors, $t$ statistics, significance levels, and confidence intervals. areg, however, does not report the coefficients for rep78, and, in fact, they are not even calculated. This computational trick makes the problem manageable when $m$ is large. areg reports a test that the coefficients associated with rep78 are jointly zero. Here this test has a significance level of $35.6 \%$. This $F$ test for rep78 is the same that we would obtain after regress if we were to specify test 1. rep78 2.rep78 3.rep78 4.rep78; see $[R]$ test.

The model $F$ tests reported by regress and areg also differ. The regress command reports a test that all coefficients except that of the constant are equal to zero; thus, the indicators are included in this test. The areg output shows a test that all coefficients excluding the indicators and the constant are equal to zero. This is the same test that can be obtained after regress by typing test weight gear_ratio.

## - Technical note

areg is designed for datasets with many groups, but not a number that grows with the sample size. Consider two different samples from the US population. In the first sample, we have 10,000 individuals and we want to include an indicator for each of the 50 states, whereas in the second sample we have 3 observations on each of 10,000 individuals and we want to include an indicator for each individual. areg was designed for datasets similar to the first sample in which we have a fixed number of groups, the 50 states. In the second sample, the number of groups, which is the number of individuals, grows as we include more individuals in the sample. For an estimator designed to handle the case in which the number of groups grows with the sample size, see the xtreg, fe command in $[\mathrm{XT}]$ xtreg.

Although the point estimates produced by areg and xtreg, fe are the same, the estimated VCEs differ when vce (cluster clustvarlist) is specified because the commands make different assumptions about whether the number of groups increases with the sample size.

## - Technical note

The intercept reported by areg deserves some explanation because, given $m$ mutually exclusive and exhaustive indicators, it is arbitrary. areg identifies the model by choosing the intercept that makes the prediction calculated at the means of the independent variables equal to the mean of the dependent variable: $\overline{\mathbf{y}}=\overline{\mathbf{x}} \widehat{\boldsymbol{\beta}}$.

| Variable | Obs | Mean | Std. dev. | Min | Max |
| :---: | :---: | :---: | :---: | :---: | :---: |
| mpg | 69 | 21.28986 | 5.866408 | 12 | 41 |
| yhat | 69 | 21.28986 | 4.383224 | 11.58643 | 28.07367 |

We had to include if rep78 < . in our summarize command because we have missing values in our data. areg automatically dropped those missing values (as it should) in forming the estimates, but predict with the xb option will make predictions for cases with missing rep78 because it does not know that rep78 is really part of our model.

These predicted values do not include the absorbed effects (that is, the $\mathbf{d}_{i} \gamma_{i}$ ). For predicted values that include these effects, use the xbd option of predict (see [R] areg postestimation) or see [XT] xtreg.

## > Example 2

areg, vce (robust) is a Huberized version of areg; see [P] _robust. Just as areg is equivalent to using regress with indicators, areg, vce (robust) is equivalent to using regress, vce (robust) with indicators. You can use areg, vce (robust) when you expect heteroskedastic or nonnormal errors. areg, vce(robust), like ordinary regression, assumes that the observations are independent, unless the vce (cluster clustvarlist) or vce(hc2 clustvar) option is specified. If the vce (cluster clustvarlist) or vce(hc2 clustvar) option is specified, this independence assumption is relaxed and only the clusters identified by equal values of clustvarlist or clustvar are assumed to be independent.

Assume that we were to collect data by randomly sampling 10,000 doctors (from 100 hospitals) and then sampling 10 patients of each doctor, yielding a total dataset of 100,000 patients in a cluster sample. If in some regression we wished to include effects of the hospitals to which the doctors belonged, we would want to include an indicator variable for each hospital, adding 100 variables to our model. areg could fit this model by

```
. areg depvar patient_vars, absorb(hospital) vce(cluster doctor)
```


## Stored results

areg stores the following in e() :
Scalars
e(N)
e(k_absorb)
e(mss)
e(tss)
e(df_m)
e(rss)
e(df_r)
e(r2)
e(r2_a)
e(df_a)
e(rmse)
e(11)
e(11_0)
e(N_clust)
e(F)
e(F_absorb)
$e(p)$
e(p_absorb)
e(rank)
e(converged)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(absvar)
e(apm)
number of observations
total number of absorbed categories
model sum of squares
total sum of squares
model degrees of freedom
residual sum of squares
residual degrees of freedom
$R^{2}$
adjusted $R^{2}$
degrees of freedom for absorbed effect
root mean squared error
log likelihood
log likelihood, constant-only model
number of clusters
$F$ statistic
$F$ statistic for absorbed effect (when vce(robust) is not specified)
$p$-value for model $F$ test
$p$-value for $F$ test of absorbed effect
rank of e(V)
1 if APM converged, 0 otherwise

## areg

command as typed
name of dependent variable
names of absorbed variables
alternating projection method

```
e(wtype) weight type
e(wexp)
e(title)
e(clustvar)
e(cluster#)
e(vce)
e(vcetype)
e(datasignature)
e(datasignaturevars)
e(properties)
e(predict)
e(footnote)
e(marginsnotok)
e(asbalanced) factor variables fvset as asbalanced
e(asobserved)
Matrices
e(b)
e(V)
e(V_modelbased)
e(adj_df)
e(kcluster)
e(kabsorb)
weight type
weight expression
title in estimation output
names of cluster variables
cluster combination #
vcetype specified in vce()
title used to label Std. err.
the checksum
variables used in calculation of checksum
b V
program used to implement predict
program used to implement the footnote display
predictions disallowed by margins
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
variance-covariance matrix of the estimators
model-based variance
adjusted degrees of freedom when vce(hc2, dfadjust) is specified
cluster sizes, multiway clustering
number of levels for each absorbed variable
Functions
    e(sample) marks estimation sample
```

In addition to the above, the following is stored in r() :
$r(t a b l e) \quad$ matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

With one absorbed variable, areg begins by recalculating depvar and indepvars to have mean 0 within the groups specified by absorb(). The overall mean of each variable is then added back in. The adjusted depvar is then regressed on the adjusted indepvars with regress, yielding the coefficient estimates. This is a direct application of the Frisch-Waugh-Lovell theorem (see Hansen [2022, 82]). The degrees of freedom of the variance-covariance matrix of the coefficients is then adjusted to account for the absorbed variables - this calculation yields the same results (up to numerical roundoff error) as if the matrix had been calculated directly by the formulas given in $[R]$ regress.

With multiple absorbed variables in StataNow, an APM with a conjugate gradient acceleration technique is used to adjust the depvar and indepvars (Hernández-Ramos, Escalante, and Raydan 2011). Two projection methods are available: Halperin (the default) and Cimmino. You can specify which to use with the halperin or cimmino suboption within the absorb() option. We describe both methods below.

Suppose we have $h$ categorical variables, $C_{1}, \ldots, C_{h}$, that we would like to include as controls in our regression. For variable $C_{k}$, let $m_{k}$ be its number of levels and $\mathbf{d}_{k(a)}$ be its $N \times 1$ indicator vector for category $a$ of $C_{k}$. Let $\mathbf{D}_{k}$ denote the $N \times m_{k}$ matrix of indicators for variable $C_{k}$ :

$$
\mathbf{D}_{k}=\left(\begin{array}{llll}
\mathbf{d}_{k(1)} & \mathbf{d}_{k(2)} & \ldots & \mathbf{d}_{k(m)}
\end{array}\right)
$$

The orthonormal projection matrix for $C_{k}$ is therefore given by $\mathbf{P}_{k}=\mathbf{D}_{k}\left(\mathbf{D}_{k}^{\prime} \mathbf{D}_{k}\right)^{-1} \mathbf{D}_{k}^{\prime}$. Let $\mathbf{y}$ be the $N \times 1$ vector with the values of the dependent variable in the sample. Similarly, let $\mathbf{X}$ be the matrix with the values of the covariates. Thus, the product $\overline{\mathbf{y}}_{k}=\mathbf{P}_{k} \mathbf{y}$ is the projection of the dependent variable onto the column space of $\mathbf{D}_{k}$. That is, $\overline{\mathbf{y}}_{k}$ is the $N \times 1$ vector containing the (repeated) means of $y_{i}$ for each level of $C_{k}$, in the order that these levels appear in the sample. The same projection is applied to the columns of covariate matrix $\mathbf{X}$.

The Halperin algorithm first sets $\widetilde{\mathbf{y}}_{0}^{(1)}=\mathbf{y}$ and loops over the $h$ absorbed variables computing projection residuals $\widetilde{\mathbf{y}}_{k}^{(1)}=\left(\mathbf{I}-\mathbf{P}_{\mathbf{k}}\right) \widetilde{\mathbf{y}}_{k-1}^{(1)}$, for $k=1, \ldots, h$. Then, it repeats the loop with $\widetilde{\mathbf{y}}_{0}^{(j+1)}=$ $\widetilde{\mathbf{y}}_{h}^{(j)}$ until convergence at $j=\jmath$. Convergence is declared when $\left|\widetilde{\mathbf{y}}_{h}^{(\jmath)}-\widetilde{\mathbf{y}}_{0}^{(\jmath)}\right|$ is less than the specified tolerance, where the matrix norm $|\cdot|$ is defined as the largest entry in absolute value. The same computations are applied to the columns of covariate matrix $\mathbf{X}$. On convergence, the overall mean $\bar{y}$ is added to $\widetilde{\mathbf{y}}_{h}^{(\jmath)}$, and the vector of means $\overline{\mathbf{x}}$ is added to the columns of $\widetilde{\mathbf{X}}_{h}^{(\jmath)}$.

The Cimmino algorithm first sets $\widetilde{\mathbf{y}}^{(0)}=\mathbf{y}$ and then iteratively computes

$$
\widetilde{\mathbf{y}}^{(j+1)}=\left(\mathbf{I}-\frac{1}{h} \sum_{k=1}^{h} \mathbf{P}_{k}\right) \widetilde{\mathbf{y}}^{(j)}
$$

until convergence at $j+1=\jmath$. It then repeats the process for covariance matrix $\mathbf{X}$ and adds the overall means, $\overline{\mathbf{y}}$ and $\overline{\mathbf{x}}$, to $\widetilde{\mathbf{y}}^{(\jmath)}$ and to the columns of $\widetilde{\mathbf{X}}^{(\jmath)}$, respectively.

Efficient computation of the projection $\mathbf{P y}$ can be done in Mata without generating the $N \times N$ matrix $\mathbf{P}$ directly (see function panelsum(), for instance).
areg with vce(robust) or vce(cluster clustvarlist) with only one variable specified in clustvarlist works similarly, calling _robust after regress to produce the Huber/White/sandwich estimator of the variance or its clustered version. See [P] _robust, particularly Introduction and Methods and formulas. For $p$-way clustering, vce(cluster clustvarl [clustvar2 [...]]), _robust is called for each of the $2^{p}-1$ cluster combinations; details can be found in Multiway clustering of Methods and formulas in [R] regress. The model $F$ test uses the robust or cluster-robust variance estimates. There is, however, no simple computational means of obtaining a robust or cluster-robust test of the absorbed indicators. Thus, this test is not displayed when the vce (robust), vce (cluster clustvarlist), or vce(hc2 [clustvar]) option is specified.
areg with vce(hc2 [clustvar], [dfadjust]) specifies an alternative bias correction for the robust variance calculation. See Robust calculation for regress of Methods and formulas in [R] regress for a description of this VCE and the adjusted degrees of freedom.

The number of groups specified in absorb() are included in the degrees of freedom used in the finite-sample adjustment of the cluster-robust VCE estimator. This statement is valid only if the number of groups is small relative to the sample size. (Technically, the number of groups must remain fixed as the sample size grows.) For an estimator that allows the number of groups to grow with the sample size, see the xtreg, fe command in [XT] xtreg.

## Acknowledgment

We thank Sergio Correia at the Federal Reserve Board, Paulo Guimarães at the Bank of Portugal, and Fernando Rios-Avila at the Levy Economics Instute at Bard College for their work on controlling for high-dimensional categorical variables.

## References

Bell, R. M., and D. F. McCaffrey. 2002. Bias reduction in standard errors for linear regression with multi-stage samples. Survey Methodology 28: 169-181.

Blackwell, J. L., III. 2005. Estimation and testing of fixed-effect panel-data systems. Stata Journal 5: 202-207.
Correia, S., P. Guimarães, and T. Zylkin. 2020. Fast Poisson estimation with high-dimensional fixed effects. Stata Journal 20: 95-115.

Guimarães, P., and P. Portugal. 2010. A simple feasible procedure to fit models with high-dimensional fixed effects. Stata Journal 10: 628-649.
Hansen, B. E. 2022. Econometrics. Princeton, NJ: Princeton University Press.
Hernández-Ramos, L. M., R. Escalante, and M. Raydan. 2011. Unconstrained optimization techniques for the acceleration of alternating projection methods. Numerical Functional Analysis and Optimization 32: 1041-1066. https://doi.org/10.1080/01630563.2011.591954.
McCaffrey, D. F., K. Mihaly, J. R. Lockwood, and T. R. Sass. 2012. A review of Stata commands for fixed-effects estimation in normal linear models. Stata Journal 12: 406-432.

Rios-Avila, F. 2015. Feasible fitting of linear models with N fixed effects. Stata Journal 15: 881-898.
Stammann, A. 2018. Fast and feasible estimation of generalized linear models with high-dimensional $k$-way fixed effects. Unpublished manuscript. https://arxiv.org/pdf/1707.01815.pdf.

## Also see

[R] areg postestimation - Postestimation tools for areg
$[R]$ regress - Linear regression
[R] wildbootstrap - Wild cluster bootstrap inference
[MI] Estimation - Estimation commands for use with mi estimate
[XT] xtreg - Linear models for panel data ${ }^{+}$
[U] 20 Estimation and postestimation commands

## Title

areg postestimation - Postestimation tools for areg

| Postestimation commands | predict margins Remarks and examples |  |
| :--- | :--- | :--- |
| References | Also see |  |

## Postestimation commands

The following postestimation commands are available after areg:

| Command | Description |
| :--- | :--- |
| contrast | contrasts and ANOVA-style joint tests of estimates <br> estat ic |
| Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian in- <br> formation criteria (AIC, CAIC, AICc, and BIC) |  |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estimates | cataloging estimation results <br> etable |
| table of estimation results |  |
| forecast | dynamic forecasts and simulations |
| hausman | Hausman's specification test <br> point estimates, standard errors, testing, and inference for linear combinations of |
| linktest | coefficients |
| link test for model specification |  |

[^3]
## predict

## Description for predict

predict creates a new variable containing predictions such as fitted values, standard errors, residuals, and the equation-level score.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

```
predict [type] newvar [if] [in] [, statistic]
```

where $y_{j}=\mathbf{x}_{j} \mathbf{b}+d_{\text {absorbvars }}+e_{j}$ and statistic is

## statistic Description

Main
$\mathrm{xb} \quad \mathbf{x}_{j} \mathbf{b}$, fitted values; the default
stdp standard error of the prediction
$\underline{\mathrm{dr}}$ esiduals $\quad d_{\text {absorbvars }}+e_{j}=y_{j}-\mathbf{x}_{j} \mathbf{b}$
*xbd $\quad \mathbf{x}_{j} \mathbf{b}+d_{\text {absorbvars }}$
*d
*residuals
$d_{\text {absorbvars }}$
*score score; equivalent to residuals
Unstarred statistics are available both in and out of sample; type predict . . . if e (sample) ... if wanted only for the estimation sample. Starred statistics are calculated only for the estimation sample, even when if e(sample) is not specified.

## Options for predict

$\qquad$ Main
xb , the default, calculates the prediction of $\mathbf{x}_{j} \mathbf{b}$, the fitted values, by using the average effect of the absorbed variables. Also see xbd below.
stdp calculates the standard error of $\mathbf{x}_{j} \mathbf{b}$.
dresiduals calculates $y_{j}-\mathbf{x}_{j} \mathbf{b}$, which are the residuals plus the effects of the absorbed variables. xbd calculates $\mathbf{x}_{j} \mathbf{b}+d_{\text {absorbvars }}$, which are the fitted values including the effects of the absorbed variables.
d calculates $d_{\text {absorbvars }}$, the effects of the absorbed variables.
residuals calculates the residuals, that is, $y_{j}-\left(\mathbf{x}_{j} \mathbf{b}+d_{\text {absorbvars }}\right)$.
score is a synonym for residuals.

## margins

## Description for margins

margins estimates margins of response for fitted values.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [options]
```

statistic Description

| xb | $\mathbf{x}_{j} \mathbf{b}$, fitted values; the default |
| :--- | :--- |
| stdp | not allowed with margins |
| $\underline{\text { dresiduals }}$ | not allowed with margins |
| xbd | not allowed with margins |
| d | not allowed with margins |
| residuals | not allowed with margins |
| score | not allowed with margins |

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

## Remarks and examples

> Example 1
Continuing with example 1 of $[\mathrm{R}]$ areg, we refit the model with robust standard errors and then obtain linear predictions and standard errors for those linear predictions.

```
. use https://www.stata-press.com/data/r18/auto2
(1978 automobile data)
. areg mpg weight gear_ratio, absorb(rep78) vce(robust)
    (output omitted)
. predict xb_ar
(option xb assumed; fitted values)
. predict stdp_ar, stdp
```

We can obtain the same linear predictions by fitting the model with xtreg, fe, but we would first need to specify the panel structure by using xtset.

```
. xtset rep78
```

Panel variable: rep78 (unbalanced)
. xtreg mpg weight gear_ratio, fe vce(robust)
(output omitted)
. predict xb_xt
(option xb assumed; fitted values)
. predict stdp_xt, stdp
. summarize xb_ar xb_xt stdp*

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| xb_ar | 74 | 21.36805 | 4.286788 | 11.58643 | 28.07367 |
| xb_xt | 74 | 21.36805 | 4.286788 | 11.58643 | 28.07367 |
| stdp_ar | 74 | .7105649 | .1933936 | .4270821 | 1.245179 |
| stdp_xt | 74 | .8155919 | .4826332 | .0826999 | 1.709786 |

The predicted xb values above are the same for areg and xtreg, fe, but the standard errors for those linear predictions are different. The assumptions for these two estimators lead to different formulations for their standard errors. The robust variance estimates with areg are equivalent to the robust variance estimates using regress, including the panel dummies. The consistent robust variance estimates with xtreg are equivalent to those obtained by specifying vce (cluster panelvar) with that estimation command. For a theoretical discussion, see Wooldridge (2020), Stock and Watson (2008), and Arellano (2003); also see the technical note after example 3 of [XT] xtreg.

## Example 2

We would like to use linktest to check whether the dependent variable for our model is correctly specified:

```
. use https://www.stata-press.com/data/r18/auto2, clear
(1978 automobile data)
. areg mpg weight gear_ratio, absorb(rep78)
    (output omitted)
. linktest, absorb(rep78)
Linear regression, absorbing indicators Number of obs = 69
Absorbed variable: rep78 No. of categories = 5
F(2, 62) = 46.50
Prob > F = 0.0000
R-squared = 0.6939
Adj R-squared = 0.6643
Root MSE = 3.3990
```

| mpg | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | ---: | ---: |
| _hat | -.9305602 | .9537856 | -0.98 | 0.333 | -2.83715 | .9760302 |
| _hatsq | .0462785 | .0227219 | 2.04 | 0.046 | .0008582 | .0916989 |
| _cons | 19.24899 | 9.725618 | 1.98 | 0.052 | -.1922457 | 38.69022 |
| F test of absorbed indicators: $F(4,62)$ | $=1.278$ |  | Prob $>F=0.288$ |  |  |  |

The squared residuals are significant in the regression for mpg on the linear and squared residuals; therefore, the test indicates that our dependent variable does not seem to be well specified. Let's transform the dependent variable into energy consumption, gallons per mile, fit the alternative model, and check the link test again.

```
. generate gpm = 1/mpg
. areg gpm weight gear_ratio, absorb(rep78)
    (output omitted)
. linktest, absorb(rep78)
Linear regression, absorbing indicators Number of obs = 69
Absorbed variable: rep78
\begin{tabular}{llr} 
Number of obs & \(=\) & 69 \\
No. of categories & \(=\) & 5 \\
F \((2,62)\) & 72.60 \\
Prob \(>\mathrm{F}\) & \(=0.0000\) \\
R-squared & \(=0.7436\) \\
Adj R-squared & \(=0.7187\) \\
Root MSE & \(=0.0068\)
\end{tabular}
\begin{tabular}{r|rcccrr}
\hline gpm & Coefficient & Std. err. & t & P>|t| & [95\% conf. interval] \\
\hline _hat & .2842582 & .7109124 & 0.40 & 0.691 & -1.136835 & 1.705352 \\
_hatsq & 6.956965 & 6.862439 & 1.01 & 0.315 & -6.760855 & 20.67478 \\
_cons & .0175457 & .0178251 & 0.98 & 0.329 & -.0180862 & .0531777 \\
\hline F test of absorbed indicators: \(F(4,62)\) & \(=0.065\) & & Prob \(>\mathrm{F}=0.992\)
\end{tabular}
```

The link test supports the use of the transformed dependent variable.

## References

Arellano, M. 2003. Panel Data Econometrics. Oxford: Oxford University Press.
Stock, J. H., and M. W. Watson. 2008. Heteroskedasticity-robust standard errors for fixed effects panel data regression. Econometrica 76: 155-174. https://doi.org/10.1111/j.0012-9682.2008.00821.x.

Wooldridge, J. M. 2020. Introductory Econometrics: A Modern Approach. 7th ed. Boston: Cengage.

## Also see

[R] areg - Linear regression with many indicator variables ${ }^{+}$
[U] 20 Estimation and postestimation commands

## Title

betareg - Beta regression

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| Acknowledgments | References | Also see |  |

## Description

betareg estimates the parameters of a beta regression model. This model accommodates dependent variables that are greater than 0 and less than 1 , such as rates, proportions, and fractional data.

## Quick start

Beta regression of y on x 1 and x 2
betareg $y$ x1 x2
Add categorical variable a using factor-variable syntax
betareg y x1 x2 i.a
Add covariates for scale
betareg y x1 x2 i.a, scale(x1 z1)
Same as above, but use probit link for conditional mean and square-root link for conditional scale betareg y x1 x2 i.a, scale(x1 z1) link(probit) slink(root)

Beta regression of y on x 1 and x 2 with robust standard errors
betareg y x1 x 2 , vce(robust)

## Menu

Statistics $>$ Fractional outcomes $>$ Beta regression

## Syntax

betareg depvar indepvars $[$ if $][$ in $]$ [weight] [, options]
options
Model
noconstant
scale(varlist[, noconstant]) link (linkname)
slink(slinkname)
constraints (constraints)
SE/Robust
vce (vcetype)

Reporting
level(\#)
nocnsreport
display_options

Maximization
maximize_options
coeflegend

Description
suppress constant term
specify independent variables for scale
specify link function for the conditional mean; default is link(logit)
specify link function for the conditional scale; default is slink(log)
apply specified linear constraints
vcetype may be oim, robust, cluster clustvar, bootstrap, or jackknife
set confidence level; default is level (95)
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
control the maximization process; seldom used
display legend instead of statistics
linkname Description

| logit | logit |
| :--- | :--- |
| probit | probit |
| cloglog | complementary $\log -\log$ |
| $\log l o g$ | log-log |

slinkname
Description
log
root square root
identity
$\log$
identity
indepvars and varlist specified in scale() may contain factor variables; see [U] 11.4.3 Factor variables.
bayes, bootstrap, by, collect, fmm, fp, jackknife, nestreg, rolling, statsby, stepwise, and svy are allowed; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: betareg and [FMM] fmm: betareg.
Weights are not allowed with the bootstrap prefix; see [R] bootstrap.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
coeflegend does not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

$\xlongequal{\text { Model }}$
noconstant; see [R] Estimation options.
scale(varlist [, noconstant]) specifies the independent variables used to model the scale.
noconstant suppresses the constant term in the scale model. A constant term is included by default.
link(linkname) specifies the link function used for the conditional mean. linkname may be logit, probit, cloglog, or loglog. The default is link(logit).
slink(slinkname) specifies the link function used for the conditional scale. slinkname may be log, root, or identity. The default is slink(log).
constraints (constraints); see [R] Estimation options.

SE/Robust
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.

```
            Reporting
level(\#), nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.
```

Maximization
maximize_options: difficult, technique (algorithm_spec), $\underline{\text { iterate }}(\#),[$ no $] \log , \underline{\text { trace }}$, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from(init_specs); see [R] Maximize. These options are seldom used.

The following option is available with betareg but is not shown in the dialog box: coeflegend; see [R] Estimation options.

## Remarks and examples

Dependent variables such as rates, proportions, and fractional data are frequently greater than 0 and less than 1 . There are a variety of methods to model such variables, including beta regression and fractional logistic regression.

Beta regression is widely used because of its flexibility for modeling variables between 0 and 1 and because its predictions are confined to the same range. However, beta regression models are not appropriate for dependent variables with some observations exactly equal to 0 or 1 . See [R] fracreg for models when the dependent variable can equal 0 or 1 that also make predictions over the same range. The predictions from linear regression models are not constrained to the 0 to 1 interval; thus, they are not widely used for these variables.

These models have applications in a variety of disciplines, such as economics, the social sciences, and health science. For example, Castellani, Pattitoni, and Scorcu (2012) use beta regression to estimate Gini index values for the prices of art by famous and nonfamous artists. In political science, Paolino (2001) explores the advantages of beta regression and reviews its applicability for a variety of research topics such as the proportion of minority applicants deemed eligible for the Rural Housing Loans program and the proportion of a state's gay and lesbian population that is covered by antidiscrimination laws. In psychology, Smithson, Deady, and Gracik (2007) analyzed the relationship between how jurors judged the probability of a defendant's guilt and the verdict in a trial. Finally, beta regression has been used to model quality-adjusted life years in health-economics outcome studies (Hubben et al. [2008]; Basu and Manca [2012]).

Ferrari and Cribari-Neto (2004) and Smithson and Verkuilen (2006) derived the beta regression estimators implemented in betareg. Basu and Manca (2012) also discuss quasimaximum likelihood inference for these estimators. These estimators augment the inherent flexibility of the beta distribution with functional form choices, known as links.

Beta regression is a model of the mean of the dependent variable $y$ conditional on covariates $\mathbf{x}$, which we denote by $\mu_{\mathbf{x}}$. Because $y$ is in $(0,1)$, we must ensure that $\mu_{\mathbf{x}}$ is also in $(0,1)$. We do this by using the link function for the conditional mean, denoted $g(\cdot)$. This is necessary because linear combinations of the covariates are not otherwise restricted to $(0,1)$.

Algebraically,

$$
g\left(\mu_{\mathbf{x}}\right)=\mathbf{x} \boldsymbol{\beta}
$$

or, equivalently,

$$
\mu_{\mathbf{x}}=g^{-1}(\mathbf{x} \boldsymbol{\beta})
$$

where $g^{-1}(\cdot)$ is the inverse function of $g(\cdot)$. For example, the default logit link implies that

$$
\ln \left\{\mu_{\mathbf{x}} /\left(1-\mu_{\mathbf{x}}\right)\right\}=\mathbf{x} \boldsymbol{\beta}
$$

and that

$$
\mu_{\mathbf{x}}=\exp (\mathbf{x} \boldsymbol{\beta}) /\{1+\exp (\mathbf{x} \boldsymbol{\beta})\}
$$

Using a link function to keep the conditional-mean model inside an interval is common in the statistical literature; see $[R]$ glm for additional applications of link functions.

The conditional variance of the beta distribution is

$$
\operatorname{Var}(y \mid \mathbf{x})=\left\{\mu_{\mathbf{x}}\left(1-\mu_{\mathbf{x}}\right)\right\} /(1+\psi)
$$

The parameter $\psi$ is known as the scale factor because it rescales the conditional variance. We use the scale link to ensure that $\psi>0$.

## $>$ Example 1: Beta regression model of a rate

Suppose we wish to know whether offering a summer instruction program increases a school's pass rate for a mandatory state exam administered to students. The school-wide pass rate must be between 0 and 1. It is unlikely that any school will have either no students pass or all students pass, so we consider estimating the effect of the summer instruction program using beta regression.

The dataset sprogram contains fictional data on the pass rate of 1,000 schools (prate). We begin by reading in the data and verifying that prate contains no 0 s or 1 s .

```
. use https://www.stata-press.com/data/r18/sprogram
(Fictional summer program data)
```

- summarize prate

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| prate | 1,000 | .8150803 | .1233684 | .2986041 | .9973584 |

prate ranges from 0.299 to 0.997 , so we proceed with our choice of a beta regression model.
We model prate as a function of a binary indicator for whether the school offered voluntary, half-day instruction to students during the past two summers (summer). The summer program should raise the scores of disadvantaged children who otherwise would not have access to programs that maintain their skills through the summer, thereby increasing the total proportion of students who pass the exam the next year.

We include the fraction of students receiving free or reduced-price meals (freemeals) and the sum of parents' monetary donations to the school two years earlier (pdonations) as additional covariates that measure affluence of the students' parents. We estimate the parameters of this model using the default logit link for the conditional mean and log link for the conditional scale.


Log likelihood = 893.02792

| prate | Coefficient | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | :--- | ---: | ---: | ---: | ---: |
| prate |  |  |  |  |  |  |
| summer |  |  |  |  |  |  |
| Yes | .5560171 | .0480307 | 11.58 | 0.000 | .4618787 | .6501555 |
| freemeals | -.4564181 | .0834885 | -5.47 | 0.000 | -.6200525 | -.2927836 |
| pdonations | .0449706 | .0097781 | 4.60 | 0.000 | .025806 | .0641353 |
| _cons | 1.175013 | .0642797 | 18.28 | 0.000 | 1.049027 | 1.300999 |
| scale |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

The output table reports the estimated coefficients of the covariates and an estimated scale parameter. The coefficient of the factor variable for summer==1, shown as yes under summer, is significant and positive. Thus we conclude that the summer program was effective at increasing a school's pass rate.

However, we cannot determine the magnitude of the effect from these results. In general, when you use betareg, the best way to obtain interpretable effect sizes for the covariates is by using margins. See example 1 in [R] betareg postestimation for more information.

## - Technical note

The results displayed in example 1 can be written concisely in algebraic form. Because we used the default logit link, the estimated conditional mean is

$$
\widehat{\mu}_{\mathbf{x}}=\exp (\mathbf{x} \widehat{\boldsymbol{\beta}}) /\{1+\exp (\mathbf{x} \widehat{\boldsymbol{\beta}})\}
$$

where the estimated $\mathbf{x} \boldsymbol{\beta}$ is

$$
\mathbf{x} \widehat{\boldsymbol{\beta}}=-0.456 \times \text { freemeals }+0.045 \times \text { pdonations }+0.556 \times(\text { summer }==1)+1.18
$$

The estimated $\psi$ with the default $\log$ link for the scale is $\widehat{\psi}=\exp (2.38)=10.80$, which is simply substituted into the formula given above to express the conditional variance.

See Methods and formulas for the functional forms of the other links implemented in betareg for the conditional mean and scale.

## > Example 2: Modeling conditional variance

Some processes require that we model the scale parameter as a function of covariates. For example, we might believe that the proportion of students with free or reduced meals influences the variance of the estimated mean.

We augment example 1 by modeling the scale parameter as a function of freemeals.


Log likelihood $=895.41544$

| prate | Coefficient | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | :--- | ---: | ---: | ---: | ---: |
| prate |  |  |  |  |  |  |
| summer |  |  |  |  |  |  |
| Yes | .5571133 | .0480378 | 11.60 | 0.000 | .4629609 | .6512658 |
| freemeals | -.5291892 | .0896511 | -5.90 | 0.000 | -.7049021 | -.3534762 |
| pdonations | .0454228 | .0097809 | 4.64 | 0.000 | .0262527 | .0645929 |
| _cons | 1.209179 | .0649585 | 18.61 | 0.000 | 1.081863 | 1.336495 |
| scale |  |  |  |  |  |  |
| freemeals | -.3598137 | .1644214 | -2.19 | 0.029 | -.6820737 | -.0375536 |
| _cons | 2.547047 | .0882327 | 28.87 | 0.000 | 2.374114 | 2.71998 |

Again, we conclude that having a summer program increases the pass rate. The effect of an increase in the proportion of students receiving free meals on the conditional variance is ambiguous because it is in both equations. We can use margins to estimate the effect of the program on the conditional mean or the conditional variance.

The estimators in betareg are consistent and efficient when the model is correctly specified. Smithson and Verkuilen (2006) discuss model selection for beta regression and note that selecting the model that minimizes the Bayesian information criterion (BIC) will select the correct model in large samples. Selecting the model that minimizes the BIC is a general approach to model selection; see Cameron and Trivedi (2005) for more details.

## > Example 3: Model selection

We fit the models quietly and use estimates store to store the results under the names model1, model2, model3, and model4.

```
. quietly betareg prate i.summer freemeals pdonations, scale(freemeals)
. estimates store model1
. quietly betareg prate i.summer freemeals pdonations, scale(freemeals)
> link(cloglog)
. estimates store model2
. quietly betareg prate i.summer freemeals pdonations, scale(freemeals)
> slink(root)
. estimates store model3
. quietly betareg prate i.summer freemeals pdonations, scale(freemeals)
> link(cloglog) slink(root)
. estimates store model4
```

Next, we use estimates table to display the coefficients, standard errors, and the BIC for each model.
. estimates table model1 model2 model3 model4, stats(bic) se

| Variable | model1 | model2 | model3 | model4 |
| :---: | :---: | :---: | :---: | :---: |
| prate |  |  |  |  |
| summer |  |  |  |  |
| Yes | .55711332 | .27762093 | .55719742 | .27765283 |
|  | .04803785 | .02460121 | .04803698 | .02459953 |
| freemeals | -.52918917 | -.25685191 | -.5300549 | -.25729221 |
| pdonations | .08965112 | .04308385 | .08978883 | .04314336 |
|  | .04542281 | .02162612 | .04542462 | .02163225 |
| _cons | 1.2091789 | .37961457 | 1.2094991 | .00444817 |
|  | .06495845 | .03212448 | .06496946 | .03212509 |
|  |  |  |  |  |
| scale |  |  |  |  |
| freemeals | -.35981368 | -.36808486 | -.59912234 | -.61295521 |
| _cons | .16442142 | .16448631 | .27259725 | .27273851 |
|  | .0882327 | .08821157 | .15204032 | .15218042 |
| Statistics |  |  |  |  |
| bic | -1749.3843 | -1749.9903 | -1749.444 | -1750.0511 |

We select model4, the model with the complementary $\log -\log$ link for the conditional mean and the square-root link for the conditional variance, because it has the lowest BIC.

## Stored results

betareg stores the following in e():
Scalars
e(N)
e(k)
e(k_eq)
e(k_eq_model)
e(k_dv)
e(df_m)
e(11)
e(11_0)
e(N_clust)
e(chi2)
$e(p)$
e(rank)
e(ic)
e(rc)
e(converged)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(wtype)
e(wexp)
e(title)
e(linkt)
e(linkf)
e(slinkt)
e(slinkf)
e(clustvar)
e(chi2type)
e(vce)
e(vcetype)
e(opt)
$e$ (which)
e(ml_method)
e(user)
e(technique)
e(properties)
e(predict)
e(marginsok)
e(marginsnotok)
e(asbalanced)
e(asobserved)
Matrices
e(b)
$e$ (Cns)
e(ilog)
e(gradient)
e(V)
e(V_modelbased)
Functions
e(sample)
number of observations
number of parameters
number of equations in $e(b)$
number of equations in overall model test
number of dependent variables
model degrees of freedom
log likelihood
log likelihood, constant-only model
number of clusters
$\chi^{2}$
$p$-value for model test
rank of e(V)
number of iterations
return code
1 if converged, 0 otherwise
betareg
command as typed
name of dependent variable
weight type
weight expression
title in estimation output
link title in the conditional mean equation
link function in the conditional mean equation
link title in the conditional scale equation
link function in the conditional scale equation
name of cluster variable
Wald or LR; type of model $\chi^{2}$ test
vcetype specified in vce()
title used to label Std. err.
type of optimization
max or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement predict
predictions allowed by margins
predictions disallowed by margins
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
constraints matrix
iteration $\log$ (up to 20 iterations)
gradient vector
variance-covariance matrix of the estimators
model-based variance
marks estimation sample

In addition to the above, the following is stored in r() :

```
Matrices
    r(table)
```

matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r -class command is run after the estimation command.

## Methods and formulas

Beta regression models were proposed by Ferrari and Cribari-Neto (2004) and extended by Smithson and Verkuilen (2006) to allow the scale parameter to depend on covariates.

Beta regression is appropriate only for a dependent variable that is strictly greater than 0 and strictly less than 1 because the beta distribution has support only on the interval $(0,1)$. The density of a beta-distributed dependent variable $y$ conditional on covariates $\mathbf{x}$ can be written as

$$
f\left(y ; \mu_{\mathbf{x}}, \psi_{\mathbf{x}}\right)=\frac{\Gamma\left(\psi_{\mathbf{x}}\right)}{\Gamma\left(\mu_{\mathbf{x}} \psi_{\mathbf{x}}\right) \Gamma\left\{\left(1-\mu_{\mathbf{x}}\right) \psi_{\mathbf{x}}\right\}} y^{\mu_{\mathbf{x}} \psi_{\mathbf{x}}-1}(1-y)^{\left(1-\mu_{\mathbf{x}}\right) \psi_{\mathbf{x}}-1}
$$

where $\mu_{\mathbf{x}}=\mathbf{E}(y \mid \mathbf{x}), \mu_{\mathbf{x}}$ is linked to the covariates by the link function $g\left(\mu_{\mathbf{x}}\right)=\mathbf{x} \boldsymbol{\beta}, \psi_{\mathbf{x}}$ scales the conditional variance according to

$$
\operatorname{Var}(y \mid \mathbf{x})=\mu_{\mathbf{x}}\left(1-\mu_{\mathbf{x}}\right) /\left(1+\psi_{\mathbf{x}}\right)
$$

and $\psi_{\mathbf{x}}$ is linked to the covariates by link function $h\left(\psi_{\mathbf{x}}\right)=\mathbf{x} \gamma$.
This parameterization yields a log-likelihood function of

$$
\begin{aligned}
\sum_{i=1}^{N} \omega_{i}( & \ln \left\{\Gamma\left(\psi_{\mathbf{x}, i}\right)\right\}-\ln \left\{\Gamma\left(\mu_{\mathbf{x}, i} \psi_{\mathbf{x}, i}\right)\right\}-\ln \left[\Gamma\left\{\left(1-\mu_{\mathbf{x}, i}\right) \psi_{\mathbf{x}, i}\right\}\right] \\
& \left.+\left(\mu_{\mathbf{x}, i} \psi_{\mathbf{x}, i}-1\right) \ln \left(y_{i}\right)+\left\{\left(1-\mu_{\mathbf{x}, i}\right) \psi_{\mathbf{x}, i}-1\right\} \ln \left(1-y_{i}\right)\right)
\end{aligned}
$$

The definitions of the link functions are

| Name | Function |
| :--- | :--- |
| logit | $g\left(\mu_{\mathbf{x}}\right)=\ln \left\{\mu_{\mathbf{x}} /\left(1-\mu_{\mathbf{x}}\right)\right\}$ |
| probit | $g\left(\mu_{\mathbf{x}}\right)=\Phi^{-1}\left(\mu_{\mathbf{x}}\right)$ |
| cloglog | $g\left(\mu_{\mathbf{x}}\right)=\ln \left\{-\ln \left(1-\mu_{\mathbf{x}}\right)\right\}$ |
| loglog | $g\left(\mu_{\mathbf{x}}\right)=-\ln \left\{-\ln \left(\mu_{\mathbf{x}}\right)\right\}$ |

The definitions of the scale-link functions are

| Name | Function |
| :--- | :--- |
| $\log$ | $h\left(\psi_{\mathbf{x}}\right)=\ln \left(\psi_{\mathbf{x}}\right)$ |
| root | $h\left(\psi_{\mathbf{x}}\right)=\sqrt{\psi_{\mathbf{x}}}$ |
| identity | $h\left(\psi_{\mathbf{x}}\right)=\psi_{\mathbf{x}}$ |

This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce (robust) and vce (cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.

## Acknowledgments

We thank Maarten L. Buis of the University of Konstanz, Germany; Nicholas J. Cox of Durham University, UK, who is coeditor of the Stata Journal and author of Speaking Stata Graphics; and Stephen P. Jenkins of the London School of Economics and Political Science, UK, who is coeditor of the Stata Journal and author of betafit, an early implementation of beta regression in Stata.

## References

Basu, A., and A. Manca. 2012. Regression estimators for generic health-related quality of life and quality-adjusted life years. Medical Decision Making 32: 56-69. https://doi.org/10.1177/0272989X11416988.

Cameron, A. C., and P. K. Trivedi. 2005. Microeconometrics: Methods and Applications. New York: Cambridge University Press.
Castellani, M., P. Pattitoni, and A. E. Scorcu. 2012. Visual artist price heterogeneity. Economics and Business Letters 1(3): 16-22. https://doi.org/10.17811/ebl.1.3.2012.16-22.

Ferrari, S. L. P., and F. Cribari-Neto. 2004. Beta regression for modelling rates and proportions. Journal of Applied Statistics 31: 799-815. https://doi.org/10.1080/0266476042000214501.
Gray, L. A., and M. Hernández-Alava. 2018. A command for fitting mixture regression models for bounded dependent variables using the beta distribution. Stata Journal 18: 51-75.

Hubben, G. A. A., D. Bishai, P. Pechlivanoglou, A. M. Cattelan, R. Grisetti, C. Facchin, F. A. Compostella, J. M. Bos, M. J. Postma, and A. Tramarin. 2008. The societal burden of HIV/AIDS in Northern Italy: An analysis of costs and quality of life. AIDS Care: Psychological and Socio-medical Aspects of AIDS/HIV 20: 449-455. https://doi.org/10.1080/09540120701867107.

Paolino, P. 2001. Maximum likelihood estimation of models with beta-distributed dependent variables. Political Analysis 9: 325-346. https://doi.org/10.1093/oxfordjournals.pan.a004873.
Smithson, M., S. Deady, and L. Gracik. 2007. Guilty, not guilty, or . . . ? Multiple options in jury verdict choices. Journal of Behavioral Decision Making 20: 481-498. https://doi.org/10.1002/bdm.572.

Smithson, M., and J. Verkuilen. 2006. A better lemon squeezer? Maximum-likelihood regression with beta-distributed dependent variables. Psychological Methods 11: 54-71. https://doi.org/10.1037/1082-989X.11.1.54.

## Also see

[R] betareg postestimation - Postestimation tools for betareg
[R] fracreg - Fractional response regression
[R] glm - Generalized linear models
[BAYES] bayes: betareg - Bayesian beta regression
[FMM] fmm: betareg - Finite mixtures of beta regression models
[SVY] svy estimation - Estimation commands for survey data
[U] 20 Estimation and postestimation commands

## Title

betareg postestimation - Postestimation tools for betareg
Postestimation commands predict margins Remarks and examples

## Postestimation commands

The following postestimation commands are available after betareg:

| Command | Description |
| :---: | :---: |
| contrast | contrasts and ANOVA-style joint tests of estimates |
| estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estat (svy) | postestimation statistics for survey data |
| estimates | cataloging estimation results |
| etable | table of estimation results |
| * forecast | dynamic forecasts and simulations |
| * hausman | Hausman's specification test |
| lincom | point estimates, standard errors, testing, and inference for linear combinations of coefficients |
| * 1 rtest | likelihood-ratio test |
| margins | marginal means, predictive margins, marginal effects, and average marginal effects |
| marginsplot | graph the results from margins (profile plots, interaction plots, etc.) |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients |
| predict | conditional means and variances, linear predictions, etc. |
| predictnl | point estimates, standard errors, testing, and inference for generalized predictions |
| pwcompare | pairwise comparisons of estimates |
| suest | seemingly unrelated estimation |
| test | Wald tests of simple and composite linear hypotheses |
| testnl | Wald tests of nonlinear hypotheses |

[^4]
## predict

## Description for predict

predict creates a new variable containing predictions such as linear predictions, conditional means, conditional variances, and equation-level scores.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

predict [type] newvar [if] [in] [, statistic]
predict [type] stub* [if] [in], scores

## statistic Description

Main

| cmean | conditional mean of the dependent variable; the default <br> conditional variance of the dependent variable |
| :--- | :--- |
| $\frac{\mathrm{xb}}{\mathrm{xb}} \mathrm{lance}$ | linear prediction in the conditional-mean equation |
| xbscale | linear prediction in the conditional-scale equation <br> stdp |

These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

## Options for predict

$\qquad$ Main
cmean, the default, calculates the conditional mean of the dependent variable. cvariance calculates the conditional variance of the dependent variable. xb calculates the linear prediction for the conditional-mean equation.
xbscale calculates the linear prediction for the conditional-scale equation.
stdp calculates the standard error of the linear prediction for the conditional-mean equation.
scores calculates equation-level score variables. The first new variable will contain the derivative of the log likelihood with respect to the conditional-mean equation, and the second new variable will contain the derivative of the log likelihood with respect to the conditional-scale equation.

## margins

## Description for margins

margins estimates margins of response for conditional means, conditional variances, and linear predictions.

## Menu for margins

Statistics > Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...)...] [options]
```

statistic Description

| cmean | conditional mean of the dependent variable; the default <br> conditional variance of the dependent variable |
| :--- | :--- |
| $\underline{\mathrm{cvar}} \mathrm{iance}$ | linear prediction in the conditional-mean equation |
| xbscale | linear prediction in the conditional-scale equation |
| stdp | not allowed with margins |
| scores | not allowed with margins |

Statistics not allowed with margins are functions of stochastic quantities other than e(b). For the full syntax, see [R] margins.

## Remarks and examples

## > Example 1

In example 3 in [R] betareg, we selected a model for school-level data on the fraction of students passing a state-required exam. In that example, we were interested in whether a voluntary summer program increased schools' pass rates. We continue that example to estimate an average treatment effect (ATE) of the program.

After reading in the data and fitting the model, we use margins to estimate the ATE.

```
. use https://www.stata-press.com/data/r18/sprogram
(Fictional summer program data)
. betareg prate freemeals pdonations i.summer, scale(freemeals) link(cloglog)
> slink(root) vce(robust)
    (output omitted)
```

We specify vce(robust) with the estimation command and vce(unconditional) with the margins command to obtain standard errors for a population ATE instead of a sample ATE.

```
. margins r.summer, contrast(nowald) vce(unconditional)
Contrasts of predictive margins Number of obs = 1,000
```

Expression: Conditional mean of prate, predict()

|  | Unconditional <br> std. err. |  |  |  |
| :--- | :---: | :---: | :---: | :---: |
| [95\% conf. interval] |  |  |  |  |
| (Yes vs No) | .0890851 | .008626 | .0721784 | .1059918 |

The average pass rate would be about $9 \%$ higher when all schools offered the program than when no school offered the program.

## Also see

[R] betareg - Beta regression
[U] 20 Estimation and postestimation commands

## Title

binreg - Generalized linear models: Extensions to the binomial family

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

binreg fits generalized linear models for the binomial family. It estimates odds ratios, risk ratios, health ratios, and risk differences. The available links are

| Option | Implied link | Parameter |
| ---: | ---: | ---: |
| or | logit | odds ratios $=\exp (\beta)$ |
| rr | log | risk ratios $=\exp (\beta)$ |
| hr | log complement | health ratios $=\exp (\beta)$ |
| rd | identity | risk differences $=\beta$ |

Estimates of odds, risk, and health ratios are obtained by exponentiating the appropriate coefficients. The or option produces the same results as Stata's logistic command, and or coefficients yields the same results as the logit command. When no link is specified, or is assumed.

## Quick start

Report odds ratios from a model of y on x 1 and x 2 using a logit link
binreg y x1 x2, or
Use the $\log$ link and report risk ratios
binreg y x1 x2, rr
Use the identity link and report risk differences
binreg y x1 x2, rd
Same as above, but with data stored as the number of successes, ys, out of $n$ trials
binreg ys x 1 x 2 , rd $\mathrm{n}(\mathrm{n})$

## Menu

Statistics $>$ Generalized linear models $>$ GLM for the binomial family

## Syntax

binreg depvar [indepvars] [if] [in] [weight] [, options]
options
Description
Model
noconstant
or
rr
hr
rd
n(\#|varname)
exposure (varname)
offset (varname)
constraints (constraints)
mu(varname)
init (varname)
SE/Robust
vce (vcetype)
t (varname)
vfactor(\#)
disp(\#)
scale(x2|dev|\#)
Reporting
level(\#)
coefficients
nocnsreport
display_options

Maximization
irls
ml
maximize_options
fisher (\#)
search
collinear
coeflegend
suppress constant term use logit link and report odds ratios use log link and report risk ratios use log-complement link and report health ratios use identity link and report risk differences use \# or varname for number of trials include $\ln$ (varname) in model with coefficient constrained to 1 include varname in model with coefficient constrained to 1 apply specified linear constraints
use varname as the initial estimate for the mean of depvar synonym for mu(varname)
vcetype may be eim, robust, cluster clustvar, oim, opg, bootstrap, jackknife, hac kernel, jackknife1, or unbiased
variable name corresponding to time
multiply variance matrix by scalar \#
quasilikelihood multiplier
set the scale parameter; default is scale(1)
set confidence level; default is level (95)
report nonexponentiated coefficients
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
use iterated, reweighted least-squares optimization; the default use maximum likelihood optimization
control the maximization process; seldom used
Fisher scoring steps
search for good starting values
keep collinear variables
display legend instead of statistics
indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
depvar and indepvars may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bayes, bootstrap, by, collect, fp, jackknife, mi estimate, rolling, and statsby are allowed; see
[U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: binreg.
vce(bootstrap), vce(jackknife), and vce(jackknife1) are not allowed with the mi estimate prefix; see [MI] mi estimate.
Weights are not allowed with the bootstrap prefix; see [R] bootstrap.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

Model
noconstant; see [R] Estimation options.
or requests the logit link and results in odds ratios if coefficients is not specified.
rr requests the log link and results in risk ratios if coefficients is not specified.
hr requests the log-complement link and results in health ratios if coefficients is not specified. rd requests the identity link and results in risk differences.
n (\#|varname) specifies either a constant integer to use as the denominator for the binomial family or a variable that holds the denominator for each observation.
exposure(varname), offset (varname), constraints(constraints); see [R] Estimation options. constraints (constraints) is not allowed with irls.
mu (varname) specifies varname containing an initial estimate for the mean of depvar. This option can be useful if you encounter convergence difficulties. init(varname) is a synonym.

SE/Robust
vce (vcetype) specifies the type of standard error reported, which includes types that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), that are derived from asymptotic theory (oim, opg), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
vce (eim), the default, uses the expected information matrix (EIM) for the variance estimator.
binreg also allows the following:
vce (hac kernel [\#]) specifies that a heteroskedasticity- and autocorrelation-consistent (HAC) variance estimate be used. HAC refers to the general form for combining weighted matrices to form the variance estimate. There are three kernels built into binreg. kernel is a user-written program or one of

```
nwest | gallant | anderson
```

If \# is not specified, $N-2$ is assumed.
vce (jackknife1) specifies that the one-step jackknife estimate of variance be used.
vce (unbiased) specifies that the unbiased sandwich estimate of variance be used.
t(varname) specifies the variable name corresponding to time; see [TS] tsset. binreg does not always need to know $t$ (), though it does if vce (hac ...) is specified. Then you can either specify the time variable with $t$ (), or you can tsset your data before calling binreg. When the time variable is required, binreg assumes that the observations are spaced equally over time.
vfactor(\#) specifies a scalar by which to multiply the resulting variance matrix. This option allows users to match output with other packages, which may apply degrees of freedom or other small-sample corrections to estimates of variance.
disp(\#) multiplies the variance of depvar by \# and divides the deviance by \#. The resulting distributions are members of the quasilikelihood family. This option is not allowed with option ml.
scale (x2|dev|\#) overrides the default scale parameter. This option is allowed only with Hessian (information matrix) variance estimates.
By default, scale(1) is assumed for the discrete distributions (binomial, Poisson, and negative binomial), and scale (x2) is assumed for the continuous distributions (Gaussian, gamma, and inverse Gaussian).
scale ( x 2 ) specifies that the scale parameter be set to the Pearson $\chi^{2}$ (or generalized $\chi^{2}$ ) statistic divided by the residual degrees of freedom, which is recommended by McCullagh and Nelder (1989) as a good general choice for continuous distributions.
scale(dev) sets the scale parameter to the deviance divided by the residual degrees of freedom. This option provides an alternative to scale (x2) for continuous distributions and overdispersed or underdispersed discrete distributions. This option is not allowed with option ml .
scale(\#) sets the scale parameter to \#.

## Reporting <br> level (\#), noconstant; see [R] Estimation options.

coefficients displays the nonexponentiated coefficients and corresponding standard errors and confidence intervals. This option has no effect when the rd option is specified, because it always presents the nonexponentiated coefficients.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\% fint), pformat (\% fint), sformat (\% fint), and nolstretch; see [R] Estimation options.

## Maximization

irls requests iterated, reweighted least-squares (IRLS) optimization of the deviance instead of Newton-Raphson optimization of the log likelihood. This option is the default.
ml requests that optimization be carried out by using Stata's ml command; see [R] ml. maximize_options: difficult, technique (algorithm_spec), iterate (\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from(init_specs); see [R] Maximize. These options are seldom used.
Setting the optimization method to ml , with technique () set to something other than BHHH, changes the vcetype to vce (oim). Specifying technique (bhhh) changes vcetype to vce (opg).
Unless option ml is specified, only maximize_options iterate(), nolog, trace, and ltolerance() are allowed. With IRLS optimization, the convergence criterion is satisfied when the
absolute change in deviance from one iteration to the next is less than or equal to ltolerance(), where ltolerance $(1 e-6)$ is the default.
fisher (\#) specifies the number of Newton-Raphson steps that should use the Fisher scoring Hessian or EIM before switching to the observed information matrix (OIM). This option is available only if ml is specified and is useful only for Newton-Raphson optimization.
search specifies that the command search for good starting values. This option is available only if ml is specified and is useful only for Newton-Raphson optimization.

The following options are available with binreg but are not shown in the dialog box:
collinear, coeflegend; see [R] Estimation options. collinear is not allowed with irls.

## Remarks and examples

Wacholder (1986) suggests methods for estimating risk ratios and risk differences from prospective binomial data. These estimates are obtained by selecting the proper link functions in the generalized linear-model framework. (See Methods and formulas for details; also see [R] glm.)

## > Example 1

Wacholder (1986) presents an example, using data from Wright et al. (1983), of an investigation of the relationship between alcohol consumption and the risk of a low-birthweight baby. Covariates examined included whether the mother smoked (yes or no), mother's social class (three levels), and drinking frequency (light, moderate, or heavy). The data for the 18 possible categories determined by the covariates are illustrated below.

Let's first describe the data and list a few observations.
. use https://www.stata-press.com/data/r18/binreg (Alcohol and low-birthweight baby) . list

|  | category | n_lbw_~s | n_women | alcohol | smokes | social |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 1. | 1 | 11 | 84 | Heavy | Nonsmoker | 1 |
| 2. | 2 | 5 | 79 | Moderate | Nonsmoker | 1 |
| 3. | 3 | 11 | 169 | Light | Nonsmoker | 1 |
| 4. | 4 | 6 | 28 | Heavy | Smoker | 1 |
| 5. | 5 | 3 | 13 | Moderate | Smoker | 1 |
| 6. | 6 | 1 | 26 | Light | Smoker | 1 |
| 7. | 7 | 4 | 22 | Heavy | Nonsmoker | 2 |
| 8. | 8 | 3 | 25 | Moderate | Nonsmoker | 2 |
| 9. | 9 | 12 | 162 | Light | Nonsmoker | 2 |
| 10. | 10 | 4 | 17 | Heavy | Smoker | 2 |
| 11. | 11 | 2 | 7 | Moderate | Smoker | 2 |
| 12. | 12 | 6 | 38 | Light | Smoker | 2 |
| 13. | 13 | 0 | 14 | Heavy | Nonsmoker | 3 |
| 14. | 14 | 1 | 18 | Moderate | Nonsmoker | 3 |
| 15. | 15 | 12 | 91 | Light | Nonsmoker | 3 |
| 16. | 16 | 7 | 19 | Heavy | Smoker | 3 |
| 17. | 17 | 2 | 18 | Moderate | Smoker | 3 |
| 18. | 18 | 8 | 70 | Light | Smoker | 3 |

Each observation corresponds to one of the 18 covariate structures. The number of low-birthweight babies from n_women in each category is given by the $n \_l b w \_b a b i e s ~ v a r i a b l e . ~$

We begin by estimating risk ratios:


Note: _cons estimates baseline risk.
By default, Stata reports the risk ratios (the exponentiated regression coefficients) estimated by the model. We can see that the risk ratio comparing heavy drinkers with light drinkers, after adjusting for smoking and social class, is 1.974078 . That is, mothers who drink heavily during their pregnancy have approximately twice the risk of delivering low-birthweight babies as mothers who are light drinkers.

The nonexponentiated coefficients can be obtained with the coefficients option:

```
. binreg n_lbw_babies i.soc i.alc i.smo, n(n_women) rr coefficients
Iteration 1: Deviance = 14.2879
Iteration 2: Deviance = 13.607
Iteration 3: Deviance = 13.60503
Iteration 4: Deviance = 13.60503
```

Generalized linear models $\quad$ Number of obs $=18$

BIC $=-21.07943$

| n_lbw_babies | Coefficient | $\begin{gathered} \text { EIM } \\ \text { std. err. } \end{gathered}$ | z | $P>\|z\|$ | [95\% conf | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| social |  |  |  |  |  |  |
| 2 | . 2926702 | . 2333866 | 1.25 | 0.210 | -. 1647591 | . 7500994 |
| 3 | . 2997244 | . 2439066 | 1.23 | 0.219 | -. 1783238 | . 7777726 |
| alcohol |  |  |  |  |  |  |
| Moderate | . 1749248 | . 274133 | 0.64 | 0.523 | -. 362366 | . 7122156 |
| Heavy | . 6801017 | . 2158856 | 3.15 | 0.002 | . 2569737 | 1.10323 |
| smokes |  |  |  |  |  |  |
| Smoker | . 4998317 | . 2019329 | 2.48 | 0.013 | . 1040505 | . 8956129 |
| _cons | -2.764079 | . 2031606 | -13.61 | 0.000 | -3.162266 | -2.365891 |

Risk differences are obtained with the rd option:


The risk difference between heavy drinkers and light drinkers is 0.0801291 . Because the risk differences are obtained directly from the coefficients estimated by using the identity link, the coefficients option would have no effect here.

Health ratios are obtained with the hr option. The health ratios (exponentiated coefficients for the log-complement link) are reported directly.

| Iteration 1: Deviance $=21.15233$ |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Iteration 2: Deviance $=15.16467$ |  |  |  |  |  |  |
| Iteration 3: Deviance $=15.13205$ |  |  |  |  |  |  |
| Iteration 4: Deviance $=15.13114$ |  |  |  |  |  |  |
| Iteration 5: Deviance $=15.13111$ |  |  |  |  |  |  |
| Iteration 6: Deviance $=15.13111$ |  |  |  |  |  |  |
| Iteration 7: Deviance $=15.13111$ |  |  |  |  |  |  |
| Generalized linear models |  |  |  | Numb | of obs | 18 |
| Optimization : MQL Fisher scori |  |  |  | Resi | df | 12 |
|  |  |  |  | Scal | arameter | 1 |
| Deviance $=15.13110545$ |  |  |  | (1/d | Deviance | 1.260925 |
| Pearson $\quad=12.84203917$ |  |  |  | (1/d | Pearson | 1.07017 |
| Variance function: $V(u)=u *\left(1-u / n_{-} w o m e n\right)$ <br> Link function : $\mathrm{g}(\mathrm{u})=\ln \left(1-\mathrm{u} / \mathrm{n}_{\mathrm{n}}\right.$ women) |  |  |  | [Binomial] |  |  |
|  |  |  |  | [Log complement] |  |  |
| Link function |  |  |  | BIC |  | -19.55336 |
| n_lbw_babies | EIM |  |  |  |  |  |
|  | Hlth ratio | std. err. | z | $P>\|z\|$ | [95\% con | interval] |
| social |  |  |  |  |  |  |
| 2 | . 9720541 | . 024858 | -1.11 | 0.268 | . 9245342 | 1.022017 |
| 3 | . 9597182 | . 0290412 | -1.36 | 0.174 | . 9044535 | 1.01836 |
| alcohol |  |  |  |  |  |  |
| Moderate | . 9871517 | . 0278852 | -0.46 | 0.647 | . 9339831 | 1.043347 |
| Heavy | . 9134243 | . 0325726 | -2.54 | 0.011 | . 8517631 | . 9795493 |
| smokes |  |  |  |  |  |  |
| Smoker | . 9409983 | . 0296125 | -1.93 | 0.053 | . 8847125 | 1.000865 |
| _cons | . 9409945 | . 0163084 | -3.51 | 0.000 | . 9095674 | . 9735075 |

To see the nonexponentiated coefficients, we could specify the coefficients option.

## Stored results

binreg, irls stores the following in e() :
Scalars
e(N)
e(k)
e(k_eq_model)
e(df_m)
e(df)
e(phi)
e(disp)
e(bic)
e(N_clust)
e(deviance)
e(deviance_s)
e(deviance_p)
e(deviance_ps)
e(dispers)
e(dispers_s)
e(dispers_p)
e(dispers_ps)
e(vf)
e(rank)
e(rc)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(eform)
e(varfunc)
e(varfunct)
e(varfuncf)
e(link)
e(linkt)
e(linkf)
e(m)
e(wtype)
e(wexp)
e(title)
e(title_fl)
e(clustvar)
e(offset)
e(cons)
e(hac_kernel)
e(hac_lag)
e(vce)
e(vcetype)
e(opt)
e(opt1)
e(opt2)
e(properties)
e(predict)
e(marginsok)
e(marginsnotok)
e(asbalanced)
e(asobserved)
number of observations
number of parameters
number of equations in overall model test
model degrees of freedom
residual degrees of freedom
model scale parameter
dispersion parameter
model BIC
number of clusters
deviance
scaled deviance
Pearson deviance
scaled Pearson deviance
dispersion
scaled dispersion
Pearson dispersion
scaled Pearson dispersion
factor set by vfactor(), 1 if not set
rank of $e(V)$
return code

## binreg

command as typed
name of dependent variable
eform() option implied by or, rr, hr, or rd
program to calculate variance function
variance title
variance function
program to calculate link function
link title
link function
number of binomial trials
weight type
weight expression
title in estimation output
family-link title
name of cluster variable
linear offset variable
noconstant or not set
HAC kernel
HAC lag
vcetype specified in vce()
title used to label Std. err.
type of optimization
optimization title, line 1
optimization title, line 2
b V
program used to implement predict
predictions allowed by margins
predictions disallowed by margins
factor variables fvset as asbalanced
factor variables fvset as asobserved

Matrices

```
e(b)
coefficient vector
e(V) variance-covariance matrix of the estimators
e(V_modelbased) model-based variance
    e(sample) marks estimation sample
```

Functions

In addition to the above, the following is stored in $r()$ :

```
Matrices
    r(table)
```

matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any $r$-class command is run after the estimation command.
binreg, ml stores the following in e() :
Scalars
e(N)
e(k)
e(k_eq)
e(k_eq_model)
e(k_dv)
$e\left(d f \_m\right)$
e(df)
e(phi)
e(aic)
e(bic)
e(ll)
e(N_clust)
e(chi2)
e(p)
e(deviance)
e(deviance_s)
e(deviance_p)
e(deviance_ps)
e(dispers)
e(dispers_s)
e(dispers_p)
e(dispers_ps)
e(vf)
e(rank)
e(ic)
e(rc)
e(converged)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(eform)
e(varfunc)
e(varfunct)
e(varfuncf)
e(link)
e(linkt)
e(linkf)
e(m)
e(wtype)
e(wexp)
e(title)
number of observations
number of parameters
number of equations in $e(b)$
number of equations in overall model test
number of dependent variables
model degrees of freedom
residual degrees of freedom
model scale parameter
model AIC, if ml
model BIC
$\log$ likelihood, if ml
number of clusters
$\chi^{2}$
$p$-value for model test
deviance
scaled deviance
Pearson deviance
scaled Pearson deviance
dispersion
scaled dispersion
Pearson dispersion
scaled Pearson dispersion
factor set by vfactor(), 1 if not set
rank of $e(V)$
number of iterations
return code
1 if converged, 0 otherwise
binreg
command as typed
name of dependent variable
eform() option implied by or, $r r, h r$, or $r d$
program to calculate variance function
variance title
variance function
program to calculate link function
link title
link function
number of binomial trials
weight type
weight expression
title in estimation output

```
    e(title_fl)
    e(clustvar)
    e(offset)
    e(cons)
    e(hac_kernel)
    e(hac_lag)
    e(chi2type)
    e(vce)
    e(vcetype)
    e(opt)
    e(opt1)
    e(which)
    e(ml_method)
    e(user)
    e(technique)
    e(properties)
    e(predict)
    e(marginsok)
    e(marginsnotok)
    e(asbalanced)
    e(asobserved)
Matrices
    e(b)
    e(Cns)
    e(ilog)
    e(gradient)
    e(V)
    e(V_modelbased)
Functions
```

    e(sample) marks estimation sample
    In addition to the above, the following is stored in r() :
$r$ (table) matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

Let $\pi_{i}$ be the probability of success for the $i$ th observation, $i=1, \ldots, N$, and let $X \beta$ be the linear predictor. The link function relates the covariates of each observation to its respective probability through the linear predictor.

In logistic regression, the logit link is used:

$$
\ln \left(\frac{\pi}{1-\pi}\right)=X \beta
$$

The regression coefficient $\beta_{k}$ represents the change in the logarithm of the odds associated with a one-unit change in the value of the $X_{k}$ covariate; thus, $\exp \left(\beta_{k}\right)$ is the ratio of the odds associated with a change of one unit in $X_{k}$.

For risk differences, the identity link $\pi=X \beta$ is used. The regression coefficient $\beta_{k}$ represents the risk difference associated with a change of one unit in $X_{k}$. When using the identity link, you can obtain fitted probabilities outside the interval $(0,1)$. As suggested by Wacholder, at each iteration,
fitted probabilities are checked for range conditions (and put back in range if necessary). For example, if the identity link results in a fitted probability that is smaller than $1 \mathrm{e}-4$, the probability is replaced with $1 \mathrm{e}-4$ before the link function is calculated.

A similar adjustment is made for the logarithmic link, which is used for estimating the risk ratio, $\ln (\pi)=X \beta$, where $\exp \left(\beta_{k}\right)$ is the risk ratio associated with a change of one unit in $X_{k}$, and for the log-complement link used to estimate the probability of no disease or health, where $\exp \left(\beta_{k}\right)$ represents the "health ratio" associated with a change of one unit in $X_{k}$.

This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce (robust) and vce (cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.

## References

Cummings, P. 2009. Methods for estimating adjusted risk ratios. Stata Journal 9: 175-196.
Kleinbaum, D. G., and M. Klein. 2010. Logistic Regression: A Self-Learning Text. 3rd ed. New York: Springer. McCullagh, P., and J. A. Nelder. 1989. Generalized Linear Models. 2nd ed. London: Chapman and Hall/CRC.

Wacholder, S. 1986. Binomial regression in GLIM: Estimating risk ratios and risk differences. American Journal of Epidemiology 123: 174-184. https://doi.org/10.1093/oxfordjournals.aje.a114212.

Wright, J. T., I. G. Barrison, I. G. Lewis, K. D. MacRae, E. J. Waterson, P. J. Toplis, M. G. Gordon, N. F. Morris, and I. M. Murray-Lyon. 1983. Alcohol consumption, pregnancy, and low birthweight. Lancet 1: 663-665. https://doi.org/10.1016/S0140-6736(83)91964-5.

## Also see

[R] binreg postestimation - Postestimation tools for binreg
[R] glm - Generalized linear models
[BAYES] bayes: binreg - Bayesian generalized linear models: Extensions to the binomial family
[ME] mecloglog - Multilevel mixed-effects complementary log-log regression
[ME] meglm - Multilevel mixed-effects generalized linear models
[ME] melogit - Multilevel mixed-effects logistic regression
[ME] meprobit - Multilevel mixed-effects probit regression
[MI] Estimation - Estimation commands for use with mi estimate
[U] 20 Estimation and postestimation commands

## Title

binreg postestimation - Postestimation tools for binreg

Postestimation commands predict margins References Also see

## Postestimation commands

The following postestimation commands are available after binreg:

| Command | Description |
| :---: | :---: |
| contrast | contrasts and ANOVA-style joint tests of estimates |
| *estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estimates | cataloging estimation results |
| etable | table of estimation results |
| $\dagger$ forecast | dynamic forecasts and simulations |
| hausman | Hausman's specification test |
| lincom | point estimates, standard errors, testing, and inference for linear combinations of coefficients |
| linktest | link test for model specification |
| *lrtest | likelihood-ratio test |
| margins | marginal means, predictive margins, marginal effects, and average marginal effects |
| marginsplot | graph the results from margins (profile plots, interaction plots, etc.) |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients |
| predict | predictions, residuals, influence statistics, and other diagnostic measures |
| predictnl | point estimates, standard errors, testing, and inference for generalized predictions |
| pwcompare | pairwise comparisons of estimates |
| suest | seemingly unrelated estimation |
| test | Wald tests of simple and composite linear hypotheses |
| testnl | Wald tests of nonlinear hypotheses |

[^5]
## predict

## Description for predict

predict creates a new variable containing predictions such as expected values, linear predictions, standard errors, residuals, Cook's distance, diagonals, weighted averages, differences, and first derivatives.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

predict [type] newvar [if] [in] [, statistic options]
statistic Description
Main
mu expected value of $y$; the default
$\mathrm{xb} \quad$ linear prediction $\eta=\mathbf{x} \widehat{\boldsymbol{\beta}}$
eta synonym for xb
stdp standard error of the linear prediction
anscombe Anscombe (1953) residuals
cooksd
deviance
hat diagonals of the "hat" matrix
likelihood
pearson
response
score
working
Cook's distance
deviance residuals

Pearson residuals
working residuals
weighted average of the standardized deviance and standard Pearson residuals
differences between the observed and fitted outcomes
first derivative of the log likelihood with respect to $\mathbf{x}_{j} \boldsymbol{\beta}$
options
Options
nooffset
adjusted standardized studentized modified

Description
modify calculations to ignore the offset variable
adjust deviance residual to speed up convergence
multiply residual by the factor $(1-h)^{1 / 2}$
multiply residual by one over the square root of the estimated scale parameter modify denominator of residual to be a reasonable estimate of the variance of depvar

These statistics are available both in and out of sample; type predict . . . if e(sample) ... if wanted only for the estimation sample.

## Options for predict

$\qquad$
mu , the default, specifies that predict calculate the expected value of $y$, equal to $g^{-1}(\mathbf{x} \widehat{\boldsymbol{\beta}})\left[n g^{-1}(\mathbf{x} \widehat{\boldsymbol{\beta}})\right.$ for the binomial family].
xb calculates the linear prediction $\eta=\mathbf{x} \widehat{\boldsymbol{\beta}}$.
eta is a synonym for xb .
stdp calculates the standard error of the linear prediction.
anscombe calculates the Anscombe (1953) residuals to produce residuals that closely follow a normal distribution.
cooksd calculates Cook's distance, which measures the aggregate change in the estimated coefficients when each observation is left out of the estimation.
deviance calculates the deviance residuals, which are recommended by McCullagh and Nelder (1989) and others as having the best properties for examining goodness of fit of a GLM. They are approximately normally distributed if the model is correct and may be plotted against the fitted values or against a covariate to inspect the model's fit. Also see the pearson option below.
hat calculates the diagonals of the "hat" matrix, analogous to linear regression.
likelihood calculates a weighted average of the standardized deviance and standardized Pearson (described below) residuals.
pearson calculates the Pearson residuals, which often have markedly skewed distributions for nonnormal family distributions. Also see the deviance option above.
response calculates the differences between the observed and fitted outcomes.
score calculates the equation-level score, $\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)$.
working calculates the working residuals, which are response residuals weighted according to the derivative of the link function.

## Options

nooffset is relevant only if you specified offset (varname) for binreg. It modifies the calculations made by predict so that they ignore the offset variable; the linear prediction is treated as $\mathbf{x}_{j} \mathbf{b}$ rather than as $\mathbf{x}_{j} \mathbf{b}+$ offset $_{j}$.
adjusted adjusts the deviance residual to make the convergence to the limiting normal distribution faster. The adjustment deals with adding to the deviance residual a higher-order term depending on the variance function family. This option is allowed only when deviance is specified.
standardized requests that the residual be multiplied by the factor $(1-h)^{-1 / 2}$, where $h$ is the diagonal of the hat matrix. This step is done to take into account the correlation between depvar and its predicted value.
studentized requests that the residual be multiplied by one over the square root of the estimated scale parameter.
modified requests that the denominator of the residual be modified to be a reasonable estimate of the variance of depvar. The base residual is multiplied by the factor $(k / w)^{-1 / 2}$, where $k$ is either one or the user-specified dispersion parameter and $w$ is the specified weight (or one if left unspecified).

## margins

## Description for margins

margins estimates margins of response for expected values and linear predictions.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...)...] [options]
```

statistic Description
mu expected value of $y$; the default
$\mathrm{xb} \quad$ linear prediction $\eta=\mathbf{x} \widehat{\boldsymbol{\beta}}$
eta synonym for xb
stdp not allowed with margins
anscombe not allowed with margins
cooksd not allowed with margins
deviance not allowed with margins
hat not allowed with margins
likelihood not allowed with margins
pearson not allowed with margins
response not allowed with margins
score not allowed with margins
working not allowed with margins

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

## References

Anscombe, F. J. 1953. Contribution of discussion paper by H. Hotelling "New light on the correlation coefficient and its transforms". Journal of the Royal Statistical Society, Series B 15: 229-230. https://doi.org/10.1111/j.25176161.1953.tb00136.x.

McCullagh, P., and J. A. Nelder. 1989. Generalized Linear Models. 2nd ed. London: Chapman and Hall/CRC.

## Also see

[R] binreg - Generalized linear models: Extensions to the binomial family
[U] 20 Estimation and postestimation commands

## Title

biprobit - Bivariate probit regression

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

biprobit fits maximum-likelihood two-equation probit models—either a bivariate probit or a seemingly unrelated probit (limited to two equations).

## Quick start

Bivariate probit regression of y 1 and y 2 on x 1
biprobit y1 y2 x1
Bivariate probit regression of y 1 and y 2 on x 1 , x 2 , and x 3
biprobit y1 y2 x1 x2 x3
Constrain the coefficients for x 1 to equality in both equations
constraint define 1 _b[y1:x1] = _b[y2:x1]
biprobit y1 y2 x1 x2 x3, constraints(1)
Seemingly unrelated bivariate probit regression
biprobit ( $\mathrm{y} 1=\mathrm{x} 1 \mathrm{x} 2 \mathrm{x} 3$ ) ( $\mathrm{y} 2=\mathrm{x} 1 \mathrm{x} 2)$
With robust standard errors
biprobit ( $\mathrm{y} 1=\mathrm{x} 1 \mathrm{x} 2 \mathrm{x} 3$ ) ( y 2 = x 1 x 2 ), vce(robust)
Poirier partial observability model with difficult option
biprobit ( $\mathrm{y} 1=\mathrm{x} 1 \mathrm{x} 2$ ) ( $\mathrm{y} 2=\mathrm{x} 2 \mathrm{x} 3)$, partial difficult

## Menu

biprobit
Statistics $>$ Binary outcomes $>$ Bivariate probit regression

## Seemingly unrelated biprobit

Statistics $>$ Binary outcomes $>$ Seemingly unrelated bivariate probit regression

## Syntax

Bivariate probit regression
biprobit depvar ${ }_{1}$ depvar ${ }_{2}$ [indepvars] [if] [in] [weight] [, options]

Seemingly unrelated bivariate probit regression

where equation ${ }_{1}$ and equation ${ }_{2}$ are specified as
([eqname: ] depvar $[=]$ [indepvars] [, noconstant offset(varname)])
options
Model
noconstant
partial
offset1 (varname)
offset2(varname)
constraints (constraints)
SE/Robust
vce (vcetype)

Reporting
level(\#)
lrmodel
nocnsreport
display_options

Description
suppress constant term
fit partial observability model
offset variable for first equation
offset variable for second equation
apply specified linear constraints
vcetype may be oim, robust, cluster clustvar, opg, bootstrap, or jackknife
set confidence level; default is level (95)
perform the likelihood-ratio model test instead of the default Wald test do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

Maximization
maximize_options
collinear
coeflegend
control the maximization process; seldom used
keep collinear variables
display legend instead of statistics

## su_options

Description
Model
partial
constraints (constraints)
SE/Robust
vce (vcetype)

Reporting
level(\#)
lrmodel
nocnsreport
display_options

Maximization
maximize_options
collinear
coeflegend
fit partial observability model
apply specified linear constraints
vcetype may be oim, robust, cluster clustvar, opg, bootstrap, or jackknife
set confidence level; default is level (95)
perform the likelihood-ratio model test instead of the default Wald test do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
control the maximization process; seldom used
keep collinear variables
display legend instead of statistics
indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
depvar ${ }_{1}$, depvar ${ }_{2}$, indepvars, and depvar may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bayes, bootstrap, by, collect, fp, jackknife, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: biprobit.
Weights are not allowed with the bootstrap prefix; see $[R]$ bootstrap.
vce(), lrmodel, and weights are not allowed with the svy prefix; see [SVY] svy.
pweights, fweights, and iweights are allowed; see [U] 11.1.6 weight.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

$\qquad$ Model
noconstant; see [R] Estimation options.
partial specifies that the partial observability model be fit. This particular model commonly has poor convergence properties, so we recommend that you use the difficult option if you want to fit the Poirier partial observability model; see [R] Maximize.

This model computes the product of the two dependent variables so that you do not have to replace each with the product.
offset1(varname), offset2(varname), constraints (constraints); see [R] Estimation options.
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
$\qquad$ Reporting
level(\#), lrmodel, nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fmt), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate(\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from(init_specs); see [R] Maximize. These options are seldom used.

Setting the optimization type to technique (bhhh) resets the default vcetype to vce (opg).
The following options are available with biprobit but are not shown in the dialog box:
collinear, coeflegend; see [R] Estimation options.

## Remarks and examples

For a good introduction to the bivariate probit models, see Greene (2018, sec. 17.9) and Pindyck and Rubinfeld (1998). Poirier (1980) explains the partial observability model. Van de Ven and Van Pragg (1981) explain the probit model with sample selection; see $[R]$ heckprobit for details.

## > Example 1

We use the data from Pindyck and Rubinfeld (1998, 332). In this dataset, the variables are whether children attend private school (private), number of years the family has been at the present residence (years), log of property tax (logptax), log of income (loginc), and whether the head of the household voted for an increase in property taxes (vote).

We wish to model the bivariate outcomes of whether children attend private school and whether the head of the household voted for an increase in property tax based on the other covariates.
. use https://www.stata-press.com/data/r18/school
. biprobit private vote years logptax loginc
Fitting comparison equation 1 :

```
Iteration 0: Log likelihood = -31.967097
Iteration 1: Log likelihood = -31.452424
Iteration 2: Log likelihood = -31.448958
Iteration 3: Log likelihood = -31.448958
Fitting comparison equation 2:
Iteration 0: Log likelihood = -63.036914
Iteration 1: Log likelihood = -58.534843
Iteration 2: Log likelihood = -58.497292
Iteration 3: Log likelihood = -58.497288
Comparison: Log likelihood = -89.946246
Fitting full model:
Iteration 0: Log likelihood = -89.946246
Iteration 1: Log likelihood = -89.258897
Iteration 2: Log likelihood = -89.254028
Iteration 3: Log likelihood = -89.254028
Bivariate probit regression Number of obs = 95
Wald chi2(6) = 9.59
Prob > chi2 = 0.1431
```



The output shows several iteration logs. The first iteration $\log$ corresponds to running the univariate probit model for the first equation, and the second log corresponds to running the univariate probit for the second model. If $\rho=0$, the sum of the log likelihoods from these two models will equal the $\log$ likelihood of the bivariate probit model; this sum is printed in the iteration $\log$ as the comparison log likelihood.

The final iteration $\log$ is for fitting the full bivariate probit model. A likelihood-ratio test of the $\log$ likelihood for this model and the comparison $\log$ likelihood is presented at the end of the output. If we had specified the vce (robust) option, this test would be presented as a Wald test instead of as a likelihood-ratio test.

We could have fit the same model by using the seemingly unrelated syntax as

[^6]
## Stored results

biprobit stores the following in e():
Scalars

```
e(N)
e(k)
e(k_eq)
e(k_aux)
e(k_eq_model)
e(k_dv)
e(df_m)
e(ll)
e(ll_0)
e(ll_c)
e(N_clust)
e(chi2)
e(chi2_c)
e(p)
e(rho)
e(rank)
e(rank0)
e(ic)
e(rc)
e(converged)
```

Macros
e(cmd)
e(cmdline)
e(depvar)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(offset1)
e(offset2)
e(chi2type)
e(chi2_ct)
e(vce)
e(vcetype)
e (opt)
e(which)
e(ml_method)
e(user)
$e(t e c h n i q u e)$
e(properties)
d(predict)
e(marginsok)
e(marginsnotok)
e(asbalanced)
e(asobserved)

Matrices
e(b)
e(Cns)
e(ilog)
e(gradient)
e(V)
e(V_modelbased)

Functions
e(sample)
number of observations
number of parameters
number of equations in $e(b)$
number of auxiliary parameters
number of equations in overall model test
number of dependent variables
model degrees of freedom
log likelihood
log likelihood, constant-only model (lrmodel only)
log likelihood, comparison model
number of clusters
$\chi^{2}$
$\chi^{2}$ for comparison test
$p$-value for model test
$\rho$
rank of $e(V)$
rank of e(V) for constant-only model
number of iterations
return code
1 if converged, 0 otherwise
biprobit
command as typed
names of dependent variables
weight type
weight expression
title in estimation output
name of cluster variable
offset for first equation
offset for second equation
Wald or LR; type of model $\chi^{2}$ test
Wald or LR; type of model $\chi^{2}$ test corresponding to e(chi2_c)
vcetype specified in vce()
title used to label Std. err.
type of optimization
max or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement predict
predictions allowed by margins
predictions disallowed by margins
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
constraints matrix
iteration $\log$ (up to 20 iterations)
gradient vector
variance-covariance matrix of the estimators
model-based variance
marks estimation sample

In addition to the above, the following is stored in r() :
Matrices
$r(t a b l e) \quad$ matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r$ () are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

The $\log$ likelihood, $\ln L$, is given by

$$
\begin{aligned}
\xi_{j}^{\beta} & =x_{j} \beta+\text { offset }_{j}^{\beta} \\
\xi_{j}^{\gamma} & =z_{j} \gamma+\text { offset }_{j}^{\gamma} \\
q_{1 j} & = \begin{cases}1 & \text { if } y_{1 j} \neq 0 \\
-1 & \text { otherwise }\end{cases} \\
q_{2 j} & = \begin{cases}1 & \text { if } y_{2 j} \neq 0 \\
-1 & \text { otherwise }\end{cases} \\
\rho_{j}^{*} & =q_{1 j} q_{2 j} \rho \\
\ln L & =\sum_{j=1}^{n} w_{j} \ln \Phi_{2}\left(q_{1 j} \xi_{j}^{\beta}, q_{2 j} \xi_{j}^{\gamma}, \rho_{j}^{*}\right)
\end{aligned}
$$

where $\Phi_{2}()$ is the cumulative bivariate normal distribution function (with mean $\left[\begin{array}{ll}0 & 0\end{array}\right]^{\prime}$ ) and $w_{j}$ is an optional weight for observation $j$. This derivation assumes that

$$
\begin{aligned}
y_{1 j}^{*} & =x_{j} \beta+\epsilon_{1 j}+\operatorname{offset}_{j}^{\beta} \\
y_{2 j}^{*} & =z_{j} \gamma+\epsilon_{2 j}+\operatorname{offset}_{j}^{\gamma} \\
E\left(\epsilon_{1}\right) & =E\left(\epsilon_{2}\right)=0 \\
\operatorname{Var}\left(\epsilon_{1}\right) & =\operatorname{Var}\left(\epsilon_{2}\right)=1 \\
\operatorname{Cov}\left(\epsilon_{1}, \epsilon_{2}\right) & =\rho
\end{aligned}
$$

where $y_{1 j}^{*}$ and $y_{2 j}^{*}$ are the unobserved latent variables; instead, we observe only $y_{i j}=1$ if $y_{i j}^{*}>0$ and $y_{i j}=0$ otherwise (for $i=1,2$ ).

In the maximum likelihood estimation, $\rho$ is not directly estimated, but atanh $\rho$ is

$$
\operatorname{atanh} \rho=\frac{1}{2} \ln \left(\frac{1+\rho}{1-\rho}\right)
$$

From the form of the likelihood, if $\rho=0$, then the log likelihood for the bivariate probit models is equal to the sum of the $\log$ likelihoods of the two univariate probit models. A likelihood-ratio test may therefore be performed by comparing the likelihood of the full bivariate model with the sum of the log likelihoods for the univariate probit models.

This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce(robust) and vce(cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.
biprobit also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

## References

De Luca, G. 2008. SNP and SML estimation of univariate and bivariate binary-choice models. Stata Journal 8: 190-220.

Greene, W. H. 2018. Econometric Analysis. 8th ed. New York: Pearson.
Heckman, J. J. 1979. Sample selection bias as a specification error. Econometrica 47: 153-161. https://doi.org/10.2307/1912352.

Hernández-Alava, M., and S. Pudney. 2016. bicop: A command for fitting bivariate ordinal regressions with residual dependence characterized by a copula function and normal mixture marginals. Stata Journal 16: 159-184.
Lokshin, M., and Z. Sajaia. 2011. Impact of interventions on discrete outcomes: Maximum likelihood estimation of the binary choice models with binary endogenous regressors. Stata Journal 11: 368-385.
Mullahy, J. 2016. Estimation of multivariate probit models via bivariate probit. Stata Journal 16: 37-51.
Pindyck, R. S., and D. L. Rubinfeld. 1998. Econometric Models and Economic Forecasts. 4th ed. New York: McGraw-Hill.
Poirier, D. J. 1980. Partial observability in bivariate probit models. Journal of Econometrics 12: 209-217. https://doi.org/10.1016/0304-4076(80)90007-X.

Van de Ven, W. P. M. M., and B. M. S. Van Pragg. 1981. The demand for deductibles in private health insurance: A probit model with sample selection. Journal of Econometrics 17: 229-252. https://doi.org/10.1016/0304-4076(81)90028-2.

## Also see

[R] biprobit postestimation - Postestimation tools for biprobit
[ R ] mprobit - Multinomial probit regression
[R] probit - Probit regression
[BAYES] bayes: biprobit - Bayesian bivariate probit regression
[SVY] svy estimation - Estimation commands for survey data
[U] 20 Estimation and postestimation commands

## Title

biprobit postestimation - Postestimation tools for biprobit

Postestimation commands predict margins Also see

## Postestimation commands

The following postestimation commands are available after biprobit:

| Command | Description |
| :---: | :---: |
| contrast | contrasts and ANOVA-style joint tests of estimates |
| estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estat (svy) | postestimation statistics for survey data |
| estimates | cataloging estimation results |
| etable | table of estimation results |
| * hausman | Hausman's specification test |
| lincom | point estimates, standard errors, testing, and inference for linear combinations of coefficients |
| * Irtest | likelihood-ratio test |
| margins | marginal means, predictive margins, marginal effects, and average marginal effects |
| marginsplot | graph the results from margins (profile plots, interaction plots, etc.) |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients |
| predict | probabilities for joint, marginal, and conditional outcomes |
| predictnl | point estimates, standard errors, testing, and inference for generalized predictions |
| pwcompare | pairwise comparisons of estimates |
| suest | seemingly unrelated estimation |
| test | Wald tests of simple and composite linear hypotheses |
| testnl | Wald tests of nonlinear hypotheses |

[^7]
## predict

## Description for predict

predict creates a new variable containing predictions such as probabilities, linear predictions, and standard errors.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

```
predict [type] newvar [if] [in] [, statistic nooffset]
predict [type] stub* [if] [in], scores
```

statistic Description

## Main

p11
$\Phi_{2}\left(\mathbf{x}_{j} \mathbf{b}, \mathbf{z}_{j} \mathbf{g}, \rho\right)$, predicted probability $\operatorname{Pr}\left(y_{1 j}=1, y_{2 j}=1\right)$; the default
p10
$\Phi_{2}\left(\mathbf{x}_{j} \mathbf{b},-\mathbf{z}_{j} \mathbf{g},-\rho\right)$, predicted probability $\operatorname{Pr}\left(y_{1 j}=1, y_{2 j}=0\right)$
$\Phi_{2}\left(-\mathbf{x}_{j} \mathbf{b}, \mathbf{z}_{j} \mathbf{g},-\rho\right)$, predicted probability $\operatorname{Pr}\left(y_{1 j}=0, y_{2 j}=1\right)$
p00
pmarg1
$\Phi_{2}\left(-\mathbf{x}_{j} \mathbf{b},-\mathbf{z}_{j} \mathbf{g}, \rho\right)$, predicted probability $\operatorname{Pr}\left(y_{1 j}=0, y_{2 j}=0\right)$
pmarg2
$\Phi\left(\mathbf{x}_{j} \mathbf{b}\right)$, marginal success probability for equation 1
pcond1
$\Phi\left(\mathbf{z}_{j} \mathbf{g}\right)$, marginal success probability for equation 2
pcond2
xb1
$\Phi_{2}\left(\mathbf{x}_{j} \mathbf{b}, \mathbf{z}_{j} \mathbf{g}, \rho\right) / \Phi\left(\mathbf{z}_{j} \mathbf{g}\right)$, conditional probability of success for equation 1
xb2
$\mathbf{x}_{j} \mathbf{b}$, linear prediction for equation 1
stdp1
$\mathbf{z}_{j} \mathbf{g}$, linear prediction for equation 2
standard error of the linear prediction for equation 1
stdp2
standard error of the linear prediction for equation 2
where $\Phi(\cdot)$ is the standard normal-distribution function and $\Phi_{2}(\cdot)$ is the bivariate standard normal-distribution function.

These statistics are available both in and out of sample; type predict ... if e(sample)... if wanted only for the estimation sample.

## Options for predict

p11, the default, calculates the bivariate predicted probability $\operatorname{Pr}\left(y_{1 j}=1, y_{2 j}=1\right)$.
p10 calculates the bivariate predicted probability $\operatorname{Pr}\left(y_{1 j}=1, y_{2 j}=0\right)$.
p01 calculates the bivariate predicted probability $\operatorname{Pr}\left(y_{1 j}=0, y_{2 j}=1\right)$.
p00 calculates the bivariate predicted probability $\operatorname{Pr}\left(y_{1 j}=0, y_{2 j}=0\right)$.
pmarg1 calculates the univariate (marginal) predicted probability of success $\operatorname{Pr}\left(y_{1 j}=1\right)$. pmarg2 calculates the univariate (marginal) predicted probability of success $\operatorname{Pr}\left(y_{2 j}=1\right)$. pcond1 calculates the conditional (on success in equation 2 ) predicted probability of success $\operatorname{Pr}\left(y_{1 j}=1, y_{2 j}=1\right) / \operatorname{Pr}\left(y_{2 j}=1\right)$.
pcond2 calculates the conditional (on success in equation 1) predicted probability of success
$\operatorname{Pr}\left(y_{1 j}=1, y_{2 j}=1\right) / \operatorname{Pr}\left(y_{1 j}=1\right)$.
xb 1 calculates the probit linear prediction $\mathbf{x}_{j} \mathbf{b}$.
xb 2 calculates the probit linear prediction $\mathbf{z}_{j} \mathbf{g}$.
stdp1 calculates the standard error of the linear prediction for equation 1.
stdp2 calculates the standard error of the linear prediction for equation 2.
nooffset is relevant only if you specified offset1 (varname) or offset2 (varname) for biprobit. It modifies the calculations made by predict so that they ignore the offset variables; the linear predictions are treated as $\mathbf{x}_{j} \mathbf{b}$ rather than as $\mathbf{x}_{j} \mathbf{b}+\operatorname{offset}_{1 \mathrm{j}}$ and $\mathbf{z}_{j} \gamma$ rather than as $\mathbf{z}_{j} \gamma+$ offset $_{2 j}$. scores calculates equation-level score variables.

The first new variable will contain $\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)$.
The second new variable will contain $\partial \ln L / \partial\left(\mathbf{z}_{j} \gamma\right)$.
The third new variable will contain $\partial \ln L / \partial(\operatorname{atanh} \rho)$.

## margins

## Description for margins

margins estimates margins of response for probabilities and linear predictions.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...)...] [options]
```


## statistic Description

p11 $\quad \Phi_{2}\left(\mathbf{x}_{j} \mathbf{b}, \mathbf{z}_{j} \mathbf{g}, \rho\right)$, predicted probability $\operatorname{Pr}\left(y_{1 j}=1, y_{2 j}=1\right)$; the default
p10 $\Phi_{2}\left(\mathbf{x}_{j} \mathbf{b},-\mathbf{z}_{j} \mathbf{g},-\rho\right)$, predicted probability $\operatorname{Pr}\left(y_{1 j}=1, y_{2 j}=0\right)$
p01 $\quad \Phi_{2}\left(-\mathbf{x}_{j} \mathbf{b}, \mathbf{z}_{j} \mathbf{g},-\rho\right)$, predicted probability $\operatorname{Pr}\left(y_{1 j}=0, y_{2 j}=1\right)$
p00 $\quad \Phi_{2}\left(-\mathbf{x}_{j} \mathbf{b},-\mathbf{z}_{j} \mathbf{g}, \rho\right)$, predicted probability $\operatorname{Pr}\left(y_{1 j}=0, y_{2 j}=0\right)$
pmarg1 $\Phi\left(\mathbf{x}_{j} \mathbf{b}\right)$, marginal success probability for equation 1
pmarg2 $\quad \Phi\left(\mathbf{z}_{j} \mathbf{g}\right)$, marginal success probability for equation 2
pcond1 $\quad \Phi_{2}\left(\mathbf{x}_{j} \mathbf{b}, \mathbf{z}_{j} \mathbf{g}, \rho\right) / \Phi\left(\mathbf{z}_{j} \mathbf{g}\right)$, conditional probability of success for equation 1
pcond2 $\quad \Phi_{2}\left(\mathbf{x}_{j} \mathbf{b}, \mathbf{z}_{j} \mathbf{g}, \rho\right) / \Phi\left(\mathbf{x}_{j} \mathbf{b}\right)$, conditional probability of success for equation 2
xb1
$\mathbf{x}_{j} \mathbf{b}$, linear prediction for equation 1
$\mathrm{xb} 2 \quad \mathbf{z}_{j} \mathbf{g}$, linear prediction for equation 2
stdp1 not allowed with margins
stdp2 not allowed with margins

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see $[R]$ margins.

## Also see

[R] biprobit - Bivariate probit regression
[U] 20 Estimation and postestimation commands

## Title

bitest - Binomial probability test

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Option | Remarks and examples | Stored results | Methods and formulas |
| Reference | Also see |  |  |

## Description

bitest performs exact hypothesis tests for binomial random variables. The null hypothesis is that the probability of a success on a trial is $\#_{p}$. The total number of trials is the number of nonmissing values of varname (in bitest) or $\#_{N}$ (in bitesti). The number of observed successes is the number of 1 s in varname (in bitest) or $\#_{\text {succ }}$ (in bitesti). varname must contain only $0 \mathrm{~s}, 1 \mathrm{~s}$, and missing.
bitesti is the immediate form of bitest; see [U] 19 Immediate commands for a general introduction to immediate commands.

## Quick start

Exact test for probability of success $(a=1)$ is 0.4
bitest a = . 4
With additional exact probabilities
bitest a = .4, detail
Exact test that the probability of success is 0.46 , given 22 successes in 74 trials bitesti 7422.46

## Menu

## bitest

Statistics $>$ Summaries, tables, and tests $>$ Classical tests of hypotheses $>$ Binomial probability test

## bitesti

Statistics $>$ Summaries, tables, and tests $>$ Classical tests of hypotheses $>$ Binomial probability test calculator

## Syntax

Binomial probability test
bitest varname $=\#_{p}$ [if] [in] [weight] [, detail]

Immediate form of binomial probability test
bitesti $\#_{N} \#_{\text {succ }} \#_{p}[$, detail $]$
by and collect are allowed with bitest; see [U] 11.1.10 Prefix commands.
fweights are allowed with bitest; see [U] 11.1.6 weight.

## Option

## Remarks and examples

Remarks are presented under the following headings:
bitest
bitesti

## bitest

> Example 1
We test 15 university students for high levels of one measure of visual quickness which, from other evidence, we believe is present in $30 \%$ of the nonuniversity population. Included in our data is quick, taking on the values 1 ("success") or 0 ("failure") depending on the outcome of the test.

```
. use https://www.stata-press.com/data/r18/quick
. bitest quick == 0.3
Binomial probability test
\begin{tabular}{r|rrrrr} 
Variable & N & Observed \(k\) & Expected \(k\) & Assumed p & Observed p \\
\hline quick & 15 & 7 & 4.5 & 0.30000 & 0.46667
\end{tabular}
    Pr(k >= 7) = 0.131143 (one-sided test)
    Pr}(\textrm{k}<=7)=0.949987 (one-sided test
    Pr}(\textrm{k}<=1\mathrm{ or k >= 7) = 0.166410 (two-sided test)
```

The first part of the output reveals that, assuming a true probability of success of 0.3 , the expected number of successes is 4.5 and that we observed seven. Said differently, the assumed frequency under the null hypothesis $H_{0}$ is 0.3 , and the observed frequency is 0.47 .

The first line under the table is a one-sided test; it is the probability of observing seven or more successes conditional on $p=0.3$. It is a test of $H_{0}: p=0.3$ versus the alternative hypothesis $H_{\mathrm{A}}: p>0.3$. Said in English, the alternative hypothesis is that more than $30 \%$ of university students score at high levels on this test of visual quickness. The $p$-value for this hypothesis test is 0.13 .

The second line under the table is a one-sided test of $H_{0}$ versus the opposite alternative hypothesis $H_{\mathrm{A}}: p<0.3$.

The third line is the two-sided test. It is a test of $H_{0}$ versus the alternative hypothesis $H_{\mathrm{A}}: p \neq 0.3$.

## Technical note

The $p$-value of a hypothesis test is the probability (calculated assuming $H_{0}$ is true) of observing any outcome as extreme or more extreme than the observed outcome, with extreme meaning in the direction of the alternative hypothesis. In example 1 , the outcomes $k=8,9, \ldots, 15$ are clearly "more extreme" than the observed outcome $k_{\text {obs }}=7$ when considering the alternative hypothesis $H_{\mathrm{A}}: p \neq 0.3$. However, outcomes with only a few successes are also in the direction of this alternative hypothesis. For two-sided hypotheses, outcomes with $k$ successes are considered "as extreme or more extreme" than the observed outcome $k_{\text {obs }}$ if $\operatorname{Pr}(k) \leq \operatorname{Pr}\left(k_{\text {obs }}\right)$. Here $\operatorname{Pr}(k=0)$ and $\operatorname{Pr}(k=1)$ are both less than $\operatorname{Pr}(k=7)$, so they are included in the two-sided $p$-value.

The detail option allows you to see the probability (assuming that $H_{0}$ is true) of the observed successes $(k=7)$ and the probability of the boundary point $(k=1)$ of the opposite tail used for the two-sided $p$-value.

```
. bitest quick == 0.3, detail
Binomial probability test
\begin{tabular}{|c|c|c|c|c|}
\hline Variable & N Obse & ved k Expected k & Assumed p & Observed p \\
\hline quick & 15 & \(7 \quad 4.5\) & 0.30000 & 0.46667 \\
\hline \(\operatorname{Pr}(\mathrm{k}>=7)\) & \(=0.131143\) & (one-sided test) & & \\
\hline \(\operatorname{Pr}(\mathrm{k}<=7)\) & \(=0.949987\) & (one-sided test) & & \\
\hline \(\operatorname{Pr}(\mathrm{k}<=1\) or & \(=0.166410\) & (two-sided test) & & \\
\hline \(\operatorname{Pr}(\mathrm{k}==7\) ) & \(=0.081130\) & (observed) & & \\
\hline \(\operatorname{Pr}(\mathrm{k}==2)\) & \(=0.091560\) & & & \\
\hline \(\operatorname{Pr}(\mathrm{k}==1)\) & \(=0.030520\) & (opposite extreme) & & \\
\hline
\end{tabular}
```

Also shown is the probability of the point next to the boundary point. This probability, namely, $\operatorname{Pr}(k=2)=0.092$, is certainly close to the probability of the observed outcome $\operatorname{Pr}(k=7)=0.081$, so some people might argue that $k=2$ should be included in the two-sided $p$-value. Statisticians (at least some we know) would reply that the $p$-value is a precisely defined concept and that this is an arbitrary "fuzzification" of its definition. When you compute exact $p$-values according to the precise definition of a $p$-value, your type I error is never more than what you say it is - so no one can criticize you for being anticonservative. Including the point $k=2$ is being overly conservative because it makes the $p$-value larger yet. But it is your choice; being overly conservative, at least in statistics, is always safe. Know that bitest and bitesti always keep to the precise definition of a $p$-value, so if you wish to include this extra point, you must do so by hand or by using the $r()$ stored results; see Stored results below.

## bitesti

## Example 2

The binomial test is a function of two statistics and one parameter: $N$, the number of observations; $k_{\text {obs }}$, the number of observed successes; and $p$, the assumed probability of a success on a trial. For instance, in a city of $N=2,500,000$, we observe $k_{\text {obs }}=36$ cases of a particular disease when the population rate for the disease is $p=0.00001$.

| N | Observed k | Expected |  | Assumed p | Observed p |
| :---: | :---: | :---: | :---: | :---: | :---: |
| 2,500,000 | 36 |  | 25 | 0.00001 | 0.00001 |
| $\operatorname{Pr}(\mathrm{k}>=36)$ |  | 0.022458 |  | -sided te |  |
| $\operatorname{Pr}(\mathrm{k}<=36)$ | $=$ | 0.985448 |  | -sided test) |  |
| $\operatorname{Pr}(\mathrm{k}<=14$ or | $\mathrm{k}>=36)=$ | 0.034859 |  | -sided tes |  |

## Stored results

bitest and bitesti store the following in $r()$ :
Scalars

```
r(N) number }N\mathrm{ of trials
    r(P_p) assumed probability p of success
    r(k) observed number k of successes
    r(p_l) lower one-sided p-value
    r(p_u) upper one-sided p-value
    r(p) two-sided p-value
    r(k_opp) opposite extreme k
    r(P_k) probability of observed k (detail only)
    r(P_oppk) probability of opposite extreme k (detail only)
    r(k_nopp) b next to opposite extreme (detail only)
    r(P_noppk) probability of k next to opposite extreme (detail only)
```


## Methods and formulas

Let $N, k_{\text {obs }}$, and $p$ be, respectively, the number of observations, the observed number of successes, and the assumed probability of success on a trial. The expected number of successes is $N p$, and the observed probability of success on a trial is $k_{\text {obs }} / N$.
bitest and bitesti compute exact $p$-values based on the binomial distribution. The upper one-sided $p$-value is

$$
\operatorname{Pr}\left(k \geq k_{\mathrm{obs}}\right)=\sum_{m=k_{\mathrm{obs}}}^{N}\binom{N}{m} p^{m}(1-p)^{N-m}
$$

The lower one-sided $p$-value is

$$
\operatorname{Pr}\left(k \leq k_{\mathrm{obs}}\right)=\sum_{m=0}^{k_{\mathrm{obs}}}\binom{N}{m} p^{m}(1-p)^{N-m}
$$

If $k_{\text {obs }} \geq N p$, the two-sided $p$-value is

$$
\operatorname{Pr}\left(k \leq k_{\text {opp }} \text { or } k \geq k_{\text {obs }}\right)
$$

where $k_{\text {opp }}$ is the largest number $\leq N p$ such that $\operatorname{Pr}\left(k=k_{\text {opp }}\right) \leq \operatorname{Pr}\left(k=k_{\text {obs }}\right)$. If $k_{\text {obs }}<N p$, the two-sided $p$-value is

$$
\operatorname{Pr}\left(k \leq k_{\mathrm{obs}} \text { or } k \geq k_{\mathrm{opp}}\right)
$$

where $k_{\text {opp }}$ is the smallest number $\geq N p$ such that $\operatorname{Pr}\left(k=k_{\text {opp }}\right) \leq \operatorname{Pr}\left(k=k_{\text {obs }}\right)$.

## Reference

Hoel, P. G. 1984. Introduction to Mathematical Statistics. 5th ed. New York: Wiley.

## Also see

$[\mathrm{R}]$ ci - Confidence intervals for means, proportions, and variances
[R] prtest - Tests of proportions

## Title

bootstrap - Bootstrap sampling and estimation

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

bootstrap performs nonparametric bootstrap estimation of specified statistics (or expressions) for a Stata command or a user-written program. Statistics are bootstrapped by resampling the data in memory with replacement. bootstrap is designed for use with nonestimation commands, functions of coefficients, or user-written programs. To bootstrap coefficients, we recommend using the vce(bootstrap) option when allowed by the estimation command.

## Quick start

Bootstrap the mean of v1 returned by summarize in $r$ (mean) bootstrap mean=r(mean): summarize v1

Bootstrap the statistic $r$ (mystat) returned by program myprog1 bootstrap stat=r(mystat): myprog1 v1

Same as above, but use 100 replications
bootstrap stat=r(mystat), reps(100): myprog1 v1
Same as above, and save the results from each replication in mydata.dta bootstrap stat=r(mystat), reps(100) saving(mydata): myprog1 v1

Bootstrap a difference in coefficients estimated by regress

```
bootstrap diff=(_b[x2]-_b[x1]): regress y x1 x2 x3
```

Bootstrap the coefficients stored in e(b) by myprog2

```
bootstrap _b: myprog2 y x1 x2 x3
```

Same as above, but with bootstrap samples taken independently within strata identified by svar bootstrap _b, strata(svar): myprog2 y x1 x2 x3

Resample clusters defined by cvar and create newcvar identifying resampled clusters bootstrap _b, cluster(cvar) idcluster(newcvar): myprog2 y x1 x2 x3

## Menu

Statistics $>$ Resampling $>$ Bootstrap estimation

## Syntax

bootstrap exp_list [, options eform_option ] : command
options
Main
reps(\#)
Options
strata (varlist)
size(\#)
cluster (varlist)
idcluster(newvar)
saving (filename, ...)
bca
ties
mse
Reporting
level(\#)
notable
noheader
nolegend
verbose
nodots
dots(\#)
noisily
trace
title(text)
display_options
eform_option
Advanced nodrop
nowarn
force
reject (exp)
seed (\#)
group(varname)
jackknifeopts(jkopts)
coeflegend

Description
perform \# bootstrap replications; default is reps (50)
variables identifying strata
draw samples of size \#; default is $\_\mathrm{N}$
variables identifying resampling clusters
create new cluster ID variable
save results to filename; save statistics in double precision;
save results to filename every \# replications
compute acceleration for $\mathrm{BC}_{a}$ confidence intervals
adjust $\mathrm{BC} / \mathrm{BCa}$ confidence intervals for ties
use MSE formula for variance estimation
set confidence level; default is level (95)
suppress table of results
suppress table header suppress table legend
display the full table legend
suppress replication dots
display dots every \# replications
display any output from command
trace command
use text as title for bootstrap results
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
display coefficient table in exponentiated form
do not drop observations
do not warn when e(sample) is not set
do not check for weights or svy commands; seldom used identify invalid results
set random-number seed to \#

ID variable for groups within cluster ()
options for jackknife; see [R] jackknife
display legend instead of statistics
command is any command that follows standard Stata syntax. weights are not allowed in command. collect and svy are allowed; see [U] 11.1.10 Prefix commands. group(), jackknifeopts(), and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

| exp_list contains | (name: elist) elist |
| :---: | :---: |
|  | eexp |
| elist contains | $\begin{aligned} & \text { newvar }=(\exp ) \\ & (\text { exp }) \end{aligned}$ |
| eexp is | specname |
|  | [eqno]specname |
| specname is | _b |
|  | _b [] |
|  | _se |
|  | _se [] |
| eqno is | \# \# |
|  | name |

$\exp$ is a standard Stata expression; see [U] 13 Functions and expressions.

Distinguish between [], which are to be typed, and [], which indicate optional arguments.

## Options

$\qquad$ Main
reps (\#) specifies the number of bootstrap replications to be performed. The default is 50. A total of $50-200$ replications are generally adequate for estimates of standard error and thus are adequate for normal-approximation confidence intervals; see Mooney and Duval (1993, 11). Estimates of confidence intervals using the percentile or bias-corrected methods typically require 1,000 or more replications.

## Options

strata (varlist) specifies the variables that identify strata. If this option is specified, bootstrap samples are taken independently within each stratum.
size(\#) specifies the size of the samples to be drawn. The default is $\_$N, meaning to draw samples of the same size as the data. If specified, \# must be less than or equal to the number of observations within strata().

If cluster() is specified, the default size is the number of clusters in the original dataset. For unbalanced clusters, resulting sample sizes will differ from replication to replication. For cluster sampling, \# must be less than or equal to the number of clusters within strata().
cluster (varlist) specifies the variables that identify resampling clusters. If this option is specified, the sample drawn during each replication is a bootstrap sample of clusters.
idcluster (newvar) creates a new variable containing a unique identifier for each resampled cluster.
This option requires that cluster() also be specified.
saving ( filename [, suboptions]) creates a Stata data file (.dta file) consisting of (for each statistic in exp_list) a variable containing the replicates.
double specifies that the results for each replication be saved as doubles, meaning 8-byte reals. By default, they are saved as floats, meaning 4-byte reals. This option may be used without the saving() option to compute the variance estimates by using double precision.
every (\#) specifies that results be written to disk every \#th replication. every () should be specified only in conjunction with saving () when command takes a long time for each replication. This option will allow recovery of partial results should some other software crash your computer. See [ P$]$ postfile.
replace specifies that filename be overwritten if it exists. This option does not appear in the dialog box.
bca specifies that bootstrap estimate the acceleration of each statistic in exp_list. This estimate is used to construct $\mathrm{BC}_{a}$ confidence intervals. Type estat bootstrap, bca to display the $\mathrm{BC}_{a}$ confidence interval generated by the bootstrap command.
ties specifies that bootstrap adjust for ties in the replicate values when computing the median bias used to construct BC and BCa confidence intervals.
mse specifies that bootstrap compute the variance by using deviations of the replicates from the observed value of the statistics based on the entire dataset. By default, bootstrap computes the variance by using deviations from the average of the replicates.

## Reporting

level(\#); see [R] Estimation options.
notable suppresses the display of the table of results.
noheader suppresses the display of the table header. This option implies nolegend. This option may also be specified when replaying estimation results.
nolegend suppresses the display of the table legend. This option may also be specified when replaying estimation results.
verbose specifies that the full table legend be displayed. By default, coefficients and standard errors are not displayed. This option may also be specified when replaying estimation results.
nodots and dots(\#) specify whether to display replication dots. By default, one dot character is displayed for each successful replication. An " x " is displayed if command returns an error or if any value in exp_list is missing. You can also control whether dots are displayed using set dots; see [R] set.
nodots suppresses display of the replication dots.
dots (\#) displays dots every \# replications. dots (0) is a synonym for nodots.
noisily specifies that any output from command be displayed. This option implies the nodots option.
trace causes a trace of the execution of command to be displayed. This option implies the noisily option.
title (text) specifies a title to be displayed above the table of bootstrap results. The default title is the title stored in e(title) by an estimation command, or if e(title) is not filled in, Bootstrap results is used. title() may also be specified when replaying estimation results.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% $\% m t$ ), and nolstretch; see [R] Estimation options.
eform_option causes the coefficient table to be displayed in exponentiated form; see $[\mathrm{R}]$ eform_option.
command determines which of the following are allowed (eform(string) and eform are always allowed):

| eform_option | Description |
| :--- | :--- |
| eform (string) | use string for the column title |
| eform | exponentiated coefficient, string is exp (b) |
| hr | hazard ratio, string is Haz. ratio |
| shr | subhazard ratio, string is SHR |
| $\underline{\mathrm{irr}}$ | incidence-rate ratio, string is IRR <br> or |
| $\underline{\mathrm{rrr}}$ | odds ratio, string is Odds ratio <br> relative-risk ratio, string is RRR |

Advanced
nodrop prevents observations outside e(sample) and the if and in qualifiers from being dropped before the data are resampled.
nowarn suppresses the display of a warning message when command does not set e(sample).
force suppresses the restriction that command not specify weights or be a svy command. This is a rarely used option. Use it only if you know what you are doing.
reject (exp) identifies an expression that indicates when results should be rejected. When exp is true, the resulting values are reset to missing values.
seed(\#) sets the random-number seed. Specifying this option is equivalent to typing the following command prior to calling bootstrap:
. set seed \#
The following options are available with bootstrap but are not shown in the dialog box:
group (varname) re-creates varname containing a unique identifier for each group across the resampled clusters. This option requires that idcluster() also be specified.
This option is useful for maintaining unique group identifiers when sampling clusters with replacement. Suppose that cluster 1 contains 3 groups. If the idcluster (newclid) option is specified and cluster 1 is sampled multiple times, newclid uniquely identifies each copy of cluster 1 . If group (newgroupid) is also specified, newgroupid uniquely identifies each copy of each group.
jackknifeopts (jkopts) identifies options that are to be passed to jackknife when it computes the acceleration values for the $\mathrm{BC}_{a}$ confidence intervals; see [R] jackknife. This option requires the bca option and is mostly used for passing the eclass, rclass, or n(\#) option to jackknife.
coeflegend; see [R] Estimation options.

## Remarks and examples

Remarks are presented under the following headings:

## Introduction

Using bootstrap
Regression coefficients
Expressions
Combining bootstrap datasets
A note about macros
Achieved significance level
Bootstrapping a ratio
Warning messages and e(sample)
Bootstrapping statistics from data with a complex structure

## Introduction

With few assumptions, bootstrapping provides a way of estimating standard errors and other measures of statistical precision (Efron 1979; Efron and Stein 1981; Efron 1982; Efron and Tibshirani 1986; Efron and Tibshirani 1993; also see Davison and Hinkley [1997]; Guan [2003]; Mooney and Duval [1993]; Poi [2004]; and Stine [1990]). It provides a way to obtain such measures when no formula is otherwise available or when available formulas make inappropriate assumptions. Cameron and Trivedi (2022, chap. 12) discuss many bootstrapping topics and demonstrate how to do them in Stata.

To illustrate bootstrapping, suppose that you have a dataset containing $N$ observations and an estimator that, when applied to the data, produces certain statistics. You draw, with replacement, $N$ observations from the $N$-observation dataset. In this random drawing, some of the original observations will appear once, some more than once, and some not at all. Using the resampled dataset, you apply the estimator and collect the statistics. This process is repeated many times; each time, a new random sample is drawn and the statistics are recalculated.

This process builds a dataset of replicated statistics. From these data, you can calculate the standard error by using the standard formula for the sample standard deviation

$$
\widehat{\mathrm{se}}=\left\{\frac{1}{k-1} \sum\left(\widehat{\theta}_{i}-\bar{\theta}\right)^{2}\right\}^{1 / 2}
$$

where $\widehat{\theta}_{i}$ is the statistic calculated using the $i$ th bootstrap sample and $k$ is the number of replications. This formula gives an estimate of the standard error of the statistic, according to Hall and Wilson (1991). Although the average, $\bar{\theta}$, of the bootstrapped estimates is used in calculating the standard deviation, it is not used as the estimated value of the statistic itself. Instead, the original observed value of the statistic, $\widehat{\theta}$, is used, meaning the value of the statistic computed using the original $N$ observations.

You might think that $\bar{\theta}$ is a better estimate of the parameter than $\widehat{\theta}$, but it is not. If the statistic is biased, bootstrapping exaggerates the bias. In fact, the bias can be estimated as $\bar{\theta}-\widehat{\theta}$ (Efron 1982, 33). Knowing this, you might be tempted to subtract this estimate of bias from $\widehat{\theta}$ to produce an unbiased statistic. The bootstrap bias estimate has an indeterminate amount of random error, so this unbiased estimator may have greater mean squared error than the biased estimator (Mooney and Duval 1993; Hinkley 1978). Thus, $\widehat{\theta}$ is the best point estimate of the statistic.

The logic behind the bootstrap is that all measures of precision come from a statistic's sampling distribution. When the statistic is estimated on a sample of size $N$ from some population, the sampling distribution tells you the relative frequencies of the values of the statistic. The sampling distribution, in turn, is determined by the distribution of the population and the formula used to estimate the statistic.

Sometimes the sampling distribution can be derived analytically. For instance, if the underlying population is distributed normally and you calculate means, the sampling distribution for the mean is also normal but has a smaller variance than that of the population. In other cases, deriving the sampling distribution is difficult, such as when means are calculated from nonnormal populations. Sometimes, as in the case of means, it is not too difficult to derive the sampling distribution as the sample size goes to infinity $(N \rightarrow \infty)$. However, such asymptotic distributions may not perform well when applied to finite samples.

If you knew the population distribution, you could obtain the sampling distribution by simulation: you could draw random samples of size $N$, calculate the statistic, and make a tally. Bootstrapping does precisely this, but it uses the observed distribution of the sample in place of the true population distribution. Thus, the bootstrap procedure hinges on the assumption that the observed distribution is a good estimate of the underlying population distribution. In return, the bootstrap produces an estimate, called the bootstrap distribution, of the sampling distribution. From this, you can estimate the standard error of the statistic, produce confidence intervals, etc.

The accuracy with which the bootstrap distribution estimates the sampling distribution depends on the number of observations in the original sample and the number of replications in the bootstrap. A crudely estimated sampling distribution is adequate if you are going to extract, say, only a standard error. A better estimate is needed if you want to use the 2.5 th and 97.5 th percentiles of the distribution to produce a $95 \%$ confidence interval. To extract many features simultaneously about the distribution, an even better estimate is needed. Generally, replications on the order of 1,000 produce very good estimates, but only $50-200$ replications are needed for estimates of standard errors. See Poi (2004) for a method to choose the number of bootstrap replications.

## Using bootstrap

## Typing

. bootstrap exp_list, reps(\#): command
executes command multiple times, bootstrapping the statistics in exp_list by resampling observations (with replacement) from the data in memory \# times. This method is commonly referred to as the nonparametric bootstrap.
command defines the statistical command to be executed. Most Stata commands and user-written programs can be used with bootstrap, as long as they follow standard Stata syntax; see [U] 11 Language syntax. If the bca option is supplied, command must also work with jackknife; see [R] jackknife. The by prefix may not be part of command.
exp_list specifies the statistics to be collected from the execution of command. If command changes the contents in e(b), exp_list is optional and defaults to _b.

Because bootstrapping is a random process, if you want to be able to reproduce results, set the random-number seed by specifying the seed(\#) option or by typing
. set seed \#
where \# is a seed of your choosing, before running bootstrap; see [R] set seed.

## Regression coefficients

## - Example 1

Let's say that we wish to compute bootstrap estimates for the standard errors of the coefficients from the following regression:

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. regress mpg weight gear foreign
```

| Source | SS | df | MS |
| ---: | :---: | ---: | :---: |
| Model | 1629.67805 | 3 | 543.226016 |
| Residual | 813.781411 | 70 | 11.6254487 |
| Total | 2443.45946 | 73 | 33.4720474 |


| Number of obs | $=$ | 74 |
| :--- | :--- | ---: |
| $\mathrm{~F}(3,70)$ | $=$ | 46.73 |
| Prob > F | $=$ | 0.0000 |
| R-squared | $=$ | 0.6670 |
| Adj R-squared | $=$ | 0.6527 |
| Root MSE | $=$ | 3.4096 |


| mpg | Coefficient | Std. err. | t | P>\|t| | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | ---: | ---: |
| weight | -.006139 | .0007949 | -7.72 | 0.000 | -.0077245 | -.0045536 |
| gear_ratio | 1.457113 | 1.541286 | 0.95 | 0.348 | -1.616884 | 4.53111 |
| foreign | -2.221682 | 1.234961 | -1.80 | 0.076 | -4.684735 | .2413715 |
| _cons | 36.10135 | 6.285984 | 5.74 | 0.000 | 23.56435 | 48.63835 |

To run the bootstrap, we simply prefix the above regression command with the bootstrap command (specifying its options before the colon separator). We must set the random-number seed before calling bootstrap.

| . bootstrap, reps(100) seed(1): regress mpg weight gear foreign (running regress on estimation sample) |  |  |
| :---: | :---: | :---: |
|  |  |  |
|  |  |  |
| Linear regression | Number of obs | $=74$ |
|  | Replications | $=100$ |
|  | Wald chi2(3) | $=167.13$ |
|  | Prob > chi2 | $=0.0000$ |
|  | R -squared | $=0.6670$ |
|  | Adj R-squared | $=0.6527$ |
|  | Root MSE | $=3.4096$ |


|  | Observed <br> mpg <br> coefficient | Bootstrap <br> std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ |  | Normal-based <br> [95\% conf. interval] |  |
| ---: | ---: | ---: | :---: | ---: | ---: | ---: | :---: |
| weight | -.006139 | .0006063 | -10.13 | 0.000 | -.0073273 | -.0049507 |  |
| gear_ratio | 1.457113 | 1.367917 | 1.07 | 0.287 | -1.223954 | 4.138181 |  |
| foreign | -2.221682 | 1.169727 | -1.90 | 0.058 | -4.514305 | .070942 |  |
| _cons | 36.10135 | 5.20581 | 6.93 | 0.000 | 25.89815 | 46.30455 |  |

The displayed confidence interval is based on the assumption that the sampling (and hence bootstrap) distribution is approximately normal (see Methods and formulas below). Because this confidence interval is based on the standard error, it is a reasonable estimate if normality is approximately true, even for a few replications. Other types of confidence intervals are available after bootstrap; see $[R]$ bootstrap postestimation.

We could instead supply names to our expressions when we run bootstrap. For example,

```
. bootstrap diff=(_b[weight]-_b[gear]): regress mpg weight gear foreign
```

would bootstrap a statistic, named diff, equal to the difference between the coefficients on weight and gear_ratio.

## - Technical note

regress, like many estimation commands, allows the vce (bootstrap) option. For any estimation command that allows this option, we recommend using vce (bootstrap) over bootstrap because the estimation command automatically handles clustering and other model-specific details for you.

## Expressions

## > Example 2

When we use bootstrap, the list of statistics can contain complex expressions, as long as each expression is enclosed in parentheses. For example, to bootstrap the range of a variable $x$, we could type

```
. bootstrap range=(r(max)-r(min)), reps(1000): summarize x
```

Of course, we could also bootstrap the minimum and maximum and later compute the range.

```
. bootstrap max=r(max) min=r(min), reps(1000) saving(mybs): summarize x
. use mybs, clear
(bootstrap: summarize)
. generate range = max - min
. bstat range, stat(19.5637501)
```

The difference between the maximum and minimum of x in the sample is 19.5637501 .
The stat () option to bstat specifies the observed value of the statistic (range) to be summarized. This option is useful when, as shown above, the statistic of ultimate interest is not specified directly to bootstrap but instead is calculated by other means.

Here the observed values of $r(\max )$ and $r(\min )$ are stored as characteristics of the dataset created by bootstrap and are thus available for retrieval by bstat; see [R] bstat. The observed range, however, is unknown to bstat, so it must be specified.

## Combining bootstrap datasets

You can combine two datasets from separate runs of bootstrap by using append (see [D] append) and then get the bootstrap statistics for the combined datasets by running bstat. The runs must have been performed independently (having different starting random-number seeds), and the original dataset, command, and bootstrap statistics must have been all the same.

## A note about macros

In example 2, we executed the command

```
. bootstrap max=r(max) min=r(min), reps(1000) saving(mybs): summarize x
```

We did not enclose $r(\max )$ and $r(\min )$ in single quotes, as we would in most other contexts, because it would not produce what was intended:

```
. bootstrap 'r(max)' 'r(min)', reps(1000) saving(mybs): summarize x
```

To understand why, note that ' $r(\max )$ ', like any reference to a local macro, will evaluate to a literal string containing the contents of $r(\max )$ before bootstrap is even executed. Typing the command above would appear to Stata as if we had typed

```
. bootstrap 14.5441234 33.4393293, reps(1000) saving(mybs): summarize x
```

Even worse, the current contents of $r(\min )$ and $r(\max )$ could be empty, producing an even more confusing result. To avoid this outcome, refer to statistics by name (for example, $r(\max )$ ) and not by value (for example, ' $r(\max )$ ').

## Achieved significance level

## > Example 3

Suppose that we wish to estimate the achieved significance level (ASL) of a test statistic by using the bootstrap. ASL is another name for $p$-value. An example is

$$
\mathrm{ASL}=\operatorname{Pr}\left(\widehat{\theta}^{*} \geq \widehat{\theta} \mid H_{0}\right)
$$

for an upper-tailed, alternative hypothesis, where $H_{0}$ denotes the null hypothesis, $\widehat{\theta}$ is the observed value of the test statistic, and $\widehat{\theta}^{*}$ is the random variable corresponding to the test statistic, assuming that $H_{0}$ is true.

Here we will compare the mean miles per gallon ( mpg ) between foreign and domestic cars by using the two-sample $t$ test with unequal variances. The following results indicate the $p$-value to be 0.0034 for the two-sided test using Satterthwaite's approximation. Thus, assuming that mean mpg is the same for foreign and domestic cars, we would expect to observe a $t$ statistic more extreme (in absolute value) than 3.1797 in about $0.3 \%$ of all possible samples of the type that we observed. Thus, we have evidence to reject the null hypothesis that the means are equal.

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. ttest mpg, by(foreign) unequal
Two-sample \(t\) test with unequal variances
```

| Group | Obs | Mean | Std. err | Std. dev. | [95\% conf | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Domestic | 52 | 19.82692 | . 657777 | 4.743297 | 18.50638 | 21.14747 |
| Foreign | 22 | 24.77273 | 1.40951 | 6.611187 | 21.84149 | 27.70396 |
| Combined | 74 | 21.2973 | . 6725511 | 5.785503 | 19.9569 | 22.63769 |
| diff |  | -4.945804 | 1.555438 |  | -8.120053 | -1.771556 |
| diff $=$ mean(Domestic) - mean(Foreign) $t=-3.1797$ <br> H0: $\operatorname{diff}=0$ Satterthwaite's degrees of freedom $=30.5463$  <br> Ha: diff < 0 Ha: diff $!=0$ $H a: \operatorname{diff}>0$ <br> $\operatorname{Pr}(T<t)=0.0017$ $\operatorname{Pr}(\|T\|>\|t\|)=0.0034$ $\operatorname{Pr}(T>t)=0.9983$ |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

We also place the value of the test statistic in a scalar for later use.

```
. scalar tobs = r(t)
```

Efron and Tibshirani $(1993,224)$ describe an alternative to Satterthwaite's approximation that estimates the ASL by bootstrapping the statistic from the test of equal means. Their idea is to recenter the two samples to the combined sample mean so that the data now conform to the null hypothesis but that the variances within the samples remain unchanged.

```
. summarize mpg, meanonly
. scalar omean = r(mean)
. summarize mpg if foreign==O, meanonly
. replace mpg = mpg - r(mean) + scalar(omean) if foreign==0
variable mpg was int now float
(52 real changes made)
. summarize mpg if foreign==1, meanonly
. replace mpg = mpg - r(mean) + scalar(omean) if foreign==1
(22 real changes made)
. sort foreign
```

. by foreign: summarize mpg
-> foreign = Domestic

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| mpg | 52 | 21.2973 | 4.743297 | 13.47037 | 35.47038 |


| -> foreign $=$ Foreign |  |  |  |  |  |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| Variable |  | Obs | Mean | Std. dev. | Min | Max |
| mpg | 22 | 21.2973 | 6.611187 | 10.52457 | 37.52457 |  |

Each sample (foreign and domestic) is a stratum, so the bootstrapped samples must have the same number of foreign and domestic cars as the original dataset. This requirement is facilitated by the strata() option to bootstrap. By typing the following, we bootstrap the test statistic using the modified dataset and save the values in bsauto2.dta:

```
. keep mpg foreign
. set seed 1
. bootstrap t=r(t), rep(1000) strata(foreign) saving(bsauto2) nodots: ttest mpg,
> by(foreign) unequal
warning: ttest does not set e(sample), so no observations will be excluded
    from the resampling because of missing values or other reasons. To
    exclude observations, press Break, save the data, drop any
    observations that are to be excluded, and rerun bootstrap.
Bootstrap results
Number of strata = 2 Number of obs = 74
Replications = 1,000
    Command: ttest mpg, by(foreign) unequal
        t: r(t)
\begin{tabular}{l|cccccr}
\hline & \begin{tabular}{c} 
Observed \\
coefficient
\end{tabular} & \begin{tabular}{l} 
Bootstrap \\
std. err.
\end{tabular} & \(\mathbf{z}\) & \(\mathrm{P}>|\mathrm{z}|\) & \begin{tabular}{c} 
Normal-based \\
[95\% conf. interval]
\end{tabular} \\
\hline t & \(1.75 \mathrm{e}-07\) & 1.051867 & 0.00 & 1.000 & -2.061622 & 2.061622 \\
\hline
\end{tabular}
```

We can use the data in bsauto2.dta to estimate ASL via the fraction of bootstrap test statistics that are more extreme than 3.1797.

```
. use bsauto2, clear
(bootstrap: ttest)
. generate indicator = abs(t)>=abs(scalar(tobs))
. summarize indicator, meanonly
. display "ASLboot = " r(mean)
ASLboot = . }00
```

The result is $\mathrm{ASL}_{\mathrm{boot}}=0.004$. Assuming that the mean mpg is the same between foreign and domestic cars, we would expect to observe a $t$ statistic more extreme (in absolute value) than 3.1797 in about $0.4 \%$ of all possible samples of the type we observed. This finding is still strong evidence to reject the hypothesis that the means are equal.

## Bootstrapping a ratio

## > Example 4

Suppose that we wish to produce a bootstrap estimate of the ratio of two means. Because summarize stores results for only one variable, we must call summarize twice to compute the means. Actually, we could use collapse to compute the means in one call, but calling summarize twice is much faster. Thus, we will have to write a small program that will return the results we want.

We write the program below and save it to a file called ratio. ado (see [U] $\mathbf{1 7}$ Ado-files). Our program takes two variable names as input and saves them in the local macros y (first variable) and $x$ (second variable). It then computes one statistic: the mean of ' $y$ ' divided by the mean of ' $x$ '. This value is returned as a scalar in $r$ (ratio). ratio also returns the ratio of the number of observations used to the mean for each variable.

```
program myratio, rclass
    version 18.0 // (or version 18.5 for StataNow)
    args y x
    confirm var 'y'
    confirm var 'x'
    tempname ymean
    summarize 'y', meanonly
    scalar 'ymean' = r(mean)
    return scalar n_'y' = r(N)
    summarize 'x', meanonly
    return scalar n_'x' = r(N)
    return scalar ratio = 'ymean'/r(mean)
end
```

Remember to test any newly written commands before using them with bootstrap.

```
. use https://www.stata-press.com/data/r18/auto, clear
(1978 automobile data)
. summarize price
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline price & 74 & 6165.257 & 2949.496 & 3291 & 15906
\end{tabular}
. scalar mean1=r(mean)
. summarize weight
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline weight & 74 & 3019.459 & 777.1936 & 1760 & 4840
\end{tabular}
. scalar mean2=r(mean)
. di scalar(mean1)/scalar(mean2)
2.0418412
. myratio price weight
. return list
scalars:
\[
\begin{aligned}
r(\text { ratio }) & =2.041841210168278 \\
r(\text { n_weight }) & =74 \\
r(\text { n_price }) & =74
\end{aligned}
\]
    r(ratio) = 2.041841210168278
    r(n_weight) = 74
        r(n_price) = 74
```

The results of running bootstrap on our program are

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. set seed 1
. bootstrap ratio=r(ratio), reps(1000) nowarn nodots: myratio price weight
Bootstrap results Number of obs = 74
Replications = 1,000
```

    Command: myratio price weight
        ratio: r(ratio)
    |  | Observed <br> coefficient | Bootstrap <br> std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | Normal-based <br> [95\% conf. interval] |  |
| ---: | :---: | :---: | :---: | :---: | :---: | :---: |
| ratio | 2.041841 | .0953559 | 21.41 | 0.000 | 1.854947 | 2.228735 |

As mentioned previously, we should specify the saving() option if we wish to save the bootstrap dataset.

## Warning messages and e(sample)

bootstrap is not meant to be used with weighted calculations. bootstrap determines the presence of weights by parsing the prefixed command with standard syntax. However, commands like stcox and streg require that weights be specified in stset, and some user commands may allow weights to be specified by using an option instead of the standard syntax. Both cases pose a problem for bootstrap because it cannot determine the presence of weights under these circumstances. In these cases, we can only assume that you know what you are doing.
bootstrap does not know which variables of the dataset in memory matter to the calculation at hand. You can speed their execution by dropping unnecessary variables because, otherwise, they are included in each bootstrap sample.

You should thus drop observations with missing values. Leaving in missing values causes no problem in one sense because all Stata commands deal with missing values gracefully. It does, however, cause a statistical problem. Bootstrap sampling is defined as drawing, with replacement, samples of size $N$ from a set of $N$ observations. bootstrap determines $N$ by counting the number of observations in memory, not counting the number of nonmissing values on the relevant variables. The result is that too many observations are resampled; the resulting bootstrap samples, because they are drawn from a population with missing values, are of unequal sizes.

If the number of missing values relative to the sample size is small, this will make little difference. If you have many missing values, however, you should first drop the observations that contain them.

## > Example 5

To illustrate, we use the previous example but replace some of the values of price with missing values. The number of values of price used to compute the mean for each bootstrap is not constant. This is the purpose of the warning message.

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. replace price = . if inlist(_n,1,3,5,7)
(4 real changes made, 4 to missing)
. set seed 1
. bootstrap ratio=r(ratio) np=r(n_price) nw=r(n_weight), reps(100) nodots:
> myratio price weight
warning: myratio does not set e(sample), so no observations will be excluded
    from the resampling because of missing values or other reasons. To
    exclude observations, press Break, save the data, drop any
    observations that are to be excluded, and rerun bootstrap.
Bootstrap results Number of obs = 74
                                    Replications = 100
    Command: myratio price weight
        ratio: r(ratio)
            np: r(n_price)
            nw: r(n_weight)
\begin{tabular}{r|rrrrrr}
\hline & \begin{tabular}{c} 
Observed \\
coefficient
\end{tabular} & \begin{tabular}{c} 
Bootstrap \\
std. err.
\end{tabular} & z & \(\mathrm{P}>|\mathrm{z}|\) & \begin{tabular}{c} 
Normal-based \\
[95\% conf. interval]
\end{tabular} \\
\hline ratio & 2.063051 & .0981706 & 21.01 & 0.000 & 1.870641 & 2.255462 \\
np & 70 & 2.071939 & 33.78 & 0.000 & 65.93908 & 74.06092 \\
nw & 74 &. &. &. &. &. \\
\hline
\end{tabular}
```


## Bootstrapping statistics from data with a complex structure

Here we describe how to bootstrap statistics from data with a complex structure, for example, longitudinal or panel data, or matched data. bootstrap, however, is not designed to work with complex survey data. It is important to include all necessary information about the structure of the data in the bootstrap syntax to obtain correct bootstrap estimates for standard errors and confidence intervals.
bootstrap offers several options identifying the specifics of the data. These options are strata(), cluster(), idcluster(), and group(). The usage of strata() was described in example 3 above. Below, we demonstrate several examples that require specifying the other three options.

## > Example 6

Suppose that auto.dta in example 1 above are clustered by rep78. We want to obtain bootstrap estimates for the standard errors of the difference between the coefficients on weight and gear_ratio, taking into account clustering.

We supply the cluster (rep78) option to bootstrap to request resampling from clusters rather than from observations in the dataset.

```
. use https://www.stata-press.com/data/r18/auto, clear
(1978 automobile data)
. keep if rep78<.
(5 observations deleted)
. bootstrap diff=(_b[weight]-_b[gear]), seed(1) cluster(rep78): regress mpg
> weight gear foreign
(running regress on estimation sample)
Bootstrap replications (50): .........10..........20...........30...........40........
> ..50 done
Linear regression Number of obs = 69
                                    Replications = 50
    Command: regress mpg weight gear foreign
        diff: _b[weight]-_b[gear]
                            (Replications based on 5 clusters in rep78)
\begin{tabular}{c|cccccr}
\hline & \begin{tabular}{c} 
Observed \\
coefficient
\end{tabular} & \begin{tabular}{l} 
Bootstrap \\
std. err.
\end{tabular} & z & \(\mathrm{P}>|\mathrm{z}|\) & \begin{tabular}{c}
\multicolumn{2}{c}{\begin{tabular}{c} 
Normal-based \\
[95\% conf. interval]
\end{tabular}} \\
\hline diff
\end{tabular}\(-1.910396\) & 2.538558 \\
\hline
\end{tabular}
```

We drop missing values in rep78 before issuing the command because bootstrap does not allow missing values in cluster(). See the section above about using bootstrap when variables contain missing values.

We can also obtain these same results by using the following syntax:

```
. bootstrap diff=(_b[weight]-_b[gear]), seed(1): regress mpg weight gear foreign,
> vce(cluster rep78)
```

When only clustered information is provided to the command, bootstrap can pick up the vce (cluster clustvar) option from the main command and use it to resample from clusters.

## > Example 7

Suppose now that we have matched data and want to use bootstrap to obtain estimates of the standard errors of the exponentiated difference between two coefficients (or, equivalently, the ratio of two odds ratios) estimated by clogit. Consider the example of matched case-control data on birthweight of infants described in example 2 of $[R]$ clogit.

The infants are paired by being matched on mother's age. All groups, defined by the pairid variable, have $1: 1$ matching. clogit requires that the matching information, pairid, be supplied to the group() (or, equivalently, strata()) option to be used in computing the parameter estimates. Because the data are matched, we need to resample from groups rather than from the whole dataset. However, simply supplying the grouping variable pairid in cluster() is not enough with bootstrap, as it is with clustered data.

```
. use https://www.stata-press.com/data/r18/lowbirth2, clear
(Applied Logistic Regression, Hosmer \& Lemeshow)
. bootstrap ratio=exp(_b[smoke]-_b[ptd]), seed(1) cluster(pairid): clogit low
> lwt smoke ptd ht ui i.race, group(pairid)
(running clogit on estimation sample)
Bootstrap replications (50): ..........10..........20...........30........... \(40 . .\).
> .. 50 done
Bootstrap results Number of obs = 112
Replications \(=50\)
    Command: clogit low lwt smoke ptd ht ui i.race, group(pairid)
        ratio: \(\exp \left(\_b[s m o k e]-\_b[p t d]\right)\)
                            (Replications based on 56 clusters in pairid)
\begin{tabular}{r|cccccc}
\hline & \begin{tabular}{c} 
Observed \\
coefficient
\end{tabular} & \begin{tabular}{l} 
Bootstrap \\
std. err.
\end{tabular} & \(\mathbf{z}\) & \(\mathrm{P}>|\mathrm{z}|\) & \begin{tabular}{c} 
Normal-based \\
[95\% conf. interval]
\end{tabular} \\
\hline ratio & .6654095 & 2.043274 & 0.33 & 0.745 & -3.339334 & 4.670153 \\
\hline
\end{tabular}
```

For the syntax above, imagine that the first pair was sampled twice during a replication. Then, the bootstrap sample has four subjects with pairid equal to one, which clearly violates the original 1:1 matching design. As a result, the estimates of the coefficients obtained from this bootstrap sample will be incorrect.

Therefore, in addition to resampling from groups, we need to ensure that resampled groups are uniquely identified in each of the bootstrap samples. The idcluster (newcluster) option is designed for this. It requests that at each replication bootstrap create the new variable, newcluster, containing unique identifiers for all resampled groups. Thus, to make sure that the correct matching is preserved during each replication, we need to specify the grouping variable in cluster(), supply a variable name to idcluster(), and use this variable as the grouping variable with clogit, as we demonstrate below.


Note the difference between the estimates of the bootstrap standard error for the two specifications of the bootstrap syntax.

## - Technical note

Similarly, when you have panel (longitudinal) data, all resampled panels must be unique in each of the bootstrap samples to obtain correct bootstrap estimates of statistics. Therefore, both cluster(panelvar) and idcluster(newpanelvar) must be specified with bootstrap, and i (newpanelvar) must be used with the main command. Moreover, you must clear the current xtset settings by typing xtset, clear before calling bootstrap.

## > Example 8

Continuing with our birthweight data, suppose that we have more information about doctors supervising women's pregnancies. We believe that the data on the pairs of infants from the same doctor may be correlated and want to adjust standard errors for possible correlation among the pairs. clogit offers the vce (cluster clustvar) option to do this.

Let's add a cluster variable to our dataset. One thing to keep in mind is that to use vce (cluster clustvar), groups in group() must be nested within clusters.

```
. use https://www.stata-press.com/data/r18/lowbirth2, clear
(Applied Logistic Regression, Hosmer & Lemeshow)
. set seed 12345
. by pairid, sort: egen byte doctor = total(int(2*runiform()+1)*(_n == 1))
. clogit low lwt smoke ptd ht ui i.race, group(pairid) vce(cluster doctor)
Iteration 0: Log pseudolikelihood = -26.768693
Iteration 1: Log pseudolikelihood = -25.810476
Iteration 2: Log pseudolikelihood = -25.794296
Iteration 3: Log pseudolikelihood = -25.794271
Iteration 4: Log pseudolikelihood = -25.794271
Conditional (fixed-effects) logistic regression Number of obs = 112
Wald chi2(1) = .
Prob > chi2 = .
Pseudo R2 = 0.3355
(Std. err. adjusted for 2 clusters in doctor)
```

| low | Coefficient | Robust std. err. | z | $P>\|z\|$ | [95\% conf | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| lwt | -. 0183757 | . 0020314 | -9.05 | 0.000 | -. 0223571 | -. 0143942 |
| smoke | 1.400656 | . 3067525 | 4.57 | 0.000 | . 7994322 | 2.00188 |
| ptd | 1.808009 | . 2092246 | 8.64 | 0.000 | 1.397936 | 2.218082 |
| ht | 2.361152 | 1.410341 | 1.67 | 0.094 | -. 4030665 | 5.12537 |
| ui | 1.401929 | . 9406248 | 1.49 | 0.136 | -. 4416617 | 3.24552 |
| race |  |  |  |  |  |  |
| Black | . 5713643 | . 9992656 | 0.57 | 0.567 | -1.38716 | 2.529889 |
| Other | -. 0253148 | . 8453206 | -0.03 | 0.976 | -1.682113 | 1.631483 |

To obtain correct bootstrap standard errors of the exponentiated difference between the two coefficients in this example, we need to make sure that both resampled clusters and groups within resampled clusters are unique in each of the bootstrap samples. To achieve this, bootstrap needs the information about clusters in cluster(), the variable name of the new identifier for clusters in idcluster(), and the information about groups in group(). We demonstrate the corresponding syntax of bootstrap below.

```
. bootstrap ratio=exp(_b[smoke]-_b[ptd]), seed(1) cluster(doctor)
> idcluster(uidoctor) group(pairid): clogit low lwt smoke ptd ht ui i.race,
> group(pairid)
(running clogit on estimation sample)
Bootstrap replications (50): .........10.........20..........30...........40.
> .. 50 done
Bootstrap results Number of obs = 112
Replications = 50
    Command: clogit low lwt smoke ptd ht ui i.race, group(pairid)
        ratio: exp(_b[smoke]-_b[ptd])
                            (Replications based on 2 clusters in doctor)
\begin{tabular}{c|cccccc}
\hline & \begin{tabular}{c} 
Observed \\
coefficient
\end{tabular} & \begin{tabular}{c} 
Bootstrap \\
std. err.
\end{tabular} & z & \(\mathrm{P}>|\mathrm{z}|\) & \begin{tabular}{c} 
Normal-based \\
[95\% conf. interval]
\end{tabular} \\
\hline ratio & .6654095 & .1459234 & 4.56 & 0.000 & .3794048 & .9514142 \\
\hline
\end{tabular}
```

In the above syntax, although we specify group (pairid) with clogit, it is not the group identifiers of the original pairid variable that are used to compute parameter estimates from bootstrap samples. The way bootstrap works is that, at each replication, the clusters defined by doctor are resampled and the new variable, uidoctor, uniquely identifying resampled clusters is created. After that, another new variable uniquely identifying the (uidoctor, group) combination is created and renamed to have the same name as the grouping variable, pairid. This newly defined grouping variable is then used by clogit to obtain the parameter estimates from this bootstrap sample of clusters. After all replications are performed, the original values of the grouping variable are restored.

## Technical note

The same logic must be used when running bootstrap with commands designed for panel (longitudinal) data that allow specifying the cluster (clustervar) option. To ensure that the combination of (clustervar, panelvar) values are unique in each of the bootstrap samples, cluster (clustervar), idcluster (newclustervar), and group (panelvar) must be specified with bootstrap, and i (panelvar) must be used with the main command.

## - Technical note

When the bootstrap prefix is used with a user-defined program and when the expression list is _b, bootstrap calls
set coeftabresults off
before entering the replication loop to prevent Stata from performing unnecessary calculations. This means that, provided option noisily is not specified, estimation commands will not build or post the coefficient table matrix $r$ (table).

If your program calls an estimation command and needs $r(t a b l e)$ to exist to perform properly, then your program will need to call

```
set coeftabresults on
```

before calling other estimation commands.

Bradley Efron (1938- ) was born in 1938 in Minnesota and studied mathematics and statistics at Caltech and Stanford; he has lived in northern California since 1960. He has worked on empirical Bayes, survival analysis, exponential families, bootstrap and jackknife methods, and confidence intervals, in conjunction with applied work in biostatistics, astronomy, and physics.

Efron is a member of the US National Academy of Sciences and was awarded the US National Medal of Science in 2005. He is by any standards one of the world's leading statisticians: his work ranges from deep and elegant contributions in theoretical statistics to pathbreaking involvement in a variety of practical applications.

## Stored results

bootstrap stores the following in e():
Scalars
e(N)
e(N_reps)
e(N_misreps)
e(N_strata)
e(N_clust)
e(k_eq)
e(k_exp)
e(k_eexp)
e(k_extra)
e(level)
e(bs_version)
e(rank)
Macros
e(cmdname)
e(cmd)
e(command)
e(cmdline)
e(prefix)
e(title)
e(strata)
e(cluster)
e(rngstate)
e(size)
e(exp\#)
e(ties)
e(mse)
e(vce)
e(vcetype)
e(properties)
Matrices
e(b)
e(b_bs)
e(reps)
e(bias)
e(se)
e(zo)
e(accel)
e(ci_normal)
e(ci_percentile)
e(ci_bc)
e(ci_bca)
e(V)
e(V_modelbased)
sample size
number of complete replications
number of incomplete replications
number of strata
number of clusters
number of equations in $e(b)$
number of standard expressions
number of extended expressions (i.e., _b)
number of extra equations beyond the original ones from e(b)
confidence level for bootstrap CIs
version for bootstrap results
rank of e(V)
command name from command
same as e(cmdname) or bootstrap
command
command as typed
bootstrap
title in estimation output
strata variables
cluster variables
random-number state used
from the size(\#) option
expression for the \#th statistic
ties, if specified
mse, if specified
bootstrap
title used to label Std. err.
b V
observed statistics
bootstrap estimates
number of nonmissing results
estimated biases
estimated standard errors
median biases
estimated accelerations
normal-approximation CIs
percentile CIs
bias-corrected CIs
bias-corrected and accelerated CIs
bootstrap variance-covariance matrix
model-based variance

When exp_list is _b, bootstrap will also carry forward most of the results already in e() from command.
In addition to the above, the following is stored in r() :

Matrices $r$ (table)
matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

Let $\widehat{\theta}$ be the observed value of the statistic, that is, the value of the statistic calculated with the original dataset. Let $i=1,2, \ldots, k$ denote the bootstrap samples, and let $\widehat{\theta}_{i}$ be the value of the statistic from the $i$ th bootstrap sample.

When the mse option is specified, the standard error is estimated as

$$
\widehat{\mathrm{se}}_{\mathrm{MSE}}=\left\{\frac{1}{k} \sum_{i=1}^{k}\left(\widehat{\theta}_{i}-\widehat{\theta}\right)^{2}\right\}^{1 / 2}
$$

Otherwise, the standard error is estimated as

$$
\widehat{\mathrm{se}}=\left\{\frac{1}{k-1} \sum_{i=1}^{k}\left(\widehat{\theta}_{i}-\bar{\theta}\right)^{2}\right\}^{1 / 2}
$$

where

$$
\bar{\theta}=\frac{1}{k} \sum_{i=1}^{k} \widehat{\theta}_{i}
$$

The variance-covariance matrix is similarly computed. The bias is estimated as

$$
\widehat{\mathrm{bias}}=\bar{\theta}-\widehat{\theta}
$$

Confidence intervals with nominal coverage rates $1-\alpha$ are calculated according to the following formulas. The normal-approximation method yields the confidence intervals

$$
\left[\widehat{\theta}-z_{1-\alpha / 2} \widehat{\mathrm{se}}, \widehat{\theta}+z_{1-\alpha / 2} \widehat{\mathrm{se}}\right]
$$

where $z_{1-\alpha / 2}$ is the $(1-\alpha / 2)$ th quantile of the standard normal distribution. If the mse option is specified, bootstrap will report the normal confidence interval using $\widehat{\operatorname{se}}_{\text {MSE }}$ instead of $\widehat{\text { sed }}$. estat bootstrap only uses $\widehat{\text { se }}$ in the normal confidence interval.

The percentile method yields the confidence intervals

$$
\left[\theta_{\alpha / 2}^{*}, \theta_{1-\alpha / 2}^{*}\right]
$$

where $\theta_{p}^{*}$ is the $p$ th quantile (the $100 p$ th percentile) of the bootstrap distribution $\left(\widehat{\theta}_{1}, \ldots, \widehat{\theta}_{k}\right)$.

Let

$$
z_{0}=\Phi^{-1}\left\{\#\left(\widehat{\theta}_{i} \leq \widehat{\theta}\right) / k\right\}
$$

where $\#\left(\widehat{\theta}_{i} \leq \widehat{\theta}\right)$ is the number of elements of the bootstrap distribution that are less than or equal to the observed statistic and $\Phi$ is the standard cumulative normal. $z_{0}$ is known as the median bias of $\widehat{\theta}$. When the ties option is specified, $z_{0}$ is estimated as $\#\left(\widehat{\theta}_{i}<\widehat{\theta}\right)+\#\left(\widehat{\theta}_{i}=\widehat{\theta}\right) / 2$, which is the number of elements of the bootstrap distribution that are less than the observed statistic plus half the number of elements that are equal to the observed statistic.

Let

$$
a=\frac{\sum_{i=1}^{n}\left(\bar{\theta}_{(\cdot)}-\widehat{\theta}_{(i)}\right)^{3}}{6\left\{\sum_{i=1}^{n}\left(\bar{\theta}_{(\cdot)}-\widehat{\theta}_{(i)}\right)^{2}\right\}^{3 / 2}}
$$

where $\widehat{\theta}_{(i)}$ are the leave-one-out (jackknife) estimates of $\widehat{\theta}$ and $\bar{\theta}_{(\cdot)}$ is their mean. This expression is known as the jackknife estimate of acceleration for $\widehat{\theta}$. Let

$$
\begin{aligned}
& p_{1}=\Phi\left\{z_{0}+\frac{z_{0}-z_{1-\alpha / 2}}{1-a\left(z_{0}-z_{1-\alpha / 2}\right)}\right\} \\
& p_{2}=\Phi\left\{z_{0}+\frac{z_{0}+z_{1-\alpha / 2}}{1-a\left(z_{0}+z_{1-\alpha / 2}\right)}\right\}
\end{aligned}
$$

where $z_{1-\alpha / 2}$ is the $(1-\alpha / 2)$ th quantile of the normal distribution. The bias-corrected and accelerated $\left(\mathrm{BC}_{a}\right)$ method yields confidence intervals

$$
\left[\theta_{p_{1}}^{*}, \theta_{p_{2}}^{*}\right]
$$

where $\theta_{p}^{*}$ is the $p$ th quantile of the bootstrap distribution as defined previously. The bias-corrected (but not accelerated) method is a special case of $\mathrm{BC}_{a}$ with $a=0$.

## References

Ängquist, L. 2010. Stata tip 92: Manual implementation of permutations and bootstraps. Stata Journal 10: 686-688.
Cameron, A. C., and P. K. Trivedi. 2022. Microeconometrics Using Stata. 2nd ed. College Station, TX: Stata Press.
Davison, A. C., and D. V. Hinkley. 1997. Bootstrap Methods and Their Application. Cambridge: Cambridge University Press.
Dorta, M., and G. Sánchez. 2021. Bootstrap unit-root test for random walk with drift: The bsrwalkdrift command. Stata Journal 21: 39-50.

Efron, B. 1979. Bootstrap methods: Another look at the jackknife. Annals of Statistics 7: 1-26. https://doi.org/10.1214/aos/1176344552.
_ . 1982. The Jackknife, the Bootstrap and Other Resampling Plans. Philadelphia: Society for Industrial and Applied Mathematics.
Efron, B., and C. Stein. 1981. The jackknife estimate of variance. Annals of Statistics 9: 586-596. https://doi.org/10.1214/aos/1176345462.

Efron, B., and R. J. Tibshirani. 1986. Bootstrap methods for standard errors, confidence intervals, and other measures of statistical accuracy. Statistical Science 1: 54-77. https://doi.org/10.1214/ss/1177013815.
-_. 1993. An Introduction to the Bootstrap. New York: Chapman and Hall/CRC.
Field, C. A., and A. H. Welsh. 2007. Bootstrapping clustered data. Journal of the Royal Statistical Society, Series B 69: 369-390. https://doi.org/10.1111/j.1467-9868.2007.00593.x.
Guan, W. 2003. From the help desk: Bootstrapped standard errors. Stata Journal 3: 71-80.

Hall, P., and S. R. Wilson. 1991. Two guidelines for bootstrap hypothesis testing. Biometrics 47: 757-762. https://doi.org/10.2307/2532163.

Hinkley, D. V. 1978. Improving the jackknife with special reference to correlation estimation. Biometrika 65: 13-22. https://doi.org/10.1093/biomet/65.1.13.
Holmes, S., C. N. Morris, and R. J. Tibshirani. 2003. Bradley Efron: A conversation with good friends. Statistical Science 18: 268-281. https://doi.org/10.1214/ss/1063994981.
Huber, C. 2013. Measures of effect size in Stata 13. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2013/09/05/measures-of-effect-size-in-stata-13/.
Mooney, C. Z., and R. D. Duval. 1993. Bootstrapping: A Nonparametric Approach to Statistical Inference. Newbury Park, CA: Sage.

Ng, E. S.-W., R. Grieve, and J. R. Carpenter. 2013. Two-stage nonparametric bootstrap sampling with shrinkage correction for clustered data. Stata Journal 13: 141-164.

Poi, B. P. 2004. From the help desk: Some bootstrapping techniques. Stata Journal 4: 312-328.
Roodman, D., J. G. MacKinnon, M. Ø. Nielsen, and M. D. Webb. 2019. Fast and wild: Bootstrap inference in Stata using boottest. Stata Journal 19: 4-60.
Royston, P., and W. Sauerbrei. 2009. Bootstrap assessment of the stability of multivariable models. Stata Journal 9: 547-570.

Ruhe, C. 2019. Bootstrap pointwise confidence intervals for covariate-adjusted survivor functions in the Cox model. Stata Journal 19: 185-199.
Schwarz, C. 2019. Isemantica: A command for text similarity based on latent semantic analysis. Stata Journal 19: 129-142.

Stine, R. 1990. An introduction to bootstrap methods: Examples and ideas. In Modern Methods of Data Analysis, ed. J. Fox and J. S. Long, 353-373. Newbury Park, CA: Sage.

## Also see

[R] bootstrap postestimation - Postestimation tools for bootstrap
[R] jackknife - Jackknife estimation
[R] permute - Permutation tests
[R] simulate - Monte Carlo simulations
$[R]$ set rngstream - Specify the stream for the stream random-number generator
[SVY] svy bootstrap - Bootstrap for survey data
[U] 13.5 Accessing coefficients and standard errors
[U] 13.6 Accessing results from Stata commands
[U] 20 Estimation and postestimation commands

## Title

bootstrap postestimation - Postestimation tools for bootstrap

| Postestimation commands | predict | margins |
| :--- | :--- | :--- |
| estat | Remarks and examples | Also see |

## Postestimation commands

The following postestimation command is of special interest after bootstrap:
Command Description
estat bootstrap percentile-based and bias-corrected CI tables

The following standard postestimation commands are also available:

| Command | Description |
| :--- | :--- |
| contrast | contrasts and ANOVA-style joint tests of estimates <br> estat ic |
| Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian in- <br> formation criteria (AIC, CAIC, AICc, and BIC) |  |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) <br> cataloging estimation results |
| estimates | table of estimation results |
| etable | Hausman's specification test |
| hausman | point estimates, standard errors, testing, and inference for linear combinations of |
| lincom | coefficients |
| marginal means, predictive margins, marginal effects, and average marginal effects |  |
| marginsplot | graph the results from margins (profile plots, interaction plots, etc.) <br> point estimates, standard errors, testing, and inference for nonlinear combinations <br> of coefficients |
| nlcom | predictions, residuals, influence statistics, and other diagnostic measures <br> point estimates, standard errors, testing, and inference for generalized predictions |
| predict | pairwise comparisons of estimates |
| predictnl | Wald tests of simple and composite linear hypotheses |
| pwcompare | Wald tests of nonlinear hypotheses |

The postestimation command is allowed if it may be used after command.

## predict

The syntax of predict (and even if predict is allowed) following bootstrap depends upon the command used with bootstrap. If predict is not allowed, neither is predictnl.

## margins

The syntax of margins (and even if margins is allowed) following bootstrap depends upon the command used with bootstrap.

## estat

## Description for estat

estat bootstrap displays a table of confidence intervals for each statistic from a bootstrap analysis.

## Menu for estat

Statistics $>$ Postestimation

## Syntax for estat

estat bootstrap [, options]
options
Description
bc
bca
normal
percentile
all
noheader
nolegend
verbose
normal-based CIs
percentile CIs
all available CIs
bias-corrected CIs; the default
bias-corrected and accelerated ( $\mathrm{BC}_{a}$ ) CIs
suppress table header
suppress table legend
display the full table legend
bc , bca, normal, and percentile may be used together.

## Options for estat

bc is the default and displays bias-corrected confidence intervals.
bca displays bias-corrected and accelerated confidence intervals. This option assumes that you also specified the bca option on the bootstrap prefix command.
normal displays normal approximation confidence intervals.
percentile displays percentile confidence intervals.
all displays all available confidence intervals.
noheader suppresses display of the table header. This option implies nolegend.
nolegend suppresses display of the table legend, which identifies the rows of the table with the expressions they represent.
verbose requests that the full table legend be displayed.

## Remarks and examples

> Example 1
The estat bootstrap postestimation command produces a table containing the observed value of the statistic, an estimate of its bias, the bootstrap standard error, and up to four different confidence intervals.

If we were interested merely in getting bootstrap standard errors for the model coefficients, we could use the bootstrap prefix with our estimation command. If we were interested in performing a thorough bootstrap analysis of the model coefficients, we could use the estat bootstrap postestimation command after fitting the model with the bootstrap prefix.

Using example 1 from $[\mathrm{R}]$ bootstrap, we need many more replications for the confidence interval types other than the normal based, so let's rerun the estimation command. We will reset the randomnumber seed-in case we wish to reproduce the results-increase the number of replications, and save the bootstrap distribution as a dataset called bsauto.dta.

| ```(1978 automobile data) set seed 1 bootstrap _b, reps(1000) (output omitted) . estat bootstrap, all Linear regression``` |  | saving(bsauto) bca: regress mpg wei <br> Number of obs Replications |  |  | t gear foreign$\begin{array}{lr} = & 74 \\ = & 1000 \end{array}$ |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  |  |  |  |
| mpg | Observed coefficient | Bias | Bootstrap std. err. | [95\% conf. | interval] |  |
| weight | -. 00613903 | . 0000686 | . 00065005 | -. 0074131 | -. 004865 | (N) |
|  |  |  |  | -. 0073115 | -. 0048083 | (P) |
|  |  |  |  | -. 0073757 | -. 0048444 | (BC) |
|  |  |  |  | -. 0075498 | -. 0049202 | (BCa) |
|  | 1.4571134 | . 0297538 | 1.4471522 | -1.379253 | 4.29348 | (N) |
|  |  |  |  | -1.18779 | 4.540121 | (P) |
|  |  |  |  | -1.185389 | 4.540121 | (BC) |
|  |  |  |  | -1.131393 | 4.58386 | (BCa) |
| foreign | -2.2216815 | . 1029615 | 1.2606565 | -4.692523 | . 2491598 | (N) |
|  |  |  |  | -4.513954 | . 5011647 | (P) |
|  |  |  |  | -4.608057 | . 4208305 | (BC) |
|  |  |  |  | -4.614719 | . 3925043 | (BCa) |
| _cons | 36.101353 | -. 3122698 | 5.4303717 | 25.45802 | 46.74469 | (N) |
|  |  |  |  | 24.55211 | 46.0322 | (P) |
|  |  |  |  | 24.90078 | 46.05819 | (BC) |
|  |  |  |  | 24.99072 | 46.40419 | (BCa) |

Key: N: Normal
P: Percentile
BC: Bias-corrected
$\mathrm{BCa}:$ Bias-corrected and accelerated
The estimated standard errors here differ from our previous estimates using only 100 replications by, respectively, $7 \%, 6 \%, 8 \%$, and $4 \%$; see example 1 of [R] bootstrap. So much for our advice that $50-200$ replications are good enough to estimate standard errors. Well, the more replications the better-that advice you should believe.

Which of the methods to compute confidence intervals should we use? If the statistic is unbiased, the percentile ( P ) and bias-corrected ( BC ) methods should give similar results. The bias-corrected confidence interval will be the same as the percentile confidence interval when the observed value of the statistic is equal to the median of the bootstrap distribution. Thus, for unbiased statistics, the two methods should give similar results as the number of replications becomes large. For biased statistics, the bias-corrected method should yield confidence intervals with better coverage probability (closer to the nominal value of $95 \%$ or whatever was specified) than the percentile method. For statistics with variances that vary as a function of the parameter of interest, the bias-corrected and accelerated method $\left(\mathrm{BC}_{a}\right)$ will typically have better coverage probability than the others.

When the bootstrap distribution is approximately normal, all of these methods should give similar confidence intervals as the number of replications becomes large. If we examine the normality of these bootstrap distributions using, say, the pnorm command (see [R] Diagnostic plots), we see that they closely follow a normal distribution. In this case, the normal approximation would also be a valid choice. The chief advantage of the normal-approximation method is that it (supposedly) requires fewer replications than the other methods. Of course, it should be used only when the bootstrap distribution exhibits normality.

We can load bsauto.dta containing the bootstrap distributions for these coefficients:


We can now run other commands, such as pnorm, on the bootstrap distributions. As with all standard estimation commands, we can use the bootstrap command to replay its output table. The default variable names assigned to the statistics in exp_list are _bs_1, _bs_2, ..., and each variable is labeled with the associated expression. The naming convention for the extended expressions _b and _se is to prepend _ $\mathrm{b}_{-}$and _se_, respectively, onto the name of each element of the coefficient vector. Here the first coefficient is _b [weight], so bootstrap named it _b_weight.

## Also see

[R] bootstrap - Bootstrap sampling and estimation
[U] 20 Estimation and postestimation commands

## Title

boxcox - Box-Cox regression models

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

boxcox finds the maximum likelihood estimates of the parameters of the Box-Cox transform, the coefficients on the independent variables, and the standard deviation of the normally distributed errors. Any depvar or indepvars to be transformed must be strictly positive. Options can be used to control which variables remain untransformed.

## Quick start

Box-Cox transform of y in a model of y as a function of x 1
boxcox y x1
Same as above
boxcox y x1, model(lhsonly)
Likelihood-ratio test for each scale-variant parameter
boxcox y x1, lrtest
Different transform for each side and adding covariates x 2 and x 3
boxcox y x1 x2 x3, model(theta)
Same transform for both sides, and include x 3 as an untransformed variable transformation
boxcox y x1 x2, model(lambda) notrans(x3)

## Menu

Statistics $>$ Linear models and related $>$ Box-Cox regression

## Syntax

## boxcox depvar [indepvars] [if] [in] [weight] [, options]

options
Description
Model
noconstant
model(1hsonly)
model (rhsonly)
model (lambda)
model(theta)
notrans (varlist)
Reporting
level(\#)
lrtest
Maximization
[no]log
nologlr
maximize_options
suppress constant term
left-hand-side Box-Cox model; the default
right-hand-side Box-Cox model
both sides Box-Cox model with same parameter both sides Box-Cox model with different parameters do not transform specified independent variables
depvar and indepvars may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bootstrap, by, collect, jackknife, rolling, statsby, and xi are allowed; see [U] 11.1.10 Prefix commands.
Weights are not allowed with the bootstrap prefix; see $[\mathrm{R}]$ bootstrap.
fweights and iweights are allowed; see [U] 11.1.6 weight.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

noconstant; see [R] Estimation options.
model (lhsonly $\mid$ rhsonly $\mid$ lambda $\mid$ theta) specifies which of the four models to fit.
model(lhsonly) applies the Box-Cox transform to depvar only. model (lhsonly) is the default. model (rhsonly) applies the transform to the indepvars only.
model (lambda) applies the transform to both depvar and indepvars, and they are transformed by the same parameter.
model(theta) applies the transform to both depvar and indepvars, but this time, each side is transformed by a separate parameter.
notrans (varlist) specifies that the variables in varlist not be transformed when included in the model. You can specify notrans (varlist) with model (lhsonly), but the results will be the same as specifying the variables in varlist in indepvars.
lrtest specifies that a likelihood-ratio test of significance be performed and reported for each independent variable.
$\qquad$ Maximization
$\log$ and nolog specify whether to display the iteration log. The iteration $\log$ is displayed by default unless you used set iterlog off to suppress it; see set iterlog in [R] set iter. These options control the iteration log produced by the full model and, if option lrtest is specified, by the fitted restricted models.
nologlr suppresses the iteration $\log$ when fitting the restricted models required by the lrtest option. maximize_options: iterate(\#) and from(init_specs); see [R] Maximize.

| Model | Initial value specification |
| :--- | :--- |
| lhsonly | from $\left(\theta_{0}\right.$, copy $)$ |
| rhsonly | from $\left(\lambda_{0}\right.$, copy $)$ |
| lambda | from $\left(\lambda_{0}\right.$, copy $)$ |
| theta | from $\left(\lambda_{0} \theta_{0}\right.$, copy $)$ |

## Remarks and examples

Remarks are presented under the following headings:

Introduction<br>Theta model<br>Lambda model<br>Left-hand-side-only model<br>Right-hand-side-only model

## Introduction

The Box-Cox transform

$$
y^{(\lambda)}=\frac{y^{\lambda}-1}{\lambda}
$$

has been widely used in applied data analysis. Box and Cox (1964) developed the transformation and argued that the transformation could make the residuals more closely normal and less heteroskedastic. Cook and Weisberg (1982) discuss the transform in this light. Because the transform embeds several popular functional forms, it has received some attention as a method for testing functional forms, in particular,

$$
y^{(\lambda)}= \begin{cases}y-1 & \text { if } \lambda=1 \\ \ln (y) & \text { if } \lambda=0 \\ 1-1 / y & \text { if } \lambda=-1\end{cases}
$$

Davidson and MacKinnon (1993) discuss this use of the transform. Atkinson (1985) also gives a good general treatment.

## Theta model

boxcox obtains the maximum likelihood estimates of the parameters for four different models. The most general of the models, the theta model, is

$$
y_{j}^{(\theta)}=\beta_{0}+\beta_{1} x_{1 j}^{(\lambda)}+\beta_{2} x_{2 j}^{(\lambda)}+\cdots+\beta_{k} x_{k j}^{(\lambda)}+\gamma_{1} z_{1 j}+\gamma_{2} z_{2 j}+\cdots+\gamma_{l} z_{l j}+\epsilon_{j}
$$

where $\epsilon \sim N\left(0, \sigma^{2}\right)$. Here the dependent variable, $y$, is subject to a Box-Cox transform with parameter $\theta$. Each of the indepvars, $x_{1}, x_{2}, \ldots, x_{k}$, is transformed by a Box-Cox transform with parameter $\lambda$. The $z_{1}, z_{2}, \ldots, z_{l}$ specified in the notrans() option are independent variables that are not transformed.

Box and Cox (1964) argued that this transformation would leave behind residuals that more closely follow a normal distribution than those produced by a simple linear regression model. Bear in mind that the normality of $\epsilon$ is assumed and that boxcox obtains maximum likelihood estimates of the $k+l+4$ parameters under this assumption. boxcox does not choose $\lambda$ and $\theta$ so that the residuals are approximately normally distributed. If you are interested in this type of transformation to normality, see the official Stata commands lnskew0 and bcskew0 in [R] Inskew0. However, those commands work on a more restrictive model in which none of the independent variables is transformed.

## > Example 1

Below, we fit a theta model to a nonrepresentative extract of the Second National Health and Nutrition Examination Survey (NHANES II) dataset discussed in McDowell et al. (1981).

We model individual-level diastolic blood pressure (bpdiast) as a function of the transformed variables body mass index (bmi) and cholesterol level (tcresult) and of the untransformed variables age (age) and sex (sex).
. use https://www.stata-press.com/data/r18/nhanes2
. boxcox bpdiast bmi tcresult, notrans(age sex) model(theta) lrtest
Fitting comparison model
$\begin{array}{ll}\text { Iteration 0: } & \text { Log likelihood }=-41178.61 \\ \text { Iteration 1: } & \text { Log likelihood }=-41032.51 \\ \text { Iteration 2: } & \text { Log likelihood }=-41032.488 \\ \text { Iteration 3: } & \text { Log likelihood }=-41032.488\end{array}$
Fitting full model
Iteration 0: Log likelihood $=-39928.606$
Iteration 1: Log likelihood $=-39775.026$
Iteration 2: Log likelihood $=-39774.987$
Iteration 3: Log likelihood = -39774.987
Fitting comparison models for LR tests

| Iteration 0: | Log likelihood $=-39947.144$ |  |
| :--- | :--- | ---: |
| Iteration 1: | Log likelihood $=$ | -39934.55 |
| Iteration 2: | Log likelihood $=-39934.516$ |  |
| Iteration 3: | Log likelihood $=-39934.516$ |  |
| Iteration 0: | Log likelihood $=-39906.96$ |  |
| Iteration 1: | Log likelihood $=-39896.63$ |  |
| Iteration 2: | Log likelihood $=-39896.629$ |  |
| Iteration 0: | Log likelihood $=-40464.599$ |  |
| Iteration 1: | Log likelihood $=-40459.765$ |  |
| Iteration 2: | Log likelihood $=-40459.604$ |  |
| Iteration 3: | Log likelihood $=-40459.604$ |  |
| Iteration 0: | Log likelihood $=-39829.859$ |  |
| Iteration 1: | Log likelihood $=-39815.576$ |  |
| Iteration 2: | Log likelihood $=-39815.575$ |  |

Log likelihood = -39774.987

| Number of obs | $=$ | 10,351 |
| :--- | :--- | ---: |
| LR chi2 $(5)$ | $=$ | 2515.00 |
| Prob $>$ chi2 | $=$ | 0.000 |


| bpdiast | Coefficient | Std. err. | $z$ | $\mathrm{P}>\|\mathrm{z}\|$ | $[95 \%$ conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | :---: | :---: |
| /lambda | .6383286 | .1577601 | 4.05 | 0.000 | .3291245 | .9475327 |
| /theta | .1988197 | .0454088 | 4.38 | 0.000 | .1098201 | .2878193 |

Estimates of scale-variant parameters

|  | Coefficient chi2(df) |  | P>chi2(df) | df of chi2 |
| :---: | :---: | :---: | :---: | :---: |
| Notrans |  |  |  |  |
| age | . 003811 | 319.060 | 0.000 | 1 |
| sex | -. 1054887 | 243.284 | 0.000 | 1 |
| _cons | 5.835555 |  |  |  |
| Trans |  |  |  |  |
| bmi | . 0872041 | 1369.235 | 0.000 | 1 |
| tcresult | . 004734 | 81.177 | 0.000 | 1 |
| /sigma | . 3348267 |  |  |  |


| Test <br> H0: | Restricted <br> log likelihood | chi2 | Prob > chi2 |
| :--- | :---: | ---: | :---: |
| theta=lambda $=-1$ | -40162.898 | 775.82 | 0.000 |
| theta=lambda $=$ | 0 | -39790.945 | 31.92 |
| theta=lambda $=$ | 1 | -39928.606 | 307.24 |

The output is composed of the iteration logs and three distinct tables. The first table contains a standard header for a maximum likelihood estimator and a standard output table for the BoxCox transform parameters. The second table contains the estimates of the scale-variant parameters. The third table contains the output from likelihood-ratio tests on three standard functional form specifications.

The right-hand-side and the left-hand-side transformations each add to the regression fit at the $1 \%$ significance level and are both positive but less than 1 . All the variables have significant impacts on diastolic blood pressure, bpdiast. As expected, the transformed variables-the body mass index, bmi, and cholesterol level, tcresult-contribute to higher blood pressure. The last output table shows that the linear, multiplicative inverse, and $\log$ specifications are strongly rejected.

## - Technical note

Spitzer (1984) showed that the Wald tests of the joint significance of the coefficients of the right-hand-side variables, either transformed or untransformed, are not invariant to changes in the scale of the transformed dependent variable. Davidson and MacKinnon (1993) also discuss this point. This problem demonstrates that Wald statistics can be manipulated in nonlinear models. Lafontaine and White (1986) analyze this problem numerically, and Phillips and Park (1988) analyze it by using Edgeworth expansions. See Drukker (2000) for a more detailed discussion of this issue. Because the parameter estimates and their Wald tests are not scale invariant, no Wald tests or confidence intervals are reported for these parameters. However, when the lrtest option is specified, likelihood-ratio tests are performed and reported. Schlesselman (1971) showed that, if a constant is included in the model, the parameter estimates of the Box-Cox transforms are scale invariant. For this reason, we strongly recommend that you not use the noconstant option.

The lrtest option does not perform a likelihood-ratio test on the constant, so no value for this statistic is reported. Unless the data are properly scaled, the restricted model does not often converge. For this reason, no likelihood-ratio test on the constant is performed by the lrtest option. However, if you have a special interest in performing this test, you can do so by fitting the constrained model separately. If problems with convergence are encountered, rescaling the data by their means may help.

## Lambda model

A less general model than the one above is called the lambda model. It specifies that the same parameter be used in both the left-hand-side and right-hand-side transformations. Specifically,

$$
y_{j}^{(\lambda)}=\beta_{0}+\beta_{1} x_{1 j}^{(\lambda)}+\beta_{2} x_{2 j}^{(\lambda)}+\cdots+\beta_{k} x_{k j}^{(\lambda)}+\gamma_{1} z_{1 j}+\gamma_{2} z_{2 j}+\cdots+\gamma_{l} z_{l j}+\epsilon_{j}
$$

where $\epsilon \sim N\left(0, \sigma^{2}\right)$. Here the depvar variable, $y$, and each of the indepvars, $x_{1}, x_{2}, \ldots, x_{k}$, is transformed by a Box-Cox transform with the common parameter $\lambda$. Again, the $z_{1}, z_{2}, \ldots, z_{l}$ are independent variables that are not transformed.

## Left-hand-side-only model

Even more restrictive than a common transformation parameter is transforming the dependent variable only. Because the dependent variable is on the left-hand side of the equation, this model is known as the lhsonly model. Here you are estimating the parameters of the model

$$
y_{j}^{(\theta)}=\beta_{0}+\beta_{1} x_{1 j}+\beta_{2} x_{2 j}+\cdots+\beta_{k} x_{k j}+\epsilon_{j}
$$

where $\epsilon \sim N\left(0, \sigma^{2}\right)$. Here only the depvar, $y$, is transformed by a Box-Cox transform with the parameter $\theta$.

## > Example 2

In this example, we model the transform of diastolic blood pressure as a linear combination of the untransformed body mass index, cholesterol level, age, and sex.

| . boxcox bpdiast bmi tcresult age sex, model <br> Fitting comparison model <br> Fitting full model <br> Fitting comparison models for LR tests |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  |  |  |  |
| Number of ob LR chi2(4) |  |  |  |  |  |  |
| Log likelihood $=-39777.709$ Prob > chi2 |  |  |  |  |  |  |
| bpdiast | Coefficient | Std. err | . $\quad \mathrm{z}$ | $\mathrm{P}>\|z\|$ |  | [95\% |
| /theta | . 2073268 | . 0452895 | 4.58 | 0.000 |  | . 118 |
| Estimates of scale-variant parameters |  |  |  |  |  |  |
|  | Coefficient chi2(df) P>chi2(df) df of chi2 |  |  |  |  |  |
| Notrans |  |  |  |  |  |  |
| bmi | $\begin{array}{lll}.0272628 & 1375.841 & 0.000\end{array}$ |  |  |  |  |  |
| tcresult | . $0006929 \quad 82.380 \quad 0.000$ |  |  |  |  |  |
| age | . $0040141 \quad 334.117 \quad 0.000$ |  |  |  |  |  |
| sex | $\begin{array}{llll}-.1122274 & 263.219 & 0.000 & 1\end{array}$ |  |  |  |  |  |
| _cons | 6.302855 |  |  |  |  |  |
| /sigma | . 3476615 |  |  |  |  |  |
| Test HO : | Restricted log likelihood | LR statistic |  | Prob > chi2 |  |  |
|  |  |  | hi2 Pro |  |  |  |
| $\begin{aligned} \text { theta } & =-1 \\ \text { theta } & =0 \\ \text { theta } & =1 \end{aligned}$ | $\begin{aligned} & -40146.678 \\ & -39788.241 \\ & -39928.606 \end{aligned}$ | 737.94 |  | 0.000 |  |  |
|  |  | 21.06 |  | 0.000 |  |  |
|  |  | 301.79 |  | 0.000 |  |  |

The maximum likelihood estimate of the transformation parameter for this model is positive and significant. Once again, all the scale-variant parameters are significant, and we find a positive impact of body mass index (bmi) and cholesterol levels (tcresult) on the transformed diastolic blood pressure (bpdiast). This model rejects the linear, multiplicative inverse, and log specifications.

## Right-hand-side-only model

The fourth model leaves the depvar alone and transforms a subset of the indepvars using the parameter $\lambda$. This is the rhsonly model. In this model, the depvar, $y$, is given by

$$
y_{j}=\beta_{0}+\beta_{1} x_{1 j}^{(\lambda)}+\beta_{2} x_{2 j}^{(\lambda)}+\cdots+\beta_{k} x_{k j}^{(\lambda)}+\gamma_{1} z_{1 j}+\gamma_{2} z_{2 j}+\cdots+\gamma_{l} z_{l j}+\epsilon_{j}
$$

where $\epsilon \sim N\left(0, \sigma^{2}\right)$. Here each of the indepvars, $x_{1}, x_{2}, \ldots, x_{k}$, is transformed by a Box-Cox transform with the parameter $\lambda$. Again, the $z_{1}, z_{2}, \ldots, z_{l}$ are independent variables that are not transformed.

## > Example 3

Now, we consider a rhsonly model in which the regressors sex and age are not transformed.


Estimates of scale-variant parameters

|  | Coefficient chi2(df) |  | $\mathrm{P}>\mathrm{chi2}(\mathrm{df})$ | df of chi2 |
| :---: | :---: | :---: | :---: | :---: |
| Notrans |  |  |  |  |
| sex | -3.544042 | 235.020 | 0.000 | 1 |
| age | . 128809 | 311.754 | 0.000 | 1 |
| _cons | 50.01498 |  |  |  |
| Trans |  |  |  |  |
| bmi | 1.418215 | 1396.709 | 0.000 | 1 |
| tcresult | . 0462964 | 78.500 | 0.000 | 1 |
| /sigma | 11.4557 |  |  |  |


| Test <br> H0: | Restricted <br> log likelihood | LR statistic <br> chi2 | Prob > chi2 |
| :---: | :---: | :---: | :---: |
| lambda $=-1$ | -39989.331 | 122.24 | 0.000 |
| lambda $=$ | 0 | -39942.945 | 29.47 |
| lambda $=$ | 1 | -39928.606 | 0.79 |

The maximum likelihood estimate of the transformation parameter in this model is positive and significant at the $1 \%$ level. The transformed bmi coefficient behaves as expected, and the remaining scale-variant parameters are significant at the $1 \%$ level. This model rejects the multiplicative inverse and $\log$ specifications strongly. However, we cannot reject the hypothesis that the model is linear.

## Stored results

boxcox stores the following in e() :
Scalars
$e(N) \quad$ number of observations
e(11) $\quad \log$ likelihood
$e$ (chi2) LR statistic of full vs. comparison
$e\left(\mathrm{df} \_\mathrm{m}\right) \quad$ full model degrees of freedom
$\mathrm{e}(110) \quad \log$ likelihood of the restricted model
$e\left(d f \_r\right) \quad$ restricted model degrees of freedom
e(ll_t1) $\quad$ log likelihood of model $\lambda=\theta=1$
$e($ chi2_t1) LR of $\lambda=\theta=1$ vs. full model
e (p_t1) $\quad p$-value of $\lambda=\theta=1$ vs. full model
$\mathrm{e}\left(11 \_\right.$tm1) $\quad \log$ likelihood of model $\lambda=\theta=-1$
$e($ chi2_tm1) LR of $\lambda=\theta=-1$ vs. full model
$\mathrm{e}($ p_tm1) $\quad p$-value of $\lambda=\theta=-1$ vs. full model
e(ll_t0) $\quad$ log likelihood of model $\lambda=\theta=0$
e(chi2_t0) LR of $\lambda=\theta=0$ vs. full model
e (p_t0) $\quad p$-value of $\lambda=\theta=0$ vs. full model
$e$ (rank) rank of e(V)
$e(i c) \quad$ number of iterations
$e(r c)$ return code
Macros
$e(\mathrm{cmd})$
boxcox
e(cmdline) command as typed
e(depvar) name of dependent variable
e(model) lhsonly, rhsonly, lambda, or theta
$e$ (wtype) weight type
e(wexp)
$e$ (ntrans)
e(chi2type)
weight expression
yes if untransformed indepvars
$e$ (lrtest) lrtest, if requested
e(properties)
b V
e(predict)
e(marginsnotok)
program used to implement predict
predictions disallowed by margins
Matrices
e(b)
e(V)
$\mathrm{e}(\mathrm{pm}) \quad p$-values for LR tests on indepvars
$e(d f) \quad$ degrees of freedom of LR tests on indepvars
e(chi2m)
coefficient vector
variance-covariance matrix of the estimators (see note below)

LR statistics for tests on indepvars
Functions
e(sample) marks estimation sample
$e(V)$ contains all zeros, except for the elements that correspond to the parameters of the Box-Cox transform.

## Methods and formulas

In the internal computations,

$$
y^{(\lambda)}= \begin{cases}\frac{y^{\lambda}-1}{\lambda} & \text { if }|\lambda|>10^{-10} \\ \ln (y) & \text { otherwise }\end{cases}
$$

The unconcentrated log likelihood for the theta model is

$$
\ln L=\left(\frac{-N}{2}\right)\left\{\ln (2 \pi)+\ln \left(\sigma^{2}\right)\right\}+(\theta-1) \sum_{i=1}^{N} \ln \left(y_{i}\right)-\left(\frac{1}{2 \sigma^{2}}\right) \mathrm{SSR}
$$

where

$$
\operatorname{SSR}=\sum_{i=1}^{N}\left(y_{i}^{(\theta)}-\beta_{0}+\beta_{1} x_{i 1}^{(\lambda)}+\beta_{2} x_{i 2}^{(\lambda)}+\cdots+\beta_{k} x_{i k}^{(\lambda)}+\gamma_{1} z_{i 1}+\gamma_{2} z_{i 2}+\cdots+\gamma_{l} z_{i l}\right)^{2}
$$

Writing the SSR in matrix form,

$$
\operatorname{SSR}=\left(\mathbf{y}^{(\theta)}-\mathbf{X}^{(\lambda)} \mathbf{b}^{\prime}-\mathbf{Z g}^{\prime}\right)^{\prime}\left(\mathbf{y}^{(\theta)}-\mathbf{X}^{(\lambda)} \mathbf{b}^{\prime}-\mathbf{Z g}^{\prime}\right)
$$

where $\mathbf{y}^{(\theta)}$ is an $N \times 1$ vector of elementwise transformed data, $\mathbf{X}^{(\lambda)}$ is an $N \times k$ matrix of elementwise transformed data, $\mathbf{Z}$ is an $N \times l$ matrix of untransformed data, $\mathbf{b}$ is a $1 \times k$ vector of coefficients, and $\mathbf{g}$ is a $1 \times l$ vector of coefficients. Letting

$$
\mathbf{W}_{\lambda}=\left(\mathbf{X}^{(\lambda)} \mathbf{Z}\right)
$$

be the horizontal concatenation of $\mathbf{X}^{(\lambda)}$ and $\mathbf{Z}$ and

$$
\mathbf{d}^{\prime}=\binom{\mathbf{b}^{\prime}}{\mathbf{g}^{\prime}}
$$

be the vertical concatenation of the coefficients yields

$$
\operatorname{SSR}=\left(\mathbf{y}^{(\theta)}-\mathbf{W}_{\lambda} \mathbf{d}^{\prime}\right)^{\prime}\left(\mathbf{y}^{(\theta)}-\mathbf{W}_{\lambda} \mathbf{d}^{\prime}\right)
$$

For given values of $\lambda$ and $\theta$, the solutions for $\mathbf{d}^{\prime}$ and $\sigma^{2}$ are

$$
\widehat{\mathbf{d}}^{\prime}=\left(W_{\lambda}^{\prime} W_{\lambda}\right)^{-1} W_{\lambda}^{\prime} y^{(\theta)}
$$

and

$$
\widehat{\sigma}^{2}=\frac{1}{N}\left(\mathbf{y}^{(\theta)}-W_{\lambda} \widehat{d}^{\prime}\right)^{\prime}\left(y^{(\theta)}-W_{\lambda}{\widehat{d^{\prime}}}^{\prime}\right)
$$

Substituting these solutions into the log-likelihood function yields the concentrated log-likelihood function

$$
\ln L_{c}=\left(-\frac{N}{2}\right)\left\{\ln (2 \pi)+1+\ln \left(\widehat{\sigma}^{2}\right)\right\}+(\theta-1) \sum_{i=1}^{N} \ln \left(y_{i}\right)
$$

Similar calculations yield the concentrated log-likelihood function for the lambda model,

$$
\ln L_{c}=\left(-\frac{N}{2}\right)\left\{\ln (2 \pi)+1+\ln \left(\widehat{\sigma}^{2}\right)\right\}+(\lambda-1) \sum_{i=1}^{N} \ln \left(y_{i}\right)
$$

the lhsonly model,

$$
\ln L_{c}=\left(-\frac{N}{2}\right)\left\{\ln (2 \pi)+1+\ln \left(\widehat{\sigma}^{2}\right)\right\}+(\theta-1) \sum_{i=1}^{N} \ln \left(y_{i}\right)
$$

and the rhsonly model,

$$
\ln L_{c}=\left(-\frac{N}{2}\right)\left\{\ln (2 \pi)+1+\ln \left(\widehat{\sigma}^{2}\right)\right\}
$$

where $\widehat{\sigma}^{2}$ is specific to each model and is defined analogously to that in the theta model.

## References

Atkinson, A. C. 1985. Plots, Transformations, and Regression: An Introduction to Graphical Methods of Diagnostic Regression Analysis. Oxford: Oxford University Press.

Box, G. E. P., and D. R. Cox. 1964. An analysis of transformations. Journal of the Royal Statistical Society, Series B 26: 211-252.

Carroll, R. J., and D. Ruppert. 1988. Transformation and Weighting in Regression. New York: Chapman and Hall.
Cook, R. D., and S. Weisberg. 1982. Residuals and Influence in Regression. New York: Chapman and Hall/CRC.
Davidson, R., and J. G. MacKinnon. 1993. Estimation and Inference in Econometrics. New York: Oxford University Press.
Drukker, D. M. 2000. sg131: On the manipulability of Wald tests in Box-Cox regression models. Stata Technical Bulletin 54: 36-42. Reprinted in Stata Technical Bulletin Reprints, vol. 9, pp. 319-327. College Station, TX: Stata Press.

Lafontaine, F., and K. J. White. 1986. Obtaining any Wald statistic you want. Economics Letters 21: 35-40. https://doi.org/10.1016/0165-1765(86)90117-5.
Lindsey, C., and S. J. Sheather. 2010a. Power transformation via multivariate Box-Cox. Stata Journal 10: 69-81.
——. 2010b. Optimal power transformation via inverse response plots. Stata Journal 10: 200-214.
McDowell, A., A. Engel, J. T. Massey, and K. Maurer. 1981. Plan and operation of the Second National Health and Nutrition Examination Survey, 1976-1980. Vital and Health Statistics 1(15): 1-144.
Phillips, P. C. B., and J. Y. Park. 1988. On the formulation of Wald tests of nonlinear restrictions. Econometrica 56: 1065-1083. https://doi.org/10.2307/1911359.

Schlesselman, J. J. 1971. Power families: A note on the Box and Cox transformation. Journal of the Royal Statistical Society, Series B 33: 307-311. https://doi.org/10.1111/j.2517-6161.1971.tb00882.x.

Spitzer, J. J. 1984. Variance estimates in models with the Box-Cox transformation: Implications for estimation and hypothesis testing. Review of Economics and Statistics 66: 645-652. https://doi.org/10.2307/1935988.

## Also see <br> [R] boxcox postestimation - Postestimation tools for boxcox <br> [R] Inskew0 - Find zero-skewness log or Box-Cox transform <br> $[R]$ regress - Linear regression

[U] 20 Estimation and postestimation commands

## Title

boxcox postestimation - Postestimation tools for boxcox

| Postestimation commands | predict | Remarks and examples |
| :--- | :--- | :--- |
| References | Also see |  |

## Postestimation commands

The following postestimation commands are available after boxcox:

| Command | Description |
| :---: | :---: |
| estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estimates | cataloging estimation results |
| etable | table of estimation results |
| hausman | Hausman's specification test |
| *lincom | point estimates, standard errors, testing, and inference for linear combinations of coefficients |
| * nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients |
| predict | predictions and residuals |
| * test | Wald tests of simple and composite linear hypotheses |
| * testnl | Wald tests of nonlinear hypotheses |

[^8]
## predict

## Description for predict

predict creates a new variable containing predictions such as predicted values and residuals.

## Menu for predict

Statistics > Postestimation

## Syntax for predict

```
predict [type] newvar [if] [in] [, statistic options]
```

statistic Description
Main
yhat predicted value of $y$; the default residuals residuals
options Description
Options
smearing compute statistic using smearing method; the default
btransform compute statistic using back-transform method

These statistics are available both in and out of sample; type predict . . . if e (sample) ... if wanted only for the estimation sample.

## Options for predict

$\qquad$ Main
yhat, the default, calculates the predicted value of the dependent variable.
residuals calculates the residuals, that is, the observed value minus the predicted value.
$\qquad$ Options
smearing calculates the statistics yhat and residuals using the smearing method proposed by Duan (1983) (see Methods and formulas for a description of this method). smearing is the default. btransform calculates the statistics yhat and residuals using the back-transform method (see Methods and formulas for a description of this method).

## Remarks and examples

Below, we present two examples that illustrate how to use the smearing and btransform options.
> Example 1: Predictions with the smearing option
In this example, we calculate the predicted values of diastolic blood pressure, bpdiast, that arise from the theta model calculated in example 1 of $[R]$ boxcox.

```
. use https://www.stata-press.com/data/r18/nhanes2
. boxcox bpdiast bmi tcresult, notrans(age sex) model(theta) lrtest
    (output omitted)
. predict yhat
(statistic yhat and option smearing are assumed)
```

In the expression above, yhat is the name we gave to the estimates of the conditional expectation. Given that we did not specify any statistic or option, the corresponding defaults yhat and smearing were assumed.

As the summary table below illustrates, the mean of the dependent variable is close to the mean of the predicted value yhat. This indicates that the theta model does a good job approximating the true value of diastolic blood pressure, bpdiast.
. summarize bpdiast yhat

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| bpdiast | 10,351 | 81.715 | 12.92722 | 35 | 150 |
| yhat | 10,351 | 81.71406 | 5.983486 | 66.93709 | 110.5283 |

Similarly, we could have asked that residuals be calculated. Here we again use the default smearing option:

```
. predict resid, residuals
(option smearing assumed to compute residuals)
```

> Example 2: Predictions with the btransform option
In this example, we illustrate the tradeoffs involved by using the btransform option as opposed to the default smearing option. Continuing with example 1, we compute the predicted values using the back-transform method.

```
. predict yhatb, btransform
(statistic yhat assumed)
```

We now compute the predicted values using the smearing option and summarize both computations.
. predict yhats
(statistic yhat and option smearing are assumed)
• summarize bpdiast yhats yhatb
Variable

As can be seen from the mean and the standard deviation of the summary table, the predicted values using the back-transform method give biased estimates but are less variable than those coming from the smearing method. However, the efficiency loss is small compared with the bias reduction.

## - Technical note

boxcox estimates variances only for the $\lambda$ and $\theta$ parameters (see the technical note in [R] boxcox), so the extent to which postestimation commands can be used following boxcox is limited. Formulas used in lincom, nlcom, test, and testnl are dependent on the estimated variances. Therefore, the use of these commands is limited and generally applicable only to inferences on the $\lambda$ and $\theta$ coefficients.

## Methods and formulas

The computation of the expected value of the dependent variable conditional on the regressors for the Box-Cox model does not follow the logic of the standard linear regression model because the random disturbance does not vanish from the conditional expectation and must be accounted for. To show this, we will revisit the lhsonly model described by

$$
y_{j}^{(\lambda)}=\beta_{0}+\beta_{1} x_{1 j}+\beta_{2} x_{2 j}+\cdots+\beta_{(k-1)} x_{(k-1) j}+\epsilon_{j}
$$

where

$$
y^{(\lambda)}=\frac{y^{\lambda}-1}{\lambda}
$$

and

$$
y^{(\lambda)}= \begin{cases}y-1 & \text { if } \lambda=1 \\ \ln (y) & \text { if } \lambda=0 \\ 1-1 / y & \text { if } \lambda=-1\end{cases}
$$

For the presentation below, let $\mathbf{y}^{(\lambda)}$ be an $N \times 1$ vector of elementwise transformed data, $\mathbf{X}$ be an $N \times k$ matrix of regressors, $\boldsymbol{\beta}$ be a $k \times 1$ vector of parameters, and $\iota$ be an $n \times 1$ vector of ones.

If we were interested in $E\left(\mathbf{y}^{(\lambda)} \mid \mathbf{X}\right)$, then the conventional logic would follow, and we would obtain predictions as $y^{(\widehat{\lambda})}=\mathbf{X} \widehat{\boldsymbol{\beta}}$, where $\widehat{\boldsymbol{\beta}}$ is the estimate of $\boldsymbol{\beta}$. However, to estimate the conditional expectation of $\mathbf{y}$, we need to isolate it on the left-hand side of the model. In the case of the lhsonly model, this yields

$$
\mathbf{y}=\{\lambda(\mathbf{X} \widehat{\boldsymbol{\beta}}+\boldsymbol{\epsilon})+\boldsymbol{\iota}\}^{1 / \lambda}
$$

The conditional expectation is then defined by

$$
E(\mathbf{y} \mid \mathbf{X})=\int\{\lambda(\mathbf{X} \boldsymbol{\beta}+\boldsymbol{\epsilon})+\boldsymbol{\iota}\}^{1 / \lambda} d F(\boldsymbol{\epsilon} \mid \mathbf{X})
$$

In the expression above, $d F(\epsilon \mid \mathbf{X})$ corresponds to the cdf of $\epsilon$ conditional on the regressors. It is also clear that the random disturbance does not vanish.

To address this issue, the default methodology used by predict computes this integral using the smearing method proposed by Duan (1983) to implement a two-step estimator, as was suggested by Abrevaya (2002).

In the first step, we get an estimate for $\boldsymbol{\epsilon}$ defined as

$$
\widehat{\boldsymbol{\epsilon}}=\mathbf{y}^{(\widehat{\lambda})}-\mathbf{X} \widehat{\boldsymbol{\beta}}
$$

In the second step, for each $j$ we compute our predicted values as the sum:

$$
\widehat{y}_{j}=\frac{1}{N} \sum_{i=1}^{N}\left\{\widehat{\lambda}\left(\mathbf{x}_{j} \widehat{\boldsymbol{\beta}}+\widehat{\epsilon}_{i}\right)+1\right\}^{1 / \widehat{\lambda}}
$$

In the expression above, $\mathbf{x}_{j}$ is the $j$ th row of the matrix $\mathbf{X}$ (in other words, the values of the covariates for individual $j$ ), and $\widehat{\epsilon}_{i}$ is the residual for individual $i$. The result of this summation gives us the conditional expectation of the dependent variable for individual $j$. Given that this operation is performed for each individual $j$, the methodology is computationally intensive.

The back-transform method can be understood as a naïve estimate that disregards the random disturbance. The predictions using this approach are given by

$$
\widehat{y}_{j}=\left(\widehat{\lambda} \mathbf{x}_{j} \widehat{\boldsymbol{\beta}}+1\right)^{1 / \widehat{\lambda}}
$$

## References

Abrevaya, J. 2002. Computing marginal effects in the Box-Cox model. Econometric Reviews 21: 383-393. https://doi.org/10.1081/ETC-120015789.

Duan, N. 1983. Smearing estimate: A nonparametric retransformation method. Journal of the American Statistical Association 78: 605-610. https://doi.org/10.1080/01621459.1983.10478017.

## Also see

[R] boxcox - Box-Cox regression models
[U] 20 Estimation and postestimation commands

## Title

brier - Brier score decomposition

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Option | Remarks and examples | Stored results | Methods and formulas |
| Acknowledgment | References | Also see |  |

## Description

brier computes the Yates, Sanders, and Murphy decompositions of the Brier mean probability score. The Brier score is a measure of disagreement between the observed outcome and a forecast (prediction).

## Quick start

Brier score decomposition for binary outcome y and predicted probability pvar brier y pvar

Same as above, but use 5 groups rather than 10 to compute the decomposition brier y pvar, group(5)

## Menu

Statistics $>$ Epidemiology and related $>$ Other $>$ Brier score decomposition

## Syntax

brier outcomevar forecastvar [if] [in] [, group(\#)]
outcomevar is a binary variable indicating the outcome of the experiment.
forecastvar is the corresponding probability of a positive outcome and must be between 0 and 1 .
by and collect are allowed; see [U] 11.1.10 Prefix commands.

## Option

> Main
> group(\#) specifies the number of groups that will be used to compute the decomposition. group(10) is the default.

## Remarks and examples

You have a binary ( $0 / 1$ ) response and a formula that predicts the corresponding probabilities of having observed a positive outcome (1). If the probabilities were obtained from logistic regression, there are many methods that assess goodness of fit (see, for instance, [R] estat gof). However, the probabilities might be computed from a published formula or from a model fit on another sample, both completely unrelated to the data at hand, or perhaps the forecasts are not from a formula at all. In any case, you now have a test dataset consisting of the forecast probabilities and observed outcomes. Your test dataset might, for instance, record predictions made by a meteorologist on the probability of rain along with a variable recording whether it actually rained.

The Brier score is an aggregate measure of disagreement between the observed outcome and a prediction - the average squared error difference. The Brier score decomposition is a partition of the Brier score into components that suggest reasons for discrepancy. These reasons fall roughly into three groups: 1) lack of overall calibration between the average predicted probability and the actual probability of the event in your data, 2) misfit of the data in groups defined within your sample, and 3 ) inability to match actual 0 and 1 responses.
Problem 1 refers to simply overstating or understating the probabilities.
Problem 2 refers to what is standardly called a goodness-of-fit test: the data are grouped, and the predictions for the group are compared with the outcomes.
Problem 3 refers to an individual-level measure of fit. Imagine that the grouped outcomes are predicted on average correctly but that within the group, the outcomes are poorly predicted.
Using logit or probit analysis to fit your data will guarantee that there is no lack of fit due to problem 1 , and a good model fitter will be able to avoid problem 2. Problem 3 is inherent in any prediction exercise.
> Example 1
We have data on the outcomes of 20 basketball games (win) and the probability of victory predicted by a local pundit (for).
. use https://www.stata-press.com/data/r18/bball
. summarize win for
Variable

win
for

The mean probabilities of forecast and outcome are simply the mean of the predicted probabilities and the actual outcomes (wins/losses). The correlation is the product-moment correlation between them.

The Brier score measures the total difference between the event (winning) and the forecast probability of that event as an average squared difference. As a benchmark, a perfect forecaster would have a Brier score of 0 , a perfect misforecaster (predicts probability of win is 1 when loses and 0 when wins) would have a Brier score of 1 , and a fence sitter (forecasts every game as $50 / 50$ ) would have a Brier score of 0.25 . Our pundit is doing reasonably well.

Spiegelhalter's $z$ statistic is a standard normal test statistic for testing whether an individual Brier score is extreme. The ROC area is the area under the receiver operating curve, and the associated test is a test of whether it is greater than 0.5 . The more accurate the forecast probabilities, the larger the ROC area.

The Sanders-modified Brier score measures the difference between a grouped forecast measure and the event, where the data are grouped by sorting the sample on the forecast and dividing it into approximately equally sized groups. The difference between the modified and the unmodified score is typically minimal. For this and the other statistics that require grouping-the Sanders and Murphy resolutions and reliability-in-the-small-to be well defined, group boundaries are chosen so as not to allocate observations with the same forecast probability to different groups. This task is done by grouping on the forecast using xtile, $n(\#)$, with \# being the number of groups; see [D] pctile.

The Sanders resolution measures error that arises from statistical considerations in evaluating the forecast for a group. A group with all positive or all negative outcomes would have a Sanders resolution of 0 ; it would most certainly be feasible to predict exactly what happened to each member of the group. If the group had $40 \%$ positive responses, on the other hand, a forecast that assigned $p=0.4$ to each member of the group would be a good one, and yet, there would be "errors" in
the squared difference sense. The "error" would be $(1-0.4)^{2}$ or $(0-0.4)^{2}$ for each member. The Sanders resolution is the average across groups of such "expected" errors. The 0.1400 value in our data from an overall Brier score of 0.1828 or 0.1861 suggests that a substantial portion of the "error" in our data is inherent.

Outcome index variance is just the variance of the outcome variable. This is the expected value of the Brier score if all the forecast probabilities were merely the average observed outcome. Remember that a fence sitter has an expected Brier score of 0.25 ; a smarter fence sitter (who would guess $p=0.65$ for these data) would have a Brier score of 0.2275 .

The Murphy resolution measures the variation in the average outcomes across groups. If all groups have the same frequency of positive outcomes, little information in any forecast is possible, and the Murphy resolution is 0 . If groups differ markedly, the Murphy resolution is as large as 0.25 . The 0.0875 means that there is some variation but not a lot, and 0.0875 is probably higher than in most real cases. If you had groups in your data that varied between $40 \%$ and $60 \%$ positive outcomes, the Murphy resolution would be 0.01 ; between $30 \%$ and $70 \%$, it would be 0.04 .

Reliability-in-the-small measures the error that comes from the average forecast within group not measuring the average outcome within group-a classical goodness-of-fit measure, with 0 meaning a perfect fit and 1 meaning a complete lack of fit. The calculated value of 0.0461 shows some amount of lack of fit. Remember, the number is squared, and we are saying that probabilities could be just more than $\sqrt{0.0461}=0.215$ or $21.5 \%$ off.

Forecast variance measures the amount of discrimination being attempted - that is, the variation in the forecasted probabilities. A small number indicates a fence sitter making constant predictions. If the forecasts were from a logistic regression model, forecast variance would tend to increase with the amount of information available. Our pundit shows considerable forecast variance of 0.0438 (standard deviation $\sqrt{0.0438}=0.2093$ ), which is in line with the reliability-in-the-small, suggesting that the forecaster is attempting as much variation as is available in these data.

Excess forecast variance is the amount of actual forecast variance over a theoretical minimum. The theoretical minimum - called the minimum forecast variance-corresponds to forecasts of $p_{0}$ for observations ultimately observed to be negative responses and $p_{1}$ for observations ultimately observed to be positive outcomes. Moreover, $p_{0}$ and $p_{1}$ are set to the average forecasts made for the ultimate negative and positive outcomes. These predictions would be just as good as the predictions the forecaster did make, and any variation in the actual forecast probabilities above this is useless. If this number is large, above $1 \%-2 \%$, then the forecaster may be attempting more than is possible. The 0.0285 in our data suggests this possibility.

Reliability-in-the-large measures the discrepancy between the mean forecast and the observed fraction of positive outcomes. This discrepancy will be 0 for forecasts made by most statistical models-at least when measured on the same sample used for estimation-because they, by design, reproduce sample means. For our human pundit, the 0.0294 says that there is a $\sqrt{0.0294}$, or 17 -percentage-point, difference. (This difference can also be found by calculating the difference in the averages of the observed outcomes and forecast probabilities: $0.65-0.4785=0.17$.) That difference, however, is not significant, as we would see if we typed ttest win=for; see [R] ttest. If these data were larger and the bias persisted, this difference would be a critical shortcoming of the forecast.

Twice the forecast-outcome covariance is a measure of how accurately the forecast corresponds to the outcome. The concept is similar to that of $R^{2}$ in linear regression.

## Stored results

brier stores the following in $r()$ :
Scalars

```
r(p_roc) one-sided p-value for ROC area test
r(roc_area) ROC area
r(z) Spiegelhalter's z statistic
r(p) one-sided p-value for Spiegelhalter's z test
r(brier) Brier score
r(brier_s) Sanders-modified Brier score
r(sanders) Sanders resolution
r(oiv) outcome index variance
r(murphy) Murphy resolution
r(relinsm) reliability-in-the-small
r(Var_f) forecast variance
r(Var_fex) excess forecast variance
r(Var_fmin) minimum forecast variance
r(relinla) reliability-in-the-large
r(cov_2f) 2 forecast-outcome-covariance
```


## Methods and formulas

See Wilks (2019) or Schmid and Griffith (2005) for a discussion of the Brier score.
Let $d_{j}, j=1, \ldots, N$, be the observed outcomes with $d_{j}=0$ or $d_{j}=1$, and let $f_{j}$ be the corresponding forecasted probabilities that $d_{j}$ is $1,0 \leq f_{j} \leq 1$. Assume that the data are ordered so that $f_{j+1} \geq f_{j}$ (brier sorts the data to obtain this order). Divide the data into $K$ nearly equally sized groups, with group 1 containing observations 1 through $j_{2}-1$, group 2 containing observations $j_{2}$ through $j_{3}-1$, and so on.

Define

$$
\begin{aligned}
\bar{f}_{0} & =\text { average } f_{j} \text { among } d_{j}=0 \\
\bar{f}_{1} & =\text { average } f_{j} \text { among } d_{j}=1 \\
\bar{f} & =\text { average } f_{j} \\
\bar{d} & =\text { average } d_{j} \\
\widetilde{f}_{k} & =\text { average } f_{j} \text { in group } k \\
\widetilde{d}_{k} & =\text { average } d_{j} \text { in group } k \\
\widetilde{n}_{k} & =\text { number of observations in group } k
\end{aligned}
$$

The Brier score is $\sum_{j}\left(d_{j}-f_{j}\right)^{2} / N$.
The Sanders-modified Brier score is $\sum_{j}\left(d_{j}-\widetilde{f}_{k(j)}\right)^{2} / N$.

Let $p_{j}$ denote the true but unknown probability that $d_{j}=1$. Under the null hypothesis that $p_{j}=$ $f_{j}$ for all $j$, Spiegelhalter (1986) determined that the expectation and variance of the Brier score is given by the following:

$$
\begin{aligned}
E(\text { Brier }) & =\frac{1}{N} \sum_{j=1}^{N} f_{j}\left(1-f_{j}\right) \\
\operatorname{Var}(\text { Brier }) & =\frac{1}{N^{2}} \sum_{j=1}^{N} f_{j}\left(1-f_{j}\right)\left(1-2 f_{j}\right)^{2}
\end{aligned}
$$

Denoting the observed value of the Brier score by $O$ (Brier), Spiegelhalter's $z$ statistic is given by

$$
Z=\frac{O(\text { Brier })-E(\text { Brier })}{\sqrt{\operatorname{Var}(\text { Brier })}}
$$

The corresponding $p$-value is given by the upper-tail probability of $Z$ under the standard normal distribution.

The area under the ROC curve is estimated by applying the trapezoidal rule to the empirical ROC curve. This area is Wilcoxon's test statistic, so the corresponding $p$-value is just that of a one-sided Wilcoxon test of the null hypothesis that the distribution of predictions is constant across the two outcomes.

The Sanders resolution is $\sum_{k} \widetilde{n}_{k}\left\{\widetilde{d}_{k}\left(1-\widetilde{d}_{k}\right)\right\} / N$.
The outcome index variance is $\bar{d}(1-\bar{d})$.
The Murphy resolution is $\sum_{k} \widetilde{n}_{k}\left(\widetilde{d}_{k}-\bar{d}\right)^{2} / N$.
Reliability-in-the-small is $\sum_{k} \widetilde{n}_{k}\left(\widetilde{d}_{k}-\widetilde{f}_{k}\right)^{2} / N$.
The forecast variance is $\sum_{j}\left(f_{j}-\bar{f}\right)^{2} / N$.
The minimum forecast variance is $\left\{\sum_{j \in F}\left(f_{j}-\bar{f}_{0}\right)^{2}+\sum_{j \in S}\left(f_{j}-\bar{f}_{1}\right)^{2}\right\} / N$, where $F$ is the set of observations for which $d_{j}=0$ and $S$ is the complement.

The excess forecast variance is the difference between the forecast variance and the minimum forecast variance.

Reliability-in-the-large is $(\bar{f}-\bar{d})^{2}$.
Twice the outcome covariance is $2\left(\bar{f}_{1}-\bar{f}_{0}\right) \bar{d}(1-\bar{d})$.

Glenn Wilson Brier (1913-1998) was an American meteorological statistician who, after obtaining degrees in physics and statistics, was for many years head of meteorological statistics at the US Weather Bureau in Washington, DC. In the latter part of his career, he was associated with Colorado State University. Brier worked especially on verification and evaluation of predictions and forecasts, statistical decision making, the statistical theory of turbulence, the analysis of weather modification experiments, and the application of permutation techniques.

## Acknowledgment

We thank Richard Goldstein for his contributions to this improved version of brier．

## References

Brier，G．W．1950．Verification of forecasts expressed in terms of probability．Monthly Weather Review 78：1－3． https：／／doi．org／10．1175／1520－0493（1950）078〈0001：VOFEIT $\rangle 2.0 . C O ; 2$.

Goldstein，R．1996．sg55：Extensions to the brier command．Stata Technical Bulletin 32：21－22．Reprinted in Stata Technical Bulletin Reprints，vol．6，pp．133－134．College Station，TX：Stata Press．
Hadorn，D．C．，E．B．Keeler，W．H．Rogers，and R．H．Brook．1993．Assessing the Performance of Mortality Prediction Models．Santa Monica，CA：Rand．

Holloway，L．，and P．W．Mielke，Jr．1998．Glenn Wilson Brier 1913－1998．Bulletin of the American Meteorological Society 79：1438－1439．
Jolliffe，I．T．，and D．B．Stephenson，ed．2012．Forecast Verification：A Practitioner＇s Guide in Atmospheric Science． 2nd ed．Chichester，UK：Wiley．

Murphy，A．H．1973．A new vector partition of the probability score．Journal of Applied Meteorology 12：595－600． https：／／doi．org／10．1175／1520－0450（1973）012〈0595：ANVPOT $\rangle 2.0 . C O ; 2$.
＿＿．1997．Forecast verification．In Economic Value of Weather and Climate Forecasts，ed．R．W．Katz and A．H． Murphy，19－74．Cambridge：Cambridge University Press．

Redelmeier，D．A．，D．A．Bloch，and D．H．Hickam．1991．Assessing predictive accuracy：How to compare Brier scores．Journal of Clinical Epidemiology 44：1141－1146．https：／／doi．org／10．1016／0895－4356（91）90146－z．
Sanders，F．1963．On subjective probability forecasting．Journal of Applied Meteorology 2：191－201．https：／／doi．org／ 10．1175／1520－0450（1963）002〈0191：OSPF $\rangle 2.0 . \mathrm{CO} ; 2$ ．
Schmid，C．H．，and J．L．Griffith．2005．Multivariate classification rules：Calibration and discrimination．In Vol． 5 of Encyclopedia of Biostatistics，ed．P．Armitage and T．Colton，vol．2，3492－3494．Chichester，UK：Wiley． https：／／doi．org／10．1002／0470011815．b2a13049．
Spiegelhalter，D．J．1986．Probabilistic prediction in patient management and clinical trials．Statistics in Medicine 5： 421－433．https：／／doi．org／10．1002／sim． 4780050506.

Von Storch，H．，and F．W．Zwiers．1999．Statistical Analysis in Climate Research．Cambridge：Cambridge University Press．
Wilks，D．S．2019．Statistical Methods in the Atmospheric Sciences．4th ed．Amsterdam：Elsevier．
Yates，J．F．1982．External correspondence：Decompositions of the mean probability score．Organizational Behavior and Human Performance 30：132－156．https：／／doi．org／10．1016／0030－5073（82）90237－9．

## Also see

［R］logistic－Logistic regression，reporting odds ratios
［R］logit－Logistic regression，reporting coefficients
［R］probit－Probit regression

## Title

bsample - Sampling with replacement

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | References | Also see |

## Description

bsample replaces the data in memory with a bootstrap sample (random sample with replacement) drawn from the current dataset. Clusters can be optionally sampled during each replication in place of observations. Bootstrap samples can also be selected within strata.

## Quick start

Bootstrap sample with the same number of observations as the current dataset bsample

Same as above, but restrict to just 100 observations
bsample 100
Stratified bootstrap sample of 100 observations at each level of svar bsample 100, strata(svar)

Bootstrap sample of 10 clusters identified by values of cvar bsample 10, cluster(cvar)

Same as above, but create a new unique ID code for sampled clusters and store it in cvar2 bsample 10, cluster(cvar) idcluster(cvar2)

## Menu

Statistics $>$ Resampling $>$ Draw bootstrap sample

## Syntax

```
bsample [exp][if][in] [,options]
```

where $\exp$ is a standard Stata expression specifying the size of the sample; see [U] 13 Functions and expressions.
exp must be less than or equal to $\_\mathrm{N}$ (the number of observations; [U] 13.4 System variables (_variables)) when neither the cluster() nor the strata() option is specified. _N is the default when $\exp$ is not specified.

Observations that do not meet the optional if and in criteria are dropped from the resulting dataset.

options

Description
strata (varlist)
cluster (varlist)
idcluster (newvar)
weight (varname)

## Options

strata (varlist) specifies the variables identifying strata. If strata() is specified, bootstrap samples are selected within each stratum, and $\exp$ must be less than or equal to $\_\mathrm{N}$ within the defined strata.
cluster (varlist) specifies the variables identifying resampling clusters. If cluster() is specified, the sample drawn during each replication is a bootstrap sample of clusters, and exp must be less than or equal to $N_{c}$ (the number of clusters identified by the cluster() option). If strata() is also specified, exp must be less than or equal to the number of within-strata clusters.
idcluster (newvar) creates a new variable containing a unique identifier for each resampled cluster. weight (varname) specifies a variable in which the sampling frequencies will be placed. varname must be an existing variable, which will be replaced. After bsample, varname can be used as an fweight in any Stata command that accepts fweights, which can speed up resampling for commands like regress and summarize. This option cannot be combined with idcluster().
By default, bsample replaces the data in memory with the sampled observations; however, specifying the weight () option causes only the specified varname to be changed.

## Remarks and examples

Below is a series of examples illustrating how bsample is used with various sampling schemes.

## > Example 1: Bootstrap sampling

We have data on the characteristics of hospital patients and wish to draw a bootstrap sample of 200 patients. We type

```
. use https://www.stata-press.com/data/r18/bsample1
. bsample 200
. count
200
```

> Example 2: Stratified samples with equal sizes
Among the variables in our dataset is female, an indicator for the female patients. To get a bootstrap sample of 200 female patients and 200 male patients, we type

```
. use https://www.stata-press.com/data/r18/bsample1, clear
. bsample 200, strata(female)
. tabulate female
\begin{tabular}{r|rrr}
\begin{tabular}{r} 
Indicator \\
for female
\end{tabular} & Freq. & Percent & Cum. \\
\hline Male & 200 & 50.00 & 50.00 \\
Female & 200 & 50.00 & 100.00 \\
\hline Total & 400 & 100.00 &
\end{tabular}
```


## > Example 3: Stratified samples with unequal sizes

To sample 300 females and 200 males, we must generate a variable that is 300 for females and 200 for males and then use this variable in $\exp$ when we call bsample.

```
. use https://www.stata-press.com/data/r18/bsample1, clear
. generate nsamp = cond(female,300,200)
. bsample nsamp, strata(female)
. tabulate female
\begin{tabular}{r|rrr}
\begin{tabular}{r} 
Indicator \\
for female
\end{tabular} & Freq. & Percent & Cum. \\
\hline \begin{tabular}{r} 
Male
\end{tabular} & 200 & 40.00 & 40.00 \\
Female & 300 & 60.00 & 100.00 \\
\hline Total & 500 & 100.00 &
\end{tabular}
```

> Example 4: Stratified samples with proportional sizes
Our original dataset has 2,392 males and 3,418 females.
. use https://www.stata-press.com/data/r18/bsample1, clear
. tabulate female

| Indicator <br> for female | Freq. | Percent | Cum. |
| ---: | ---: | ---: | ---: |
| Male | 2,392 | 41.17 | 41.17 |
| Female | 3,418 | 58.83 | 100.00 |
| Total | 5,810 | 100.00 |  |

To sample $10 \%$ from females and males, we type

```
. bsample round(0.1*_N), strata(female)
```

bsample requires that the specified size of the sample be an integer, so we use the round () function to obtain the nearest integer to $0.1 \times 2392$ and $0.1 \times 3418$. Our sample now has 239 males and 342 females:

| . tabulate female |  |  |  |
| ---: | ---: | ---: | ---: |
| Indicator <br> for female | Freq. | Percent | Cum. |
| Male 239 41.14 41.14 <br> Female 342 58.86 100.00 <br> Total 581 100.00 . |  |  |  |

## Example 5: Samples satisfying a condition

For a bootstrap sample of 200 female patients, we type

```
. use https://www.stata-press.com/data/r18/bsample1, clear
. bsample 200 if female
. tabulate female
\begin{tabular}{r|rrr}
\begin{tabular}{r} 
Indicator \\
for female
\end{tabular} & Freq. & Percent & Cum. \\
\hline Female & 200 & 100.00 & 100.00 \\
\hline Total & 200 & 100.00 &
\end{tabular}
```

> Example 6: Generating frequency weights
To identify the sampled observations using frequency weights instead of dropping unsampled observations, we use the weight() option (we will need to supply it an existing variable name) and type

```
. use https://www.stata-press.com/data/r18/bsample1, clear
. set seed 1234
. generate fw = .
(5,810 missing values generated)
. bsample 200 if female, weight(fw)
. tabulate fw female
\begin{tabular}{r|rr|r} 
& \begin{tabular}{r} 
Indicator for female \\
Male
\end{tabular} & Female & Total \\
\hline 0 & 2,392 & 3,222 & 5,614 \\
1 & 0 & 192 & 192 \\
2 & 0 & 4 & 4 \\
\hline Total & 2,392 & 3,418 & 5,810
\end{tabular}
```

Note that $(192 \times 1)+(4 \times 2)=200$.

## > Example 7: Oversampling observations

bsample requires the expression in $\exp$ to evaluate to a number that is less than or equal to the number of observations. To sample twice as many male and female patients as there are already in memory, we must expand the data before using bsample. For example,

```
. use https://www.stata-press.com/data/r18/bsample1, clear
. set seed 1234
- expand 2
(5,810 observations created)
. bsample, strata(female)
. tabulate female
\begin{tabular}{r|rcr}
\begin{tabular}{r} 
Indicator \\
for female
\end{tabular} & Freq. & Percent & Cum. \\
\hline Male & 4,784 & 41.17 & 41.17 \\
Female & 6,836 & 58.83 & 100.00 \\
\hline Total & 11,620 & 100.00 &
\end{tabular}
```

$>$ Example 8: Stratified oversampling with unequal sizes
To sample twice as many female patients as male patients, we must expand the records for the female patients because there are less than twice as many of them as there are male patients, but first put the number of observed male patients in a local macro. After expanding the female records, we generate a variable that contains the number of observations to sample within the two groups.

```
. use https://www.stata-press.com/data/r18/bsample1, clear
. set seed 1234
. count if !female
    2,392
. local nmale = r(N)
. expand 2 if female
(3,418 observations created)
. generate nsamp = cond(female, 2*'nmale','nmale')
. bsample nsamp, strata(female)
. tabulate female
\begin{tabular}{r|rrr}
\begin{tabular}{r} 
Indicator \\
for female
\end{tabular} & Freq. & Percent & Cum. \\
\hline Male & 2,392 & 33.33 & 33.33 \\
Female & 4,784 & 66.67 & 100.00 \\
\hline Total & 7,176 & 100.00 &
\end{tabular}
```


## > Example 9: Oversampling of clusters

For clustered data, sampling more clusters than are present in the original dataset requires more than just expanding the data. To illustrate, suppose we wanted a bootstrap sample of eight clusters from a dataset consisting of five clusters of observations.

```
. use https://www.stata-press.com/data/r18/bsample2, clear
```

. tabstat $x$, stat( $n$ mean) by (group)
Summary for variables: x
Group variable: group

| group | N | Mean |
| ---: | ---: | ---: |
| A | 15 | -.3073028 |
| B | 10 | -.00984 |
| C | 11 | .0810985 |
| D | 11 | -.1989179 |
| E | 29 | -.095203 |
| Total | 76 | -.1153269 |

bsample will complain if we simply expand the dataset.

```
. use https://www.stata-press.com/data/r18/bsample2
. expand 3
(152 observations created)
. bsample 8, cluster(group)
resample size must not be greater than number of clusters
r(498);
```

Expanding the data will only partly solve the problem. We also need a new variable that uniquely identifies the copied clusters. We use the expandcl command to accomplish both these tasks; see [D] expandcl.

```
. use https://www.stata-press.com/data/r18/bsample2, clear
. set seed 1234
. expandcl 2, generate(expgroup) cluster(group)
(76 observations created)
. tabstat x, stat(n mean) by(expgroup)
Summary for variables: x
Group variable: expgroup (New cluster ID from expandcl)
\begin{tabular}{r|rr} 
expgroup & N & Mean \\
\hline 1 & \(15-.3073028\) \\
2 & 15 & -.3073028 \\
3 & 10 & -.00984 \\
4 & 10 & -.00984 \\
5 & 11 & .0810985 \\
6 & 11 & .0810985 \\
7 & \(11-.1989179\) \\
8 & 11 & -.1989179 \\
9 & 29 & -.095203 \\
10 & 29 & -.095203 \\
\hline Total & \(152-.1153269\) \\
\hline
\end{tabular}
```

. generate $\mathrm{fw}=$.
(152 missing values generated)
. bsample 8, cluster(expgroup) weight(fw)
. tabulate fw group

| fw | A | B | Croup | D | E | Total |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| 0 | 15 | 10 | 22 | 11 | 58 | 116 |
| 1 | 0 | 0 | 0 | 11 | 0 | 11 |
| 2 | 15 | 0 | 0 | 0 | 0 | 15 |
| 5 | 0 | 10 | 0 | 0 | 0 | 10 |
| Total | 30 | 20 | 22 | 22 | 58 | 152 |

The results from tabulate on the generated frequency weight variable versus the original cluster ID (group) show us that the bootstrap sample contains one copy of cluster A, one copy of cluster B, two copies of cluster $C$, two copies of cluster $D$, and two copies of cluster $E(1+1+2+2+2=8)$.

## > Example 10: Stratified oversampling of clusters

Suppose that we have a dataset containing two strata with five clusters in each stratum, but the cluster identifiers are not unique between the strata. To get a stratified bootstrap sample with eight clusters in each stratum, we first use expandcl to expand the data and get a new cluster ID variable. We use cluster (strid group) in the call to expandcl; this action will uniquely identify the $2 * 5=10$ clusters across the strata.
. use https://www.stata-press.com/data/r18/bsample2, clear
. set seed 1234
. tabulate group strid

| group | strid <br> 1 | 2 | Total |
| ---: | ---: | ---: | ---: |
| A | 7 | 8 | 15 |
| B | 5 | 5 | 10 |
| C | 5 | 6 | 11 |
| D | 5 | 6 | 11 |
| E | 14 | 15 | 29 |
| Total | 36 | 40 | 76 |

. expandcl 2, generate(expgroup) cluster(strid group)
(76 observations created)
Now, we can use bsample with the expanded data, stratum ID variable, and new cluster ID variable.

```
. generate fw = .
(152 missing values generated)
```

. bsample 8, cluster(expgroup) str(strid) weight(fw)
. by strid, sort: tabulate fw group


The results from by strid: tabulate on the generated frequency weight variable versus the original cluster ID (group) show us how many times each cluster was sampled for each stratum. For stratum 1 , the bootstrap sample contains two copies of cluster A, one copy of cluster B, two copies of cluster C, one copy of cluster $D$, and two copies of cluster $E(2+1+2+1+2=8)$. For stratum 2 , the bootstrap sample contains one copy of cluster A, zero copies of cluster B, three copies of cluster C, one copy of cluster $D$, and three copies of cluster $E(1+0+3+1+3=8)$.

## References

Gould, W. W. 2012a. Using Stata's random-number generators, part 2: Drawing without replacement. The Stata Blog: Not Elsewhere Classified.
https://blog.stata.com/2012/08/03/using-statas-random-number-generators-part-2-drawing-without-replacement/.
__. 2012b. Using Stata's random-number generators, part 3: Drawing with replacement. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2012/08/29/using-statas-random-number-generators-part-3-drawing-with-replacement/.

## Also see

[R] bootstrap - Bootstrap sampling and estimation
[R] bstat - Report bootstrap results
[R] simulate - Monte Carlo simulations
[D] sample - Draw random sample
[D] splitsample - Split data into random samples

## Title

bstat — Report bootstrap results

| Description | Menu | Syntax | Options |
| :--- | :--- | :--- | :--- |
| Remarks and examples | Stored results | References | Also see |

## Description

bstat is a programmer's command that computes and displays estimation results from bootstrap statistics. For each variable in varlist, bstat computes a covariance matrix, estimates bias, and constructs normal confidence intervals (CIs), percentile CIs, bias-corrected (BC) CIs, and bias-corrected and accelerated $\left(\mathrm{BC}_{a}\right)$ CIs using a bootstrap dataset in memory or on disk. The computed CIs can be displayed using estat bootstrap; see [R] bootstrap postestimation.
bstat without varlist replays results from the last bootstrap estimation when results are stored in e().

## Menu

Statistics $>$ Resampling $>$ Report bootstrap results

## Syntax

Bootstrap statistics from variables
bstat [varlist] [if] [in] [, options]
Bootstrap statistics from file
bstat [namelist][using filename] [if] [in] [, options]
options Description
Main

* stat (vector) observed values for each statistic
* accel (vector) acceleration values for each statistic
*ties adjust $\mathrm{BC} / \mathrm{BCa}$ confidence intervals for ties
*mse use MSE formula for variance estimation
Reporting
level(\#)
n (\#) \# of observations from which bootstrap samples were taken
notable suppress table of results
noheader suppress table header
nolegend suppress table legend
verbose display the full table legend
title (text) use text as title for bootstrap results
display_options
control column formats and line width
*Starred options and qualifiers using, if, and in require a bootstrap dataset.
collect is allowed; see [U] 11.1.10 Prefix commands.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.


## Options

stat (vector) specifies the observed value of each statistic (that is, the value of the statistic using the original dataset).
accel (vector) specifies the acceleration of each statistic, which is used to construct $\mathrm{BC}_{a}$ CIs.
ties specifies that bstat adjust for ties in the replicate values when computing the median bias used to construct BC and BCa CIs.
mse specifies that bstat compute the variance by using deviations of the replicates from the observed value of the statistics. By default, bstat computes the variance by using deviations from the average of the replicates.
$\qquad$ Reporting
level (\#); see [R] Estimation options.
$\mathrm{n}(\#)$ specifies the number of observations from which bootstrap samples were taken. This value is used in no calculations but improves the table header when this information is not saved in the bootstrap dataset.
notable suppresses the display of the output table.
noheader suppresses the display of the table header. This option implies nolegend. nolegend suppresses the display of the table legend.
verbose specifies that the full table legend be displayed. By default, coefficients and standard errors are not displayed.
title (text) specifies a title to be displayed above the table of bootstrap results; the default title is Bootstrap results.
display_options: cformat (\%fmt), pformat (\%fmt), sformat (\%fmt), and nolstretch; see [R] Estimation options.

## Remarks and examples

Remarks are presented under the following headings:
Bootstrap datasets
Creating a bootstrap dataset

## Bootstrap datasets

Although bstat allows you to specify the observed value and acceleration of each bootstrap statistic via the stat() and accel() options, programmers may be interested in what bstat uses when these options are not supplied.

When working from a bootstrap dataset, bstat first checks the data characteristics (see $[\mathrm{P}]$ char) that it understands:
_dta[bs_version] identifies the version of the bootstrap dataset. This characteristic may be empty (not defined), 2, or 3; otherwise, bstat will quit and display an error message. This version tells bstat which other characteristics to look for in the bootstrap dataset.
bstat uses the following characteristics from version 3 bootstrap datasets:

```
_dta[N]
_dta[N_strata]
_dta[N_cluster]
_dta[command]
varname[observed]
varname[acceleration]
varname[expression]
```

bstat uses the following characteristics from version 2 bootstrap datasets:
_dta [N]
_dta[N_strata]
_dta[N_cluster]
varname[observed]
varname[acceleration]
An empty bootstrap dataset version implies that the dataset was created by the bstrap command in a version of Stata earlier than Stata 8. Here bstat expects varname [bstrap] to contain the observed value of the statistic identified by varname (varname [observed] in version 2). All other characteristics are ignored.
_dta [N] is the number of observations in the observed dataset. This characteristic may be overruled by specifying the n () option.
_dta[N_strata] is the number of strata in the observed dataset.
_dta[N_cluster] is the number of clusters in the observed dataset.
_dta[command] is the command used to compute the observed values of the statistics.
varname [observed] is the observed value of the statistic identified by varname. To specify a different value, use the stat() option.
varname[acceleration] is the estimate of acceleration for the statistic identified by varname. To specify a different value, use the accel() option.
varname [expression] is the expression or label that describes the statistic identified by varname.

## Creating a bootstrap dataset

Suppose that we are interested in obtaining bootstrap statistics by resampling the residuals from a regression (which is not possible with the bootstrap command). After loading some data, we run a regression, save some results relevant to the bstat command, and save the residuals in a new variable, res.

. matrix $b=e(b)$
. local $\mathrm{n}=\mathrm{e}(\mathrm{N})$
. predict res, residuals

We can resample the residual values in res by generating a random observation ID (rid), generate a new response variable ( y ), and run the original regression with the new response variables.

```
. set seed 54321
. generate rid = int(_N*runiform())+1
. matrix score double y = b
. replace y = y + res[rid]
(74 real changes made)
```

. regress y weight length

| Source | SS | df | MS |
| ---: | :---: | ---: | :---: |
| Model | 1695.70314 | 2 | 847.851568 |
| Residual | 601.341031 | 71 | 8.46959199 |
| Total | 2297.04417 | 73 | 31.4663585 |


| Number of obs | $=$ | 74 |
| :--- | :--- | ---: |
| F $(2,71)$ | $=$ | 100.11 |
| Prob > F | $=$ | 0.0000 |
| R-squared | $=$ | 0.7382 |
| Adj R-squared | $=$ | 0.7308 |
| Root MSE | $=$ | 2.9103 |


| y | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| weight | -.0029676 | .0013521 | -2.19 | 0.031 | -.0056636 | -.0002716 |
| length | -.1158425 | .047194 | -2.45 | 0.017 | -.2099446 | -.0217404 |
| _cons | 51.72451 | 5.190075 | 9.97 | 0.000 | 41.3758 | 62.07323 |

Instead of programming this resampling inside a loop, it is much more convenient to write a short program and use the simulate command; see [R] simulate. In the following, mysim_r requires the user to specify a coefficient vector and a residual variable. mysim_r then retrieves the list of predictor variables (removing _cons from the list), generates a new temporary response variable with the resampled residuals, and regresses the new response variable on the predictors.

```
program mysim_r
    version 18.0 // (or version 18.5 for StataNow)
    syntax name(name=bvector), res(varname)
    tempvar y rid
    local xvars : colnames 'bvector'
    local cons _cons
    local xvars : list xvars - cons
    matrix score double 'y' = 'bvector'
    generate long 'rid' = int(_N*runiform()) + 1
    replace 'y' = 'y' + 'res'['rid']
    regress 'y' 'xvars'
end
```

We can now give mysim_r a test run, but we first set the random-number seed (to reproduce results).

```
. set seed 54321
. mysim_r b, res(res)
(74 real changes made)
```

| Source | SS | df | MS |
| ---: | :---: | ---: | :---: |
| Model | 1695.70314 | 2 | 847.851568 |
| Residual | 601.341031 | 71 | 8.46959199 |
| Total | 2297.04417 | 73 | 31.4663585 |


| Number of obs | $=$ | 74 |
| :--- | :--- | ---: |
| $\mathrm{~F}(2,71)$ | $=$ | 100.11 |
| Prob > F | $=$ | 0.0000 |
| R-squared | $=$ | 0.7382 |
| Adj R-squared | $=$ | 0.7308 |
| Root MSE | $=$ | 2.9103 |


| _-000000 | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{tl}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | ---: | ---: |
| weight | -.0029676 | .0013521 | -2.19 | 0.031 | -.0056636 | -.0002716 |
| length | -.1158425 | .047194 | -2.45 | 0.017 | -.2099446 | -.0217404 |
| _cons | 51.72451 | 5.190075 | 9.97 | 0.000 | 41.3758 | 62.07323 |

Now that we have a program that will compute the results we want, we can use simulate to generate a bootstrap dataset and bstat to display the results.

```
. set seed 54321
. simulate, reps(200) nodots: mysim_r b, res(res)
    Command: mysim_r b, res(res)
. bstat, stat(b) n('n')
Bootstrap results Number of obs = 74
```

Replications $=200$

|  | Observed <br> coefficient | Bootstrap <br> std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ |  | Normal-based <br> [95\% conf. interval] |
| ---: | ---: | :--- | :---: | :---: | ---: | ---: |
| _b_weight | -.0038515 | .0014673 | -2.62 | 0.009 | -.0067274 | -.0009756 |
| _b_length | -.0795935 | .0509772 | -1.56 | 0.118 | -.1795069 | .0203199 |
| _b_cons | 47.88487 | 5.650947 | 8.47 | 0.000 | 36.80922 | 58.96053 |

Finally, we see that simulate created some of the data characteristics recognized by bstat. All we need to do is correctly specify the version of the bootstrap dataset, and bstat will automatically use the relevant data characteristics.
. char list
_dta[rngstate]: XAA000000000000d431c5e5401775ee9b9e24b2604d4885..
_dta[command]:
_b_weight[is_eexp]:
_b_weight[colname]:
_b_weight[coleq]:
_b_weight[expression]:
_b_length[is_eexp]:
_b_length[colname]:
_b_length[coleq]:
_b_length [expression]:
_b_cons[is_eexp]:
_b_cons[colname]:
_b_cons [coleq]:
_b_cons [expression]:

- char _dta[bs_version] 3
. bstat, stat(b) n('n')
Bootstrap results $\quad$ Number of obs $=74$ Replications $=200$
Command: mysim_r b, res(res)

|  | Observed <br> coefficient | Bootstrap <br> std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ |  | Normal-based <br> [95\% conf. interval] |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| weight | -.0038515 | .0014673 | -2.62 | 0.009 | -.0067274 | -.0009756 |
| length | -.0795935 | .0509772 | -1.56 | 0.118 | -.1795069 | .0203199 |
| _cons | 47.88487 | 5.650947 | 8.47 | 0.000 | 36.80922 | 58.96053 |

See Poi (2004) for another example of residual resampling.

## Stored results

bstat stores the following in e() :
Scalars

$$
e(N)
$$

e(N_reps)
e(N_misreps)
e(N_strata)
e(N_clust)
e(k_aux)
e(k_eq)
e(k_exp)
e(k_eexp)
e(k_extra)
e(level)
e(bs_version)
$e$ (rank)
sample size
number of complete replications
number of incomplete replications
number of strata
number of clusters
number of auxiliary parameters
number of equations in $e(b)$
number of standard expressions
number of extended expressions (i.e., _b)
number of extra equations beyond the original ones from e(b)
confidence level for bootstrap CIs
version for bootstrap results
rank of e(V)
Macros
e(cmd)
bstat
e(command)
e(cmdline)
$e($ title)
e(exp\#)
e(prefix)
e(ties)
e(mse)
from _dta [command]
command as typed
title in estimation output
expression for the \#th statistic
bootstrap
ties, if specified
mse, if specified
e(vce)
e(vcetype)
bootstrap
e(properties)
title used to label Std. err.
b V
Matrices
e(b)
e(b_bs)
e(reps)
e(bias)
e(se)
e(z0)
e(accel)
e(ci_normal)
e(ci_percentile)
e(ci_bc)
e(ci_bca)
e(V)
observed statistics
bootstrap estimates
number of nonmissing results
estimated biases
estimated standard errors
median biases
estimated accelerations
normal-approximation CIs
percentile CIs
bias-corrected CIs
bias-corrected and accelerated CIs
bootstrap variance-covariance matrix

## References

Ng, E. S.-W., R. Grieve, and J. R. Carpenter. 2013. Two-stage nonparametric bootstrap sampling with shrinkage correction for clustered data. Stata Journal 13: 141-164.
Poi, B. P. 2004. From the help desk: Some bootstrapping techniques. Stata Journal 4: 312-328.

## Also see

[R] bootstrap postestimation - Postestimation tools for bootstrap
$[R]$ bootstrap - Bootstrap sampling and estimation
[R] bsample - Sampling with replacement

## Title

centile - Report centile and confidence interval

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| Acknowledgment | References | Also see |  |

## Description

centile estimates specified centiles and calculates confidence intervals. If no varlist is specified, centile calculates centiles for all the variables in the dataset. If no centiles are specified, medians are reported.

By default, centile uses a binomial method for obtaining confidence intervals that makes no assumptions about the underlying distribution of the variable.

## Quick start

50th percentile with $95 \%$ confidence intervals for v1 and v2
centile v1 v2
For all variables in the dataset
centile
25th, 50th, and 75th percentiles of v1
centile v1, centile (25 50 75)
10th, 20th, 30th, ..., 90th percentiles of v1
centile v1, centile(10(10)90)
Force confidence limits to fall on sample values
centile v1 v2, cci
Confidence intervals based on standard errors for a normal-distribution quantile
centile v1 v2, normal
Centile and confidence intervals based on mean and standard deviation
centile v1 v2, meansd
Replace data in memory with centiles for groups defined by categorical variable cvar
statsby, by (cvar) clear: centile v1, centile(25 50 75)

## Menu

Statistics $>$ Summaries, tables, and tests $>$ Summary and descriptive statistics $>$ Centiles with Cls

## Syntax

$$
\text { centile }[\text { varlist }][\text { if }][\text { in }][, \text { options }]
$$

options
Main
centile(numlist)
Options
cci
normal
meansd
level(\#)

Description
report specified centiles; default is centile(50)
binomial exact; conservative confidence interval normal, based on observed centiles
normal, based on mean and standard deviation
set confidence level; default is level (95)
by, collect, and statsby are allowed; see [U] 11.1.10 Prefix commands.

## Options

$\qquad$
centile(numlist) specifies the centiles to be reported. The default is to display the 50th centile. Specifying centile(5) requests that the fifth centile be reported. Specifying centile (5 50 95 ) requests that the 5 th, 50 th, and 95 th centiles be reported. Specifying centile (10 (10) 90) requests that the 10 th, 20 th, $\ldots, 90$ th centiles be reported; see [U] 11.1.8 numlist.

Options
cci (conservative confidence interval) forces the confidence limits to fall exactly on sample values. Confidence intervals displayed with the cci option are slightly wider than those with the default (nocci) option.
normal causes the confidence interval to be calculated by using a formula for the standard error of a normal-distribution quantile given by Kendall and Stuart (1969, 237). The normal option is useful when you want empirical centiles - that is, centiles based on sample order statistics rather than on the mean and standard deviation - and are willing to assume normality.
meansd causes the centile and confidence interval to be calculated based on the sample mean and standard deviation, and it assumes normality.
level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [R] level.

## Remarks and examples

The $q$ th centile of a continuous random variable, $X$, is defined as the value of $C_{q}$, which fulfills the condition $\operatorname{Pr}\left(X \leq C_{q}\right)=q / 100$. The value of $q$ must be in the range $0<q<100$, though $q$ is not necessarily an integer. By default, centile estimates $C_{q}$ for the variables in varlist and for the values of $q$ given in centile (numlist). It makes no assumptions about the distribution of $X$ and, if necessary, uses linear interpolation between neighboring sample values. Extreme centiles (for example, the 99th centile in samples smaller than 100) are fixed at the minimum or maximum sample value. An "exact" confidence interval for $C_{q}$ is also given, using the binomial-based method described below in Methods and formulas and in Conover (1999, 143-148). Again linear interpolation is used to improve the accuracy of the estimated confidence limits, but extremes are fixed at the minimum or maximum sample value.

You can prevent centile from interpolating when calculating binomial-based confidence intervals by specifying cci. The resulting intervals are generally wider than with the default; that is, the coverage (confidence level) tends to be greater than the nominal value (given as usual by level(\#), by default $95 \%$ ).

If the data are believed to be normally distributed (a common case), there are two alternative methods for estimating centiles. If normal is specified, $C_{q}$ is calculated, as just described, but its confidence interval is based on a formula for the standard error (se) of a normal-distribution quantile given by Kendall and Stuart (1969, 237). If meansd is alternatively specified, $C_{q}$ is estimated as $\bar{x}+z_{q} \times s$, where $\bar{x}$ and $s$ are the sample mean and standard deviation, respectively, and $z_{q}$ is the $q$ th centile of the standard normal distribution (for example, $z_{95}=1.645$ ). The confidence interval is derived from the se of the estimate of $C_{q}$.

## Example 1

Using auto.dta, we estimate the 5th, 50th, and 95th centiles of the price variable:

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. format price %8.2fc
. centile price, centile(5 50 95)
```

| Variable | Obs | Percentile | Centile | Binom. interp. |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
|  | [95\% conf. interval] |  |  |  |  |
| price | 74 | 5 | $3,727.75$ | $3,291.23$ | $3,914.16$ |
|  |  | 50 | $5,006.50$ | $4,593.57$ | $5,717.90$ |
|  |  | 95 | $13,498.00$ | $11,061.53$ | $15,865.30$ |

summarize produces somewhat different results from centile; see Methods and formulas.
. summarize price, detail
Price

|  | Percentiles | Smallest |  |  |
| ---: | ---: | ---: | :--- | ---: |
| $1 \%$ | 3291 | 3291 |  | 74 |
| $5 \%$ | 3748 | 3299 |  | 74 |
| $10 \%$ | 3895 | 3667 | Obs |  |
| $25 \%$ | 4195 | 3748 | Sum of wgt. |  |
| $50 \%$ | 5006.5 |  | Mean | 6165.257 |
|  |  | Largest | Std. dev. | 2949.496 |
| $75 \%$ | 6342 | 13466 |  | 8699526 |
| $90 \%$ | 11385 | 13594 | Variance | 1.653434 |
| $95 \%$ | 13466 | 14500 | Skewness | 1.3 |
| $99 \%$ | 15906 | 15906 | Kurtosis | 4.819188 |

The confidence limits produced by using the cci option are slightly wider than those produced without this option:

```
. centile price, c(5 50 95) cci
```

| Variable | Obs | Percentile | Centile | Binomial exact <br> [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| price | 74 | 5 | $3,727.75$ | $3,291.00$ | $3,955.00$ |
|  |  | 50 | $5,006.50$ | $4,589.00$ | $5,719.00$ |
|  |  | 95 | $13,498.00$ | $10,372.00$ | $15,906.00$ |

If we are willing to assume that price is normally distributed, we could include either the normal or the meansd option:

```
. centile price, c(5 50 95) normal
```

| Variable | - Normal, based on observed centiles - |  |  |  |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
|  | Obs | Percentile | Centile | [95\% conf. interval] |  |
| price | 74 | 5 | $3,727.75$ | $3,211.19$ | $4,244.31$ |
|  |  | 50 | $5,006.50$ | $4,096.68$ | $5,916.32$ |
|  |  | 95 | $13,498.00$ | $5,426.81$ | $21,569.19$ |

. centile price, c(5 50 95) meansd

| Variable | Obs | Percentile | Normal, based on mean and std. dev. Centile <br> [95\% conf. interval] |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: |
| price | 74 | 5 | 1,313.77 | 278.93 | 2,348.61 |
|  |  | 50 | 6,165.26 | 5,493.24 | 6,837.27 |
|  |  | 95 | 11,016.75 | 9,981.90 | 12,051.59 |

With the normal option, the centile estimates are, by definition, the same as before. The confidence intervals for the 5th and 50th centiles are similar to the previous ones, but the interval for the 95th centile is different. The results using the meansd option also differ from both previous sets of estimates.

We can use sktest (see [R] sktest) to check the correctness of the normality assumption:
. sktest price
Skewness and kurtosis tests for normality

| Variable | Obs | $\operatorname{Pr}$ (skewness) | Pr(kurtosis) | Adj chi2(2) | Prob>chi2 |
| ---: | ---: | ---: | ---: | ---: | ---: |
| price | 74 | 0.0000 | 0.0127 | 21.77 | 0.0000 |

sktest reveals that price is definitely not normally distributed, so the normal assumption is not reasonable, and the normal and meansd options are not appropriate for these data. We should rely on the results from the default choice, which does not assume normality. If the data are normally distributed, however, the precision of the estimated centiles and their confidence intervals will be ordered (best) meansd > normal > [default] (worst). The normal option is useful when we really do want empirical centiles (that is, centiles based on sample order statistics rather than on the mean and standard deviation) but are willing to assume normality.

## Stored results

centile stores the following in $r()$ :
Scalars
$r(N) \quad$ number of observations
$r$ (n_cent) number of centiles requested
r(c_\#) value of \# centile
r(lb_\#) \#-requested centile lower confidence bound
r(ub_\#) \#-requested centile upper confidence bound
Macros
r(centiles) centiles requested

## Methods and formulas

Methods and formulas are presented under the following headings:
Default case
Normal case
meansd case

## Default case

The calculation is based on the method of Mood and Graybill (1963, 408). Let $x_{1} \leq x_{2} \leq \cdots \leq x_{n}$ be a sample of size $n$ arranged in ascending order. Denote the estimated $q$ th centile of the $x$ 's as $c_{q}$. We require that $0<q<100$. Let $R=(n+1) q / 100$ have integer part $r$ and fractional part $f$; that is, $r=\operatorname{int}(R)$ and $f=R-r$. (If $R$ is itself an integer, then $r=R$ and $f=0$.) Note that $0 \leq r \leq n$. For convenience, define $x_{0}=x_{1}$ and $x_{n+1}=x_{n} . C_{q}$ is estimated by

$$
c_{q}=x_{r}+f \times\left(x_{r+1}-x_{r}\right)
$$

that is, $c_{q}$ is a weighted average of $x_{r}$ and $x_{r+1}$. Loosely speaking, a (conservative) $p \%$ confidence interval for $C_{q}$ involves finding the observations ranked $t$ and $u$, which correspond, respectively, to the $\alpha=(100-p) / 200$ and $1-\alpha$ quantiles of a binomial distribution with parameters $n$ and $q / 100$, that is, $\mathrm{B}(n, q / 100)$. More precisely, define the $i$ th value $(i=0, \ldots, n$ ) of the cumulative binomial distribution function as $F_{i}=\operatorname{Pr}(S \leq i)$, where $S$ has distribution $B(n, q / 100)$. For convenience, let $F_{-1}=0$ and $F_{n+1}=1 . t$ is found such that $F_{t} \leq \alpha$ and $F_{t+1}>\alpha$, and $u$ is found such that $1-F_{u} \leq \alpha$ and $1-F_{u-1}>\alpha$.

With the cci option in force, the (conservative) confidence interval is $\left(x_{t+1}, x_{u+1}\right)$, and its actual coverage probability is $F_{u}-F_{t}$.

The default case uses linear interpolation on the $F_{i}$ as follows. Let

$$
\begin{aligned}
g & =\left(\alpha-F_{t}\right) /\left(F_{t+1}-F_{t}\right) \\
h & =\left\{\alpha-\left(1-F_{u}\right)\right\} /\left\{\left(1-F_{u-1}\right)-\left(1-F_{u}\right)\right\} \\
& =\left(\alpha-1+F_{u}\right) /\left(F_{u}-F_{u-1}\right)
\end{aligned}
$$

The interpolated lower and upper confidence limits $\left(c_{q L}, c_{q U}\right)$ for $C_{q}$ are

$$
\begin{aligned}
c_{q L} & =x_{t+1}+g \times\left(x_{t+2}-x_{t+1}\right) \\
c_{q U} & =x_{u+1}-h \times\left(x_{u+1}-x_{u}\right)
\end{aligned}
$$

Suppose that we want a $95 \%$ confidence interval for the median of a sample of size $13 . n=13$, $q=50, p=95, \alpha=0.025, R=14 \times 50 / 100=7$, and $f=0$. Therefore, the median is the 7th observation. Some example data, $x_{i}$, and the values of $F_{i}$ are as follows:

| $i$ | $F_{i}$ | $1-F_{i}$ | $x_{i}$ | $i$ | $F_{i}$ | $1-F_{i}$ | $x_{i}$ |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| 0 | 0.0001 | 0.9999 | - | 7 | 0.7095 | 0.2905 | 33 |
| 1 | 0.0017 | 0.9983 | 5 | 8 | 0.8666 | 0.1334 | 37 |
| 2 | 0.0112 | 0.9888 | 7 | 9 | 0.9539 | 0.0461 | 45 |
| 3 | 0.0461 | 0.9539 | 10 | 10 | 0.9888 | 0.0112 | 59 |
| 4 | 0.1334 | 0.8666 | 15 | 11 | 0.9983 | 0.0017 | 77 |
| 5 | 0.2905 | 0.7095 | 23 | 12 | 0.9999 | 0.0001 | 104 |
| 6 | 0.5000 | 0.5000 | 28 | 13 | 1.0000 | 0.0000 | 211 |

The median is $x_{7}=33$. Also, $F_{2} \leq 0.025$ and $F_{3}>0.025$, so $t=2 ; 1-F_{10} \leq 0.025$ and $1-F_{9}>0.025$, so $u=10$. The conservative confidence interval is therefore

$$
\left(c_{50 L}, c_{50 U}\right)=\left(x_{t+1}, x_{u+1}\right)=\left(x_{3}, x_{11}\right)=(10,77)
$$

with actual coverage $F_{10}-F_{2}=0.9888-0.0112=0.9776$ ( $97.8 \%$ confidence). For the interpolation calculation, we have

$$
\begin{aligned}
& g=(0.025-0.0112) /(0.0461-0.0112)=0.395 \\
& h=(0.025-1+0.9888) /(0.9888-0.9539)=0.395
\end{aligned}
$$

So,

$$
\begin{array}{r}
c_{50 L}=x_{3}+0.395 \times\left(x_{4}-x_{3}\right)=10+0.395 \times 5=11.98 \\
c_{50 U}=x_{11}-0.395 \times\left(x_{11}-x_{10}\right)=77-0.395 \times 18=69.89
\end{array}
$$

## Normal case

The value of $c_{q}$ is as above. Its se is given by the formula

$$
s_{q}=\sqrt{q(100-q)} /\left\{100 \sqrt{n} Z\left(c_{q} ; \bar{x}, s\right)\right\}
$$

where $\bar{x}$ and $s$ are the mean and standard deviation of the $x_{i}$, and

$$
Z(Y ; \mu, \sigma)=\left(1 / \sqrt{2 \pi \sigma^{2}}\right) e^{-(Y-\mu)^{2} / 2 \sigma^{2}}
$$

is the density function of a normally distributed variable $Y$ with mean $\mu$ and standard deviation $\sigma$. The confidence interval for $C_{q}$ is $\left(c_{q}-z_{100(1-\alpha)} s_{q}, c_{q}+z_{100(1-\alpha)} s_{q}\right)$.

## meansd case

The value of $c_{q}$ is $\bar{x}+z_{q} \times s$. Its se is given by the formula

$$
s_{q}^{\star}=s \sqrt{1 / n+z_{q}^{2} /(2 n-2)}
$$

The confidence interval for $C_{q}$ is $\left(c_{q}-z_{100(1-\alpha)} \times s_{q}^{\star}, c_{q}+z_{100(1-\alpha)} \times s_{q}^{\star}\right)$.

## Acknowledgment

centile was written by Patrick Royston of the MRC Clinical Trials Unit, London, and coauthor of the Stata Press book Flexible Parametric Survival Analysis Using Stata: Beyond the Cox Model.

## References

Conover, W. J. 1999. Practical Nonparametric Statistics. 3rd ed. New York: Wiley
Kendall, M. G., and A. Stuart. 1969. The Advanced Theory of Statistics, Vol. 1: Distribution Theory. 3rd ed. London: Griffin.

Mood, A. M., and F. A. Graybill. 1963. Introduction to the Theory of Statistics. 2nd ed. New York: McGraw-Hill.
Stuart, A., and J. K. Ord. 1994. Kendall's Advanced Theory of Statistics: Distribution Theory, Vol. 1. 6th ed. London: Arnold.

## Also see

$[\mathrm{R}]$ ci - Confidence intervals for means, proportions, and variances
[R] summarize - Summary statistics
[D] pctile - Create variable containing percentiles

## Title

churdle - Cragg hurdle regression

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

churdle fits a linear or exponential hurdle model for a bounded dependent variable. The hurdle model combines a selection model that determines the boundary points of the dependent variable with an outcome model that determines its nonbounded values. Separate independent covariates are permitted for each model.

## Quick start

Linear hurdle model of y 1 on x 1 and x 2 , specifying that y 1 is truncated at 0 with x 1 and x 3 predicting selection

```
churdle linear y1 x1 x2, select(x1 x3) ll(0)
```

Add an upper truncation limit of 40

```
churdle linear y1 x1 x2, select(x1 x3) ll(0) ul(40)
```

Same as above, with the upper truncation limit specified in trunc

```
churdle linear y1 x1 x2, select(x1 x3) ll(0) ul(trunc)
```

Same as above, and use $x 3$ to model the variance of the selection model

```
churdle linear y1 x1 x2, select(x1 x3, het(x3)) ll(0) ul(trunc)
```

Same as above, and use $x 4$ to model the variance of the outcome model

```
churdle linear y1 x1 x2, select(x1 x3, het(x3)) ll(0) ///
    ul(trunc) het(x4)
```

Exponential hurdle model of y 2 on x 1 and x 2 , specifying that y 2 is truncated at 4 with x 1 and x 3 predicting selection

```
churdle exponential y2 x1 x2, select(x1 x3) ll(4)
```


## Menu

Statistics $>$ Linear models and related $>$ Hurdle regression

## Syntax

Basic syntax

```
churdle linear depvar, select(varlists) {ll(...)|ul(...)}
churdle exponential depvar, select(varlists) ll(...)
```

Full syntax for churdle linear

```
churdle linear depvar [indepvars] [if] [in] [weight],
    select(varlist \(\left[\right.\), noconstant het (varlist \(\left.\left.{ }_{o}\right)\right]\) )
    \{ll(\#|varname) | ul(\#|varname) \(\}\) [options]
```

Full syntax for churdle exponential

```
churdle exponential depvar [indepvars] [if] [in] [weight],
    select(varlists[, noconstant het(varlisto)]) ll(#|varname) [options]
```

options Description

Model

* select ()
$\ddagger 11$ (\#|varname)
$\ddagger \mathrm{ul}$ (\#|varname)
noconstant
constraints (constraints)
het (varlist)
SE/Robust vce(vcetype)

Reporting level(\#) nocnsreport display_options
specify independent variables and options for selection model
lower truncation limit
upper truncation limit
suppress constant term
apply specified linear constraints
specify variables to model the variance
vcetype may be oim, robust, cluster clustvar, bootstrap, or jackknife
set confidence level; default is level (95)
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

Maximization
maximize_options
coeflegend
control the maximization process; seldom used
display legend instead of statistics
*select () is required.
The full specification is select (varlist $\left[\right.$, noconstant het (varlist ${ }_{o}$ )]).
noconstant specifies that the constant be excluded from the selection model.
het $\left(\right.$ varlist $\left._{o}\right)$ specifies the variables in the error-variance function of the selection model.
$\ddagger_{\text {You must specify at least one of ul(\#|varname) or } 11(\# \mid \text { varname) for the linear model and must specify }}$ 11 (\#|varname) for the exponential model.
indepvars, varlist $_{s}$, and varlist ${ }_{O}$ may contain factor variables; see [U] 11.4.3 Factor variables.
bootstrap, by, collect, fp, jackknife, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands. Weights are not allowed with the bootstrap prefix; see $[R]$ bootstrap.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
coeflegend does not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

$\qquad$ Model
select (varlist $\left[\right.$, noconstant het $\left(\right.$ varlist $\left.\left.\left._{o}\right)\right]\right)$ specifies the variables and options for the selection model. select () is required.
11 (\#|varname) and ul(\#|varname) indicate the lower and upper limits, respectively, for the dependent variable. You must specify one or both for the linear model and must specify a lower limit for the exponential model. Observations with depvar $\leq 11$ () have a lower bound; observations with depvar $\geq \mathrm{ul}$ () have an upper bound; and the remaining observations are in the continuous region.
noconstant, constraints (constraints); see [R] Estimation options.
het (varlist) specifies the variables in the error-variance function of the outcome model.

## SE/Robust

vce(vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster, clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vee_option.

## Reporting

level(\#), nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate(\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from (init_specs); see [R] Maximize. These options are seldom used.

The following option is available with churdle but is not shown in the dialog box: coeflegend; see [R] Estimation options.

## Remarks and examples

churdle fits a linear or an exponential hurdle model. It combines a selection model that determines the boundary points of the dependent variable with an outcome model that determines its nonbounded values. Hurdle models treat these boundary values as observed instead of censored. That is to say, observations where the dependent variable is equal to one of the boundary values are not the result of our inability to observe the distribution above or below a certain point; see Wooldridge (2010) chapter 17 for a thorough discussion of this point.

These models were proposed by Cragg (1971) to explain the demand for durable goods. In the Cragg model, individuals purchase zero or a positive amount of the durable good, with different factors determining each of these choices. This may be generalized to other individual decisions, such as money donated to charity, cigarette consumption, and time spent volunteering.

Hurdle models are characterized by the relationship $y_{i}=s_{i} h_{i}^{*}$, where $y_{i}$ is the observed value of the dependent variable.

The selection variable, $s_{i}$, is 1 if the dependent variable is not bounded and 0 otherwise. In the Cragg model, the lower limit that binds the dependent variable is 0 so the selection model is

$$
s_{i}= \begin{cases}1 & \text { if } \mathbf{z}_{i} \gamma+\epsilon_{i}>0 \\ 0 & \text { otherwise }\end{cases}
$$

where $\mathbf{z}_{i}$ is a vector of explanatory variables, $\gamma$ is a vector of coefficients, and $\epsilon_{i}$ is a standard normal error term. churdle allows a different lower limit to be specified in 11 () and, for the linear model, an upper limit in ul(). Conditional heteroskedasticity of the random error $\epsilon_{i}$ is allowed if suboption het () is specified in select().

The continuous latent variable $h_{i}^{*}$ is observed only if $s_{i}=1$. The outcome model can be either the linear model or the exponential model, as proposed in Cragg (1971):

$$
\begin{aligned}
& h_{i}^{*}=\mathbf{x}_{i} \boldsymbol{\beta}+\nu_{i} \\
& h_{i}^{*}=\exp \left(\mathbf{x}_{i} \boldsymbol{\beta}+\nu_{i}\right) \quad \text { (exponential) }
\end{aligned}
$$

where $\mathbf{x}_{i}$ is a vector of explanatory variables, $\boldsymbol{\beta}$ is a vector of coefficients, and $\nu_{i}$ is an error term.
For the linear model, $\nu_{i}$ has a truncated normal distribution with lower truncation point $-\mathbf{x}_{i} \boldsymbol{\beta}$. For the exponential model, $\nu_{i}$ has a normal distribution. churdle extends the Cragg hurdle models to allow for conditional heteroskedasticity of the random error $\nu_{i}$ if the user specifies the het () option.

The parameters and regressors in the models for $h_{i}^{*}$ and for $s_{i}$ may differ.
> Example 1: Linear hurdle model
Consider a dataset that contains the number of hours an individual exercises per day (hours), their age (age), whether they are single (single), hours they work per day (whours), whether they smoke (smoke), their weight in kilograms (weight), their distance from the nearest gym (distance), and their average commute from work (commute).


Figure 1.
Figure 1 shows that $43.9 \%$ of the individuals in the sample do not exercise and that the hours exercised varies among individuals that decide to exercise.

We model the decision to exercise or not as a function of commute, whours, and age. These variables are written in select(). Once a decision to exercise is made, the time an individual exercises is modeled as a linear function of age, smoke, distance, and single.

```
. use https://www.stata-press.com/data/r18/fitness
(Fictional fitness data)
. churdle linear hours age i.smoke distance i.single,
> select(commute whours age) ll(0)
Iteration 0: Log likelihood = -23657.236
Iteration 1: Log likelihood = -23344.182
Iteration 2: Log likelihood = -23340.051
Iteration 3: Log likelihood = -23340.044
Iteration 4: Log likelihood = -23340.044
Cragg hurdle regression Number of obs = 19,831
```

    LR chi2(4) \(=9059.26\)
    Prob > chi2 \(=0.0000\)
    Pseudo R2 \(=0.1625\)
    | hours | Coefficient | Std. err. | z | $P>\|z\|$ | [95\% conf | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| hours |  |  |  |  |  |  |
| age | . 0015116 | . 000763 | 1.98 | 0.048 | . 0000162 | . 003007 |
| smoke |  |  |  |  |  |  |
| Smoking | -1.06646 | . 0460578 | -23.15 | 0.000 | -1.156731 | -. 9761879 |
| distance | -. 1333868 | . 0126344 | -10.56 | 0.000 | -. 1581497 | -. 1086238 |
| single |  |  |  |  |  |  |
| Single | . 9940893 | . 0258775 | 38.42 | 0.000 | . 9433703 | 1.044808 |
| _cons | . 9138855 | . 0396227 | 23.06 | 0.000 | . 8362264 | . 9915447 |
| selection_ll |  |  |  |  |  |  |
| commute | -. 2953345 | . 0624665 | -4.73 | 0.000 | -. 4177666 | -. 1729024 |
| whours | . 0022974 | . 0069306 | 0.33 | 0.740 | -. 0112864 | . 0158811 |
| age | -. 0485347 | . 0006501 | -74.65 | 0.000 | -. 049809 | -. 0472604 |
| _cons | 2.649945 | . 0499795 | 53.02 | 0.000 | 2.551987 | 2.747903 |
| lnsigma |  |  |  |  |  |  |
| _cons | . 0083199 | . 0099648 | 0.83 | 0.404 | -. 0112107 | . 0278506 |
| /sigma | 1.008355 | . 010048 |  |  | . 9888519 | 1.028242 |

The coefficients in the outcome model for the latent variable appear under hours. Because we only specified a lower limit to bind the dependent variable, the output shows parameter estimates for a single selection model under selection_ll. Information about the estimated standard deviation of the error term in the outcome model appears under lnsigma and /sigma.

The coefficient estimates are not directly interpretable. To obtain the effect of a covariate on the model, we need to use the margins command; see [R] churdle postestimation. Consider the effect of age:
. margins, dydx(age)
Average marginal effects $\quad$ Number of obs $=19,831$
Model VCE: OIM
Expression: Conditional mean estimates of dependent variable, predict()
dy/dx wrt: age

|  | Delta-method |  |  |  |  |  |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: |
|  | dy/dx | std. err. | z | P>\|z| | [95\% conf. interval] |  |
| age | -.0216855 | .000289 | -75.03 | 0.000 | -.022252 | -.021119 |

Each additional year of age is associated with about -0.02 fewer hours, or 1.2 minutes, of exercise.

## $>$ Example 2: Linear hurdle with models for the outcome and selection variances

In this example, we illustrate the possibility of fitting a heteroskedastic probit for the selection and latent model. In both cases, this is done by specifying age and single as the variables that affect the conditional variance. As in example 1, we have separate parameters for the outcome model and lower-limit selection model.
. churdle linear hours age i.smoke distance i.single,
> select(commute whours age, het(age single)) ll(0) het(age single) nolog
Cragg hurdle regression

Log likelihood $=-23339.355$$\quad$ Number of obs $=19,831$

| hours | Coefficient | Std. err. | z | $P>\|z\|$ | [95\% con | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| hours |  |  |  |  |  |  |
| age | . 0012559 | . 0008198 | 1.53 | 0.126 | -. 0003508 | . 0028626 |
| smoke |  |  |  |  |  |  |
| Smoking | -1.065564 | . 0457657 | -23.28 | 0.000 | -1.155263 | -. 9758649 |
| distance | -. 1332939 | . 0126102 | -10.57 | 0.000 | -. 1580094 | -. 1085783 |
| single |  |  |  |  |  |  |
| Single | 1.002511 | . 032535 | 30.81 | 0.000 | . 9387436 | 1.066278 |
| _cons | . 9166356 | . 0388318 | 23.61 | 0.000 | . 8405268 | . 9927445 |
| selection_ll |  |  |  |  |  |  |
| commute | -. 2959986 | . 0641594 | -4.61 | 0.000 | -. 4217488 | -. 1702484 |
| whours | . 0024514 | . 0069769 | 0.35 | 0.725 | -. 0112231 | . 0161259 |
| age | -. 048886 | . 0021405 | -22.84 | 0.000 | -. 0530814 | -. 0446906 |
| _cons | 2.669613 | . 1139478 | 23.43 | 0.000 | 2.44628 | 2.892947 |
| Insigma |  |  |  |  |  |  |
| age | . 0003537 | . 0004026 | 0.88 | 0.380 | -. 0004354 | . 0011427 |
| single | -. 0080667 | . 019253 | -0.42 | 0.675 | -. 0458019 | . 0296685 |
| lnsigma_ll |  |  |  |  |  |  |
| age | -. 0002035 | . 0008424 | -0.24 | 0.809 | -. 0018546 | . 0014475 |
| single | . 0268271 | . 0270133 | 0.99 | 0.321 | -. 0261179 | . 0797721 |

The coefficients on age and single have no effect on the conditional variance of the outcome model or on the conditional variance of the selection model. Thus, there is no evidence that the variance depends on age and marital status.

## > Example 3: Exponential hurdle model

Returning to example 1, if we believe that the conditional mean of the latent variable has an exponential form instead of a linear form, we use churdle exponential.
. churdle exponential hours age i.smoke distance i.single,
$>$ select(commute whours age) ll(0) nolog
Cragg hurdle regression

Log likelihood $=-15666.195$$\quad$| Number of obs $=19,831$ |
| :--- |

| hours | Coefficient | Std. err | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| hours |  |  |  |  |  |  |
| age | . 0008368 | . 0005341 | 1.57 | 0.117 | -. 00021 | . 0018836 |
| smoke |  |  |  |  |  |  |
| Smoking | -. 6431348 | . 0258509 | -24.88 | 0.000 | -. 6938016 | -. 592468 |
| distance | -. 0772879 | . 0079132 | -9.77 | 0.000 | -. 0927976 | -. 0617783 |
| single |  |  |  |  |  |  |
| Single | . 5975111 | . 016108 | 37.09 | 0.000 | . 5659401 | . 6290821 |
| _cons | -. 0770619 | . 0254833 | -3.02 | 0.002 | -. 1270082 | -. 0271157 |
| selection_11 |  |  |  |  |  |  |
| commute | -. 2953345 | . 0624665 | -4.73 | 0.000 | -. 4177666 | -. 1729024 |
| whours | . 0022974 | . 0069306 | 0.33 | 0.740 | -. 0112864 | . 0158811 |
| age | -. 0485347 | . 0006501 | -74.65 | 0.000 | -. 049809 | -. 0472604 |
| _cons | 2.649945 | . 0499795 | 53.02 | 0.000 | 2.551987 | 2.747903 |
| lnsigma |  |  |  |  |  |  |
| _cons | -. 186917 | . 0067067 | -27.87 | 0.000 | -. 200062 | -. 1737721 |
| /sigma | . 8295126 | . 0055633 |  |  | . 81868 | . 8404884 |

What was said previously regarding the interpretation of the effects of the different regressors also holds true for churdle exponential. We again use margins to estimate the effect of age on time spent exercising.

| Average marginal effects Model VCE: OIM |  |  | Number of obs $=19,831$ |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Expression: Conditional mean estimates of dependent variable, predict() dy/dx wrt: age |  |  |  |  |  |  |
|  | dy/dx | lta-meth std. err | z | $P>\|z\|$ | [95\% conf | interval] |
| age | -. 0245582 | . 0004805 | -51.11 | 0.000 | -. 0255 | -. 0236164 |

With the exponential outcome model of the latent variable, our estimate is that each additional year of age decreases exercise time by about 0.025 hours, or 1.5 minutes.

## Stored results

churdle stores the following in e():
Scalars

```
e(N)
e(k_eq_model)
e(df_m)
e(r2_p)
e(chi2)
e(ll)
e(1l_0)
e(N_clust)
e(p)
e(rank)
e(converged)
```

number of observations
number of equations in overall model test
model degrees of freedom
pseudo- $R^{2}$
$\chi^{2}$
log likelihood
log likelihood, constant-only model
number of clusters
$p$-value for model test
rank of $e(v)$
1 if converged, 0 otherwise
Macros
e(cmd)
e(cmdline)
e(depvar)
e(estimator)
e(model)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(chi2type)
e(vce)
e(vcetype)
e(opt)
e(which)
e(technique)
e(properties)
churdle
command as typed
name of dependent variable
linear or exponential
Linear or Exponential
weight type
weight expression
title in estimation output
name of cluster variable
Wald or LR; type of model $\chi^{2}$ test
vcetype specified in vce()
title used to label Std. err.
type of optimization
max or min; whether optimizer is to perform maximization or minimization
maximization technique
b V
e(predict)
program used to implement predict
e(marginsnotok)
predictions disallowed by margins
e(asbalanced)
e(asobserved)
factor variables fvset as asbalanced
factor variables fvset as asobserved
Matrices
e(b)
coefficient vector
e(Cns) constraints matrix
e(ilog) iteration $\log$
e(V)
variance-covariance matrix of the estimators
model-based variance
Functions
e(sample) marks estimation sample
In addition to the above, the following is stored in r() :

Matrices
$r$ (table)
matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

Let $\ell \ell$ refer to the lower limit and $u \ell$ to the upper limit. Also let the probabilities of being at these limits be given by

$$
\begin{aligned}
\operatorname{Pr}\left(y_{i}\right. & \left.=\ell \ell \mid \mathbf{z}_{i}\right)
\end{aligned}=\Phi\left(\ell \ell-\mathbf{z}_{i}^{\prime} \gamma_{\ell \ell}\right), ~=\Phi\left(\mathbf{z}_{i}^{\prime} \gamma_{u \ell}-u \ell\right) .
$$

where $\mathbf{z}_{i}$ are the covariates of the selection model for individual $i$, which may be distinct from the covariates $\mathbf{x}_{i}$ for the latent model; $\mathbf{\Phi}$ corresponds to the standard normal cumulative distribution function; $\gamma_{\ell \ell}$ is the parameter vector for the lower-limit selection model; and $\gamma_{u \ell}$ is the parameter vector for the upper-limit selection model.

Under the assumptions that $\nu_{i}$ has a truncated normal distribution with lower truncation point $\ell \ell-\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}$ and upper truncation point $u \ell-\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}$ and has a homoskedastic variance, the log-likelihood function is given by

$$
\begin{aligned}
\ln \mathbf{L}= & \sum_{i=1}^{n}\left(y_{i} \leq \ell \ell\right) \log \Phi\left(\ell \ell-\mathbf{z}_{i}^{\prime} \boldsymbol{\gamma}_{\ell \ell}\right)+\left(y_{i} \geq u \ell\right) \log \left\{1-\Phi\left(u \ell-\mathbf{z}_{i}^{\prime} \boldsymbol{\gamma}_{u \ell}\right)\right\} \\
& +\left(u \ell>y_{i}>\ell \ell\right)\left[\log \left\{\Phi\left(u \ell-\mathbf{z}_{i}^{\prime} \boldsymbol{\gamma}_{u \ell}\right)-\Phi\left(\ell \ell-\mathbf{z}_{i}^{\prime} \boldsymbol{\gamma}_{\ell \ell}\right)\right\}\right] \\
& -\left(u \ell>y_{i}>\ell \ell\right)\left[\log \left\{\Phi\left(\frac{u \ell-\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}}{\sigma}\right)-\Phi\left(\frac{\ell \ell-\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}}{\sigma}\right)\right\}\right] \\
& +\left(u \ell>y_{i}>\ell \ell\right)\left[\log \left\{\phi\left(\frac{y_{i}-\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}}{\sigma}\right)\right\}-\log (\sigma)\right]
\end{aligned}
$$

Without the homoskedasticity assumption, the heteroskedasticity can be modeled using the form $\sigma^{2}\left(\mathbf{w}_{i}\right)=\exp \left(2 \mathbf{w}_{i}^{\prime} \boldsymbol{\theta}\right)$, where $\mathbf{w}_{i}$ are the variables that affect the conditional variance of $\nu_{i}$. The $\log$-likelihood function is obtained by replacing $\sigma$ with $\exp \left(\mathbf{w}_{i}^{\prime} \boldsymbol{\theta}\right)$.

The log-likelihood function for the exponential model is given by

$$
\begin{aligned}
\ln \mathbf{L}= & \sum_{i=1}^{n}\left(y_{i} \leq \ell \ell\right) \log \Phi\left(\ell \ell-\mathbf{z}_{i}^{\prime} \boldsymbol{\gamma}\right)+\left(y_{i}>\ell \ell\right)\left[\log \left\{1-\Phi\left(\ell \ell-\mathbf{z}_{i}^{\prime} \boldsymbol{\gamma}\right)\right\}\right] \\
& \left.+\left(y_{i}>\ell \ell\right)\left\{\log \left\{\boldsymbol{\phi}\left[\log \left(y_{i}-\ell \ell\right)-\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\right) / \sigma\right]\right\}-\log (\sigma)-\log \left(y_{i}-\ell \ell\right)\right\}
\end{aligned}
$$

Analogous to the linear case, we can model heteroskedasticity by $\sigma^{2}\left(\mathbf{w}_{i}\right)=\exp \left(2 \mathbf{w}_{i}^{\prime} \boldsymbol{\theta}\right)$.
Estimation of both of the aforementioned likelihood functions is done by maximum likelihood.

## References

Belotti, F., P. Deb, W. G. Manning, and E. C. Norton. 2015. twopm: Two-part models. Stata Journal 15: 3-20.
Cragg, J. G. 1971. Some statistical models for limited dependent variables with application to the demand for durable goods. Econometrica 39: 829-844. https://doi.org/10.2307/1909582.
Deb, P., E. C. Norton, and W. G. Manning. 2017. Health Econometrics Using Stata. College Station, TX: Stata Press. Engel, C., and P. G. Moffatt. 2014. dhreg, xtdhreg, and bootdhreg: Commands to implement double-hurdle regression. Stata Journal 14: 778-797.

Farbmacher, H. 2011. Estimation of hurdle models for overdispersed count data. Stata Journal 11: 82-94.
García, B. 2013. Implementation of a double-hurdle model. Stata Journal 13: 776-794.

Gray, L. A., and M. Hernández-Alava. 2018. A command for fitting mixture regression models for bounded dependent variables using the beta distribution. Stata Journal 18: 51-75.
Marchenko, Y. V. 2015. Bayesian modeling: Beyond Stata's built-in models. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2015/05/26/bayesian-modeling-beyond-statas-built-in-models/.
Sánchez-Peñalver, A. 2019. Estimation methods in the presence of corner solutions. Stata Journal 19: 87-111.
Wooldridge, J. M. 2010. Econometric Analysis of Cross Section and Panel Data. 2nd ed. Cambridge, MA: MIT Press.
Wulff, J. N. 2019. Generalized two-part fractional regression with cmp. Stata Journal 19: 375-389.

## Also see

$[\mathrm{R}]$ churdle postestimation - Postestimation tools for churdle
[R] intreg - Interval regression
[R] tobit - Tobit regression
[SVY] svy estimation - Estimation commands for survey data
[U] 20 Estimation and postestimation commands

## Title

churdle postestimation - Postestimation tools for churdle

| Postestimation commands | predict | margins |
| :--- | :--- | :--- |
| Remarks and examples | Methods and formulas | Also see |

## Postestimation commands

The following standard postestimation commands are available after churdle:

| Command | Description |
| :---: | :---: |
| contrast | contrasts and ANOVA-style joint tests of estimates |
| estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estat (svy) | postestimation statistics for survey data |
| estimates | cataloging estimation results |
| etable | table of estimation results |
| * forecast | dynamic forecasts and simulations |
| * hausman | Hausman's specification test |
| lincom | point estimates, standard errors, testing, and inference for linear combinations of coefficients |
| * 1 rtest | likelihood-ratio test |
| margins | marginal means, predictive margins, marginal effects, and average marginal effects |
| marginsplot | graph the results from margins (profile plots, interaction plots, etc.) |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients |
| predict | conditional means, residuals, etc. |
| predictnl | point estimates, standard errors, testing, and inference for generalized predictions |
| pwcompare | pairwise comparisons of estimates |
| suest | seemingly unrelated estimation |
| test | Wald tests of simple and composite linear hypotheses |
| testnl | Wald tests of nonlinear hypotheses |

[^9]
## predict

## Description for predict

predict creates a new variable containing predictions such as conditional expectation of depvar, residuals, linear predictions, standard errors, and probabilities.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

```
predict [type] newvar [if] [in] [, statistic equation(eqno)]
predict [type] stub* [if] [in], scores
```


## statistic <br> Description

## Main

| ystar | conditional expectation of depvar; the default |
| :--- | :--- |
| $\underline{\text { residuals }}$ | residuals |
| $\mathrm{ys} \operatorname{star}(a, b)$ | $E\left(y_{j}^{*}\right), y_{j}^{*}=\max \left\{a, \min \left(y_{j}, b\right)\right\}$ |
| xb | linear prediction |
| stdp | standard error of the linear prediction |
| $\operatorname{pr}(a, b)$ | $\operatorname{Pr}\left(a<y_{j}<b\right)$ |
| $\mathrm{e}(a, b)$ | $E\left(y_{j} \mid a<y_{j}<b\right)$ |

These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.
where $a$ and $b$ may be numbers or variables; $a$ missing ( $a \geq$.) means $-\infty$, and $b$ missing ( $b \geq$.) means $+\infty$; see [U] 12.2.1 Missing values. For churdle exponential, $b$ is . (missing).

## Options for predict

Main
ystar, the default, calculates the conditional expectation of the dependent variable.
residuals calculates the residuals.
ystar $(a, b)$ calculates $E\left(y_{j}^{*}\right) . a$ and $b$ are specified as they are for $\operatorname{pr}()$. If $a$ and $b$ are equal to the lower and upper bounds specified in churdle, then $E\left(y_{j}^{*}\right)$ is equivalent to the predicted value of the dependent variable ystar.
xb calculates the linear prediction.
stdp calculates the standard error of the linear prediction.
$\operatorname{pr}(a, b)$ calculates $\operatorname{Pr}\left(a<y_{j}<b\right)$, the probability that $y_{j} \mid \mathbf{x}_{i}$ would be observed in the interval $(a, b)$.
$a$ and $b$ may be specified as numbers or variable names; $l b$ and $u b$ are variable names;
$\operatorname{pr}(20,30)$ calculates $\operatorname{Pr}\left(20<y_{j}<30\right) ; \operatorname{pr}(l b, u b)$ calculates $\operatorname{Pr}\left(l b<y_{j}<u b\right)$;
and $\operatorname{pr}(20, u b)$ calculates $\operatorname{Pr}\left(20<y_{j}<u b\right)$.
$a$ missing $(a \geq$.$) means 11 ; \operatorname{pr}(., 30)$ calculates $\operatorname{Pr}\left(l l<y_{j}<30\right)$;
$\operatorname{pr}(l b, 30)$ calculates $\operatorname{Pr}\left(l l<y_{j}<30\right)$ in observations for which $l b \geq$. and calculates $\operatorname{Pr}\left(l b<y_{j}<30\right)$ elsewhere.
$b$ missing ( $b \geq$. ) means $\infty$; $\operatorname{pr}(20,$.$) calculates \operatorname{Pr}\left(\infty>y_{j}>20\right)$;
$\operatorname{pr}(20, u b)$ calculates $\operatorname{Pr}\left(\infty>y_{j}>20\right)$ in observations for which $u b \geq$. and calculates $\operatorname{Pr}\left(u b>y_{j}>20\right)$ elsewhere. For churdle linear, $u l$ is $\infty$.
$\mathrm{e}(a, b)$ calculates $E\left(y_{j} \mid a<y_{j}<b\right)$, the expected value of $y_{j} \mid \mathbf{x}_{j}$ conditional on $y_{j} \mid \mathbf{x}_{j}$ being in the interval $(a, b)$, meaning that $y_{j} \mid \mathbf{x}_{j}$ is bounded. $a$ and $b$ are specified as they are for pr() .
equation (eqno) specifies the equation for which predictions are to be made for the xb and stdp options. equation() should contain either one equation name or one of \#1, \#2, ... with \#1 meaning the first equation, \#2 meaning the second equation, etc.

If you do not specify equation(), the results are the same as if you specified equation(\#1).
scores calculates the equation-level score variables. If you specify one new variable, the scores for the latent-variable equation are computed. If you specify a variable list, the scores for each equation are calculated. The following scores may be obtained:
the first new variable will contain $\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)$,
the second new variable will contain $\partial \ln L / \partial\left(\mathbf{z}_{j} \gamma_{l l}\right)$,
the third new variable will contain $\partial \ln L / \partial\left(\mathbf{z}_{j} \gamma_{u l}\right)$,
the fourth new variable will contain $\partial \ln L / \partial(\sigma)$,
the fifth new variable will contain $\partial \ln L / \partial\left(\sigma_{l l}\right)$, and
the sixth new variable will contain $\partial \ln L / \partial\left(\sigma_{u l}\right)$.

## margins

## Description for margins

margins estimates margins of response for conditional expectations, linear predictions, and probabilities.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...) ...] [options]
```

statistic
Description
Main

| ystar | conditional expectation of depvar; the default |
| :--- | :--- |
| $\frac{\mathrm{ys}}{\mathrm{xb}} \mathrm{tar}(a, b)$ | $E\left(y_{j}^{*}\right), y_{j}^{*}=\max \left\{a, \min \left(y_{j}, b\right)\right\}$; for churdle exponential $b$ is. |
| $\mathrm{pr}(a, b)$ | linear prediction |
| $\mathrm{e}(a, b)$ | $\operatorname{Pr}\left(a<y_{j}<b\right) ;$ for churdle exponential $b$ is. |
| residuals | $E\left(y_{j} \mid a<y_{j}<b\right) ;$ for churdle exponential $b$ is. |
| stdp | not allowed with margins |
|  | not allowed with margins |

Statistics not allowed with margins are functions of stochastic quantities other than e(b). For the full syntax, see [R] margins.

## Remarks and examples

$>$ Example 1: Predictions for depvar
Below, we use the parameters estimated in example 1 of $[\mathrm{R}]$ churdle to calculate the average hours exercised given the covariates.

```
. use https://www.stata-press.com/data/r18/fitness
(Fictional fitness data)
. churdle linear hours age i.smoke distance i.single,
> select(commute whours age) ll(0)
    (output omitted)
. predict hourshat
(statistic ystar assumed)
```

We might also be interested in estimating the average number of hours exercised given that an individual exercises. Below we estimate this quantity and compare it with the predicted and true values of the dependent variable for all the observations.

- predict exercises, e(0,.)
. summarize hours hourshat exercises

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| hours | 19,831 | .8800172 | 1.051221 | 0 | 5.308835 |
| hourshat | 19,831 | .8786302 | .4915214 | .0754708 | 1.904694 |
| exercises | 19,831 | 1.580729 | .3998335 | .5630298 | 2.079012 |

As expected, we observe that the sample-average predictions are higher for those who exercise.

## > Example 2: Marginal effects

Suppose we want to study whether single individuals exercise more on average than married individuals. Below, we use margins to estimate the average effect of being single on hours spent exercising in the population.

```
. margins, dydx(1.single)
Average marginal effects Number of obs = 19,831
Model VCE: OIM
Expression: Conditional mean estimates of dependent variable, predict()
dy/dx wrt: 1.single
```

|  | Delta-method |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | dy/dxstd. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |  |
| single <br> Single | .3858462 | .0091398 | 42.22 | 0.000 | .3679324 | .4037599 |

Note: dy/dx for factor levels is the discrete change from the base level.
The average effect of moving each individual from not single to single is an increase in exercise of 0.39 hours, or about 23 minutes.

The statistics $\operatorname{pr}(a, b), \mathrm{e}(a, b)$, and ystar $(a, b)$ produce counterfactual predictions.
These statistics predict what would be observed if the limits $\ell \ell$ and $u \ell$ were the specified $a$ and $b$ and the estimated parameters did not change, even though the parameters of the model were estimated using the limits $\ell \ell$ and $u \ell$.

For example, suppose we model contributions to a retirement plan in a world where the government requires a minimum contribution of $2 \%$ so $\ell \ell=2$. After estimating the model parameters, we could predict the average contribution, given the covariates, when the government raises the minimum contribution to $3 \%$ with the statistic ystar (3,.).

## Methods and formulas

Let $\ell \ell$ refer to the lower limit and $u \ell$ to the upper limit. Also let the probabilities of being at these limits be given by

$$
\begin{aligned}
\operatorname{Pr}\left(y_{i}=\ell \ell \mid \mathbf{z}_{i}\right) & =\Phi\left(\ell \ell-\mathbf{z}_{i}^{\prime} \gamma_{\ell \ell}\right) \\
\operatorname{Pr}\left(y_{i}=u \ell \mid \mathbf{z}_{i}\right) & =\Phi\left(\mathbf{z}_{i}^{\prime} \gamma_{u \ell}-u \ell\right)
\end{aligned}
$$

where $\mathbf{z}_{i}$ are the covariates of the selection model for individual $i$, which may be distinct from the covariates $\mathbf{x}_{i}$ for the outcome model; $\mathbf{\Phi}$ is the standard normal cumulative distribution function; $\gamma_{\ell \ell}$ is the parameter vector for the lower-limit selection model; and $\gamma_{u \ell}$ is the parameter vector for the upper-limit selection model.

We will limit the exposition below to the case with a lower and an upper limit.
In churdle linear, ystar is equivalent to $E\left(y_{i} \mid \mathbf{x}_{i}\right)$ and is given by

$$
\begin{aligned}
E\left(y_{i} \mid \mathbf{x}_{i}, \mathbf{z}_{i}\right)= & \Phi\left(\mathbf{z}_{i}^{\prime} \boldsymbol{\gamma}_{u \ell}-u \ell\right) u \ell+\Phi\left(\ell \ell-\mathbf{z}_{i}^{\prime} \boldsymbol{\gamma}_{\ell \ell}\right) \ell \ell \\
& +\left\{\Phi\left(u \ell-\mathbf{z}_{i}^{\prime} \boldsymbol{\gamma}\right)-\Phi\left(\ell \ell-\mathbf{z}_{i}^{\prime} \boldsymbol{\gamma}\right)\right\}\left\{\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}+\sigma \frac{\phi\left(\frac{\ell \ell-\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}}{\sigma}\right)-\phi\left(\frac{u \ell-\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}}{\sigma}\right)}{\Phi\left(\frac{u \ell-\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}}{\sigma}\right)-\Phi\left(\frac{\ell \ell-\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}}{\sigma}\right)}\right\}
\end{aligned}
$$

$\operatorname{pr}(a, b)$ is given by

$$
\operatorname{Pr}\left(a<y_{i}<b \mid \mathbf{z}_{i}\right)=\Phi\left(b-\mathbf{z}_{i}^{\prime} \boldsymbol{\beta}\right)-\Phi\left(a-\mathbf{z}_{i}^{\prime} \boldsymbol{\beta}\right)
$$

$\mathrm{e}(a, b)$ is given by

$$
E\left(a<y_{i}<b \mid \mathbf{x}_{i}\right)=\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}+\sigma \frac{\phi\left(\frac{a-\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}}{\sigma}\right)-\phi\left(\frac{b-\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}}{\sigma}\right)}{\Phi\left(\frac{b-\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}}{\sigma}\right)-\Phi\left(\frac{a-\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}}{\sigma}\right)}
$$

and ystar $(a, b)$ is given by

$$
\begin{aligned}
E\left(y^{*}\right)= & \Phi\left(\mathbf{z}_{i}^{\prime} \gamma_{u \ell}-b\right) b+\Phi\left(a-\mathbf{z}_{i}^{\prime} \boldsymbol{\gamma}_{\ell \ell}\right) a \\
& +\operatorname{Pr}\left(a<y_{i}<b \mid \mathbf{x}_{i}\right) E\left(a<y_{i}<b \mid \mathbf{x}_{i}\right)
\end{aligned}
$$

For churdle exponential, ystar is equivalent to

$$
\begin{aligned}
E\left(y_{i} \mid \mathbf{x}_{i}\right)= & \Phi\left(\ell \ell-\mathbf{z}_{i}^{\prime} \boldsymbol{\gamma}_{\ell \ell}\right) \ell \ell \\
& +\left\{1-\Phi\left(\ell \ell-\mathbf{z}_{i}^{\prime} \boldsymbol{\gamma}\right)\right\} \exp \left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}+\sigma^{2} / 2\right)\left[\frac{1-\Phi\left\{\frac{\ln (\ell \ell)-\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}}{\sigma}-\sigma\right\}}{1-\Phi\left\{\frac{\ln (\ell \ell)-\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}}{\sigma}\right\}}\right]
\end{aligned}
$$

$\mathrm{p}(a,$.$) is given by$

$$
\operatorname{Pr}\left(a<y_{i} \mid \mathbf{x}_{i}\right)=1-\Phi\left(a-\mathbf{z}_{i}^{\prime} \gamma_{\ell \ell}\right)
$$

$\mathrm{e}(a,$.$) is given by$

$$
E\left(a<y_{i} \mid \mathbf{x}_{i}\right)=\exp \left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}+\sigma^{2} / 2\right)\left[\frac{1-\Phi\left\{\frac{\ln (a)-\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}}{\sigma}-\sigma\right\}}{1-\Phi\left\{\frac{\ln (a)-\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}}{\sigma}\right\}}\right]
$$

and ystar $(a,$.$) is given by$

$$
E\left(y^{*}\right)=a \Phi\left(a-\mathbf{z}_{i}^{\prime} \boldsymbol{\gamma}_{\ell \ell}\right)+\operatorname{Pr}\left(a<y_{i} \mid \mathbf{x}_{i}\right) E\left(a<y_{i} \mid \mathbf{x}_{i}\right)
$$

## Also see

[R] churdle - Cragg hurdle regression
[U] 20 Estimation and postestimation commands

## Title

ci - Confidence intervals for means, proportions, and variances

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| Acknowledgment | References | Also see |  |

## Description

ci computes confidence intervals for population means, proportions, variances, and standard deviations.
cii is the immediate form of ci; see [U] 19 Immediate commands for a general discussion of immediate commands.

## Quick start

Confidence intervals for means of normally distributed variables v1, v2, and v3 ci means v1-v3

Confidence interval for mean of Poisson-distributed variable v4
ci means v4, poisson
Confidence interval for rate of v 4 with total exposure recorded in v 5
ci means v4, poisson exposure(v5)
Confidence interval for proportion of binary variable v6
ci proportions v6
Confidence intervals for variances of $\mathrm{v} 1, \mathrm{v} 2$, and v 3
ci variances v1-v3
Same as above, but Bonett confidence intervals are produced
ci variances v1-v3, bonett
$90 \%$ Bonett confidence intervals for standard deviations of v1, v2, and v3
ci variances v1-v3, sd bonett level(90)
Confidence interval for a mean based on a sample with 85 observations, a sample mean of 10 , and a standard deviation of 3
cii means 85103
$90 \%$ confidence interval for rate from a sample with 4,379 deaths over 11,394 person-years cii means 11394 4379, poisson level(90)

Agresti-Coull confidence interval for proportion based on a sample with 2,377 observations and 136 successes

```
cii proportions 2377 136, agresti
```

Bonett confidence interval for variance based on a sample with 20 observations, sample variance of 9 , and estimated kurtosis of 1.8

```
cii variances 20 9 1.8, bonett
```

Same as above, but with confidence interval for standard deviation cii variances 2031.8 , sd bonett

## Menu

ci
Statistics $>$ Summaries, tables, and tests $>$ Summary and descriptive statistics $>$ Confidence intervals

## cii for a normal mean

Statistics $>$ Summaries, tables, and tests $>$ Summary and descriptive statistics $>$ Normal mean Cl calculator

## cii for a Poisson mean

Statistics $>$ Summaries, tables, and tests $>$ Summary and descriptive statistics $>$ Poisson mean Cl calculator

## cii for a proportion

Statistics $>$ Summaries, tables, and tests $>$ Summary and descriptive statistics $>$ Proportion Cl calculator

## cii for a variance

Statistics $>$ Summaries, tables, and tests $>$ Summary and descriptive statistics $>$ Variance Cl calculator

## cii for a standard deviation

Statistics $>$ Summaries, tables, and tests $>$ Summary and descriptive statistics $>$ Standard deviation Cl calculator

## Syntax

Confidence intervals for means, normal distribution
ci means $[$ varlist $][$ if $][$ in $][$ weight $][$, options $]$
cii means $\#_{\text {obs }} \#_{\text {mean }} \#_{\text {sd }}[$, level (\#) $]$

Confidence intervals for means, Poisson distribution
ci means [varlist] [if] [in] [weight], poisson [ exposure(varname) options]
cii means $\#_{\text {exposure }} \#_{\text {events }}$, poisson [level(\#)]

Confidence intervals for proportions ci proportions [varlist] [if] [in] [weight] [, prop_options options] cii proportions $\#_{\text {obs }} \#_{\text {succ }}[$, prop_options level(\#)]

Confidence intervals for variances
ci variances [varlist][if][in][weight][, bonett options]
cii variances $\#_{\text {obs }} \#_{\text {variance }}[$, level (\#) $]$
cii variances $\#_{\text {obs }} \#_{\text {variance }} \#_{\text {kurtosis }}$, bonett [level(\#)]

Confidence intervals for standard deviations
ci variances $[$ varlist $][$ if $][$ in $][$ weight], sd [bonett options]
cii variances $\#_{\text {obs }} \#_{\text {sd }}$, sd [level(\#)]
cii variances $\#_{\text {obs }} \#_{\text {sd }} \#_{\text {kurtosis }}$, sd bonett [ level (\#) $]$
$\#_{\text {obs }}$ must be a positive integer. \#exposure, $\#_{\text {sd }}$, and $\#_{\text {variance }}$ must be a positive number. \# succ and $\#_{\text {events }}$ must be a nonnegative integer or between 0 and 1 . If the number is between 0 and 1 , Stata interprets it as the fraction of successes or events and converts it to an integer number representing the number of successes or events. The computation then proceeds as if two integers had been specified. If option bonett is specified, you must additionally specify $\#_{\text {kurtosis }}$ with cii variances.
exact
wald
wilson
agresti
jeffreys
calculate exact confidence intervals; the default calculate Wald confidence intervals calculate Wilson confidence intervals calculate Agresti-Coull confidence intervals calculate Jeffreys confidence intervals

## options

Description
level(\#)
separator(\#)
total
set confidence level; default is level (95)
draw separator line after every \# variables; default is separator (5) add output for all groups combined (for use with by only)
by, collect, and statsby are allowed with ci, and collect is allowed with cii; see [U] 11.1.10 Prefix commands. aweights are allowed with ci means for normal data, and fweights are allowed with all ci subcommands; see
[U] 11.1.6 weight.

## Options

Options are presented under the following headings:
Options for ci and cii means
Options for ci and cii proportions
Options for ci and cii variances

## Options for ci and cii means

$\qquad$ Main
poisson specifies that the variables (or numbers for cii) are Poisson-distributed counts; exact Poisson confidence intervals will be calculated. By default, confidence intervals for means are calculated based on a normal distribution.
exposure (varname) is used only with poisson. You do not need to specify poisson if you specify exposure(); poisson is assumed. varname contains the total exposure (typically a time or an area) during which the number of events recorded in varlist was observed.
level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [R] level.
separator(\#) specifies how often separation lines should be inserted into the output. The default is separator (5), meaning that a line is drawn after every five variables. separator (10) would draw the line after every 10 variables. separator ( 0 ) suppresses the separation line.
total is used with the by prefix. It requests that in addition to output for each by-group, output be added for all groups combined.

## Options for ci and cii proportions

$\qquad$
exact, wald, wilson, agresti, and jeffreys specify how binomial confidence intervals are to be calculated.
exact is the default and specifies exact (also known in the literature as Clopper-Pearson [1934]) binomial confidence intervals.
wald specifies calculation of Wald confidence intervals.
wilson specifies calculation of Wilson confidence intervals.
agresti specifies calculation of Agresti-Coull confidence intervals.
jeffreys specifies calculation of Jeffreys confidence intervals.
See Brown, Cai, and DasGupta (2001) for a discussion and comparison of the different binomial confidence intervals.
level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [R] level.
separator (\#) specifies how often separation lines should be inserted into the output. The default is separator (5), meaning that a line is drawn after every five variables. separator (10) would draw the line after every 10 variables. separator ( 0 ) suppresses the separation line.
total is used with the by prefix. It requests that in addition to output for each by-group, output be added for all groups combined.

## Options for ci and cii variances

$\qquad$ Main
sd specifies that confidence intervals for standard deviations be calculated. The default is to compute confidence intervals for variances.
bonett specifies that Bonett confidence intervals be calculated. The default is to compute normal-based confidence intervals, which assume normality for the data.
level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [R] level.
separator (\#) specifies how often separation lines should be inserted into the output. The default is separator (5), meaning that a line is drawn after every five variables. separator (10) would draw the line after every 10 variables. separator ( 0 ) suppresses the separation line.
total is used with the by prefix. It requests that in addition to output for each by-group, output be added for all groups combined.

## Remarks and examples

Remarks are presented under the following headings:
Confidence intervals for means Normal-based confidence intervals
Poisson confidence intervals
Confidence intervals for proportions
Confidence intervals for variances
Immediate form

## Confidence intervals for means

ci means computes a confidence interval for the population mean for each of the variables in varlist.

## Normal-based confidence intervals

## > Example 1: Normal-based confidence intervals

Without the poisson option, ci means produces normal-based confidence intervals that are correct if the variable is normally distributed and asymptotically correct for all other distributions satisfying the conditions of the central limit theorem.

```
. use https://www.stata-press.com/data/r18/auto
```

(1978 automobile data)
. ci means mpg price

| Variable | Obs | Mean | Std. err. | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| mpg | 74 | 21.2973 | .6725511 | 19.9569 | 22.63769 |
| price | 74 | 6165.257 | 342.8719 | 5481.914 | 6848.6 |

The standard error of the mean of mpg is 0.67 , and the $95 \%$ confidence interval is [19.96, 22.64]. We can obtain wider confidence intervals, $99 \%$, by typing
. ci means mpg price, level(99)

| Variable | Obs | Mean | Std. err. | [99\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| mpg | 74 | 21.2973 | .6725511 | 19.51849 | 23.07611 |
| price | 74 | 6165.257 | 342.8719 | 5258.405 | 7072.108 |

## > Example 2: The by prefix

The by prefix breaks out the confidence intervals according to by-group; total adds an overall summary. For instance,
. by foreign: ci means mpg, total

| Variable | Obs | Mean | Std. err. | [95\% conf | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: |
| mpg | 52 | 19.82692 | . 657777 | 18.50638 | 21.14747 |
| -> foreign = Foreign |  |  |  |  |  |
| Variable | Obs | Mean | Std. err. | [95\% conf | interval] |
| mpg | 22 | 24.77273 | 1.40951 | 21.84149 | 27.70396 |
| -> Total |  |  |  |  |  |
| Variable | Obs | Mean | Std. err. | [95\% conf | interval] |
| mpg | 74 | 21.2973 | . 6725511 | 19.9569 | 22.63769 |

> Example 3: Controlling the format
You can control the formatting of the numbers in the output by specifying a display format for the variable; see [U] 12.5 Formats: Controlling how data are displayed. For instance,
. format mpg \%9.2f

| ci means mpg |
| ---: | ---: | ---: | ---: | ---: | ---: |
| Variable |

mpg

## Poisson confidence intervals

If you specify the poisson option, ci means assumes count data and computes exact Poisson confidence intervals.
> Example 4: Poisson confidence intervals
We have data on the number of bacterial colonies on a Petri dish. The dish has been divided into 36 small squares, and the number of colonies in each square has been counted. Each observation in our dataset represents a square on the dish. The variable count records the number of colonies in each square counted, which varies from 0 to 5 .

```
. use https://www.stata-press.com/data/r18/petri, clear
. ci means count, poisson
```

| Variable | Exposure | Mean | Std. err. | Poisson exact <br> [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| count | 36 | 2.333333 | .2545875 | 1.861158 | 2.888825 |

ci reports that the average number of colonies per square is 2.33 . If the expected number of colonies per square were as low as 1.86 , the probability of observing 2.33 or more colonies per square would be $2.5 \%$. If the expected number were as large as 2.89 , the probability of observing 2.33 or fewer colonies per square would be $2.5 \%$.

## > Example 5: Option exposure()

The number of "observations"-how finely the Petri dish is divided-makes no difference. The Poisson distribution is a function only of the count. In example 4, we observed a total of $2.33 \times 36=84$ colonies and a confidence interval of $[1.86 \times 36,2.89 \times 36]=[67,104]$. We would obtain the same $[67,104]$ confidence interval if our dish were divided into, say, 49 squares rather than 36.

For the counts, it is not even important that all the squares be of the same size. For rates, however, such differences do matter but in an easy-to-calculate way. Rates are obtained from counts by dividing by exposure, which is typically a number multiplied by either time or an area. For our Petri dishes, we divide by an area to obtain a rate, but if our example were cast in terms of being infected by a disease, we might divide by person-years to obtain the rate. Rates are convenient because they are easier to compare: we might have 2.3 colonies per square inch or 0.0005 infections per person-year.

So let's assume that we wish to obtain the number of colonies per square inch and, moreover, that not all the "squares" on our dish are of equal size. We have a variable called area that records the area of each square:

```
. ci means count, exposure(area)
```

| Variable | Exposure | Mean | Std. err. | Poisson exact <br> [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| count | 3 | 28 | 3.055051 | 22.3339 | 34.66591 |

The rates are now in more familiar terms. In our sample, there are 28 colonies per square inch, and the $95 \%$ confidence interval is [22.3, 34.7]. When we did not specify exposure(), ci means with option poisson assumed that each observation contributed 1 to exposure.

- Technical note

If there were no colonies on our dish, ci means with option poisson would calculate a one-sided confidence interval:
. use https://www.stata-press.com/data/r18/petrinone
. ci means count, poisson

| Variable | Exposure | Mean | Std. err. | Poisson exact <br> [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| count | 36 | 0 | 0 | 0 | $.1024689 *$ |

(*) one-sided, $97.5 \%$ confidence interval

## Confidence intervals for proportions

The ci proportions command assumes binary ( $0 / 1$ ) data and computes binomial confidence intervals.
> Example 6: Exact binomial (Clopper-Pearson) confidence interval
We have data on employees, including a variable marking whether the employee was promoted last year.
. use https://www.stata-press.com/data/r18/promo
. ci proportions promoted

| Variable | Obs Proportion | Std. err. | Binomial exact <br> [95\% conf. interval] |  |  |
| :---: | ---: | :---: | :---: | :---: | ---: |
| promoted | 20 | .1 | .067082 | .0123485 | .3169827 |

The exact binomial, also known as the Clopper-Pearson (1934) interval, is computed by default.
Nominally, the interpretation of a $95 \%$ confidence interval is that under repeated samples or experiments, $95 \%$ of the resultant intervals would contain the unknown parameter in question. However, for binomial data, the actual coverage probability, regardless of method, usually differs from that interpretation. This result occurs because of the discreteness of the binomial distribution, which produces only a finite set of outcomes, meaning that coverage probabilities are subject to discrete jumps and that the exact nominal level cannot always be achieved. Therefore, the term "exact confidence interval" refers to its being derived from the binomial distribution, the distribution exactly generating the data, rather than resulting in exactly the nominal coverage.

For the Clopper-Pearson interval, the actual coverage probability is guaranteed to be greater than or equal to the nominal confidence level, here $95 \%$. Because of the way it is calculated-see Methods and formulas-it may also be interpreted as follows: If the true probability of being promoted were 0.012 , the chances of observing a result as extreme or more extreme than the result observed ( $20 \times 0.1=2$ or more promotions) would be $2.5 \%$. If the true probability of being promoted were 0.317 , the chances of observing a result as extreme or more extreme than the result observed (two or fewer promotions) would be $2.5 \%$.

## > Example 7: Other confidence intervals

The Clopper-Pearson interval is desirable because it guarantees nominal coverage; however, by dropping this restriction, you may obtain accurate intervals that are not as conservative. In this vein, you might opt for the Wilson (1927) interval,
. ci proportions promoted, wilson

| Variable | Obs Proportion | Std. err. | Wilson |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: |
| [95\% conf. interval] |  |  |  |  |  |
| promoted | 20 | .1 | .067082 | .0278665 | .3010336 |

the Agresti-Coull (1998) interval,
. ci proportions promoted, agresti

| Variable | Obs | Proportion | Std. err. | Agresti-Coull <br> [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: | :---: |
| promoted | 20 | .1 | .067082 | .0156562 | .3132439 |

or the Bayesian-derived Jeffreys interval (Brown, Cai, and DasGupta 2001),
. ci proportions promoted, jeffreys

| Variable | Obs | Jroportion |  | Std. err. | [95\% conf. interval] |
| :---: | ---: | :---: | :---: | ---: | ---: |
| promoted | 20 | .1 | .067082 | .0213725 | .2838533 |

Picking the best interval is a matter of balancing accuracy (coverage) against precision (average interval length) and depends on sample size and success probability. Brown, Cai, and DasGupta (2001) recommend the Wilson or Jeffreys interval for small sample sizes $(\leq 40)$ yet favor the Agresti-Coull interval for its simplicity, decent performance for sample sizes less than or equal to 40 , and performance comparable to Wilson or Jeffreys for sample sizes greater than 40 . They also deem the ClopperPearson interval to be "wastefully conservative and [...] not a good choice for practical use", unless of course one requires, at a minimum, the nominal coverage level.

Finally, the binomial Wald confidence interval is obtained by specifying the wald option. The Wald interval is the one taught in most introductory statistics courses and, for the above, is simply, for level $1-\alpha$, Proportion $\pm z_{\alpha / 2}$ (Std. err.), where $z_{\alpha / 2}$ is the $1-\alpha / 2$ quantile of the standard normal. Because its overall poor performance makes it impractical, the Wald interval is available mainly for pedagogical purposes. The binomial Wald interval is also similar to the interval produced by treating binary data as normal data and using ci means, with two exceptions. First, the calculation of the standard error in ci proportions uses denominator $n$ rather than $n-1$, used for normal data in ci means. Second, confidence intervals for normal data are based on the $t$ distribution rather than the standard normal. Of course, both discrepancies vanish as sample size increases.

- Technical note

Let's repeat example 6, but this time with data in which there are no promotions over the observed period:

```
. use https://www.stata-press.com/data/r18/promonone
. ci proportions promoted
```

| Variable | Obs Proportion | Std. err. | Binomial exact <br> [95\% conf. interval] |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| promoted | 20 | 0 | 0 | 0 | $.1684335 *$ |
| (*) one-sided, $97.5 \%$ confidence interval |  |  |  |  |  |

The confidence interval is $[0,0.168]$, and this is the confidence interval that most books publish. It is not, however, a true $95 \%$ confidence interval because the lower tail has vanished. As Stata notes, it is a one-sided, $97.5 \%$ confidence interval. If you wanted to put $5 \%$ in the right tail, you could type ci proportions promoted, level(90).

## - Technical note

ci proportions ignores any variables that do not take on the values 0 and 1 exclusively. For instance, with our automobile dataset,

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. ci proportions mpg foreign
\begin{tabular}{c|ccccc} 
Variable & Obs & & \multicolumn{2}{c}{\begin{tabular}{c} 
Binomial exact \\
[95\% conf. interval]
\end{tabular}} \\
\hline foreign & 74 & .2972973 & .0531331 & .196584 & .4148353 \\
Note: The results are produced only for binary (0/1) & variables.
\end{tabular}
Note: The results are produced only for binary (0/1) variables.
```

We also requested the confidence interval for mpg , but Stata ignored us. It does that so you can type ci proportions and obtain correct confidence intervals for all the variables that are $0 / 1$ in your data.

## Confidence intervals for variances

The ci variances command computes confidence intervals for the variances or, if the sd option is specified, for the standard deviations. The default is a normal-based confidence interval that assumes the data are normal and uses a $\chi^{2}$ distribution to construct the confidence intervals. If normality is suspect, you may consider using the bonett option to compute Bonett (2006) confidence intervals, which are more robust to nonnormality.
> Example 8: Normal-based confidence intervals
So far, we have restricted our attention to confidence intervals for means and proportions. Typically, when people think of statistical inference, they usually have in mind inferences concerning population means. However, the population parameter of interest will vary from one situation to another. In many scenarios, the population variance is as important as the population mean. For example, in a quality
control study, a machine that fills 16 -ounce canned peas is investigated at regular time intervals. A random sample of $n=8$ containers is selected every hour. Ideally, the amount of peas in a can should vary only slightly about the 16 -ounce value. If the variance was large, then a large proportion of cans would be either underfilled, thus cheating the customer, or overfilled, thus resulting in economic loss to the manufacturing company. Suppose that the weights of 16 -ounce cans filled by the machine are normally distributed. The acceptable variability in the weights is expected to be 0.09 with the respective standard deviation of 0.3 ounces. To monitor the machine's performance, we can compute confidence intervals for the variance of the weights of cans:

| . use https://www.stata-press.com/data/r18/peas_normdist <br> (Weights of canned peas, normal distribution) |
| :--- |
| . ci variances weight <br> Variable |
| weight |

The command reports the sample estimate of the variance of 0.39 with the $95 \%$ confidence interval of $[0.17,1.61]$.

Instead of the variance, we may be interested in confidence intervals for the standard deviation. We can specify the sd option to compute such confidence intervals.

```
. ci variances weight, sd
```

| Variable | Obs | Std. dev. | [95\% conf. interval] |  |
| ---: | ---: | :---: | ---: | :---: |
| weight | 8 | .6235711 | .4122891 | 1.269137 |

The $95 \%$ confidence interval for the standard deviation of the weights is [0.41, 1.27]. Because the desired value for the standard deviation, 0.3 ounces, falls outside the interval, the machine may require some tuning.

Confidence intervals in example 8 are based on the assumption that the random sample is selected from a population having a normal distribution. Nonnormality of the population distribution, in the form of skewness or heavy tails, can have a drastic impact on the asymptotic coverage probability of the normal-based confidence intervals. This is the case even for distributions that are similar to normal. Scheffé $(1959,336)$ showed that the normal-based interval has an asymptotic coverage probability of about $0.76,0.63,0.60$, and 0.51 for the logistic, $t$ with seven degrees of freedom, Laplace, and $t$ with five degrees of freedom distributions, respectively. Miller $(1997,264)$ describes this situation as "catastrophic" because these distributions are symmetric and not easily distinguishable from a normal distribution unless the sample size is large. Hence, it is judicious to evaluate the normality of the data prior to constructing the normal-based confidence intervals for variances or standard deviations.

Bonett (2006) proposed a confidence interval that performs well in small samples under moderate departures from normality. His interval performs only slightly worse than the exact normal-based confidence interval when sampling from a normal distribution. A larger sample size provides Bonett confidence intervals with greater protection against nonnormality.

## > Example 9: Bonett confidence interval for normal data

We will repeat example 8 and construct a Bonett confidence interval for the standard deviation by specifying the bonett option. The results are similar, and both examples lead to the same inferential conclusion.
. ci variances weight, sd bonett

| Variable | Obs | Std. dev. | [95\% conf. interval] |  |
| ---: | :---: | :---: | :---: | :---: |
| weight | 8 | .6235711 | .3997041 | 1.288498 |

The Bonett confidence interval is wider than the normal-based confidence interval in example 8. For normal data, Bonett (2006) suggested that if Bonett confidence interval is used for a sample of size $n+3$, then its average width will be about the same as the average width of the normal-based confidence interval from a sample size of $n$. Sampling three more observations may be a small price to pay because Bonett confidence intervals perform substantially better than the normal-based confidence intervals for nonnormal data.

## > Example 10: Bonett confidence interval for nonnormal data

The following data have been generated from a $t$ distribution with five degrees of freedom to illustrate the effect of wrongfully using the normal-based confidence interval when the data-generating process is not normal.
. use https://www.stata-press.com/data/r18/peas_tdist

| (Weights of canned peas, t distribution) |
| :--- |
| - ci variances weight, sd |
| Variable |$\quad$| Obs | Std. dev. | [95\% conf. interval] |  |  |
| ---: | :--- | :--- | :--- | :--- |
| weight | 8 | 2.226558 | 1.472143 | 4.531652 |

The standard deviation of a $t$ distribution with five degrees of freedom is $\sqrt{5 / 3} \approx 1.29$ and falls outside the confidence interval limits. If we suspect that data may not be normal, the Bonett confidence interval is typically a better choice:
. ci variances weight, sd bonett

| Variable | Obs | Std. dev. | Bonett |  |
| ---: | ---: | :--- | :--- | :--- |
| [95\% conf. interval] |  |  |  |  |

The value 1.29 is within the limits of the Bonett confidence interval [1.14, 5.77]

## Immediate form

So far, we computed confidence intervals for various parameters using data in memory. We can also compute confidence intervals using only data summaries, without any data in memory. Each of the considered ci commands has an immediate cii version that computes the respective confidence intervals using data summaries.

## > Example 11: Confidence interval for a normal mean

We are reading a soon-to-be-published paper by a colleague. In it is a table showing the number of observations, mean, and standard deviation of the 1980 median family income for the Northeast and West. We correctly think that the paper would be much improved if it included the confidence intervals. The paper claims that for 166 cities in the Northeast, the average of median family income is $\$ 19,509$ with a standard deviation of $\$ 4,379$ :

For the Northeast:

```
. cii means 166 19509 4379
```

| Variable | Obs | Mean | Std. err. | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: | :---: |
|  | 166 | 19509 | 339.8763 | 18837.93 | 20180.07 |

For the West:

| cii means 256 <br> Variable | Obs |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: |
|  | 25003 | Mean | Std. err. | [95\% conf. interval] |  |
|  | 256 | 22557 | 312.6875 | 21941.22 | 23172.78 |

> Example 12: Confidence interval for a Poisson mean
The number of reported traffic accidents in Santa Monica over a 24 -hour period is 27 . We need know nothing else to compute a confidence interval for the mean number of accidents for a day:

```
. cii means 1 27, poisson
```

| Variable | Exposure | Mean | Std. err. | Poisson exact <br> [95\% conf. interval] |  |
| :---: | ---: | :---: | :---: | :---: | :---: |
|  | 1 | 27 | 5.196152 | 17.79317 | 39.28358 |

## > Example 13: Confidence interval for a proportion

We flip a coin 10 times, and it comes up heads only once. We are shocked and decide to obtain a $99 \%$ confidence interval for this coin:
. cii proportions 10 1, level(99)

| Variable | Obs |  | Binomial exact <br> [99\% conf. interval] |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: |
|  | 10 | .1 | .0948683 | .0005011 | .5442871 |

## > Example 14: Confidence interval for a variance

A company fills 32-ounce tomato juice jars with a quantity of juice having a normal distribution with a claimed variance not exceeding 0.2 . A random sample of 15 jars is collected to evaluate this claim. The sample variance is 0.5 :
. cii variances 150.5

| Variable | Obs | Variance | [95\% conf. interval] |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: |
|  | 15 | .5 | .2680047 | 1.243621 |  |

Because the advertised value of 0.2 does not fall inside the confidence interval, the company is allowing too much variation in the amount of tomato juice per jar.
> Example 15: Confidence interval for a standard deviation
Suppose the director of statistical development at a statistical software company is a big soccer fan and requires all developers to play on the company team in the city's local soccer league. Ten developers are randomly selected to participate in the game. To ensure an advantage over other teams, the director requires each of the 10 developers to cover 6 miles on average each game. Being merciful, she will tolerate a standard deviation of 0.3 miles across different players, arguing that this will keep the team's performance consistent. The distance covered by each player is measured using a pedometer. At the end of the game, the sample standard deviation of the distances covered by the 10 players was 0.56 miles:

```
. cii variances 10 0.56, sd
```

| Variable | Obs | Std. dev. | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: |
|  | 10 | .56 | .3851877 | 1.022342 |

Because the confidence interval does not include the designated value for the standard deviation, 0.3 miles, it is clear the team is not meeting standards, and an unpleasant meeting is planned.

## > Example 16: Confidence interval for a standard deviation of nonnormal data

Continuing with example 15, a clever statistician points out that distances covered by company players in a soccer match do not follow the normal distribution because some players, mostly econometricians, walk on the field, while others, mostly statisticians, do all the running. Therefore, the normal-based confidence interval (which assumes normality) is not valid. Instead, we should use the Bonett confidence interval, which additionally requires an estimate of kurtosis; see Methods and formulas. If kurtosis is estimated to be 5, we would obtain the following:

```
. cii variances 10 0.56 5, sd bonett
```

| Variable | Obs | Std. dev. | Bonett |  |
| :---: | :---: | :---: | :---: | :---: |
|  | [95\% conf. interval] |  |  |  |
|  | 10 | .56 | .2689449 | 1.45029 |

The Bonett confidence interval now contains the specified value for the standard deviation, 0.3 miles. The director of statistics concludes that overall team performance is acceptable. An uncomfortable meeting is still planned but for a smaller group.

## Stored results

ci means and cii means store the following in r() :
Scalars
$r(N) \quad$ number of observations or, if poisson is specified, exposure
$r$ (mean)
$r$ (se)
r(lb)
$r(u b)$
r(level) mean
estimate of standard error
lower bound of confidence interval
upper bound of confidence interval
confidence level of confidence interval
Macros
$r$ (citype) normal or poisson; type of confidence interval
r(exposure)
ci proportions and cii proportions store the following in $r()$ :

Scalars
$r(N)$
$r$ (proportion)
$r(\mathrm{se})$
r(lb)
$r(u b)$
$r$ (level)
Macros
r(citype)
number of observations
proportion
estimate of standard error
lower bound of confidence interval upper bound of confidence interval confidence level of confidence interval
exact, wald, wilson, agresti, or jeffreys; type of confidence interval
ci variances and cii variances store the following in $r()$ :
Scalars
$r(N) \quad$ number of observations
$r$ (Var) variance
$r$ (sd) standard deviation, if sd is specified
$r$ (kurtosis) kurtosis, only if bonett is specified
$r(l b) \quad$ lower bound of confidence interval
$r$ (ub)
$r$ (level) upper bound of confidence interval confidence level of confidence interval
Macros
r(citype) normal or bonett, type of confidence interval

## Methods and formulas

Methods and formulas are presented under the following headings:
Normal mean
Poisson mean
Binomial proportion
Variance and standard deviation

## Normal mean

Define $n, \bar{x}$, and $s^{2}$ as, respectively, the number of observations, (weighted) average, and (unbiased) estimated variance of the variable in question; see $[R]$ summarize.

The standard error of the mean, $s_{\mu}$, is defined as $\sqrt{s^{2} / n}$.
Let $\alpha$ be $1-l / 100$, where $l$ is the confidence level specified by the user in the level() option. Define $t_{\alpha / 2}$ as the two-sided $t$ statistic corresponding to a significance level of $\alpha$ with $n-1$ degrees of freedom; $t_{\alpha / 2}$ is obtained from Stata as invttail ( $n-1,0.5 * \alpha$ ). The lower and upper confidence bounds are, respectively, $\bar{x}-s_{\mu} t_{\alpha / 2}$ and $\bar{x}+s_{\mu} t_{\alpha / 2}$.

## Poisson mean

Given the total cases, $k$, the estimate of the expected count $\lambda$ is $k$, and its standard error is $\sqrt{k}$. ci means with option poisson calculates the exact confidence interval [ $\lambda_{1}, \lambda_{2}$ ] such that

$$
\operatorname{Pr}\left(K \geq k \mid \lambda=\lambda_{1}\right)=\alpha / 2
$$

and

$$
\operatorname{Pr}\left(K \leq k \mid \lambda=\lambda_{2}\right)=\alpha / 2
$$

where $K$ is Poisson with mean $\lambda$. Solution is obtained by Newton's method. If $k=0$, the calculation of $\lambda_{1}$ is skipped. All values are then reported as rates, which are the above numbers divided by the total exposure.

## Binomial proportion

Given $k$ successes of $n$ trials, the estimated probability of a success is $\widehat{p}=k / n$ with standard error $\sqrt{\widehat{p}(1-\widehat{p}) / n}$. ci calculates the exact (Clopper-Pearson) confidence interval $\left[p_{1}, p_{2}\right]$ such that

$$
\operatorname{Pr}\left(K \geq k \mid p=p_{1}\right)=\alpha / 2
$$

and

$$
\operatorname{Pr}\left(K \leq k \mid p=p_{2}\right)=\alpha / 2
$$

where $K$ is distributed as $\operatorname{binomial}(n, p)$. The endpoints may be obtained directly by using Stata's invbinomial() function. If $k=0$ or $k=n$, the calculation of the appropriate tail is skipped.

The Wald interval is $\widehat{p} \pm z_{\alpha / 2} \sqrt{\widehat{p}(1-\widehat{p}) / n}$, where $z_{\alpha / 2}$ is the $1-\alpha / 2$ quantile of the standard normal. The interval is obtained by inverting the acceptance region of the large-sample Wald test of $H_{0}: p=p_{0}$ versus the two-sided alternative. That is, the confidence interval is the set of all $p_{0}$ such that

$$
\left|\frac{\widehat{p}-p_{0}}{\sqrt{n^{-1} \widehat{p}(1-\widehat{p})}}\right| \leq z_{\alpha / 2}
$$

The Wilson interval is a variation on the Wald interval, using the null standard error $\sqrt{n^{-1} p_{0}\left(1-p_{0}\right)}$ in place of the estimated standard error $\sqrt{n^{-1} \widehat{p}(1-\widehat{p})}$ in the above expression. Inverting this acceptance region is more complicated yet results in the closed form

$$
\frac{k+z_{\alpha / 2}^{2} / 2}{n+z_{\alpha / 2}^{2}} \pm \frac{z_{\alpha / 2} n^{1 / 2}}{n+z_{\alpha / 2}^{2}}\left\{\widehat{p}(1-\widehat{p})+\frac{z_{\alpha / 2}^{2}}{4 n}\right\}^{1 / 2}
$$

The Agresti-Coull interval is basically a Wald interval that borrows its center from the Wilson interval. Defining $\widetilde{k}=k+z_{\alpha / 2}^{2} / 2, \widetilde{n}=n+z_{\alpha / 2}^{2}$, and (hence) $\widetilde{p}=\widetilde{k} / \widetilde{n}$, the Agresti-Coull interval is

$$
\widetilde{p} \pm z_{\alpha / 2} \sqrt{\widetilde{p}(1-\widetilde{p}) / \widetilde{n}}
$$

When $\alpha=0.05, z_{\alpha / 2}$ is near enough to 2 that $\widetilde{p}$ can be thought of as a typical estimate of proportion where two successes and two failures have been added to the sample (Agresti and Coull 1998). This typical estimate of proportion makes the Agresti-Coull interval an easy-to-present alternative for introductory statistics students.

The Jeffreys interval is a Bayesian credible interval and is based on the Jeffreys prior, which is the $\operatorname{Beta}(1 / 2,1 / 2)$ distribution. Assigning this prior to $p$ results in a posterior distribution for $p$ that is Beta with parameters $k+1 / 2$ and $n-k+1 / 2$. The Jeffreys interval is then taken to be the $1-\alpha$ central posterior probability interval, namely, the $\alpha / 2$ and $1-\alpha / 2$ quantiles of the $\operatorname{Beta}(k+1 / 2, n-k+1 / 2)$ distribution. These quantiles may be obtained directly by using Stata's invibeta() function. See [BAYES] bayesstats summary for more details about credible intervals.

## Variance and standard deviation

Let $X_{1}, \ldots, X_{n}$ be a random sample and assume that $X_{i} \sim N\left(\mu, \sigma^{2}\right)$. Because $(n-1) s^{2} / \sigma^{2} \sim$ $\chi_{n-1}^{2}$, we have $\operatorname{Pr}\left\{\chi_{n-1, \alpha / 2}^{2} \leq(n-1) s^{2} / \sigma^{2} \leq \chi_{n-1,1-\alpha / 2}^{2}\right\}=1-\alpha$, where $\chi_{n-1, \alpha / 2}^{2}$ and $\chi_{n-1,1-\alpha / 2}^{2}$ are the $\alpha / 2$ and $1-\alpha / 2$ quantiles of the $\chi_{n-1}^{2}$ distribution. Thus, the normal-based confidence interval for the population variance $\sigma^{2}$ with $100(1-\alpha) \%$ confidence level is given by

$$
I_{\text {normal }}=\left[\frac{(n-1) s^{2}}{\chi_{n-1,1-\alpha / 2}^{2}}, \frac{(n-1) s^{2}}{\chi_{n-1, \alpha / 2}^{2}}\right]
$$

$\chi_{n-1,1-\alpha / 2}^{2}$ and $\chi_{n-1, \alpha / 2}^{2}$ are obtained from Stata as invchi2tail ( $n-1,0.5 * \alpha$ ) and invchi2( $n-$ $1,0.5 * \alpha)$, respectively.

The normal-based confidence interval is very sensitive to minor departures from the normality assumption, and its performance does not improve with increasing sample size. For scenarios in which the population distribution is not normal, the actual coverage probability of the normal-based confidence interval can be drastically lower than the nominal confidence level $\alpha$.

Bonett (2006) proposed an alternative to the normal-based confidence interval that is nearly exact under normality and has coverage probability close to $1-\alpha$ under moderate nonnormality. It also has $1-\alpha$ asymptotic coverage probability for nonnormal distributions with finite fourth moment. Instead of assuming that $X_{i} \sim N\left(\mu, \sigma^{2}\right)$, Bonett's approach requires continuous i.i.d. random variables with finite fourth moment. The variance of $s^{2}$ may be expressed as $\sigma^{4}\left\{\gamma_{4}-(n-3) /(n-1)\right\} / n$ (see Casella and Berger [2002, ex. 5.8, 257]), where $\gamma_{4}=\mu^{4} / \sigma^{4}$ is the kurtosis and $\mu^{4}=E\left(X_{i}-\mu\right)^{4}$ is the population fourth central moment. The variance-stabilizing transformation $\ln \left(s^{2}\right)$ and the delta method can be used to construct an asymptotic $100(1-\alpha) \%$ confidence interval for $\sigma^{2}$,

$$
\left[\exp \left\{\ln \left(s^{2}\right)-z_{\alpha / 2} s e\right\}, \exp \left\{\ln \left(s^{2}\right)+z_{\alpha / 2} s e\right\}\right]
$$

where se $=\left\{\widehat{\gamma}_{4}-(n-3) /(n-1)\right\} / n \approx \operatorname{Var}\left\{\ln \left(s^{2}\right)\right\}$ and $\widehat{\gamma}_{4}$ is an estimate of the kurtosis. Bonett introduced three adjustments to improve the small-sample properties of the above confidence interval. First, he swapped the inner and outer denominator in the expression for se and changed it to $\left\{\widehat{\gamma}_{4}-(n-3) / n\right\} /(n-1)$. This was suggested by Shoemaker (2003) who used it to improve the small-sample performance of his variance test. Second, with regard to the estimation of kurtosis, Bonett proposed $\widehat{\gamma}_{4}=n \sum\left(X_{i}-m\right)^{4} /\left\{\sum\left(X_{i}-\bar{X}\right)^{2}\right\}^{2}$, where $m$ is a trimmed mean with a trimproportion equal to $1 /\left\{2(n-4)^{1 / 2}\right\}$. This kurtosis estimator reduces the negative bias in symmetric and skewed heavy-tailed distributions. Last, he empirically derived a small-sample correction factor $c=n /\left(n-z_{\alpha / 2}\right)$ that helps equalize the tail probabilities. These modifications yield

$$
I_{\text {Bonett }}=\left[\exp \left\{\ln \left(c s^{2}\right)-z_{\alpha / 2} s e\right\}, \exp \left\{\ln \left(c s^{2}\right)+z_{\alpha / 2} s e\right\}\right]
$$

where $z_{\alpha / 2}$ is the $1-\alpha / 2$ quantile of the standard normal and se $=c\left[\left\{\widehat{\gamma}_{4}-(n-3) / n\right\} /(n-1)\right]$.
Taking the square root of the endpoints of both intervals gives confidence intervals for the standard deviation $\sigma$.

Edwin Bidwell (E. B.) Wilson (1879-1964) majored in mathematics at Harvard and studied and taught at Yale and mit before returning to Harvard in 1922. He worked in mathematics, physics, and statistics. His method for binomial intervals can be considered a precursor, for a particular problem, of Neyman's concept of confidence intervals.
Jerzy Neyman (1894-1981) was born in Bendery, Russia, now Moldavia. He studied and then taught at Kharkov University, moving from physics to mathematics. In 1921, Neyman moved to Poland, where he worked in statistics at Bydgoszcz and then Warsaw. Neyman received a Rockefeller Fellowship to work with Karl Pearson at University College London. There he collaborated with Egon Pearson, Karl's son, on the theory of hypothesis testing. Life in Poland became progressively more difficult, and Neyman returned to UCL to work there from 1934 to 1938. At this time, he published on the theory of confidence intervals. He then was offered a post in California at Berkeley, where he settled. Neyman established an outstanding statistics department and remained highly active in research, including applications in astronomy, meteorology, and medicine. He was one of the great statisticians of the 20th century.

## Acknowledgment

We thank Nicholas J. Cox of the Department of Geography at Durham University, UK, who is coeditor of the Stata Journal and author of Speaking Stata Graphics for his assistance with the jeffreys and wilson options.

## References

Agresti, A., and B. A. Coull. 1998. Approximate is better than "exact" for interval estimation of binomial proportions. American Statistician 52: 119-126. https://doi.org/10.1080/00031305.1998.10480550.
Bonett, D. G. 2006. Approximate confidence interval for standard deviation of nonnormal distributions. Computational Statistics and Data Analysis 50: 775-782. https://doi.org/10.1016/j.csda.2004.10.003.

Brown, L. D., T. T. Cai, and A. DasGupta. 2001. Interval estimation for a binomial proportion. Statistical Science 16: 101-133. https://doi.org/10.1214/ss/1009213286.

Campbell, M. J., D. Machin, and S. J. Walters. 2007. Medical Statistics: A Textbook for the Health Sciences. 4th ed. Chichester, UK: Wiley.

Casella, G., and R. L. Berger. 2002. Statistical Inference. 2nd ed. Pacific Grove, CA: Duxbury.
Clopper, C. J., and E. S. Pearson. 1934. The use of confidence or fiducial limits illustrated in the case of the binomial. Biometrika 26: 404-413. https://doi.org/10.1093/biomet/26.4.404.

Cook, A. 1990. Sir Harold Jeffreys, 2 April 1891-18 March 1989. Biographical Memoirs of Fellows of the Royal Society 36: 303-333. https://doi.org/10.1098/rsbm.1990.0034.

Earnest, A. 2017. Essentials of a Successful Biostatistical Collaboration. Boca Raton, FL: CRC Press.
Jeffreys, H. 1946. An invariant form for the prior probability in estimation problems. Proceedings of the Royal Society of London, A ser., 186: 453-461. https://doi.org/10.1098/rspa.1946.0056.

Lash, T. L., T. J. VanderWeele, S. Haneuse, and K. J. Rothman. 2021. Modern Epidemiology. 4th ed. Philadelphia: Wolters Kluwer.

Lindley, D. V. 2001. Harold Jeffreys. In Statisticians of the Centuries, ed. C. C. Heyde and E. Seneta, 402-405. New York: Springer.

Miller, R. G., Jr. 1997. Beyond ANOVA: Basics of Applied Statistics. London: Chapman and Hall.
Reid, C. 1982. Neyman-from Life. New York: Springer.
Scheffé, H. 1959. The Analysis of Variance. New York: Wiley.
Shoemaker, L. H. 2003. Fixing the $F$ test for equal variances. American Statistician 57: 105-114. https://doi.org/10.1198/0003130031441.

Stigler, S. M. 1997. Wilson, Edwin Bidwell. In Leading Personalities in Statistical Sciences: From the Seventeenth Century to the Present, ed. N. L. Johnson and S. Kotz, 344-346. New York: Wiley.
Ward, B. W. 2019. kg_nchs: A command for Korn-Graubard confidence intervals and National Center for Health Statistics’ Data Presentation Standards for Proportions. Stata Journal 19: 510-522.
Wilson, E. B. 1927. Probable inference, the law of succession, and statistical inference. Journal of the American Statistical Association 22: 209-212. https://doi.org/10.2307/2276774.

## Also see

[R] ameans - Arithmetic, geometric, and harmonic means
[R] bitest - Binomial probability test
[R] centile - Report centile and confidence interval
[R] prtest - Tests of proportions
[R] sdtest - Variance-comparison tests
[R] summarize - Summary statistics
$[\mathrm{R}]$ ttest $-t$ tests (mean-comparison tests)
[D] pctile - Create variable containing percentiles

## Title

clogit — Conditional (fixed-effects) logistic regression

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

clogit fits a conditional logistic regression model for matched case-control data, also known as a fixed-effects logit model for panel data. clogit can compute robust and cluster-robust standard errors and adjust results for complex survey designs.

See [CM] cmclogit if you want to fit McFadden's choice model (McFadden 1974).

## Quick start

Conditional logistic regression model of y on x with matched case-control pairs data identified by idvar

```
clogit y x, group(idvar)
```

Fixed-effects logistic regression model with panels identified by idvar
clogit y x, group(idvar)
Add categorical variable a and report results as odds ratios
clogit y x i.a, group(idvar) or
Same as above, but using sampling probability weight wvar
clogit y x i.a [pweight = wvar], group(idvar) or

## Menu

Statistics $>$ Binary outcomes $>$ Conditional logistic regression

## Syntax

clogit depvar [indepvars] [if] [in] [weight], group(varname) [options]
depvar is treated as binary regardless of values; depvar equal to nonzero and nonmissing (typically equal to 1) indicates a positive outcome, whereas depvar equal to 0 indicates a negative outcome.

## options

Description
Model

* group (varname)
offset (varname)
constraints (constraints)
matched group variable
include varname in model with coefficient constrained to 1
apply specified linear constraints
SE/Robust
vce(vcetype)
nonest
vcetype may be oim, robust, cluster clustvar, opg, bootstrap, or jackknife
do not check that panels are nested within clusters
Reporting
level(\#)
or
nocnsreport
display_options
set confidence level; default is level (95)
report odds ratios
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

Maximization
maximize_options
collinear
coeflegend
control the maximization process; seldom used
keep collinear variables
display legend instead of statistics

* group (varname) is required.
indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
bayes, bootstrap, by, collect, fp, jackknife, mfp, mi estimate, nestreg, rolling, statsby, stepwise, and svy are allowed; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: clogit.
vce (bootstrap) and vce (jackknife) are not allowed with the mi estimate prefix; see [MI] mi estimate.
Weights are not allowed with the bootstrap prefix; see [R] bootstrap.
vce(), nonest, and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed (see [U] 11.1.6 weight), but they are interpreted to apply to groups as a whole, not to individual observations. See Use of weights below.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.


## Options

## Model

group (varname) is required; it specifies an identifier variable (numeric or string) for the matched groups. strata(varname) is a synonym for group().
offset (varname), constraints (constraints); see [R] Estimation options.
vce(vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
nonest, available only with vce(cluster clustvar), prevents checking that matched groups are nested within clusters. It is the user's responsibility to verify that the standard errors are theoretically correct.

Reporting
level (\#); see [R] Estimation options.
or reports the estimated coefficients transformed to odds ratios, that is, $e^{b}$ rather than $b$. Standard errors and confidence intervals are similarly transformed. This option affects how results are displayed, not how they are estimated. or may be specified at estimation or when replaying previously estimated results.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% $\% m t$ ), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate (\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from (init-specs); see [R] Maximize. These options are seldom used.

Setting the optimization type to technique (bhhh) resets the default vcetype to vce (opg).
The following options are available with clogit but are not shown in the dialog box:
collinear, coeflegend; see [R] Estimation options.

## Remarks and examples

Remarks are presented under the following headings:

Introduction<br>Matched case-control data<br>Use of weights<br>Fixed-effects logit

## Introduction

clogit fits maximum likelihood models with a dichotomous dependent variable coded as $0 / 1$ (more precisely, clogit interprets 0 and not 0 to indicate the dichotomy). Conditional logistic analysis differs from regular logistic regression in that the data are grouped and the likelihood is calculated relative to each group; that is, a conditional likelihood is used. See Methods and formulas at the end of this entry.

Biostatisticians and epidemiologists call these models conditional logistic regression for matched case-control groups (see, for example, Hosmer, Lemeshow, and Sturdivant [2013, chap. 7]) and fit them when analyzing matched case-control studies with $1: 1$ matching, $1: k_{2 i}$ matching, or $k_{1 i}: k_{2 i}$ matching, where $i$ denotes the $i$ th matched group for $i=1,2, \ldots, n$, where $n$ is the total number of groups. clogit fits a model appropriate for all of these matching schemes or for any mix of the schemes because the matching $k_{1 i}: k_{2 i}$ can vary from group to group. clogit always uses the true conditional likelihood, not an approximation. Biostatisticians and epidemiologists sometimes refer to the matched groups as "strata", but we will stick to the more generic term "group".

Economists and other social scientists typically call the model fit by clogit a fixed-effects logit model for panel data (see, for example, Chamberlain [1980]). The data used to fit a fixed-effects logit model look exactly like the data biostatisticians and epidemiologists call $k_{1 i}: k_{2 i}$ matched case-control data. In terms of how the data are arranged, $k_{1 i}: k_{2 i}$ matching means that in the $i$ th group, the dependent variable is 1 a total of $k_{1 i}$ times and 0 a total of $k_{2 i}$ times. There are a total of $T_{i}=k_{1 i}+k_{2 i}$ observations for the $i$ th group. This data arrangement is what economists and other social scientists call "panel data", "longitudinal data", or "cross-sectional time-series data".

So no matter what terminology you use, the computation and the use of the clogit command is the same. The following example shows how your data should be arranged to use clogit.

## > Example 1

Suppose that we have grouped data with the variable id containing a unique identifier for each group. Our outcome variable, $y$, contains 0 s and 1 s . If we were biostatisticians, $\mathrm{y}=1$ would indicate a case, $\mathrm{y}=0$ would be a control, and id would be an identifier variable that indicates the groups of matched case-control subjects.

If we were economists, $\mathrm{y}=1$ might indicate that a person was unemployed at any time during a year and $\mathrm{y}=0$, that a person was employed all year, and id would be an identifier variable for persons.

If we list the first few observations of this dataset, it looks like

```
. use https://www.stata-press.com/data/r18/clogitid
. list y x1 x2 id in 1/11
```

|  | y | x1 | x 2 | id |
| :---: | :---: | :---: | :---: | :---: |
| 1. | 0 | 0 | 4 | 1014 |
| 2. | 0 | 1 | 4 | 1014 |
| 3. | 0 | 1 | 6 | 1014 |
| 4. | 1 | 1 | 8 | 1014 |
| 5. | 0 | 0 | 1 | 1017 |
| 6. | 0 | 0 | 7 | 1017 |
| 7. | 1 | 1 | 10 | 1017 |
| 8. | 0 | 0 | 1 | 1019 |
| 9. | 0 | 1 | 7 | 1019 |
| 10. | 1 | 1 | 7 | 1019 |
| 11. | 1 | 1 | 9 | 1019 |

Pretending that we are biostatisticians, we describe our data as follows. The first group (id $=1014$ ) consists of four matched persons: 1 case $(\mathrm{y}=1)$ and three controls $(\mathrm{y}=0)$, that is, $1: 3$ matching. The second group has $1: 2$ matching, and the third $2: 2$.

Pretending that we are economists, we describe our data as follows. The first group consists of 4 observations (one per year) for person 1014. This person had a period of unemployment during 1 year of 4 . The second person had a period of unemployment during 1 year of 3 , and the third had a period of 2 years of 4 .

Our independent variables are x 1 and x 2 . To fit the conditional (fixed-effects) logistic model, we type

```
. clogit y x1 x2, group(id)
note: multiple positive outcomes within groups encountered.
Iteration 0: Log likelihood = -123.42828
Iteration 1: Log likelihood = -123.41386
Iteration 2: Log likelihood = -123.41386
Conditional (fixed-effects) logistic regression Number of obs = 369
    LR chi2(2) = 9.07
    Prob > chi2 = 0.0107
    Pseudo R2 = 0.0355
```

| y | Coefficient | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| x1 | .653363 | .2875215 | 2.27 | 0.023 | .0898312 | 1.216895 |
| x 2 | .0659169 | .0449555 | 1.47 | 0.143 | -.0221943 | .1540281 |

## Technical note

The message "note: multiple positive outcomes within groups encountered" at the top of the clogit output for the previous example merely informs us that we have $k_{1 i}: k_{2 i}$ matching with $k_{1 i}>1$ for at least one group. If your data should be $1: k_{2 i}$ matched, this message tells you that there is an error in the data somewhere.

We can see the distribution of $k_{1 i}$ and $T_{i}=k_{1 i}+k_{2 i}$ for the data of the example 1 by using the following steps:

```
. by id, sort: generate k1 = sum(y)
. by id: replace k1 = . if _n < _N
(303 real changes made, 303 to missing)
. by id: generate T = sum(y<.)
. by id: replace T = . if _n < _N
(303 real changes made, 303 to missing)
. tabulate k1
\begin{tabular}{r|rrr} 
k1 & Freq. & Percent & Cum. \\
\hline 1 & 48 & 72.73 & 72.73 \\
2 & 12 & 18.18 & 90.91 \\
3 & 4 & 6.06 & 96.97 \\
4 & 2 & 3.03 & 100.00 \\
\hline Total & 66 & 100.00 &
\end{tabular}
```

. tabulate T

| T | Freq. | Percent | Cum. |
| ---: | ---: | ---: | ---: |
| 2 | 5 | 7.58 | 7.58 |
| 3 | 5 | 7.58 | 15.15 |
| 4 | 12 | 18.18 | 33.33 |
| 5 | 11 | 16.67 | 50.00 |
| 6 | 13 | 19.70 | 69.70 |
| 7 | 8 | 12.12 | 81.82 |
| 8 | 3 | 4.55 | 86.36 |
| 9 | 7 | 10.61 | 96.97 |
| 10 | 2 | 3.03 | 100.00 |
| Total | 66 | 100.00 |  |

We see that $k_{1 i}$ ranges from 1 to 4 and $T_{i}$ ranges from 2 to 10 for these data.

## - Technical note

For $k_{1 i}: k_{2 i}$ matching (and hence in the general case of fixed-effects logit), clogit uses a recursive algorithm to compute the likelihood, which means that there are no limits on the size of $T_{i}$. However, computation time is proportional to $\sum T_{i} \min \left(k_{1 i}, k_{2 i}\right)$, so clogit will take roughly 10 times longer to fit a model with 10:10 matching than one with $1: 10$ matching. But clogit is fast, so computation time becomes an issue only when $\min \left(k_{1 i}, k_{2 i}\right)$ is around 100 or more. See Methods and formulas for details.

## Matched case-control data

Here we give a more detailed example of matched case-control data.

## $>$ Example 2

Hosmer, Lemeshow, and Sturdivant (2013, 24) present data on matched pairs of infants, each pair having one with low birthweight and another with regular birthweight. The data are matched on age of the mother. Several possible maternal exposures are considered: race (three categories), smoking status, presence of hypertension, presence of uterine irritability, previous preterm delivery, and weight at the last menstrual period.

```
. use https://www.stata-press.com/data/r18/lowbirth2, clear
(Applied Logistic Regression, Hosmer & Lemeshow)
. describe
Contains data from https://www.stata-press.com/data/r18/lowbirth2.dta
    Observations: 112 Applied Logistic Regression,
                                    Hosmer & Lemeshow
            Variables: 9 30 Jan 2022 08:46
```

| Variable <br> name | Storage <br> type | Display <br> format | Value <br> label | Variable label |
| :--- | :--- | :--- | :--- | :--- |
| pairid | byte | $\% 8.0 \mathrm{~g}$ |  | Case-control pair ID <br> low byte |
| age | byte | $\% 8.0 \mathrm{~g}$ |  | Baby has low birthweight |
| lwt | int | $\% 8.0 \mathrm{~g}$ |  | Age of mother |
| smoke | byte | $\% 8.0 \mathrm{~g}$ |  | Mother's last menstrual weight |
| ptd | byte | $\% 8.0 \mathrm{~g}$ |  | Mother smoked during pregnancy |
| ht | byte | $\% 8.0 \mathrm{~g}$ |  | Mother had previous preterm baby |
| ui | byte | $\% 8.0 \mathrm{~g}$ |  | Mother has hypertension |
| race | byte | $\% 9.0 \mathrm{~g}$ | race | Uterine irritability |
|  |  |  |  | Race of mother |

Sorted by:
We list the case-control indicator variable, low; the match identifier variable, pairid; and two of the covariates, lwt and smoke, for the first 10 observations.
. list low lwt smoke pairid in $1 / 10$

|  | low | lwt | smoke | pairid |
| ---: | ---: | ---: | ---: | ---: |
| 1. | 0 | 135 | 0 | 1 |
| 2. | 1 | 101 | 1 | 1 |
| 3. | 0 | 98 | 0 | 2 |
| 4. | 1 | 115 | 0 | 2 |
| 5. | 0 | 95 | 0 | 3 |
| 6. | 1 | 130 | 0 | 3 |
| 7. | 0 | 103 | 0 | 4 |
| 8. | 1 | 130 | 1 | 4 |
| 9. | 0 | 122 | 1 | 5 |
| 10. | 1 | 110 | 1 | 5 |

We fit a conditional logistic model of low birthweight on mother's weight, race, smoking behavior, and history.

| Conditional (fixed-effects) logistic regression | Number of obs | $=112$ |
| :---: | :---: | :---: |
|  | LR chi2(7) | $=26.04$ |
|  | Prob > chi2 | $=0.0005$ |
| Log likelihood = -25.794271 | Pseudo R2 | $=0.3355$ |


| low | Coefficient | Std. err. | $z$ | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| lwt | -.0183757 | .0100806 | -1.82 | 0.068 | -.0381333 | .0013819 |
| smoke | 1.400656 | .6278396 | 2.23 | 0.026 | .1701131 | 2.631199 |
| ptd | 1.808009 | .7886502 | 2.29 | 0.022 | .2622828 | 3.353735 |
| ht | 2.361152 | 1.086128 | 2.17 | 0.030 | .2323796 | 4.489924 |
| ui | 1.401929 | .6961585 | 2.01 | 0.044 | .0374836 | 2.766375 |
| race |  |  |  |  |  |  |
| Black | .5713643 | .689645 | 0.83 | 0.407 | -.7803149 | 1.923044 |
| Other | -.0253148 | .6992044 | -0.04 | 0.971 | -1.39573 | 1.345101 |

We might prefer to see results presented as odds ratios. We could have specified the or option when we first fit the model, or we can now redisplay results and specify or:
. clogit, or


| low | Odds ratio | Std. err. | $\mathbf{z}$ | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| lwt | .9817921 | .009897 | -1.82 | 0.068 | .9625847 | 1.001383 |
| smoke | 4.057862 | 2.547686 | 2.23 | 0.026 | 1.185439 | 13.89042 |
| ptd | 6.098293 | 4.80942 | 2.29 | 0.022 | 1.299894 | 28.60938 |
| ht | 10.60316 | 11.51639 | 2.17 | 0.030 | 1.261599 | 89.11467 |
| ui | 4.06303 | 2.828513 | 2.01 | 0.044 | 1.038195 | 15.90088 |
| race |  |  |  |  |  |  |
| Black | 1.770681 | 1.221141 | 0.83 | 0.407 | .4582617 | 6.84175 |
| Other | .975003 | .6817263 | -0.04 | 0.971 | .2476522 | 3.838573 |

Smoking, previous preterm delivery, hypertension, uterine irritability, and possibly the mother's weight all contribute to low birthweight. Race of black and race of other are statistically insignificant when compared with the race of white omitted group, although the race of black effect is large. We can test the joint statistical significance of race being black (2.race) and race being other (3.race) by using test:

```
. test 2.race 3.race
( 1) [low]2.race = 0
( 2) [low]3.race = 0
    chi2( 2) = 0.88
    Prob > chi2 = 0.6436
```

For a more complete description of test, see [R] test. test presents results in coefficients rather than odds ratios. Jointly testing that the coefficients on 2 . race and 3 . race are 0 is equivalent to jointly testing that the odds ratios are 1 .

Here one case was matched to one control, that is, $1: 1$ matching. From clogit's point of view, that was not important - $k_{1}$ cases could have been matched to $k_{2}$ controls ( $k_{1}: k_{2}$ matching), and we would have fit the model in the same way. Furthermore, the matching can change from group to group, which we have denoted as $k_{1 i}: k_{2 i}$ matching, where $i$ denotes the group. clogit does not care. To fit the conditional logistic regression model, we specified the group (varname) option, group(pairid). The case and control are stored in separate observations. clogit knew that they were linked (in the same group) because the related observations share the same value of pairid.

## - Technical note

clogit provides a way to extend McNemar's test to multiple controls per case ( $1: k_{2 i}$ matching) and to multiple controls matched with multiple cases ( $k_{1 i}: k_{2 i}$ matching).

In Stata, McNemar's test is calculated by the mcc command; see [R] Epitab. The mcc command, however, requires that the matched case and control appear in one observation, so the data will need to be manipulated from 1 to 2 observations per stratum before using clogit. Alternatively, if you begin with clogit's 2-observations-per-group organization, you will have to change it to 1 observation per group if you wish to use mcc. In either case, reshape provides an easy way to change the organization of the data. We will demonstrate its use below, but we direct you to [D] reshape for a more thorough discussion.

In example 2, we used clogit to analyze the relationship between low birthweight and various characteristics of the mother. Assume that we now want to assess the relationship between low birthweight and smoking, ignoring the mother's other characteristics. Using clogit, we obtain the following results:

```
. clogit low smoke, group(pairid) or
Iteration 0: Log likelihood = -35.425931
Iteration 1: Log likelihood = -35.419283
Iteration 2: Log likelihood = -35.419282
Conditional (fixed-effects) logistic regression Number of obs = 112
    LR chi2(1) = 6.79
Prob > chi2 = 0.0091
Pseudo R2 = 0.0875
```

| low | Odds ratio | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :--- | :---: | :---: | :---: | :---: |
| smoke | 2.75 | 1.135369 | 2.45 | 0.014 | 1.224347 | 6.176763 |

Let's compare our estimated odds ratio and $95 \%$ confidence interval with that produced by mcc. We begin by reshaping the data:

```
. keep low smoke pairid
. reshape wide smoke, i(pairid) j(low 0 1)
\begin{tabular}{lrll} 
Data & Long & \(\rightarrow\) & Wide \\
\hline Number of observations & 112 & \(->\) & 56 \\
Number of variables & 3 & \(\rightarrow\) & 3 \\
j variable (2 values) & low & \(\rightarrow\) & (dropped) \\
xij variables: & smoke & \(\rightarrow\) & smoke0 smoke1 \\
\hline
\end{tabular}
```

We now have the variables smoke0 (formed from smoke and low $=0$ ), recording 1 if the control mother smoked and 0 otherwise; and smoke1 (formed from smoke and low $=1$ ), recording 1 if the case mother smoked and 0 otherwise. We can now use mcc:
. mcc smoke1 smoke0


Both methods estimated the same odds ratio, and the $95 \%$ confidence intervals are similar. clogit produced a confidence interval of $[1.22,6.18]$, whereas mcc produced a confidence interval of [1.18, 7.14 ].

## Use of weights

With clogit, weights apply to groups as a whole, not to individual observations. For example, if there is a group in your dataset with a frequency weight of 3, there are a total of three groups in your sample with the same values of the dependent and independent variables as this one group. Weights must have the same value for all observations belonging to the same group; otherwise, an error message will be displayed.

## > Example 3

We use the example from the above discussion of the mcc command. Here we have a total of 56 matched case-control groups, each with one case matched to one control. We had 8 matched pairs in which both the case and the control are exposed, 22 pairs in which the case is exposed and the control is unexposed, 8 pairs in which the case is unexposed and the control is exposed, and 18 pairs in which they are both unexposed.

With weights, it is easy to enter these data into Stata and run clogit.

```
. clear
. input id case exposed weight
                    id case exposed weight
    1. 11 1 8
    2. 1 0 1 8
    3. 2 1 1 22
    4. 2 0 0 22
    5. 3 1 0 8
    6. 3 0 1 8
    7. 4 1 0 18
    8. 4 0 0 18
    9. end
. clogit case exposed [w=weight], group(id) or
(frequency weights assumed)
Iteration 0: Log likelihood = -35.425931
Iteration 1: Log likelihood = -35.419283
Iteration 2: Log likelihood = -35.419282
Conditional (fixed-effects) logistic regression
Log likelihood = -35.419282
Number of obs = 112
LR chi2(1) = 6.79
Prob > chi2 = 0.0091
Pseudo R2 = 0.0875
```

| case | Odds ratio | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| exposed | 2.75 | 1.135369 | 2.45 | 0.014 | 1.224347 | 6.176763 |

## Fixed-effects logit

The fixed-effects logit model can be written as

$$
\operatorname{Pr}\left(y_{i t}=1 \mid \mathbf{x}_{i t}\right)=F\left(\alpha_{i}+\mathbf{x}_{i t} \boldsymbol{\beta}\right)
$$

where $F$ is the cumulative logistic distribution

$$
F(z)=\frac{\exp (z)}{1+\exp (z)}
$$

$i=1,2, \ldots, n$ denotes the independent units (called "groups" by clogit), and $t=1,2, \ldots, T_{i}$ denotes the observations for the $i$ th unit (group).

Fitting this model by using a full maximum-likelihood approach leads to difficulties, however. When $T_{i}$ is fixed, the maximum likelihood estimates for $\alpha_{i}$ and $\boldsymbol{\beta}$ are inconsistent (Andersen 1970; Chamberlain 1980). This difficulty can be circumvented by looking at the probability of $\mathbf{y}_{i}=$ $\left(y_{i 1}, \ldots, y_{i T_{i}}\right)$ conditional on $\sum_{t=1}^{T_{i}} y_{i t}$. This conditional probability does not involve the $\alpha_{i}$, so they are never estimated when the resulting conditional likelihood is used. See Hamerle and Ronning (1995) for a succinct and lucid development. See Methods and formulas for the estimation equation.

## > Example 4

We are studying unionization of women in the United States by using the union dataset; see [XT] xt. We fit the fixed-effects logit model:

```
. use https://www.stata-press.com/data/r18/union, clear
(NLS Women 14-24 in 1968)
. clogit union age grade not_smsa south black, group(idcode)
note: multiple positive outcomes within groups encountered.
note: 2,744 groups (14,165 obs) omitted because of all positive or
    all negative outcomes.
note: black omitted because of no within-group variance.
Iteration 0: Log likelihood = -4521.3385
Iteration 1: Log likelihood = -4516.1404
Iteration 2: Log likelihood = -4516.1385
Iteration 3: Log likelihood = -4516.1385
Conditional (fixed-effects) logistic regression Number of obs = 12,035
                                    LR chi2(4) = 68.09
                                    Prob > chi2 = 0.0000
Log likelihood = -4516.1385
Pseudo R2 = 0.0075
```

| union | Coefficient | Std. err. | $\mathbf{z}$ | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | ---: | ---: |
| age | .0170301 | .004146 | 4.11 | 0.000 | .0089042 | .0251561 |
| grade | .0853572 | .0418781 | 2.04 | 0.042 | .0032777 | .1674368 |
| not_smsa | .0083678 | .1127963 | 0.07 | 0.941 | -.2127088 | .2294445 |
| south | -.748023 | .1251752 | -5.98 | 0.000 | -.9933619 | -.5026842 |
| black | 0 | (omitted) |  |  |  |  |

We received three messages at the top of the output. The first one, "multiple positive outcomes within groups encountered", we expected. Our data do indeed have multiple positive outcomes (union $=1$ ) in many groups. (Here a group consists of all the observations for a particular individual.)

The second message tells us that 2,744 groups were "omitted" by clogit. When either union $=0$ or union $=1$ for all observations for an individual, this individual's contribution to the log likelihood is zero. Although these are perfectly valid observations in every sense, they have no effect on the estimation, so they are not included in the total "Number of obs". Hence, the reported "Number of obs" gives the effective sample size of the estimation. Here it is 12,035 observations - only $46 \%$ of the total 26,200 .

We can easily check that there are indeed 2,744 groups with union either all 0 or all 1 . We will generate a variable that contains the fraction of observations for each individual who has union $=1$.

```
. by idcode, sort: generate fraction = sum(union)/sum(union < .)
. by idcode: replace fraction = . if _n < _N
(21,766 real changes made, 21,766 to missing)
. tabulate fraction
\begin{tabular}{r|rrr} 
fraction & Freq. & Percent & Cum. \\
\hline 0 & 2,481 & 55.95 & 55.95 \\
.0833333 & 30 & 0.68 & 56.63 \\
.0909091 & 33 & 0.74 & 57.37 \\
.1 & 53 & 1.20 & 58.57 \\
(output omitted ) & & & \\
.9 & 10 & 0.23 & 93.59 \\
.9090909 & 11 & 0.25 & 93.84 \\
.9166667 & 10 & 0.23 & 94.07 \\
1 & 263 & 5.93 & 100.00 \\
\hline Total & 4,434 & 100.00 &
\end{tabular}
```

Because $2481+263=2744$, we confirm what clogit did.
The third warning message from clogit said "black omitted because of no within-group variance". Obviously, race stays constant for an individual across time. Any such variables are collinear with the $\alpha_{i}$ (that is, the fixed effects), and just as the $\alpha_{i}$ drop out of the conditional likelihood, so do all variables that are unchanging within groups. Thus, they cannot be estimated with the conditional fixed-effects model.

There are other estimators implemented in Stata that we could use with these data, such as

```
cloglog ... , vce(cluster idcode)
logit ... , vce(cluster idcode)
probit ... , vce(cluster idcode)
scobit ... , vce(cluster idcode)
xtcloglog ...
xtgee ... , family(binomial) link(logit) corr(exchangeable)
xtlogit ...
xtprobit ...
```

See [R] cloglog, [R] logit, [R] probit, [R] scobit, [XT] xtcloglog, [XT] xtgee, [XT] xtlogit, and [XT] xtprobit for details.

## Stored results

clogit stores the following in e():
Scalars
e(N)
e(N_drop)
e(N_group_drop)
e(k)
$e\left(k \_e q\right)$
e(k_eq_model)
e(k_dv)
e(df_m)
e(r2_p)
e(11)
e(ll_0)
e(N_clust)
e(chi2)
e (p)
e(rank)
e(ic)
e(rc)
e(converged)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(group)
$e$ (multiple)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(offset)
e(chi2type)
e(vce)
e(vcetype)
e(opt)
e(which)
e(ml_method)
e(user)
e(technique)
e(properties)
e(predict)
e(marginsok)
e(marginsnotok)
$e$ (marginsdefault)
e(asbalanced)
e(asobserved)
Matrices

```
e(b)
e(Cns)
e(ilog)
e(gradient)
e(V)
e(V_modelbased)
```

Functions
e(sample)
number of observations
number of observations dropped because of all positive or all negative outcomes
number of groups dropped because of all positive or all negative outcomes
number of parameters
number of equations in $e(b)$
number of equations in overall model test
number of dependent variables
model degrees of freedom
pseudo- $R^{2}$
log likelihood
log likelihood, constant-only model
number of clusters
$\chi^{2}$
$p$-value for model test
rank of e(V)
number of iterations
return code
1 if converged, 0 otherwise
clogit
command as typed
name of dependent variable
name of group() variable
multiple if multiple positive outcomes within group
weight type
weight expression
title in estimation output
name of cluster variable
linear offset variable
Wald or LR; type of model $\chi^{2}$ test
vcetype specified in vce()
title used to label Std. err.
type of optimization
max or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement predict
predictions allowed by margins
predictions disallowed by margins
default predict() specification for margins
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
constraints matrix
iteration $\log$ (up to 20 iterations)
gradient vector
variance-covariance matrix of the estimators
model-based variance
marks estimation sample

In addition to the above, the following is stored in r() :

Matrices
r(table)
matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

Breslow and Day (1980, 247-279), Collett (2003, 251-267), and Hosmer, Lemeshow, and Sturdivant (2013, 243-268) provide a biostatistical point of view on conditional logistic regression. Hamerle and Ronning (1995) give a succinct and lucid review of fixed-effects logit; Chamberlain (1980) is a standard reference for this model. Greene (2018, chap. 18) provides a straightforward textbook description of conditional logistic regression from an economist's point of view, as well as a brief description of choice models.

Let $i=1,2, \ldots, n$ denote the groups and let $t=1,2, \ldots, T_{i}$ denote the observations for the $i$ th group. Let $y_{i t}$ be the dependent variable taking on values 0 or 1 . Let $\mathbf{y}_{i}=\left(y_{i 1}, \ldots, y_{i T_{i}}\right)$ be the outcomes for the $i$ th group as a whole. Let $\mathbf{x}_{i t}$ be a row vector of covariates. Let

$$
k_{1 i}=\sum_{t=1}^{T_{i}} y_{i t}
$$

be the observed number of ones for the dependent variable in the $i$ th group. Biostatisticians would say that there are $k_{1 i}$ cases matched to $k_{2 i}=T_{i}-k_{1 i}$ controls in the $i$ th group.

We consider the probability of a possible value of $\mathbf{y}_{i}$ conditional on $\sum_{t=1}^{T_{i}} y_{i t}=k_{1 i}$ (Hamerle and Ronning 1995, eq. 8.33; Hosmer, Lemeshow, and Sturdivant 2013, eq. 7.4),

$$
\operatorname{Pr}\left(\mathbf{y}_{i} \mid \sum_{t=1}^{T_{i}} y_{i t}=k_{1 i}\right)=\frac{\exp \left(\sum_{t=1}^{T_{i}} y_{i t} \mathbf{x}_{i t} \boldsymbol{\beta}\right)}{\sum_{\mathbf{d}_{i} \in S_{i}} \exp \left(\sum_{t=1}^{T_{i}} d_{i t} \mathbf{x}_{i t} \boldsymbol{\beta}\right)}
$$

where $d_{i t}$ is equal to 0 or 1 with $\sum_{t=1}^{T_{i}} d_{i t}=k_{1 i}$, and $S_{i}$ is the set of all possible combinations of $k_{1 i}$ ones and $k_{2 i}$ zeros. Clearly, there are $\binom{T_{i}}{k_{1 i}}$ such combinations, but we need not count all of these combinations to compute the denominator of the above equation. It can be computed recursively.

Denote the denominator by

$$
f_{i}\left(T_{i}, k_{1 i}\right)=\sum_{\mathbf{d}_{i} \in S_{i}} \exp \left(\sum_{t=1}^{T_{i}} d_{i t} \mathbf{x}_{i t} \boldsymbol{\beta}\right)
$$

Consider, computationally, how $f_{i}$ changes as we go from a total of 1 observation in the group to 2 observations to 3 , etc. Doing this, we derive the recursive formula

$$
f_{i}(T, k)=f_{i}(T-1, k)+f_{i}(T-1, k-1) \exp \left(\mathbf{x}_{i T} \boldsymbol{\beta}\right)
$$

where we define $f_{i}(T, k)=0$ if $T<k$ and $f_{i}(T, 0)=1$.

The conditional log-likelihood is

$$
\ln L=\sum_{i=1}^{n}\left\{\sum_{t=1}^{T_{i}} y_{i t} \mathbf{x}_{i t} \boldsymbol{\beta}-\log f_{i}\left(T_{i}, k_{1 i}\right)\right\}
$$

The derivatives of the conditional log-likelihood can also be computed recursively by taking derivatives of the recursive formula for $f_{i}$.

Computation time is roughly proportional to

$$
p^{2} \sum_{i=1}^{n} T_{i} \min \left(k_{1 i}, k_{2 i}\right)
$$

where $p$ is the number of independent variables in the model. If $\min \left(k_{1 i}, k_{2 i}\right)$ is small, computation time is not an issue. But if it is large-say, 100 or more-patience may be required.

If $T_{i}$ is large for all groups, the bias of the unconditional fixed-effects estimator is not a concern, and we can confidently use logit with an indicator variable for each group (provided, of course, that the number of groups is held within a Stata matrix; see [R] Limits).

This command supports the clustered version of the Huber/White/sandwich estimator of the variance using vce (robust) and vce (cluster clustvar). See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas. Specifying vce (robust) is equivalent to specifying vce(cluster groupvar), where groupvar is the variable for the matched groups.
clogit also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

## References

Andersen, E. B. 1970. Asymptotic properties of conditional maximum-likelihood estimators. Journal of the Royal Statistical Society, Series B 32: 283-301. https://doi.org/10.1111/j.2517-6161.1970.tb00842.x.
Breslow, N. E., and N. E. Day. 1980. Statistical Methods in Cancer Research: Vol. 1-The Analysis of Case-Control Studies. Lyon: IARC.

Chamberlain, G. 1980. Analysis of covariance with qualitative data. Review of Economic Studies 47: 225-238. https://doi.org/10.2307/2297110.
Collett, D. 2003. Modelling Binary Data. 2nd ed. London: Chapman and Hall/CRC.
Greene, W. H. 2018. Econometric Analysis. 8th ed. New York: Pearson.
Hamerle, A., and G. Ronning. 1995. Panel analysis for qualitative variables. In Handbook of Statistical Modeling for the Social and Behavioral Sciences, ed. G. Arminger, C. C. Clogg, and M. E. Sobel, 401-451. New York: Plenum. https://doi.org/10.1007/978-1-4899-1292-3_8.

Hole, A. R. 2007. Fitting mixed logit models by using maximum simulated likelihood. Stata Journal 7: 388-401.
Hosmer, D. W., Jr., S. A. Lemeshow, and R. X. Sturdivant. 2013. Applied Logistic Regression. 3rd ed. Hoboken, NJ: Wiley.
Long, J. S., and J. Freese. 2014. Regression Models for Categorical Dependent Variables Using Stata. 3rd ed. College Station, TX: Stata Press.

McFadden, D. L. 1974. Conditional logit analysis of qualitative choice behavior. In Frontiers in Econometrics, ed. P. Zarembka, 105-142. New York: Academic Press.

## Also see

[R] clogit postestimation - Postestimation tools for clogit
[R] logistic - Logistic regression, reporting odds ratios
[R] mlogit - Multinomial (polytomous) logistic regression
[R] ologit - Ordered logistic regression
[R] scobit - Skewed logistic regression
[BAYES] bayes: clogit - Bayesian conditional logistic regression
[CM] emclogit - Conditional logit (McFadden's) choice model
[CM] nlogit - Nested logit regression
[MI] Estimation - Estimation commands for use with mi estimate
[SVY] svy estimation - Estimation commands for survey data
[XT] xtgee - GEE population-averaged panel-data models
[XT] xtlogit - Fixed-effects, random-effects, and population-averaged logit models
[XT] xtmlogit - Fixed-effects and random-effects multinomial logit models
[U] 20 Estimation and postestimation commands

## Title

clogit postestimation - Postestimation tools for clogit

| Postestimation commands | predict | margins |
| :--- | :--- | :--- |
| Remarks and examples | Methods and formulas | Reference |

## Postestimation commands

The following standard postestimation commands are available after clogit:

| Command | Description |
| :---: | :---: |
| contrast | contrasts and ANOVA-style joint tests of estimates |
| estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estat (svy) | postestimation statistics for survey data |
| estimates | cataloging estimation results |
| etable | table of estimation results |
| * hausman | Hausman's specification test |
| lincom | point estimates, standard errors, testing, and inference for linear combinations of coefficients |
| linktest | link test for model specification |
| * lrtest | likelihood-ratio test |
| margins | marginal means, predictive margins, marginal effects, and average marginal effects |
| marginsplot | graph the results from margins (profile plots, interaction plots, etc.) |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients |
| predict | probabilities, influence statistics, residuals, etc. |
| predictnl | point estimates, standard errors, testing, and inference for generalized predictions |
| pwcompare | pairwise comparisons of estimates |
| suest | seemingly unrelated estimation |
| test | Wald tests of simple and composite linear hypotheses |
| testnl | Wald tests of nonlinear hypotheses |

[^10]
## predict

## Description for predict

predict creates a new variable containing predictions such as probabilities, linear predictions, standard errors, influence statistics, lack-of-fit statistics, Hosmer and Lemeshow leverages, Pearson residuals, and the equation-level score.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

```
predict [type] newvar [if] [in] [, statistic nooffset]
```

statistic Description

Main
pc1 probability of a positive outcome; the default
$\mathrm{puO} \quad$ probability of a positive outcome, assuming fixed effect is zero
$\mathrm{xb} \quad$ linear prediction
stdp standard error of the linear prediction

* dbeta Delta- $\boldsymbol{\beta}$ influence statistic
* dx2 Delta- $\chi^{2}$ lack-of-fit statistic
*gdbeta Delta- $\boldsymbol{\beta}$ influence statistic for each group
* gdx 2

Delta- $\chi^{2}$ lack-of-fit statistic for each group
*hat Hosmer and Lemeshow leverage

* residuals Pearson residuals
* rstandard standardized Pearson residuals
score first derivative of the log likelihood with respect to $\mathbf{x}_{j} \boldsymbol{\beta}$
Unstarred statistics are available both in and out of sample; type predict ... if e (sample) ... if wanted only for the estimation sample. Starred statistics are calculated only for the estimation sample, even when if e(sample) is not specified.

Starred statistics are available for multiple controls per case-matching design only. They are not available if vce(robust), vce(cluster clustvar), or pweights were specified with clogit.
dbeta, dx2, gdbeta, gdx2, hat, and rstandard are not available if constraints() was specified with clogit.

## Options for predict

$\qquad$ Main
pc 1 , the default, calculates the probability of a positive outcome conditional on one positive outcome within group.
puO calculates the probability of a positive outcome, assuming that the fixed effect is zero. xb calculates the linear prediction.
stdp calculates the standard error of the linear prediction.
dbeta calculates the Delta- $\boldsymbol{\beta}$ influence statistic, a standardized measure of the difference in the coefficient vector that is due to deletion of the observation.
dx 2 calculates the Delta- $\chi^{2}$ influence statistic, reflecting the decrease in the Pearson $\chi^{2}$ that is due to deletion of the observation.
gdbeta calculates the approximation to the Pregibon stratum-specific Delta- $\boldsymbol{\beta}$ influence statistic, a standardized measure of the difference in the coefficient vector that is due to deletion of the entire stratum.
gdx2 calculates the approximation to the Pregibon stratum-specific Delta- $\chi^{2}$ influence statistic, reflecting the decrease in the Pearson $\chi^{2}$ that is due to deletion of the entire stratum.
hat calculates the Hosmer and Lemeshow leverage or the diagonal element of the hat matrix.
residuals calculates the Pearson residuals.
rstandard calculates the standardized Pearson residuals.
score calculates the equation-level score, $\partial \ln L / \partial\left(\mathbf{x}_{i t} \boldsymbol{\beta}\right)$.
nooffset is relevant only if you specified offset (varname) for clogit. It modifies the calculations made by predict so that they ignore the offset variable; the linear prediction is treated as $\mathbf{x}_{j} \mathbf{b}$ rather than as $\mathbf{x}_{j} \mathbf{b}+$ offset $_{j}$. This option cannot be specified with dbeta, dx2, gdbeta, gdx2, hat, and rstandard.

## margins

## Description for margins

margins estimates margins of response for probabilities and linear predictions.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...) ...] [options]
```

statistic Description
pu0 probability of a positive outcome, assuming fixed effect is zero; the default
$\mathrm{xb} \quad$ linear prediction
pc1 not allowed with margins
stdp not allowed with margins
dbeta not allowed with margins
dx2 not allowed with margins
gdbeta not allowed with margins
gdx2 not allowed with margins
hat not allowed with margins
residuals not allowed with margins
rstandard not allowed with margins
score not allowed with margins

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

## Remarks and examples

predict may be used after clogit to obtain predicted values of the index $\mathbf{x}_{i t} \boldsymbol{\beta}$. Predicted probabilities for conditional logistic regression must be interpreted carefully. Probabilities are estimated for each group as a whole, not for individual observations. Furthermore, the probabilities are conditional on the number of positive outcomes in the group (that is, the number of cases and the number of controls), or it is assumed that the fixed effect is zero. predict may also be used to obtain influence and lack-of-fit statistics for an individual observation and for the whole group, to compute Pearson, standardized Pearson residuals, and leverage values.
predict may be used for both within-sample and out-of-sample predictions.
> Example 1
Suppose that we have $1: k_{2 i}$ matched data and that we have previously fit the following model:

```
. use https://www.stata-press.com/data/r18/clogitid
. clogit y x1 x2, group(id)
    (output omitted)
```

To obtain the predicted values of the index, we could type predict idx, xb to create a new variable called idx. From idx, we could then calculate the predicted probabilities. Easier, however, would be to type

```
. predict phat
(option pc1 assumed; probability of success given one success within group)
```

phat would then contain the predicted probabilities.
As noted previously, the predicted probabilities are really predicted probabilities for the group as a whole (that is, they are the predicted probability of observing $y_{i t}=1$ and $y_{i t^{\prime}}=0$ for all $t^{\prime} \neq t$ ). Thus, if we want to obtain the predicted probabilities for the estimation sample, it is important that, when we make the calculation, predictions be restricted to the same sample on which we estimated the data. We cannot predict the probabilities and then just keep the relevant ones because the entire sample determines each probability. Thus, assuming that we are not attempting to make out-of-sample predictions, we type

```
. predict phat2 if e(sample)
(option pc1 assumed; probability of success given one success within group)
```


## Methods and formulas

Recall that $i=1, \ldots, n$ denote the groups and $t=1, \ldots, T_{i}$ denote the observations for the $i$ th group.
predict produces probabilities of a positive outcome within group conditional on there being one positive outcome (pc1),

$$
\operatorname{Pr}\left(y_{i t}=1 \mid \sum_{t=1}^{T_{i}} y_{i t}=1\right)=\frac{\exp \left(\mathbf{x}_{i t} \boldsymbol{\beta}\right)}{\sum_{t=1}^{T_{i}} \exp \left(\mathbf{x}_{i t} \boldsymbol{\beta}\right)}
$$

or predict calculates the unconditional pu0:

$$
\operatorname{Pr}\left(y_{i t}=1\right)=\frac{\exp \left(\mathbf{x}_{i t} \boldsymbol{\beta}\right)}{1+\exp \left(\mathbf{x}_{i t} \boldsymbol{\beta}\right)}
$$

Let $N=\sum_{j=1}^{n} T_{j}$ denote the total number of observations, $p$ denote the number of covariates, and $\widehat{\theta}_{i t}$ denote the conditional predicted probabilities of a positive outcome ( pc 1 ).

For the multiple control per case ( $1: k_{2 i}$ ) matching, Hosmer, Lemeshow, and Sturdivant (2013, 248-251) propose the following diagnostics:

The Pearson residual is

$$
r_{i t}=\frac{\left(y_{i t}-\widehat{\theta}_{i t}\right)}{\sqrt{\widehat{\theta}_{i t}}}
$$

The leverage (hat) value is defined as

$$
h_{i t}=\widehat{\theta}_{i t} \widetilde{\mathbf{x}}_{i t}^{T}\left(\widetilde{\mathbf{X}}^{T} \mathbf{U} \widetilde{\mathbf{X}}\right)^{-1} \widetilde{\mathbf{x}}_{i t}
$$

where $\widetilde{\mathbf{x}}_{i t}=\mathbf{x}_{i t}-\sum_{j=1}^{T_{i}} \mathbf{x}_{i j} \widehat{\theta}_{i j}$ is the $1 \times p$ row vector of centered by a weighted stratum-specific mean covariate values, $\mathbf{U}_{N}=\operatorname{diag}\left\{\widehat{\theta}_{i t}\right\}$, and the rows of $\widetilde{\mathbf{X}}_{N \times p}$ are composed of $\widetilde{\mathbf{x}}_{i t}$ values.

The standardized Pearson residual is

$$
r_{s i t}=\frac{r_{i t}}{\sqrt{1-h_{i t}}}
$$

The lack-of-fit and influence diagnostics for an individual observation are (respectively) computed as

$$
\Delta \chi_{i t}^{2}=r_{s i t}^{2}
$$

and

$$
\Delta \widehat{\beta}_{i t}=\Delta \chi_{i t}^{2} \frac{h_{i t}}{1-h_{i t}}
$$

The lack-of-fit and influence diagnostics for the groups are the group-specific totals of the respective individual diagnostics shown above.

## Reference

Hosmer, D. W., J.., S. A. Lemeshow, and R. X. Sturdivant. 2013. Applied Logistic Regression. 3rd ed. Hoboken, NJ: Wiley.

## Also see

$[\mathrm{R}]$ clogit - Conditional (fixed-effects) logistic regression
[U] 20 Estimation and postestimation commands

## Title

cloglog - Complementary log-log regression

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| Acknowledgment | References | Also see |  |

## Description

cloglog fits a complementary $\log -\log$ model for a binary dependent variable, typically with one of the outcomes rare relative to the other. It can also be used to fit a gompit model. cloglog can compute robust and cluster-robust standard errors and adjust results for complex survey designs.

## Quick start

Complementary $\log -\log$ model of y on x 1 and x 2
cloglog y x1 x2
With robust standard errors
cloglog y x1 x2, vce(robust)
Adjust for complex survey design using svyset data
svy: cloglog y x1 x2

## Menu

Statistics $>$ Binary outcomes $>$ Complementary log-log regression

## Syntax

cloglog depvar [indepvars] [if] [in] [weight] [, options]
options
Model
noconstant
offset (varname)
asis
constraints (constraints)
SE/Robust
vce (vcetype)

Reporting
level(\#)
eform
nocnsreport
display_options

Description
suppress constant term
include varname in model with coefficient constrained to 1
retain perfect predictor variables
apply specified linear constraints
vcetype may be oim, robust, cluster clustvar, opg, bootstrap, or jackknife
set confidence level; default is level (95)
report exponentiated coefficients
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
control the maximization process; seldom used
keep collinear variables
display legend instead of statistics
indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
depvar and indepvars may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bayes, bootstrap, by, collect, fmm, fp, jackknife, mi estimate, nestreg, rolling, statsby, stepwise, and svy are allowed; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: cloglog and [FMM] fmm: cloglog.
vce(bootstrap) and vce(jackknife) are not allowed with the mi estimate prefix; see [MI] mi estimate.
Weights are not allowed with the bootstrap prefix; see $[R]$ bootstrap.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

[^11]SE/Robust
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
$\qquad$ Reporting
level (\#); see [R] Estimation options.
eform displays the exponentiated coefficients and corresponding standard errors and confidence intervals.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap (\#), fvwrapon(style), cformat( $\%$ fmt), pformat ( $\%$ fint), sformat ( $\% f m t$ ), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate(\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance (\#), nonrtolerance, and from(init_specs); see [R] Maximize. These options are seldom used.

Setting the optimization type to technique (bhhh) resets the default vcetype to vce (opg).
The following options are available with cloglog but are not shown in the dialog box: collinear, coeflegend; see [R] Estimation options.

## Remarks and examples

Remarks are presented under the following headings:
Introduction to complementary log-log regression Robust standard errors

## Introduction to complementary log-log regression

cloglog fits maximum likelihood models with dichotomous dependent variables coded as $0 / 1$ (or, more precisely, coded as 0 and not 0 ).
> Example 1
We have data on the make, weight, and mileage rating of 22 foreign and 52 domestic automobiles. We wish to fit a model explaining whether a car is foreign based on its weight and mileage. Here is an overview of our data:

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. keep make mpg weight foreign
. describe
Contains data from https://www.stata-press.com/data/r18/auto.dta
    Observations: 74 1978 automobile data
        Variables: 4 13 Apr 2022 17:45
        (_dta has notes)
\begin{tabular}{lccll}
\hline \begin{tabular}{r} 
Variable \\
name
\end{tabular} & \begin{tabular}{r} 
Storage \\
type
\end{tabular} & \begin{tabular}{c} 
Display \\
format
\end{tabular} & \begin{tabular}{l} 
Value \\
label
\end{tabular} & Variable label \\
\hline make & str18 & \(\%-18 \mathrm{~s}\) & & Make and model \\
mpg & int & \(\% 8.0 \mathrm{~g}\) & & Mileage (mpg) \\
weight & int & \(\% 8.0 \mathrm{gc}\) & & \begin{tabular}{l} 
Weight (lbs.) \\
foreign
\end{tabular} \\
byte & \(\% 8.0 \mathrm{~g}\) & origin & Car origin \\
\hline
\end{tabular}
Sorted by: foreign
            Note: Dataset has changed since last saved.
. inspect foreign
foreign: Car origin
Number of observations
Mlll
\begin{tabular}{lrrr}
\cline { 2 - 4 } & Total & Integers & Nonintegers \\
Negative & - & - & - \\
Zero & 52 & 52 & - \\
Positive & 22 & 22 & - \\
\cline { 2 - 3 } \cline { 2 - 4 } & 74 & 74 & - \\
Total & - & &
\end{tabular}
(2 unique values)
        foreign is labeled and all values are documented in the label.
```

The variable foreign takes on two unique values, 0 and 1 . The value 0 denotes a domestic car, and 1 denotes a foreign car.

The model that we wish to fit is

$$
\operatorname{Pr}(\text { foreign }=1)=F\left(\beta_{0}+\beta_{1} \text { weight }+\beta_{2} \mathrm{mpg}\right)
$$

where $F(z)=1-\exp \{-\exp (z)\}$.

To fit this model, we type

```
. cloglog foreign weight mpg
Iteration 0: Log likelihood = -34.054593
Iteration 1: Log likelihood = -27.869915
Iteration 2: Log likelihood = -27.742997
Iteration 3: Log likelihood = -27.742769
Iteration 4: Log likelihood = -27.742769
```

Complementary log-log regression $\quad$ Number of obs 74
Zero outcomes $=52$
Nonzero outcomes $=22$
LR chi2(2) $=34.58$
Log likelihood $=-27.742769$
Prob > chi2 $=0.0000$

| foreign | Coefficient | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | ---: | :---: | :---: | ---: | ---: |
| weight | -.0029153 | .0006974 | -4.18 | 0.000 | -.0042823 | -.0015483 |
| mpg | -.1422911 | .076387 | -1.86 | 0.062 | -.2920069 | .0074247 |
| _cons | 10.09694 | 3.351841 | 3.01 | 0.003 | 3.527448 | 16.66642 |

We find that heavier cars are less likely to be foreign and that cars yielding better gas mileage are also less likely to be foreign, at least when holding the weight of the car constant.

See [R] Maximize for an explanation of the output.

## Technical note

Stata interprets a value of 0 as a negative outcome (failure) and treats all other values (except missing) as positive outcomes (successes). Thus, if your dependent variable takes on the values 0 and 1,0 is interpreted as failure and 1 as success. If your dependent variable takes on the values 0,1 , and 2,0 is still interpreted as failure, but both 1 and 2 are treated as successes.

If you prefer a more formal mathematical statement, when you type cloglog $y x$, Stata fits the model

$$
\operatorname{Pr}\left(y_{j} \neq 0 \mid \mathbf{x}_{j}\right)=1-\exp \left\{-\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}\right)\right\}
$$

## Robust standard errors

If you specify the vce (robust) option, cloglog reports robust standard errors, as described in [U] 20.22 Obtaining robust variance estimates. For the model of foreign on weight and mpg, the robust calculation increases the standard error of the coefficient on mpg by $44 \%$ :


Without vce (robust), the standard error for the coefficient on mpg was reported to be 0.076 , with a resulting confidence interval of $[-0.29,0.01]$.

The vce(cluster clustvar) option can relax the independence assumption required by the complementary log-log estimator to being just independence between clusters. To demonstrate this ability, we will switch to a different dataset.

We are studying unionization of women in the United States by using the union dataset; see [XT] xt. We fit the following model, ignoring that women are observed an average of 5.9 times each in this dataset:


The reported standard errors in this model are probably meaningless. Women are observed repeatedly, and so the observations are not independent. Looking at the coefficients, we find a large southern
effect against unionization and a different time trend for the south. The vce(cluster clustvar) option provides a way to fit this model and obtains correct standard errors:


These standard errors are larger than those reported by the inappropriate conventional calculation. By comparison, another way we could fit this model is with an equal-correlation population-averaged complementary log-log model:

| GEE population-averaged model | Number of obs | = 26,200 |
| :---: | :---: | :---: |
| Group variable: idcode | Number of groups | $=4,434$ |
| Family: Binomial | Obs per group: |  |
| Link: Complementary log-log | min | $=1$ |
| Correlation: exchangeable | avg | $=5.9$ |
|  | max | 12 |
|  | Wald chi2(6) | $=234.66$ |
| Scale parameter = 1 | Prob > chi2 | $=0.0000$ |


| union | Coefficient | Std. err. | $z$ | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| age | .0153737 | .0081156 | 1.89 | 0.058 | -.0005326 | .03128 |
| grade | .0549518 | .0095093 | 5.78 | 0.000 | .0363139 | .0735897 |
| not_smsa | -.1045232 | .0431082 | -2.42 | 0.015 | -.1890138 | -.0200326 |
| 1.south | -1.714868 | .3384558 | -5.07 | 0.000 | -2.378229 | -1.051507 |
| year | -.0115881 | .0084125 | -1.38 | 0.168 | -.0280763 | .0049001 |
| south\#c.year |  |  |  |  |  |  |
| 1 | .0149796 | .0041687 | 3.59 | 0.000 | .0068091 | .0231501 |
| _cons | -1.488278 | .4468005 | -3.33 | 0.001 | -2.363991 | -.6125652 |

The coefficient estimates are similar, but these standard errors are smaller than those produced by cloglog, vce (cluster clustvar). This finding is as we would expect. If the within-panel correlation assumptions are valid, the population-averaged estimator should be more efficient.

In addition to this estimator, we may use the xtgee command to fit a panel estimator (with complementary $\log -\log$ link) and any number of assumptions on the within-idcode correlation.
cloglog, vce(cluster clustvar) is robust to assumptions about within-cluster correlation. That is, it inefficiently sums within cluster for the standard error calculation rather than attempting to exploit what might be assumed about the within-cluster correlation (as do the xtgee population-averaged models).

## Stored results

cloglog stores the following in e():
Scalars

```
e(N)
    e(k)
    e(k_eq)
    e(k_eq_model)
    e(k_dv)
    e(N_f)
    e(N_s)
    e(df_m)
    e(ll)
    e(ll_0)
    e(N_clust)
    e(chi2)
    e(p)
    e(rank)
    e(ic)
    e(rc)
    e(converged)
```

Macros
e(cmd)
e(cmdline)
e(depvar)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(offset)
e(chi2type)
e(vce)
e(vcetype)
e(opt)
e(which)
e(ml_method)
e(user)
$e$ (technique)
e(properties)
e(predict)
e(marginsok)
e(marginsnotok)
e(asbalanced)
e(asobserved)
number of observations
number of parameters
number of equations in $e(b)$
number of equations in overall model test
number of dependent variables
number of zero outcomes
number of nonzero outcomes
model degrees of freedom
log likelihood
log likelihood, constant-only model
number of clusters
$\chi^{2}$
$p$-value for model test
rank of e(V)
number of iterations
return code
1 if converged, 0 otherwise
cloglog
command as typed
name of dependent variable
weight type
weight expression
title in estimation output
name of cluster variable
linear offset variable
Wald or LR; type of model $\chi^{2}$ test
vcetype specified in vce()
title used to label Std. err.
type of optimization
max or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement predict
predictions allowed by margins
predictions disallowed by margins
factor variables fvset as asbalanced
factor variables fvset as asobserved

```
Matrices
    e(b) coefficient vector
    e(Cns) constraints matrix
    e(ilog) iteration log (up to 20 iterations)
    e(gradient) gradient vector
    e(V) variance-covariance matrix of the estimators
    e(V_modelbased) model-based variance
Functions
    e(sample) marks estimation sample
```

In addition to the above, the following is stored in r() :

## Matrices

$r$ (table) matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

Complementary log-log analysis (related to the gompit model, so named because of its relationship to the Gompertz distribution) is an alternative to logit and probit analysis, but it is unlike these other estimators in that the transformation is not symmetric. Typically, this model is used when the positive (or negative) outcome is rare.

The log-likelihood function for complementary $\log -\log$ is

$$
\ln L=\sum_{j \in S} w_{j} \ln F\left(\mathbf{x}_{j} \mathbf{b}\right)+\sum_{j \notin S} w_{j} \ln \left\{1-F\left(\mathbf{x}_{j} \mathbf{b}\right)\right\}
$$

where $S$ is the set of all observations $j$ such that $y_{j} \neq 0, F(z)=1-\exp \{-\exp (z)\}$, and $w_{j}$ denotes the optional weights. $\ln L$ is maximized as described in [ R$]$ Maximize.

We can fit a gompit model by reversing the success-failure sense of the dependent variable and using cloglog.

This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce (robust) and vce(cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas. The scores are calculated as $\mathbf{u}_{j}=$ $\left[\exp \left(\mathbf{x}_{j} \mathbf{b}\right) \exp \left\{-\exp \left(\mathbf{x}_{j} \mathbf{b}\right)\right\} / F\left(\mathbf{x}_{j} \mathbf{b}\right)\right] \mathbf{x}_{j}$ for the positive outcomes and $\left\{-\exp \left(\mathbf{x}_{j} \mathbf{b}\right)\right\} \mathbf{x}_{j}$ for the negative outcomes.
cloglog also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

## Acknowledgment

We thank Joseph Hilbe (1944-2017), coauthor of the Stata Press book Generalized Linear Models and Extensions, for providing the inspiration for the cloglog command.

## References

Long, J. S. 1997. Regression Models for Categorical and Limited Dependent Variables. Thousand Oaks, CA: Sage.
Long, J. S., and J. Freese. 2014. Regression Models for Categorical Dependent Variables Using Stata. 3rd ed. College Station, TX: Stata Press.
Xu, J., and J. S. Long. 2005. Confidence intervals for predicted outcomes in regression models for categorical outcomes. Stata Journal 5: 537-559.

## Also see

[R] cloglog postestimation - Postestimation tools for cloglog
[R] clogit - Conditional (fixed-effects) logistic regression
[R] glm - Generalized linear models
[R] logistic - Logistic regression, reporting odds ratios
[R] scobit - Skewed logistic regression
[BAYES] bayes: cloglog - Bayesian complementary log-log regression
[FMM] fmm: cloglog - Finite mixtures of complementary log-log regression models
[ME] mecloglog - Multilevel mixed-effects complementary log-log regression
[MI] Estimation - Estimation commands for use with mi estimate
[SVY] svy estimation - Estimation commands for survey data
[XT] xtcloglog — Random-effects and population-averaged cloglog models
[U] 20 Estimation and postestimation commands

## Title

cloglog postestimation - Postestimation tools for cloglog
Postestimation commands predict margins Remarks and examples
Also see

## Postestimation commands

The following postestimation commands are available after cloglog:

| Command | Description |
| :---: | :---: |
| contrast | contrasts and ANOVA-style joint tests of estimates |
| estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estat (svy) | postestimation statistics for survey data |
| estimates | cataloging estimation results |
| etable | table of estimation results |
| * forecast | dynamic forecasts and simulations |
| * hausman | Hausman's specification test |
| lincom | point estimates, standard errors, testing, and inference for linear combinations of coefficients |
| linktest | link test for model specification |
| * Irtest | likelihood-ratio test |
| margins | marginal means, predictive margins, marginal effects, and average marginal effects |
| marginsplot | graph the results from margins (profile plots, interaction plots, etc.) |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients |
| predict | probabilities, linear predictions and their SEs, etc. |
| predictnl | point estimates, standard errors, testing, and inference for generalized predictions |
| pwcompare | pairwise comparisons of estimates |
| suest | seemingly unrelated estimation |
| test | Wald tests of simple and composite linear hypotheses |
| testnl | Wald tests of nonlinear hypotheses |

[^12]
## predict

## Description for predict

predict creates a new variable containing predictions such as probabilities, linear predictions, standard errors, and the equation-level score.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

predict [type] newvar [if][in] [, statistic nooffset]
statistic
Description
Main
$\mathrm{pr} \quad$ probability of a positive outcome; the default
$\mathrm{xb} \quad$ linear prediction
stdp standard error of the linear prediction
score first derivative of the $\log$ likelihood with respect to $\mathbf{x}_{j} \boldsymbol{\beta}$
These statistics are available both in and out of sample; type predict ... if e(sample)... if wanted only for the estimation sample.

## Options for predict

Main
pr , the default, calculates the probability of a positive outcome.
xb calculates the linear prediction.
stdp calculates the standard error of the linear prediction.
score calculates the equation-level score, $\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)$.
nooffset is relevant only if you specified offset (varname) for cloglog. It modifies the calculations made by predict so that they ignore the offset variable; the linear prediction is treated as $\mathbf{x}_{j} \mathbf{b}$ rather than as $\mathbf{x}_{j} \mathbf{b}+$ offset $_{j}$.

## margins

## Description for margins

margins estimates margins of response for probabilities and linear predictions.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

margins [marginlist] [, options]

statistic Description
$\mathrm{pr} \quad$ probability of a positive outcome; the default
xb linear prediction
stdp not allowed with margins
score not allowed with margins

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see $[R]$ margins.

## Remarks and examples

Once you have fit a model, you can obtain the predicted probabilities by using the predict command for both the estimation sample and other samples; see [U] 20 Estimation and postestimation commands and $[\mathrm{R}]$ predict. Here we will make only a few comments.
predict without arguments calculates the predicted probability of a positive outcome. With the xb option, it calculates the linear combination $\mathbf{x}_{j} \mathbf{b}$, where $\mathbf{x}_{j}$ are the independent variables in the $j$ th observation and $\mathbf{b}$ is the estimated parameter vector.

With the stdp option, predict calculates the standard error of the linear prediction, which is not adjusted for replicated covariate patterns in the data.
> Example 1
In example 1 in [ R$]$ cloglog, we fit the complementary $\log -\log$ model cloglog foreign weight mpg . To obtain predicted probabilities,

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. cloglog foreign weight mpg
    (output omitted)
- predict p
(option pr assumed; Pr(foreign))
. summarize foreign p
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline foreign & 74 & .2972973 & .4601885 & 0 & 1 \\
\(p\) & 74 & .2928348 & .29732 & .0032726 & .9446067
\end{tabular}
```


## Also see

[R] cloglog - Complementary log-log regression
[U] 20 Estimation and postestimation commands

## Title

cls - Clear Results window

Description Syntax

## Description

cls clears the Results window, causing all text to be removed. This operation cannot be undone.

## Syntax

cls

## Title

cnsreg - Constrained linear regression

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

cnsreg fits constrained linear regression models.

## Quick start

Linear regression with coefficients for x 1 and x 2 constrained to equality
constraint $1 \mathrm{x} 1=\mathrm{x} 2$
cnsreg y x1 x2 x3, constraints(1)
Add constraint $x 2=x 3$ to impose $x 1=x 2=x 3$
constraint $2 \times 2=x 3$
cnsreg y x1 x2 x3, constraints(1 2)
Constrain the coefficient for x 4 to be -1
constraint $3 \times 4=-1$
cnsreg y x1 x2 x3 x4, constraints(1-3)

## Menu

Statistics $>$ Linear models and related $>$ Constrained linear regression

## Syntax

cnsreg depvar indepvars [if][in] [weight], constraints(constraints) [options] options Description

## Model

* constraints (constraints)
noconstant
SE/Robust
vce (vcetype)

Reporting
level(\#)
nocnsreport
display_options
mse1
collinear
coeflegend
apply specified linear constraints
suppress constant term
vcetype may be ols, robust, cluster clustvar, bootstrap, or jackknife
set confidence level; default is level (95)
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
force MSE to be 1
keep collinear variables
display legend instead of statistics
*constraints (constraints) is required.
indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
depvar and indepvars may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bootstrap, by, collect, fp, jackknife, mi estimate, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands.
vce(bootstrap) and vce(jackknife) are not allowed with the mi estimate prefix; see [MI] mi estimate.
With the fp prefix (see $[\mathrm{R}] \mathrm{fp}$ ), constraints cannot be specified for the variable containing fractional polynomial terms. Weights are not allowed with the bootstrap prefix; see $[\mathrm{R}]$ bootstrap.
aweights are not allowed with the jackknife prefix; see $[R]$ jackknife.
vce(), mse1, and weights are not allowed with the svy prefix; see [SVY] svy.
aweights, fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
mse1, collinear, and coeflegend do not appear in the dialog.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

Model
constraints(constraints), noconstant; see [R] Estimation options.

## SE/Robust

vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (ols), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
vce(ols), the default, uses the standard variance estimator for ordinary least-squares regression.
level(\#), nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.

The following options are available with ensreg but are not shown in the dialog box:
mse1 is used only in programs and ado-files that use cnsreg to fit models other than constrained linear regression. mse1 sets the mean squared error to 1 , thus forcing the variance-covariance matrix of the estimators to be $\left(\mathbf{X}^{\prime} \mathbf{D X}\right)^{-1}$ (see Methods and formulas in $[\mathrm{R}]$ regress) and affecting calculated standard errors. Degrees of freedom for $t$ statistics are calculated as $n$ rather than $n-p+c$, where $p$ is the total number of parameters (prior to restrictions and including the constant) and $c$ is the number of constraints.
mse1 is not allowed with the svy prefix.
collinear, coeflegend; see [R] Estimation options.

## Remarks and examples

For a discussion of constrained linear regression, see Greene (2018, 126-127); Hill, Griffiths, and Lim (2018, 271-273); or Davidson and MacKinnon (1993, 17).

## > Example 1: One constraint

In principle, we can obtain constrained linear regression estimates by modifying the list of independent variables. For instance, if we wanted to fit the model

$$
\operatorname{mpg}=\beta_{0}+\beta_{1} \text { price }+\beta_{2} \text { weight }+u
$$

and constrain $\beta_{1}=\beta_{2}$, we could write

$$
\operatorname{mpg}=\beta_{0}+\beta_{1}(\text { price }+ \text { weight })+u
$$

and run a regression of mpg on price + weight. The estimated coefficient on the sum would be the constrained estimate of $\beta_{1}$ and $\beta_{2}$. Using cnsreg, however, is easier:

| (1978 automobile data) <br> . constraint 1 price = weight <br> . cnsreg mpg price weight, constraint(1) |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Constrained linear regression |  |  |  |  | $\begin{array}{llr} \text { Number of obs } & =74 \\ \mathrm{~F}(1,72) & =37.59 \\ \text { Prob }>\mathrm{F} & =0.0000 \\ \text { Root MSE } & =4.7220 \end{array}$ |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
| ( 1) price - weight $=0$ |  |  |  |  |  |  |
| mpg | Coefficient | Std. err. | t | $P>\|t\|$ | [95\% conf | interval] |
| price | -. 0009875 | . 0001611 | -6.13 | 0.000 | -. 0013086 | -. 0006664 |
| weight | -. 0009875 | . 0001611 | -6.13 | 0.000 | -. 0013086 | -. 0006664 |
| _cons | 30.36718 | 1.577958 | 19.24 | 0.000 | 27.22158 | 33.51278 |

We define constraints by using the constraint command; see [R] constraint. We fit the model with cnsreg and specify the constraint number or numbers in the constraints() option.

Just to show that the results above are correct, here is the result of applying the constraint by hand:

```
. generate x = price + weight
. regress mpg x
```

| Source | SS | df | MS | Number of obs | = | 74 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | $\mathrm{F}(1,72)$ | = | 37.59 |
| Model | 838.065767 | 1 | 838.065767 | Prob > F | = | 0.0000 |
| Residual | 1605.39369 | 72 | 22.2971346 | R -squared | = | 0.3430 |
|  |  |  |  | Adj R-squared | = | 0.3339 |
| Total | 2443.45946 | 73 | 33.4720474 | Root MSE | = | 4.722 |


| mpg | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | :---: | :---: | ---: | ---: |
| x | -.0009875 | .0001611 | -6.13 | 0.000 | -.0013086 | -.0006664 |
| _cons | 30.36718 | 1.577958 | 19.24 | 0.000 | 27.22158 | 33.51278 |

## $>$ Example 2: Multiple constraints

Models can be fit subject to multiple simultaneous constraints. We simply define the constraints and then include the constraint numbers in the constraints() option. For instance, say that we wish to fit the model

$$
\begin{aligned}
\operatorname{mpg}= & \beta_{0}+\beta_{1} \text { price }+\beta_{2} \text { weight }+\beta_{3} \text { displ }+\beta_{4} \text { gear_ratio }+\beta_{5} \text { foreign }+ \\
& \beta_{6} \text { length }+u
\end{aligned}
$$

subject to the constraints

$$
\begin{aligned}
& \beta_{1}=\beta_{2}=\beta_{3}=\beta_{6} \\
& \beta_{4}=-\beta_{5}=\beta_{0} / 20
\end{aligned}
$$

(This model, like the one in example 1, is admittedly senseless.) We fit the model by typing

```
. constraint 1 price=weight
. constraint 2 displ=weight
. constraint 3 length=weight
. constraint 5 gear_ratio = -foreign
. constraint 6 gear_ratio = _cons/20
```

```
    ( 1) price - weight = 0
    ( 2) - weight + displacement = 0
    ( 3) - weight + length = 0
    ( 4) gear_ratio + foreign = 0
    ( 5) gear_ratio - .05*_cons = 0
```

. cnsreg mpg price weight displ gear_ratio foreign length, $c(1-3,5-6)$
Constrained linear regression $\quad$ Number of obs $=\quad 74$
$\mathrm{F}(2,72)=785.20$
Prob > F $\quad=0.0000$
Root MSE $=4.6823$

| mpg | Coefficient | Std. err. | t | P>\|t| | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | ---: | ---: |
| price | -.000923 | .0001534 | -6.02 | 0.000 | -.0012288 | -.0006172 |
| weight | -.000923 | .0001534 | -6.02 | 0.000 | -.0012288 | -.0006172 |
| displacement | -.000923 | .0001534 | -6.02 | 0.000 | -.0012288 | -.0006172 |
| gear_ratio | 1.326114 | .0687589 | 19.29 | 0.000 | 1.189046 | 1.463183 |
| foreign | -1.326114 | .0687589 | -19.29 | 0.000 | -1.463183 | -1.189046 |
| length | -.000923 | .0001534 | -6.02 | 0.000 | -.0012288 | -.0006172 |
| _cons | 26.52229 | 1.375178 | 19.29 | 0.000 | 23.78092 | 29.26365 |

There are many ways we could have specified the constraints() option (which we abbreviated $c()$ above $)$. We typed $c(1-3,5-6)$, meaning that we want constraints 1 through 3 and 5 and 6 ; those numbers correspond to the constraints we defined. The only reason we did not use the number 4 was to emphasize that constraints do not have to be consecutively numbered. We typed $c(1-3,5-6)$, but we could have typed $c(1,2,3,5,6)$ or $c(1-3,5,6)$ or $c(1-2,3,5,6)$ or even $c(1-6)$, which would have worked as long as constraint 4 was not defined. If we had previously defined a constraint 4 , then $c(1-6)$ would have included it.

## Stored results

cnsreg stores the following in e() :
Scalars
e(N)
$e\left(d f \_m\right)$
e(df_r)
e(F)
e(p)
e(rmse)
e(11)
e(N_clust)
e(rank)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(vce)
e(vcetype)
e(properties)
e(predict)
number of observations model degrees of freedom residual degrees of freedom $F$ statistic
$p$-value for model test root mean squared error log likelihood number of clusters rank of $e(V)$

## cnsreg

command as typed
name of dependent variable
weight type
weight expression
title in estimation output name of cluster variable vcetype specified in vce() title used to label Std. err.
b V
program used to implement predict

```
        e(marginsok)
        e(marginsok)
        e(asobserved)
Matrices
    e(b)
    e(Cns)
    e(V)
    e(V_modelbased)
Functions
    e(sample)
```

    predictions allowed by margins
        factor variables fvset as asbalanced
        factor variables fvset as asobserved
    coefficient vector
constraints matrix
variance-covariance matrix of the estimators
model-based variance
marks estimation sample

In addition to the above, the following is stored in r() :

```
Matrices
```

    \(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values,
    and confidence intervals
    Note that results stored in r() are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

Let $n$ be the number of observations, $p$ be the total number of parameters (prior to restrictions and including the constant), and $c$ be the number of constraints. The coefficients are calculated as $\mathbf{b}^{\prime}=\mathbf{T}\left\{\left(\mathbf{T}^{\prime} \mathbf{X}^{\prime} \mathbf{W} \mathbf{X} \mathbf{T}\right)^{-1}\left(\mathbf{T}^{\prime} \mathbf{X}^{\prime} \mathbf{W} \mathbf{y}-\mathbf{T}^{\prime} \mathbf{X}^{\prime} \mathbf{W} \mathbf{X a} \mathbf{a}^{\prime}\right)\right\}+\mathbf{a}^{\prime}$, where $\mathbf{T}$ and $\mathbf{a}$ are as defined in $[\mathrm{P}]$ makecns. $\mathbf{W}=\mathbf{I}$ if no weights are specified. If weights are specified, let $\mathbf{v}: 1 \times n$ be the specified weights. If fweight frequency weights are specified, $\mathbf{W}=\operatorname{diag}(\mathbf{v})$. If aweight analytic weights are specified, then $\mathbf{W}=\operatorname{diag}\left[\mathbf{v} /\left(\mathbf{1}^{\prime} \mathbf{v}\right)\left(\mathbf{1}^{\prime} \mathbf{1}\right)\right]$, meaning that the weights are normalized to sum to the number of observations.

The mean squared error is $s^{2}=\left(\mathbf{y}^{\prime} \mathbf{W} \mathbf{y}-2 \mathbf{b}^{\prime} \mathbf{X}^{\prime} \mathbf{W} \mathbf{y}+\mathbf{b}^{\prime} \mathbf{X}^{\prime} \mathbf{W} \mathbf{X} \mathbf{b}\right) /(n-p+c)$. The variance covariance matrix is $s^{2} \mathbf{T}\left(\mathbf{T}^{\prime} \mathbf{X}^{\prime} \mathbf{W} \mathbf{X} \mathbf{T}\right)^{-1} \mathbf{T}^{\prime}$.

This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce (robust) and vce(cluster clustvar), respectively. See [P] _robust, particularly Introduction and Methods and formulas.
cnsreg also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

## References

Christodoulou, D. 2020. Stata tip 137: Interpreting constraints on slopes of rank-deficient design matrices. Stata Journal 20: 493-498.

Davidson, R., and J. G. MacKinnon. 1993. Estimation and Inference in Econometrics. New York: Oxford University Press.
Greene, W. H. 2018. Econometric Analysis. 8th ed. New York: Pearson.
Hill, R. C., W. E. Griffiths, and G. C. Lim. 2018. Principles of Econometrics. 5th ed. Hoboken, NJ: Wiley.

## Also see

[R] ensreg postestimation - Postestimation tools for cnsreg
$[\mathrm{R}]$ regress - Linear regression
[MI] Estimation - Estimation commands for use with mi estimate
[SVY] svy estimation - Estimation commands for survey data
[U] 20 Estimation and postestimation commands

## Title

cnsreg postestimation - Postestimation tools for cnsreg

Postestimation commands predict margins Also see

## Postestimation commands

The following postestimation commands are available after cnsreg:

| Command | Description |
| :---: | :---: |
| contrast | contrasts and ANOVA-style joint tests of estimates |
| estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estat (svy) | postestimation statistics for survey data |
| estimates | cataloging estimation results |
| etable | table of estimation results |
| * forecast | dynamic forecasts and simulations |
| * hausman | Hausman's specification test |
| lincom | point estimates, standard errors, testing, and inference for linear combinations of coefficients |
| linktest | link test for model specification |
| * 1 rtest | likelihood-ratio test |
| margins | marginal means, predictive margins, marginal effects, and average marginal effects |
| marginsplot | graph the results from margins (profile plots, interaction plots, etc.) |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients |
| predict | predictions and their SEs, residuals, etc. |
| predictnl | point estimates, standard errors, testing, and inference for generalized predictions |
| pwcompare | pairwise comparisons of estimates |
| suest | seemingly unrelated estimation |
| test | Wald tests of simple and composite linear hypotheses |
| testnl | Wald tests of nonlinear hypotheses |

[^13]
## predict

## Description for predict

predict creates a new variable containing predictions such as linear predictions, residuals, standard errors, probabilities, and expected values.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

```
predict [type] newvar [if] [in] [, statistic]
```

statistic Description

Main
$\mathrm{xb} \quad$ linear prediction; the default
residuals residuals
stdp standard error of the prediction
stdf standard error of the forecast
$\operatorname{pr}(a, b) \quad \operatorname{Pr}\left(a<y_{j}<b\right)$
$\mathrm{e}(a, b) \quad E\left(y_{j} \mid a<y_{j}<b\right)$
$\underline{\operatorname{ystar}(a, b)} \quad E\left(y_{j}^{*}\right), y_{j}^{*}=\max \left\{a, \min \left(y_{j}, b\right)\right\}$
score equivalent to residuals

These statistics are available both in and out of sample; type predict . . . if e (sample) ... if wanted only for the estimation sample.
stdf is not allowed with svy estimation results.
where $a$ and $b$ may be numbers or variables; $a$ missing ( $a \geq$.) means $-\infty$, and $b$ missing ( $b \geq$.) means $+\infty$; see [U] 12.2.1 Missing values.

## Options for predict

$\qquad$ Main
xb , the default, calculates the linear prediction.
residuals calculates the residuals, that is, $y_{j}-\mathbf{x}_{j} \mathbf{b}$.
stdp calculates the standard error of the prediction, which can be thought of as the standard error of the predicted expected value or mean for the observation's covariate pattern. The standard error of the prediction is also referred to as the standard error of the fitted value.
stdf calculates the standard error of the forecast, which is the standard error of the point prediction for 1 observation. It is commonly referred to as the standard error of the future or forecast value. By construction, the standard errors produced by stdf are always larger than those produced by stdp; see Methods and formulas in [R] regress postestimation.
$\operatorname{pr}(a, b)$ calculates $\operatorname{Pr}\left(a<\mathbf{x}_{j} \mathbf{b}+u_{j}<b\right)$, the probability that $y_{j} \mid \mathbf{x}_{j}$ would be observed in the interval $(a, b)$.
$a$ and $b$ may be specified as numbers or variable names; $l b$ and $u b$ are variable names;
$\operatorname{pr}(20,30)$ calculates $\operatorname{Pr}\left(20<\mathbf{x}_{j} \mathbf{b}+u_{j}<30\right)$;
$\operatorname{pr}(l b, u b)$ calculates $\operatorname{Pr}\left(l b<\mathbf{x}_{j} \mathbf{b}+u_{j}<u b\right)$; and
$\operatorname{pr}(20, u b)$ calculates $\operatorname{Pr}\left(20<\mathbf{x}_{j} \mathbf{b}+u_{j}<u b\right)$.
$a$ missing $(a \geq$.$) means -\infty ; \operatorname{pr}(., 30)$ calculates $\operatorname{Pr}\left(-\infty<\mathbf{x}_{j} \mathbf{b}+u_{j}<30\right)$;
$\operatorname{pr}(l b, 30)$ calculates $\operatorname{Pr}\left(-\infty<\mathbf{x}_{j} \mathbf{b}+u_{j}<30\right)$ in observations for which $l b \geq$.
and calculates $\operatorname{Pr}\left(l b<\mathbf{x}_{j} \mathbf{b}+u_{j}<30\right)$ elsewhere.
$b$ missing ( $b \geq$.) means $+\infty$; $\operatorname{pr}(20,$.$) calculates \operatorname{Pr}\left(+\infty>\mathbf{x}_{j} \mathbf{b}+u_{j}>20\right)$;
$\operatorname{pr}(20, u b)$ calculates $\operatorname{Pr}\left(+\infty>\mathbf{x}_{j} \mathbf{b}+u_{j}>20\right)$ in observations for which $u b \geq$.
and calculates $\operatorname{Pr}\left(20<\mathbf{x}_{j} \mathbf{b}+u_{j}<u b\right)$ elsewhere.
$\mathrm{e}(a, b)$ calculates $E\left(\mathbf{x}_{j} \mathbf{b}+u_{j} \mid a<\mathbf{x}_{j} \mathbf{b}+u_{j}<b\right)$, the expected value of $y_{j} \mid \mathbf{x}_{j}$ conditional on $y_{j} \mid \mathbf{x}_{j}$ being in the interval $(a, b)$, meaning that $y_{j} \mid \mathbf{x}_{j}$ is truncated. $a$ and $b$ are specified as they are for pr() .
ystar $(a, b)$ calculates $E\left(y_{j}^{*}\right)$, where $y_{j}^{*}=a$ if $\mathbf{x}_{j} \mathbf{b}+u_{j} \leq a, y_{j}^{*}=b$ if $\mathbf{x}_{j} \mathbf{b}+u_{j} \geq b$, and $y_{j}^{*}=\mathbf{x}_{j} \mathbf{b}+u_{j}$ otherwise, meaning that $y_{j}^{*}$ is censored. $a$ and $b$ are specified as they are for pr() . score is equivalent to residuals for linear regression models.

## margins

## Description for margins

margins estimates margins of response for linear predictions, probabilities, and expected values.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...)...] [options]
```

statistic Description
$\mathrm{xb} \quad$ linear prediction; the default
$\operatorname{pr}(a, b) \quad \operatorname{Pr}\left(a<y_{j}<b\right)$
$\mathrm{e}(a, b) \quad E\left(y_{j} \mid a<y_{j}<b\right)$
ystar $(a, b) \quad E\left(y_{j}^{*}\right), y_{j}^{*}=\max \left\{a, \min \left(y_{j}, b\right)\right\}$
stdp not allowed with margins
stdf not allowed with margins
residuals not allowed with margins
score not allowed with margins
Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see $[R]$ margins.

## Also see

[R] ensreg - Constrained linear regression
[U] 20 Estimation and postestimation commands

## Title

constraint - Define and list constraints

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Remarks and examples | Reference | Also see |  |

## Description

constraint defines, lists, and drops linear constraints. Constraints are for use by models that allow constrained estimation.

Constraints are defined by the constraint command. The currently defined constraints can be listed by either constraint list or constraint dir; both do the same thing. Existing constraints can be eliminated by constraint drop.
constraint get and constraint free are programmer's commands. constraint get returns the contents of the specified constraint in macro $r$ (contents) and returns in scalar $r$ (defined) 0 or $1-1$ being returned if the constraint was defined. constraint free returns the number of a free (unused) constraint in macro $r$ (free).

## Quick start

## For single-equation models

Define constraint 1 to constrain the coefficient for x 1 to 0
constraint define $1 \times 1=0$
Same as above
constraint 1 x 1
Constrain coefficients for x 1 and x 2 to 0
constraint 2 x 1 x 2
Overwrite constraint 2 to constrain x 2 and x 3 to equality
constraint $2 \times 2=x 3$
Constrain the coefficients for factor indicators 2 .a and 3.a to equality
constraint 32 .a $=3 . \mathrm{a}$
Constrain the coefficient on x 1 to equal 1
constraint $4 \mathrm{x} 1=1$

## For multiple-equation models

Constrain coefficient for x 4 to 0 in all equations
constraint 11 x 4
Constrain coefficients for x 4 and x 5 to equality in the equation for y 2
constraint 12 [y2]x4 = [y2]x5

Constrain the coefficient for x 5 to equality in equations for y 1 and y 2
constraint 13 [y1=y2] x5
Constrain coefficient for x 1 to 0 in equation where the dependent variable equals cat2
constraint 14 [cat2] x 1
Constrain the coefficients for factor indicators $1 . \mathrm{a}$ and $1 . \mathrm{b}$ to equality in the equation for category cat3
constraint 15 [cat3]: 1.a $=1 . \mathrm{b}$
Constrain coefficients for x 1 to equality in the equations for categories cat2 and cat3 constraint 16 [cat2=cat3]: x1

Listing constraints
List existing constraints
constraint dir
Same as above
constraint list

## Menu

Statistics $>$ Other $>$ Manage constraints

## Syntax

Define constraints

```
constraint [\underline{define] # [ exp=exp coeflist]}
```

List constraints

```
constraint dir [numlist|_all]
constraint list [numlist|_all]
```

Drop constraints

```
constraint drop {numlist|_all }
```

Programmer's commands

```
constraint get #
constraint free
```

where coeflist is as defined in $[\mathrm{R}]$ test and \# is restricted to the range 1 to 1,999 , inclusive.

## Remarks and examples

Using constraints is discussed in [R] ensreg, $[R]$ mlogit, and $[R]$ reg3; this entry is concerned only with practical aspects of defining and manipulating constraints.

## > Example 1

Constraints are numbered from 1 to 1,999 , and we assign the number when we define the constraint:

```
. use https://www.stata-press.com/data/r18/sysdsn1
(Health insurance data)
. constraint 2 [indemnity]2.site = 0
```

The currently defined constraints can be listed by constraint list:
. constraint list
2: [indemnity]2.site $=0$
constraint drop drops constraints:
. constraint drop 2
. constraint list
The empty list after constraint list indicates that no constraints are defined. Below, we demonstrate the various syntaxes allowed by constraint:
. constraint 1 [Indemnity]
. constraint 10 [Indemnity]: 1.site 2.site
. constraint 11 [Indemnity]: 3.site

```
. constraint 21 [Prepaid=Uninsure]: nonwhite
. constraint 30 [Prepaid]
. constraint 31 [Insure]
. constraint list
        1: [Indemnity]
    10: [Indemnity]: 1.site 2.site
    11: [Indemnity]: 3.site
    21: [Prepaid=Uninsure]: nonwhite
    30: [Prepaid]
    31: [Insure]
. constraint drop 21-25, 31
. constraint list
        1: [Indemnity]
    10: [Indemnity]: 1.site 2.site
    11: [Indemnity]: 3.site
    30: [Prepaid]
. constraint drop _all
. constraint list
```


## Technical note

The constraint command does not check the syntax of the constraint itself because a constraint can be interpreted only in the context of a model. Thus, constraint is willing to define constraints that later will not make sense. Any errors in the constraints will be detected and mentioned at the time of estimation.

## Reference

Buis, M. L. 2012. Stata tip 108: On adding and constraining. Stata Journal 12: 342-344.

## Also see

[R] ensreg - Constrained linear regression
[P] makecns - Constrained estimation

## Title

contrast - Contrasts and linear hypothesis tests after estimation

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

contrast tests linear hypotheses and forms contrasts involving factor variables and their interactions from the most recently fit model. The tests include ANOVA-style tests of main effects, simple effects, interactions, and nested effects. contrast can use named contrasts to decompose these effects into comparisons against reference categories, comparisons of adjacent levels, comparisons against the grand mean, orthogonal polynomials, and such. Custom contrasts may also be specified.
contrast can be used with svy estimation results; see [SVY] svy postestimation.
Contrasts can also be computed for margins of linear and nonlinear responses; see [R] margins, contrast.

## Quick start

Contrasts for one-way models
Test the main effect of categorical variable a after regress y i.a or anova y a contrast a

Reference category contrasts of cell means of $y$ with the smallest value of a as the base category contrast r.a

Same as above, but specify $\mathrm{a}=3$ as the base category for comparisons
contrast rb3.a
Report tests instead of confidence intervals for each contrast
contrast r.a, pveffects
Report tests and confidence intervals for each contrast
contrast r.a, effects
Contrasts of the cell mean of $y$ for each level of a with the grand mean of $y$
contrast g.a
Same as above, but compute grand mean as a weighted average of cell means with weights based on the number of observations for each level of a

```
contrast gw.a
```

User-defined contrast comparing the cell mean of y for $\mathrm{a}=1$ with the average of the cell means for

```
a}=3\mathrm{ and a = 4
    contrast {a -1 0 .5 .5}
```


## Contrasts for two-way models

Test of the interaction effect after regress y a\#\#b or anova y a\#\#b
contrast a\#b
Test of the main and interaction effects

```
contrast a b a#b
```

Same as above

```
    contrast a##b
```

Individual reference category contrasts for the interaction of $a$ and $b$

```
    contrast r.a#r.b
```

Joint tests of the simple effects of a within each level of $b$ contrast a@b

Individual reference category contrasts for the simple effects of $a$ within each level of $b$ contrast r.a@b

Orthogonal polynomial contrasts for a within each level of $b$
contrast p.a@b
Reference category contrasts of the marginal means of $y$ for levels of a contrast r.a

Same as above, but with marginal means for a computed as a weighted average of cell means, using the marginal frequencies of $b$ rather than equal weights for each level
contrast r.a, asobserved
Contrasts of the marginal mean of $y$ for each level of a with the previous level-reverse-adjacent contrasts

```
contrast ar.a
```

Contrasts for models with continuous covariates
Test of the interaction effect after regress y a\#\#c.x or anova y a\#\#c.x
contrast a\#c.x
Reference category effects of a on the slope of $x$
contrast r.a\#c.x
Reference category effects of a on the intercept

```
    contrast r.a
```


## Contrasts for nonlinear models

Orthogonal polynomial contrasts of log odds across levels of a after logit y i.a

```
contrast p.a
```

Test the main and interaction effects after logit y a\#\#b

Simple reference category effects for $a$ within each level of $b$ contrast r.a@b

Contrasts for multiple-equation models
Test the main and interaction effects in the equation for y 2 after mvreg y1 y2 y3 $=\mathrm{a} \# \# \mathrm{~b}$ contrast $a \# \# b$, equation(y2)

Reference category contrasts of estimated marginal means of $y 3$ for levels of a contrast r.a, equation(y3)

Test for a difference in the overall estimated marginal means of $\mathrm{y} 1, \mathrm{y} 2$, and y3 contrast _eqns

Contrasts of estimated marginal means of y2 and y3 with y1 contrast r._eqns

Test whether interaction effects differ across equations
contrast a\#b\#_eqns

## Menu

Statistics $>$ Postestimation

## Syntax

```
contrast termlist [, options]
```

where termlist is a list of factor variables or interactions that appear in the current estimation results. The variables may be typed with or without contrast operators, and you may use any factor-variable syntax:

See the operators (op.) table below for the list of contrast operators.
options Description

## Main

overall
asobserved
lincom
Equations
equation(eqspec)
atequations
Advanced emptycells(empspec) noestimcheck

Reporting
level(\#) confidence level; default is level(95)
mcompare (method)
noeffects
cieffects
pveffects
effects
nowald
noatlevels
nosvyadjust
sort
post
display_options
eform_option
df (\#) use $t$ distribution with \# degrees of freedom for computing $p$-values and confidence intervals
collect is allowed; see [U] 11.1.10 Prefix commands.
$\mathrm{df}(\#)$ does not appear in the dialog box.


| Other conditional effects |  |
| :--- | :--- |
| A\#B@C | joint tests of the interaction effects of A and B within each level of C |
| A\#B@C\#D | joint tests of the interaction effects of A and B within each combination of <br> the levels of $C$ and $D$ |
| $r . A \# g$. B@C | individual contrasts for each interaction of A and B that decompose A\#B@C <br> using $r$. and $g$. |

Nested effects

A|B\#C
A\#B|C
A\#BlC\#D
$r . \mathrm{A} \mid \mathrm{B}$
$r . \mathrm{A} \mid \mathrm{B} \# \mathrm{C}$
$r . \mathrm{A} \# g . \mathrm{BlC}$

Slope effects
A\#c. $x$
A\#c.x\#c.y
A\#B\#c.x
A\#B\#c.x\#c.y
$r . A \# c . x$
joint test of the effects of $A$ on the slopes of $x$
joint test of the effects of $A$ on the slopes of the product (interaction) of $x$ and $y$ joint test of the interaction effects of $A$ and $B$ on the slopes of $x$
joint test of the interaction effects of $A$ and $B$ on the slopes of the product (interaction) of x and y
individual contrasts of A's effects on the slopes of x using $r$.
Denominators
... / term2
use term 2 as the denominator in the $F$ tests of the preceding terms
... / use the residual as the denominator in the $F$ tests of the preceding terms (the default if no other /s are specified)
$\mathrm{A}, \mathrm{B}, \mathrm{C}$, and D represent any factor variable in the current estimation results.
x and y represent any continuous variable in the current estimation results.
$r$. and $g$. represent any contrast operator. See the table below.
c. specifies that a variable be treated as continuous; see [U] 11.4.3 Factor variables.

Operators are allowed on any factor variable that does not appear to the right of @ or I. Operators decompose the effects of the associated factor variable into one-degree-of-freedom effects (contrasts).
Higher-level interactions are allowed anywhere an interaction operator (\#) appears in the table.
Time-series operators are allowed if they were used in the estimation.
_eqns designates the equations in manova, mlogit, mprobit, and mvreg and can be specified anywhere a factor variable appears.
/ is allowed only after anova, cnsreg, manova, mvreg, or regress.
operators (op.) Description
r. differences from the reference (base) level; the default
a. differences from the next level (adjacent contrasts)
ar. differences from the previous level (reverse adjacent contrasts)
As-balanced operators
g.
h.
j.
p.
q.

As-observed operators
gw.
hw.
jw.
pw.
qW.
differences from the balanced grand mean
differences from the balanced mean of subsequent levels (Helmert contrasts)
differences from the balanced mean of previous levels (reverse Helmert contrasts)
orthogonal polynomial in the level values
orthogonal polynomial in the level sequence
differences from the observation-weighted grand mean
differences from the observation-weighted mean of subsequent levels differences from the observation-weighted mean of previous levels observation-weighted orthogonal polynomial in the level values
observation-weighted orthogonal polynomial in the level sequence

One or more individual contrasts may be selected by using the op\#. or op (numlist) . syntax. For example, a3. A selects the adjacent contrast for level 3 of $A$, and $p(1 / 2)$. B selects the linear and quadratic effects of B. Also see Orthogonal polynomial contrasts and Beyond linear models.

## Custom contrasts Description

\{A numlist $\}$
\{A\#B numlist\}
user-defined contrast on the levels of factor A
user-defined contrast on the levels of interaction between A and B
Custom contrasts may be part of a term, such as \{A numlist\}\#B, \{A numlist\}@B, \{A numlist\}|B,\{A\#B numlist $\}$, and \{A numlist \}\#\{B numlist\}. The same is true of higher-order custom contrasts, such as $\{\mathrm{A} \# \mathrm{~B}$ numlist $\} @ C,\{\mathrm{~A} \# \mathrm{~B}$ numlist $\} \# r$. C , and $\{\mathrm{A} \# \mathrm{~B}$ numlist \}\#c.x.

Higher-order interactions with at most eight factor variables are allowed with custom contrasts.

## method

Description
noadjust
bonferroni [adjustall]
sidak [adjustall]
scheffe
do not adjust for multiple comparisons; the default
Bonferroni's method; adjust across all terms
Šidák's method; adjust across all terms
Scheffé's method

## Options

$\qquad$ Main
overall specifies that a joint hypothesis test over all terms be performed.
asobserved specifies that factor covariates be evaluated using the cell frequencies observed in the estimation sample. The default is to treat all factor covariates as though there were an equal number of observations in each level.
lincom specifies that user-defined contrasts be treated as linear combinations. The default is to require that all user-defined contrasts sum to zero. (Summing to zero is part of the definition of a contrast.)

## Equations

equation (eqspec) specifies the equation from which contrasts are to be computed. The default is to compute contrasts from the first equation.
atequations specifies that the contrasts be computed within each equation.

Advanced
emptycells (empspec) specifies how empty cells are handled in interactions involving factor variables that are being treated as balanced.
emptycells(strict) is the default; it specifies that contrasts involving empty cells be treated as not estimable.
emptycells (reweight) specifies that the effects of the observed cells be increased to accommodate any missing cells. This makes the contrast estimable but changes its interpretation.
noestimcheck specifies that contrast not check for estimability. By default, the requested contrasts are checked and those found not estimable are reported as such. Nonestimability is usually caused by empty cells. If noestimcheck is specified, estimates are computed in the usual way and reported even though the resulting estimates are manipulable, which is to say they can differ across equivalent models having different parameterizations.

## Reporting

level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals. mcompare (method) specifies the method for computing $p$-values and confidence intervals that account for multiple comparisons within a factor-variable term.

Most methods adjust the comparisonwise error rate, $\alpha_{c}$, to achieve a prespecified experimentwise error rate, $\alpha_{e}$.
mcompare(noadjust) is the default; it specifies no adjustment.

$$
\alpha_{c}=\alpha_{e}
$$

mcompare(bonferroni) adjusts the comparisonwise error rate based on the upper limit of the Bonferroni inequality

$$
\alpha_{e} \leq m \alpha_{c}
$$

where $m$ is the number of comparisons within the term.
The adjusted comparisonwise error rate is

$$
\alpha_{c}=\alpha_{e} / m
$$

mcompare (sidak) adjusts the comparisonwise error rate based on the upper limit of the probability inequality

$$
\alpha_{e} \leq 1-\left(1-\alpha_{c}\right)^{m}
$$

where $m$ is the number of comparisons within the term.
The adjusted comparisonwise error rate is

$$
\alpha_{c}=1-\left(1-\alpha_{e}\right)^{1 / m}
$$

This adjustment is exact when the $m$ comparisons are independent.
mcompare (scheffe) controls the experimentwise error rate using the $F$ or $\chi^{2}$ distribution with degrees of freedom equal to the rank of the term.
mcompare (method adjustall) specifies that the multiple-comparison adjustments count all comparisons across all terms rather than performing multiple comparisons term by term. This leads to more conservative adjustments when multiple variables or terms are specified in marginslist. This option is compatible only with the bonferroni and sidak methods.
noeffects suppresses the table of individual contrasts with confidence intervals. This table is produced by default when the mcompare() option is specified or when a term in termlist implies all individual contrasts.
cieffects specifies that a table containing a confidence interval for each individual contrast be reported.
pveffects specifies that a table containing a $p$-value for each individual contrast be reported.
effects specifies that a single table containing a confidence interval and $p$-value for each individual contrast be reported.
nowald suppresses the table of Wald tests.
noatlevels indicates that only the overall Wald test be reported for each term containing within or nested (@ or I) operators.
nosvyadjust is for use with svy estimation commands. It specifies that the Wald test be carried out without the default adjustment for the design degrees of freedom. That is to say the test is carried out as $W / k \sim F(k, d)$ rather than as $(d-k+1) W /(k d) \sim F(k, d-k+1)$, where $k$ is the dimension of the test and $d$ is the total number of sampled PSUs minus the total number of strata. sort specifies that the table of individual contrasts be sorted by the contrast values within each term. post causes contrast to behave like a Stata estimation (e-class) command. contrast posts the vector of estimated contrasts along with the estimated variance-covariance matrix to e(), so you can treat the estimated contrasts just as you would results from any other estimation command. For example, you could use test to perform simultaneous tests of hypotheses on the contrasts, or you could use lincom to create linear combinations.
display_options: vsquish, nofvlabel, fvwrap(\#), fvwrapon(style), cformat(\%fmt), pformat (\%fmt), sformat (\%fmt), and nolstretch.
vsquish specifies that the blank space separating factor-variable terms or time-series-operated variables from other variables in the model be suppressed.
nofvlabel displays factor-variable level values rather than attached value labels. This option overrides the fvlabel setting; see [R] set showbaselevels.
fvwrap(\#) specifies how many lines to allow when long value labels must be wrapped. Labels requiring more than \# lines are truncated. This option overrides the fvwrap setting; see [R] set showbaselevels.
fvwrapon(style) specifies whether value labels that wrap will break at word boundaries or break based on available space.
fvwrapon(word), the default, specifies that value labels break at word boundaries.
fvwrapon(width) specifies that value labels break based on available space.
This option overrides the fvwrapon setting; see [R] set showbaselevels.
cformat (\% firt) specifies how to format contrasts, standard errors, and confidence limits in the table of estimated contrasts.
pformat ( $\% \mathrm{fm} t$ ) specifies how to format $p$-values in the table of estimated contrasts. sformat (\%fmt) specifies how to format test statistics in the table of estimated contrasts. nolstretch specifies that the width of the table of estimated contrasts not be automatically widened to accommodate longer variable names. The default, lstretch, is to automatically widen the table of estimated contrasts up to the width of the Results window. Specifying lstretch or nolstretch overrides the setting given by set lstretch. If set lstretch has not been set, the default is lstretch. nolstretch is not shown in the dialog box.
eform_option specifies that the contrasts table be displayed in exponentiated form. $e^{\text {contrast }}$ is displayed rather than contrast. Standard errors and confidence intervals are also transformed. See $[\mathrm{R}]$ eform_option for the list of available options.

The following option is available with contrast but is not shown in the dialog box:
df (\#) specifies that the $t$ distribution with \# degrees of freedom be used for computing $p$-values and confidence intervals. The default is to use e(df_r) degrees of freedom or the standard normal distribution if e(df_r) is missing.

## Remarks and examples

```
Remarks are presented under the following headings:
Introduction
One-way models
    Estimated cell means
    Testing equality of cell means
    Reference category contrasts
    Reverse adjacent contrasts
    Orthogonal polynomial contrasts
Two-way models
    Estimated interaction cell means
    Simple effects
    Interaction effects
    Main effects
    Partial interaction effects
Three-way and higher-order models
Contrast operators
    Differences from a reference level (r.)
    Differences from the next level (a.)
    Differences from the previous level (ar.)
    Differences from the grand mean (g.)
    Differences from the mean of subsequent levels (h.)
    Differences from the mean of previous levels (j.)
    Orthogonal polynomials (p. and q.)
User-defined contrasts
Empty cells
Empty cells, ANOVA style
Nested effects
Multiple comparisons
Unbalanced data
    Using observed cell frequencies
    Weighted contrast operators
Testing factor effects on slopes
Chow tests
Beyond linear models
Multiple equations
Video example
```


## Introduction

contrast performs ANOVA-style tests of main effects, interactions, simple effects, and nested effects. It can easily decompose these tests into constituent contrasts using either named contrasts (codings) or user-specified contrasts. Comparing levels of factor variables-whether as main effects, interactions, or simple effects-is as easy as adding a contrast operator to the variable. The operators can compare each level with the previous level, each level with a reference level, each level with the mean of previous levels, and more.
contrast tests and estimates contrasts. A contrast of the parameters $\mu_{1}, \mu_{2}, \ldots, \mu_{p}$ is a linear combination $\sum_{i} c_{i} \mu_{i}$ whose $c_{i}$ sum to zero. A difference of population means such as $\mu_{1}-\mu_{2}$ is a contrast, as are most other comparisons of population or model quantities (Coster 2005). Some contrasts may be estimated with lincom, but contrast is much more powerful. contrast can handle multiple contrasts simultaneously, and the command's contrast operators make it easy to specify complicated linear combinations.

Both the contrast operation and the creation of the margins for comparison can be performed as though the data were balanced (typical for experimental designs) or using the observed frequencies in the estimation sample (typical for observational studies). contrast can perform these analyses on the results of almost all of Stata's estimators, not just the linear-models estimators.

Most of contrast's computations can be considered comparisons of estimated cell means from a model fit. Tests of interactions are tests of whether the cell means for the interaction are all equal. Tests of main effects are tests of whether the marginal cell means for the factor are all equal. More focused comparisons of cell means (for example, is level 2 equal to level 1) are specified using contrast operators. More formally, all of contrast's computations are comparisons of conditional expectations; cell means are one type of conditional expectation.

All contrasts can also easily be graphed; see [R] marginsplot.
For a discussion of contrasts and testing for linear models, see Searle and Gruber (2017) and Searle (1997). For discussions specifically related to experimental design, see Winer, Brown, and Michels (1991) and Milliken and Johnson (2009). Rosenthal, Rosnow, and Rubin (2000) focus on contrasts with applications in behavioral sciences. Mitchell $(2021,2015)$ and Baldwin (2019) focus on contrasts in Stata.
contrast is a flexible tool for understanding the effects of categorical covariates. If your model contains categorical covariates, and especially if it contains interactions, you will want to use contrast.

## One-way models

Suppose we have collected data on cholesterol levels for individuals from five age groups. To study the effect of age group on cholesterol, we can begin by fitting a one-way model using regress:

```
. use https://www.stata-press.com/data/r18/cholesterol
(Artificial cholesterol data)
. label list ages
ages:
            1 10-19
            2 20-29
            3 30-39
            4 40-59
            5 60-79
```

. regress chol i.agegrp

| Source | SS | df | MS | Number of obs | = | 75 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | $\mathrm{F}(4,70)$ | = | 35.02 |
| Model | 14943.3997 | 4 | 3735.84993 | Prob > F | = | 0.0000 |
| Residual | 7468.21971 | 70 | 106.688853 | R -squared | = | 0.6668 |
|  |  |  |  | Adj R-squared | = | 0.6477 |
| Total | 22411.6194 | 74 | 302.859722 | Root MSE | = | 10.329 |


| chol | Coefficient | Std. err. | t | $\mathrm{P}>\mid \mathrm{tI}$ | [95\% conf. interval] |  |
| :---: | ---: | :---: | ---: | :---: | ---: | ---: |
| agegrp |  |  |  |  |  |  |
| $20-29$ | 8.203575 | 3.771628 | 2.18 | 0.033 | .6812991 | 15.72585 |
| $30-39$ | 21.54105 | 3.771628 | 5.71 | 0.000 | 14.01878 | 29.06333 |
| $40-59$ | 30.15067 | 3.771628 | 7.99 | 0.000 | 22.6284 | 37.67295 |
| $60-79$ | 38.76221 | 3.771628 | 10.28 | 0.000 | 31.23993 | 46.28448 |
| _cons | 180.5198 | 2.666944 | 67.69 | 0.000 | 175.2007 | 185.8388 |

## Estimated cell means

margins will show us the estimated cell means for each age group based on our fitted model:
. margins agegrp
Adjusted predictions $\quad$ Number of obs $=75$
Model VCE: OLS
Expression: Linear prediction, predict()

|  | Delta-method <br> std. err. |  |  |  |  |  |  |  | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: | :---: | :---: | :---: | :---: | :---: |
| agegrp |  |  |  |  |  |  |  |  |  |  |  |
| $10-19$ | 180.5198 | 2.666944 | 67.69 | 0.000 | 175.2007 | 185.8388 |  |  |  |  |  |
| $20-29$ | 188.7233 | 2.666944 | 70.76 | 0.000 | 183.4043 | 194.0424 |  |  |  |  |  |
| $30-39$ | 202.0608 | 2.666944 | 75.76 | 0.000 | 196.7418 | 207.3799 |  |  |  |  |  |
| $40-59$ | 210.6704 | 2.666944 | 78.99 | 0.000 | 205.3514 | 215.9895 |  |  |  |  |  |
| $60-79$ | 219.282 | 2.666944 | 82.22 | 0.000 | 213.9629 | 224.601 |  |  |  |  |  |

We can graph those means with marginsplot:

```
. marginsplot
Variables that uniquely identify margins: agegrp
```

Adjusted predictions of agegrp with $95 \%$ Cls


## Testing equality of cell means

Are all the means equal? That is to say is there an effect of age group on cholesterol level? We can answer that by asking contrast to test whether the means of the age groups are identical.

```
. contrast agegrp
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | $F$ | $P>F$ |
| ---: | ---: | ---: | ---: |
| agegrp | 4 | 35.02 | 0.0000 |
| Denominator | 70 |  |  |

The means are clearly different. We could have obtained this same test directly had we fit our model using anova rather than regress.

|  | $\begin{aligned} \text { Number of obs } & = \\ \text { Root MSE } & = \end{aligned}$ | $\begin{array}{r} 75 \\ 10.329 \end{array}$ | R -squared $=$ <br> Adj R-squared $=$ |  | $\begin{aligned} & 0.6668 \\ & 0.6477 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | 14943.4 | 4 | 3735.8499 | 35.02 | 0.0000 |
| agegrp | 14943.4 | 4 | 3735.8499 | 35.02 | 0.0000 |
| Residual | 7468.2197 | 70 | 106.68885 |  |  |
| Total | 22411.619 | 74 | 302.85972 |  |  |

Achieving a more direct test result is why we recommend using anova instead of regress for models where our focus is on the categorical covariates. The models fit by anova and regress are identical; they merely parameterize the effects differently. The results of contrast will be identical regardless of which command is used to fit the model. If, however, we were fitting models whose responses are nonlinear functions of the covariates, such as logistic regression, then there would be no analogue to anova, and we would appreciate contrast's ability to quickly test main effects and interactions.

## Reference category contrasts

Now that we know that the overall effect of age group is statistically significant, we can explore the effects of each age group. One way to do that is to use the reference category operator, r.:
. contrast r.agegrp
Contrasts of marginal linear predictions
Margins: asbalanced

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | ---: | ---: | ---: |
| agegrp |  |  |  |
| $(20-29$ vs 10-19) | 1 | 4.73 | 0.0330 |
| $(40-39$ vs 10-19) | 1 | 32.62 | 0.0000 |
| $(60-79$ vs 10-19) vs 10-19) | 1 | 63.91 | 0.0000 |
| Joint | 1 | 105.62 | 0.0000 |
|  | 4 | 35.02 | 0.0000 |
| Denominator | 70 |  |  |


|  | Contrast | Std. err. | [95\% conf. interval] |  |
| :--- | ---: | :--- | ---: | ---: |
| agegrp |  |  |  |  |
| $(20-29$ vs 10-19) | 8.203575 | 3.771628 | .6812991 | 15.72585 |
| $(30-39$ vs 10-19) | 21.54105 | 3.771628 | 14.01878 | 29.06333 |
| $(40-59$ vs 10-19) | 30.15067 | 3.771628 | 22.6284 | 37.67295 |
| $(60-79$ vs 10-19) | 38.76221 | 3.771628 | 31.23993 | 46.28448 |

The cell mean of each age group is compared against the base age group (ages 10-19). The first table shows that each difference is significant. The second table gives an estimate and confidence interval for each contrast. These are the comparisons that linear regression gives with a factor covariate and no interactions. The contrasts are identical to the coefficients from our linear regression.

## Reverse adjacent contrasts

We have far more flexibility with contrast. Age group is ordinal, so it is interesting to compare each age group with the preceding age group (rather than against one reference group). We specify that analysis by using the reverse adjacent operator, ar.:

| . contrast ar.agegrp |
| :--- |
| Contrasts of marginal linear predictions |
| Margins: asbalanced |


|  | Contrast | Std. err. | [95\% conf. interval] |  |
| :--- | ---: | :--- | :--- | :--- |
| agegrp |  |  |  |  |
| $(20-29$ vs 10-19) | 8.203575 | 3.771628 | .6812991 | 15.72585 |
| $(30-39$ vs 20-29) | 13.33748 | 3.771628 | 5.815204 | 20.85976 |
| $(40-59$ vs $30-39)$ | 8.60962 | 3.771628 | 1.087345 | 16.1319 |
| $(60-79$ vs 40-59) | 8.611533 | 3.771628 | 1.089257 | 16.13381 |

The 20-29 age group's cholesterol level is 8.2 points higher than the $10-19$ age group's cholesterol level; the $30-39$ age group's level is 13.3 points higher than the $20-29$ age group's level; and so on. Each age group is statistically different from the preceding age group at the $5 \%$ level.

## Orthogonal polynomial contrasts

The relationship between age group and cholesterol level looked almost linear in our graph. We can examine that relationship further by using the orthogonal polynomial operator, p.:

```
. contrast p.agegrp, noeffects
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| :---: | :---: | ---: | ---: |
| agegrp |  |  |  |
| (linear) | 1 | 139.11 | 0.0000 |
| (quadratic) | 1 | 0.15 | 0.6962 |
| (quartic) | 1 | 0.37 | 0.5448 |
| Joint | 1 | 0.43 | 0.5153 |
| Denominator | 4 | 35.02 | 0.0000 |

Only the linear effect is statistically significant.
We can even perform the joint test that all effects beyond linear are zero. We do that by selecting all polynomial contrasts above linear-that is, polynomial contrasts 2, 3, and 4.

```
. contrast p(2 3 4).agegrp, noeffects
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| :---: | :---: | :---: | :---: |
| agegrp |  |  |  |
| (quadratic) | 1 | 0.15 | 0.6962 |
| (cubic) | 1 | 0.37 | 0.5448 |
| (quartic) | 1 | 0.43 | 0.5153 |
| Joint | 3 | 0.32 | 0.8129 |
| Denominator | 70 |  |  |

The joint test has three degrees of freedom and is clearly insignificant. A linear effect of age group seems adequate for this model.

## Two-way models

Suppose we are investigating the effects of different dosages of a blood pressure medication and believe that the effects may be different for men and women. We can fit the following ANOVA model for bpchange, the change in diastolic blood pressure. Change is defined as the after measurement minus the before measurement, so that negative values of bpchange correspond to decreases in blood pressure.

| (Artificial blood pr <br> . label list gender gender: |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: |
| 1 Male <br> 2 Female |  |  |  |  |  |
| anova bpchange dose\#\#gender |  |  |  |  |  |
|  | $\begin{array}{ll} \text { Number of obs } & = \\ \text { Root MSE } & = \end{array}$ | 30 | R -squared = |  | 0.9647 |
|  |  | 1.4677 | Adj R- | uared = | 0.9573 |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | 1411.9087 | 52 | 282.38174 | 131.09 | 0.0000 |
| dose | 963.48179 | 2 | 481.7409 | 223.64 | 0.0000 |
| gender | 355.11882 | 1 | 355.11882 | 164.85 | 0.0000 |
| dose\#gender | 93.308093 | 2 | 46.654046 | 21.66 | 0.0000 |
| Residual | 51.699253 | $24 \quad 2$ | 2.1541355 |  |  |
| Total | 1463.608 | 29 | 50.46924 |  |  |

## Estimated interaction cell means

Everything is significant, including the interaction. So increasing dosage is effective and differs by gender. Let's explore the effects. First, let's look at the estimated cell mean of blood pressure change for each combination of gender and dosage.

| Adjusted predictions |  |  |  |  | Number of obs $=30$ |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Expression: Linear prediction, predict() |  |  |  |  |  |  |
|  | Delta-method |  |  | $P>\|t\|$ | [95\% conf. interval] |  |
| dose\#gender 250\#Male | -7.35384 | . 6563742 | -11.20 | 0.000 | -8.708529 | -5.99915 |
| 250\#Female | 3.706567 | . 6563742 | 5.65 | 0.000 | 2.351877 | 5.061257 |
| 500\#Male | -13.73386 | . 6563742 | -20.92 | 0.000 | -15.08855 | -12.37917 |
| 500\#Female | -6.584167 | . 6563742 | -10.03 | 0.000 | -7.938857 | -5.229477 |
| 750\#Male | -16.82108 | . 6563742 | -25.63 | 0.000 | -18.17576 | -15.46639 |
| 750\#Female | -14.38795 | . 6563742 | -21.92 | 0.000 | -15.74264 | -13.03326 |

Our data are balanced, so these results will not be affected by the many different ways that margins can compute cell means. Moreover, because our model consists of only dose and gender, these are also the point estimates for each combination.

We can graph the results:

```
. marginsplot
Variables that uniquely identify margins: dose gender
```



The lines are not parallel, which we expected because the interaction term is significant. Males experience a greater decline in blood pressure at every dosage level, but the effect of increasing dosage is greater for females. In fact, it is not clear if we can tell the difference between male and female response at the maximum dosage.

## Simple effects

We can contrast the male and female responses within dosage to see the simple effects of gender. Because there are only two levels in gender, the choice of contrast operator is largely irrelevant. Aside from orthogonal polynomials, all operators produce the same estimates, although the effects can change signs.

- contrast r.gender@dose

Contrasts of marginal linear predictions
Margins: asbalanced

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | ---: | ---: | ---: |
| gender@dose |  |  |  |
| (Female vs Male) 250 | 1 | 141.97 | 0.0000 |
| (Female vs Male) 500 | 1 | 59.33 | 0.0000 |
| (Female vs Male) 750 | 1 | 6.87 | 0.0150 |
| Joint | 3 | 69.39 | 0.0000 |
| Denominator | 24 |  |  |


|  | Contrast | Std. err. | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: |
| gender@dose |  |  |  |  |
| (Female vs Male) 250 | 11.06041 | .9282533 | 9.144586 | 12.97623 |
| (Female vs Male) 500 | 7.149691 | .9282533 | 5.23387 | 9.065512 |
| (Female vs Male) 750 | 2.433124 | .9282533 | .5173031 | 4.348944 |

The effect for females is about 11 points higher than for males at a dosage of 250, and that shrinks to 2.4 points higher at the maximum dosage of 750 .

We can form the simple effects the other way by contrasting the effect of dose at each level of gender:
. contrast ar.dose@gender
Contrasts of marginal linear predictions
Margins: asbalanced

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | ---: | ---: | ---: |
| dose@gender |  |  |  |
| (500 vs 250) Male | 1 | 47.24 | 0.0000 |
| $(500$ vs 250) Female | 1 | 122.90 | 0.0000 |
| (750 vs 500) Male | 1 | 11.06 | 0.0028 |
| $(750$ vs 500) Female | 1 | 70.68 | 0.0000 |
| Joint | 4 | 122.65 | 0.0000 |
| Denominator |  |  |  |


|  | Contrast | Std. err. | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: |
| dose@gender |  |  |  |  |
| (500 vs 250) Male | -6.380018 | .9282533 | -8.295839 | -4.464198 |
| (500 vs 250) Female | -10.29073 | .9282533 | -12.20655 | -8.374914 |
| (750 vs 500) Male | -3.087217 | .9282533 | -5.003038 | -1.171396 |
| $(750$ vs 500) Female | -7.803784 | .9282533 | -9.719605 | -5.887963 |

Here we use the ar. reverse adjacent contrast operator so that first we are comparing a dosage of 500 with a dosage of 250 , and then we are comparing 750 with 500 . We see that increasing the dosage has a larger effect on females- 10.3 points when going from 250 to 500 compared with 6.4 points for males, and 7.8 points when going from 500 to 750 versus 3.1 points for males.

## Interaction effects

By specifying contrast operators on both factors, we can decompose the interaction effect into separate interaction contrasts.
. contrast ar.dose\#r.gender
Contrasts of marginal linear predictions
Margins: asbalanced

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| :---: | :---: | :---: | :---: |
| dose\#gender |  |  |  |
| (500 vs 250) (Female vs Male) | 1 | 8.87 | 0.0065 |
| (750 vs 500) (Female vs Male) | 1 | 12.91 | 0.0015 |
| Joint | 2 | 21.66 | 0.0000 |
| Denominator | 24 |  |  |


|  | Contrast | Std. err. | [95\% conf. interval] |  |
| ---: | :---: | :---: | :---: | :---: |
| dose\#gender <br> $(500$ vs 250) |  |  |  |  |
| $\left(\begin{array}{r}\text { Female vs Male) } \\ (750 \text { vs 500) }\end{array}\right.$ | -3.910716 | 1.312748 | -6.620095 | -1.201336 |
| $($ Female vs Male) | -4.716567 | 1.312748 | -7.425947 | -2.007187 |

Look for departures from zero to indicate an interaction effect between dose and gender. Both contrasts are significantly different from zero. Of course, we already knew the overall interaction was significant from our ANOVA results. The effect of increasing dose from 250 to 500 is 3.9 points greater in females than in males, and the effect of increasing dose from 500 to 750 is 4.7 points greater in females than in males. The confidence intervals for both estimates easily exclude zero, meaning that there is an interaction effect.

The joint test of these two interaction effects reproduces the test of interaction effects in the anova output. We can see that the $F$ statistic of 21.66 matches the statistic from our original ANOVA results.

## Main effects

We can perform tests of the main effects by listing each variable individually in contrast.

```
. contrast dose gender
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | $F$ | $P>F$ |
| ---: | ---: | ---: | ---: |
| dose | 2 | 223.64 | 0.0000 |
| gender | 1 | 164.85 | 0.0000 |
| Denominator | 24 |  |  |

The $F$ tests are equivalent to the tests of main effects in the anova output. This is true only for linear models. contrast provides an easy way to obtain main effects and other ANOVA-style tests for models whose responses are not linear in the parameters-logistic, probit, glm, etc.

If we include contrast operators on the variables, we can also decompose the main effects into individual contrasts:
. contrast ar.dose r.gender
Contrasts of marginal linear predictions
Margins: asbalanced

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | ---: | ---: | ---: |
| dose |  |  |  |
| (500 vs 250) | 1 | 161.27 | 0.0000 |
| $(750$ vs 500) | 1 | 68.83 | 0.0000 |
| Joint | 2 | 223.64 | 0.0000 |
| gender | 1 | 164.85 | 0.0000 |
| Denominator | 24 |  |  |


|  | Contrast | Std. err. | [95\% conf. interval] |  |
| ---: | ---: | :--- | ---: | ---: |
| dose |  |  |  |  |
| $(500$ vs 250) | -8.335376 | .6563742 | -9.690066 | -6.980687 |
| $(750$ vs 500) | -5.4455 | .6563742 | -6.80019 | -4.090811 |
| gender |  |  |  |  |

By specifying the ar. operator on dose, we decompose the main effect for dose into two one-degree-of-freedom contrasts, comparing the marginal mean of blood pressure change for each dosage level with that of the previous level. Because gender has only two levels, we cannot decompose this main effect any further. However, specifying a contrast operator on gender allowed us to calculate the difference in the marginal means for women and men.

## Partial interaction effects

At this point, we have looked at the total interaction effects and at the main effects of each variable. The partial interaction effects are a midpoint between these two types of effects where we collect the individual interaction effects along the levels of one of the variables and perform a joint test of those interactions. If we think of the interaction effects as forming a table, with the levels of one factor variable forming the rows and the levels of the other forming the columns, partial interaction effects are joint tests of the interactions in a row or a column. To perform these tests, we specify a contrast operator on only one of the variables in our interaction. For this particular model, these are not very interesting because our variables have only two and three levels. Therefore, the tests of the partial interaction effects reproduce the tests that we obtained for the total interaction effects. We specify a contrast operator only on dose to decompose the overall test for interaction effects into joint tests for each ar.dose contrast:

```
. contrast ar.dose#gender
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | ---: | ---: | ---: |
| dose\#gender |  |  |  |
| $(500$ vs 250) (joint) | 1 | 8.87 | 0.0065 |
| $(750$ vs 500) (joint) | 1 | 12.91 | 0.0015 |
| Joint | 2 | 21.66 | 0.0000 |
| Denominator | 24 |  |  |

The first row is a joint test of all the interaction effects involving the ( 500 vs 250 ) comparison of dosages. The second row is a joint test of all the interaction effects involving the ( 750 vs 500 ) comparison. If we look back at our output in Interaction effects, we can see that there was only one of each of these interaction effects. Therefore, each test labeled (joint) has only one degree-of-freedom.

We could have instead included a contrast operator on gender to compute the partial interaction effects along the other dimension:

```
. contrast dose#r.gender
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| :--- | :---: | :---: | ---: |
| dose\#gender | 2 | 21.66 | 0.0000 |
| Denominator | 24 |  |  |

Here we obtain a joint test of all the interaction effects involving the (Female vs Male) comparison for gender. Because gender has only two levels, the (Female vs Male) contrast is the only reference category contrast possible. Therefore, we obtain a single joint test of all the interaction effects.

Clearly, the partial interaction effects are not interesting for this particular model. However, if our factors had more levels, the partial interaction effects would produce tests that are not available in the total interaction effects. For example, if our model included factors for four dosage levels and three races, then typing
. contrast ar.dose\#race
would produce three joint tests, one for each of the reverse adjacent contrasts for dosage. Each of these tests would be a two-degree-of-freedom test because race has three levels.

## Three-way and higher-order models

All the contrasts and tests that we reviewed above for two-way models can be used with models that have more terms. For instance, we could fit a three-way full factorial model by using the anova command:

```
. use https://www.stata-press.com/data/r18/cont3way
. anova y race##sex##group
```

We could then test the simple effects of race within each level of the interaction between sex and group:

```
. contrast race@sex#group
```

To see the reference category contrasts that decompose these simple effects, type

```
. contrast r.race@sex#group
```

We could test the three-way interaction effects by typing

```
. contrast race#sex#group
```

or the interaction effects for the interaction of race and sex by typing

```
. contrast race#sex
```

To see the individual reference category contrasts that decompose this interaction effect, type

```
. contrast r.race#r.sex
```

We could even obtain joint tests for the interaction of race and sex within each level of group by typing

For tests of the main effects of each factor, we can type

```
. contrast race sex group
```

We can calculate the individual reference category contrasts that decompose these main effects:

```
. contrast r.race r.sex r.group
```

For the partial interaction effects, we could type

```
. contrast r.race#group
```

to obtain a joint test of the two-way interaction effects of race and group for each of the individual r.race contrasts.

We could type
. contrast r.race\#sex\#group
to obtain a joint test of all the three-way interaction terms for each of the individual r.race contrasts.

## Contrast operators

contrast recognizes a set of contrast operators that are used to specify commonly used contrasts. When these operators are used, contrast will report a test for each individual contrast in addition to the joint test for the term. We have already seen a few of these, like r. and ar., in the previous examples. Here we will take a closer look at each of the unweighted operators.

Here we use the cholesterol dataset and the one-way ANOVA model from the example in One-way models:

```
. use https://www.stata-press.com/data/r18/cholesterol
(Artificial cholesterol data)
. anova chol agegrp
    (output omitted)
```

The margins command reports the estimated cell means, $\widehat{\mu}_{1}, \ldots, \widehat{\mu}_{5}$, for each of the five age groups.

```
. margins agegrp
Adjusted predictions Number of obs = 75
Expression: Linear prediction, predict()
```

|  | Delta-method <br> std. err. |  |  |  | t | $\mathrm{P}>\|\mathrm{t}\|$ |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: |
|  | Margin | [95\% conf. interval] |  |  |  |  |
| agegrp |  |  |  |  |  |  |
| $10-19$ | 180.5198 | 2.666944 | 67.69 | 0.000 | 175.2007 | 185.8388 |
| $20-29$ | 188.7233 | 2.666944 | 70.76 | 0.000 | 183.4043 | 194.0424 |
| $30-39$ | 202.0608 | 2.666944 | 75.76 | 0.000 | 196.7418 | 207.3799 |
| $40-59$ | 210.6704 | 2.666944 | 78.99 | 0.000 | 205.3514 | 215.9895 |
| $60-79$ | 219.282 | 2.666944 | 82.22 | 0.000 | 213.9629 | 224.601 |

Contrast operators provide an easy way to make certain types of comparisons of these cell means. We use the ordinal factor agegrp to demonstrate these operators because some types of contrasts are meaningful only when the levels of the factor have a natural ordering. We demonstrate these contrast operators using a one-way model; however, they are equally applicable to main effects, simple effects, and interactions for more complicated models.

## Differences from a reference level (r.)

The r. operator specifies that each level of the attached factor variable be compared with a reference level. These are referred to as reference-level or reference-category contrasts (or effects), and $r$. is the reference-level operator.

In the following, we use the r. operator to test the effect of each category of age group when that category is compared with a reference category.

```
. contrast r.agegrp
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | ---: | ---: | ---: |
| agegrp |  |  |  |
| $(20-29$ vs 10-19) | 1 | 4.73 | 0.0330 |
| $(40-39$ vs 10-19) | 1 | 32.62 | 0.0000 |
| $(60-79$ vs 10-19) | 1 | 63.91 | 0.0000 |
| Jo-19) | 1 | 105.62 | 0.0000 |
| Joint | 4 | 35.02 | 0.0000 |
| Denominator | 70 |  |  |


|  | Contrast | Std. err. | [95\% conf. interval] |  |
| :--- | ---: | :--- | ---: | ---: |
| agegrp |  |  |  |  |
| $(20-29$ vs 10-19) | 8.203575 | 3.771628 | .6812991 | 15.72585 |
| $(30-39$ vs 10-19) | 21.54105 | 3.771628 | 14.01878 | 29.06333 |
| $(40-59$ vs 10-19) | 30.15067 | 3.771628 | 22.6284 | 37.67295 |
| $(60-79$ vs 10-19) | 38.76221 | 3.771628 | 31.23993 | 46.28448 |

In the first table, the row labeled (20-29 vs $10-19$ ) is a test of $\mu_{2}=\mu_{1}$, a test that the mean cholesterol levels for the $10-19$ age group and the $20-29$ age group are equal. The tests in the next three rows are defined similarly. The row labeled Joint provides the joint test for these four hypotheses, which is just the test of the main effects of age group.

The second table provides the contrasts of each category with the reference category along with confidence intervals. The contrast in the row labeled (20-29 vs 10-19) is the difference in the cell means of the second age group and the first age group, $\widehat{\mu}_{2}-\widehat{\mu}_{1}$.

The first level of a factor is the default reference level, but we can specify a different reference level by using the b . operator; see [U] 11.4.3.2 Base levels. Here we use the last age group, (60-79), instead of the first as the reference category. We also include the nowald option so that only the table of contrasts and their confidence intervals is produced.

```
. contrast rb5.agegrp, nowald
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | Contrast | Std. err. | [95\% conf. interval] |  |
| :--- | :--- | :--- | :--- | :--- |
| agegrp |  |  |  |  |
| $(10-19$ vs 60-79) | -38.76221 | 3.771628 | -46.28448 | -31.23993 |
| $(20-29$ vs 60-79) | -30.55863 | 3.771628 | -38.08091 | -23.03636 |
| $(30-39$ vs 60-79) | -17.22115 | 3.771628 | -24.74343 | -9.698877 |
| $(40-59$ vs $60-79)$ | -8.611533 | 3.771628 | -16.13381 | -1.089257 |

Now, the first row is labeled ( $10-19$ vs $60-79$ ) and is the difference in the cell means of the first and fifth age groups.

## Differences from the next level (a.)

The a. operator specifies that each level of the attached factor variable be compared with the next level. These are referred to as adjacent contrasts (or effects), and a. is the adjacent operator. This operator is meaningful only with factor variables that have a natural ordering in the levels.

We can use the a. operator to perform tests that each level of age group differs from the next adjacent level.

```
. contrast a.agegrp
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | ---: | ---: | ---: |
| agegrp |  |  |  |
| $(10-19$ vs 20-29) | 1 | 4.73 | 0.0330 |
| $(20-29$ vs 30-39) | 1 | 12.51 | 0.0007 |
| $(40-39$ vs 40-59) | 1 | 5.21 | 0.0255 |
| vs 60-79) | 1 | 5.21 | 0.0255 |
| Joint | 4 | 35.02 | 0.0000 |
| Denominator | 70 |  |  |


|  | Contrast | Std. err. | [95\% conf. interval] |  |
| :--- | ---: | :--- | ---: | ---: |
| agegrp |  |  |  |  |
| $(10-19$ vs 20-29) | -8.203575 | 3.771628 | -15.72585 | -.6812991 |
| $(20-29$ vs 30-39) | -13.33748 | 3.771628 | -20.85976 | -5.815204 |
| $(30-39$ vs 40-59) | -8.60962 | 3.771628 | -16.1319 | -1.087345 |
| $(40-59$ vs 60-79) | -8.611533 | 3.771628 | -16.13381 | -1.089257 |

In the first table, the row labeled (10-19 vs 20-29) tests the effect of belonging to the $10-19$ age group instead of the 20-29 age group. Likewise, the rows labeled ( $20-29$ vs 30-39), (30-39 vs 4059 ), and ( $40-59$ vs $60-79$ ) are tests for the effects of being in the younger of the two age groups instead of the older one.

In the second table, the contrast in the row labeled ( $10-19$ vs $20-29$ ) is the difference in the cell means of the first and second age groups, $\widehat{\mu}_{1}-\widehat{\mu}_{2}$. The contrasts in the other rows are defined similarly.

## Differences from the previous level (ar.)

The ar. operator specifies that each level of the attached factor variable be compared with the previous level. These are referred to as reverse adjacent contrasts (or effects), and ar. is the reverse adjacent operator. As with the a. operator, this operator is meaningful only with factor variables that have a natural ordering in the levels.

In the following, we use the ar. operator to report tests for the individual reverse adjacent effects of agegrp.


Here the Wald tests in the first table for the individual reverse adjacent effects are equivalent to the tests for the adjacent effects in the previous example. However, if we compare values of the contrasts in the bottom tables, we see the difference between the r. and the ar. operators. This time, the contrast in the first row is labeled (20-29 vs 10-19) and is the difference in the cell means of the second and first age groups, $\widehat{\mu}_{2}-\widehat{\mu}_{1}$. This is the estimated effect of belonging to the 20-29 age group instead of the 10-19 age group. The remaining rows make similar comparisons with the previous level.

## Differences from the grand mean (g.)

The $g$. operator specifies that each level of a factor variable be compared with the grand mean of all levels. For this operator, the grand mean is computed using a simple average of the cell means.

Here are the grand mean effects of agegrp:

```
. contrast g.agegrp
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | $F$ | $P>F$ |
| :---: | :---: | ---: | ---: |
| $(10-19$ vs mean) | 1 | 68.42 | 0.0000 |
| $(20-29$ vs mean) | 1 | 23.36 | 0.0000 |
| $(30-39$ vs mean) | 1 | 0.58 | 0.4506 |
| $(40-59$ vs mean) | 1 | 19.08 | 0.0000 |
| $(60-79$ vs mean) | 1 | 63.65 | 0.0000 |
| Joint | 4 | 35.02 | 0.0000 |
| Denominator | 70 |  |  |


|  | Contrast | Std. err. | [95\% conf. interval] |  |
| :--- | ---: | :--- | ---: | ---: |
| agegrp |  |  |  |  |
| $(10-19$ vs mean) | -19.7315 | 2.385387 | -24.48901 | -14.974 |
| $(20-29$ vs mean) | -11.52793 | 2.385387 | -16.28543 | -6.770423 |
| $(30-39$ vs mean) | 1.809552 | 2.385387 | -2.947953 | 6.567057 |
| $(40-59$ vs mean) | 10.41917 | 2.385387 | 5.661668 | 15.17668 |
| $(60-79$ vs mean) | 19.0307 | 2.385387 | 14.2732 | 23.78821 |

There are five age groups in our estimation sample. Thus, the row labeled (10-19 vs mean) tests $\mu_{1}=$ $\left(\mu_{1}+\mu_{2}+\mu_{3}+\mu_{4}+\mu_{5}\right) / 5$. The row labeled (20-29 vs mean) tests $\mu_{2}=\left(\mu_{1}+\mu_{2}+\mu_{3}+\mu_{4}+\mu_{5}\right) / 5$. The remaining rows perform similar tests for the third, fourth, and fifth age groups. In our example, the means for all age groups except the $30-39$ age group are statistically different from the grand mean.

## Differences from the mean of subsequent levels (h.)

The $h$. operator specifies that each level of the attached factor variable be compared with the mean of subsequent levels. These are referred to as Helmert contrasts (or effects), and h. is the Helmert operator. For this operator, the mean is computed using a simple average of the cell means. This operator is meaningful only with factor variables that have a natural ordering in the levels.

Here are the Helmert contrasts for agegrp:

```
. contrast h.agegrp
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | ---: | ---: | ---: |
| agegrp |  |  |  |
| $(10-19$ vs >10-19) | 1 | 68.42 | 0.0000 |
| $(20-29$ vs >20-29) | 1 | 50.79 | 0.0000 |
| $(30-39$ vs >30-39) | 1 | 15.63 | 0.0002 |
| $(40-59$ vs $60-79)$ | 1 | 5.21 | 0.0255 |
| Joint | 4 | 35.02 | 0.0000 |
| Denominator | 70 |  |  |


|  | Contrast | Std. err. | [95\% conf. interval] |  |
| :--- | :--- | :--- | :--- | :--- |
| agegrp |  |  |  |  |
| $(10-19$ vs >10-19) | -24.66438 | 2.981734 | -30.61126 | -18.7175 |
| $(20-29$ vs >20-29) | -21.94774 | 3.079522 | -28.08965 | -15.80583 |
| $(30-39$ vs >30-39) | -12.91539 | 3.266326 | -19.42987 | -6.400905 |
| $(40-59$ vs $60-79)$ | -8.611533 | 3.771628 | -16.13381 | -1.089257 |

The row labeled (10-19 vs >10-19) tests $\mu_{1}=\left(\mu_{2}+\mu_{3}+\mu_{4}+\mu_{5}\right) / 4$, that is, that the cell mean for the youngest age group is equal to the average of the cell means for the older age groups. The row labeled (20-29 vs >20-29) tests $\mu_{2}=\left(\mu_{3}+\mu_{4}+\mu_{5}\right) / 3$. The tests in the other rows are defined similarly.

## Differences from the mean of previous levels (j.)

The $j$. operator specifies that each level of the attached factor variable be compared with the mean of the previous levels. These are referred to as reverse Helmert contrasts (or effects), and $j$. is the reverse Helmert operator. For this operator, the mean is computed using a simple average of the cell means. This operator is meaningful only with factor variables that have a natural ordering in the levels.

Here are the reverse Helmert contrasts of agegrp:

```
. contrast j.agegrp
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| :---: | :---: | :---: | :---: |
| agegrp |  |  |  |
| (20-29 vs 10-19) | 1 | 4.73 | 0.0330 |
| (30-39 vs <30-39) | 1 | 28.51 | 0.0000 |
| (40-59 vs <40-59) | 1 | 43.18 | 0.0000 |
| (60-79 vs <60-79) | 1 | 63.65 | 0.0000 |
| Joint | 4 | 35.02 | 0.0000 |
| Denominator | 70 |  |  |


|  | Contrast | Std. err. | [95\% conf. interval] |  |
| :--- | ---: | :--- | ---: | ---: |
| agegrp |  |  |  |  |
| $(20-29$ vs 10-19) | 8.203575 | 3.771628 | .6812991 | 15.72585 |
| $(30-39$ vs <30-39) | 17.43927 | 3.266326 | 10.92479 | 23.95375 |
| $(40-59$ vs <40-59) | 20.2358 | 3.079522 | 14.09389 | 26.37771 |
| $(60-79$ vs <60-79) | 23.78838 | 2.981734 | 17.8415 | 29.73526 |

The row labeled (20-29 vs $10-19$ ) tests $\mu_{2}=\mu_{1}$, that is, that the cell means for the $20-29$ and the $10-19$ age groups are equal. The row labeled (30-39 vs <30-29) tests $\mu_{3}=\left(\mu_{1}+\mu_{2}\right) / 2$, that is, that the cell mean for the $30-39$ age group is equal to the average of the cell means for the $10-19$ and $20-29$ age groups. The tests in the remaining rows are defined similarly.

## Orthogonal polynomials (p. and q.)

The p. and q. operators specify that orthogonal polynomials be applied to the attached factor variable. Orthogonal polynomial contrasts allow us to partition the effects of a factor variable into linear, quadratic, cubic, and higher-order polynomial components. The p. operator applies orthogonal polynomials using the values of the factor variable. The $q$. operator applies orthogonal polynomials using the level indices. If the level values of the factor variable are equally spaced, as with our agegrp variable, then the p. and q. operators yield the same result. These operators are meaningful only with factor variables that have a natural ordering in the levels.

Because agegrp has five levels, contrast can test the linear, quadratic, cubic, and quartic effects of agegrp.

```
. contrast p.agegrp, noeffects
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| :---: | ---: | ---: | ---: |
| agegrp |  |  |  |
| (linear) | 1 | 139.11 | 0.0000 |
| (quadratic) | 1 | 0.15 | 0.6962 |
| (cubic) | 1 | 0.37 | 0.5448 |
| (quartic) | 1 | 0.43 | 0.5153 |
| Joint | 4 | 35.02 | 0.0000 |
|  |  |  |  |
| Denominator | 70 |  |  |

The row labeled (linear) tests the linear effect of agegrp, the only effect that appears to be significant in this case.

The labels for our agegrp variable show the age ranges that correspond to each level.

```
. label list ages
ages:
    1 10-19
    2 20-29
    3 30-39
    4 40-59
    5 60-79
```

Notice that these groups do not have equal widths. Now, let's refit our model using the agemidpt variable. The values of agemidpt indicate the midpoint of each age group that was defined by the agegrp variable and are, therefore, not equally spaced.
. anova chol agemidpt

|  | $\begin{aligned} \text { Number of obs } & = \\ \text { Root MSE } & = \end{aligned}$ | 10.3 | R -squared |  | $\begin{aligned} & 0.6668 \\ & 0.6477 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | 14943.4 | 4 | 3735.8499 | 35.02 | 0.0000 |
| agemidpt | 14943.4 | 4 | 3735.8499 | 35.02 | 0.0000 |
| Residual | 7468.2197 | 70 | 106.68885 |  |  |
| Total | 22411.619 | 74 | 302.85972 |  |  |

Now if we use the q. operator, we will obtain the same results as above because the level indices of agemidpt are equivalent to the values of agegrp.

```
. contrast q.agemidpt, noeffects
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| :---: | ---: | ---: | ---: |
| agemidpt |  |  |  |
| (linear) | 1 | 139.11 | 0.0000 |
| (quadratic) | 1 | 0.15 | 0.6962 |
| (cubic) | 1 | 0.37 | 0.5448 |
| (quartic) | 1 | 0.43 | 0.5153 |
| Joint | 4 | 35.02 | 0.0000 |
| Denominator | 70 |  |  |

However, if we use the p. operator, we will instead fit an orthogonal polynomial to the midpoint values.

```
. contrast p.agemidpt, noeffects
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| :---: | :---: | ---: | ---: |
| agemidpt |  |  |  |
| (linear) | 1 | 133.45 | 0.0000 |
| (quadratic) | 1 | 5.40 | 0.0230 |
| (cubic) | 1 | 0.05 | 0.8198 |
| (quartic) | 1 | 1.16 | 0.2850 |
| Joint | 4 | 35.02 | 0.0000 |
| Denominator | 70 |  |  |

Using the values of the midpoints, the quadratic effect is also significant at the $5 \%$ level.

## - Technical note

We used the noeffects option when working with orthogonal polynomial contrasts. Apart from perhaps the sign of the contrast, the values of the individual contrasts are not meaningful for orthogonal polynomial contrasts. In addition, many textbooks provide tables with contrast coefficients that can be used to compute orthogonal polynomial contrasts where the levels of a factor are equally spaced. If
we use these coefficients and calculate the contrasts manually with user-defined contrasts, as described below, the Wald tests for the polynomial terms will be equivalent, but the values of the individual contrasts will not necessarily match those that we obtain when using the polynomial contrast operator. When we use one of these contrast operators, an algorithm is used to calculate the coefficients of the polynomial contrast that will allow for unequal spacing in the levels of the factor as well as in the weights for the cell frequencies (when using pw. or qw.), as described in Methods and formulas.

## User-defined contrasts

In the previous examples, we performed tests using contrast operators. When there is not a contrast operator available to calculate the contrast in which we are interested, we can specify custom contrasts.

Here we fit a one-way model for cholesterol on the factor race, which has three levels:

margins calculates the estimated cell mean cholesterol level for each race:
. margins race
Adjusted predictions $\quad$ Number of obs $=75$
Expression: Linear prediction, predict()

|  | MarginDelta-method <br> std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| race |  |  |  |  |  |  |
| Black | 204.4279 | 3.475497 | 58.82 | 0.000 | 197.4996 | 211.3562 |
| White | 197.6132 | 3.475497 | 56.86 | 0.000 | 190.6849 | 204.5415 |
| Other | 198.7127 | 3.475497 | 57.18 | 0.000 | 191.7844 | 205.6409 |

Suppose we want to test the following linear combination:

$$
\sum_{i=1}^{3} c_{i} \mu_{i}
$$

where $\mu_{i}$ is the cell mean of chol when race is equal to its $i$ th level (the means estimated using margins above). Assuming the $c_{i}$ elements sum to zero, this linear combination is a contrast. We can specify this type of custom contrast by using the following syntax:

$$
\left\{\text { race } c_{1} c_{2} c_{3}\right\}
$$

The null hypothesis for the test of the main effects of race is

$$
H_{0_{\text {race }}}: \mu_{1}=\mu_{2}=\mu_{3}
$$

Although $H_{0_{\text {race }}}$ can be tested using any of several different contrasts on the cell means, we will test it by comparing the second and third cell means with the first. To test that the cell means for blacks and whites are equal, $\mu_{1}=\mu_{2}$, we can specify the contrast

$$
\{\text { race -1 } 100\}
$$

To test that the cell means for blacks and other races are equal, $\mu_{1}=\mu_{3}$, we can specify the contrast

$$
\{\text { race }-101\}
$$

We can use both in a single call to contrast.
. contrast \{race -110$\}$ \{race -101$\}$
Contrasts of marginal linear predictions
Margins: asbalanced

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | :---: | :---: | :---: |
| race |  |  |  |
| (1) | 1 | 1.92 | 0.1699 |
| (2) | 1 | 1.35 | 0.2488 |
| Joint | 2 | 1.11 | 0.3357 |
| Denominator | 72 |  |  |


|  | Contrast | Std. err. | [95\% conf. interval] |  |
| :--- | :--- | :--- | :--- | :--- |
| race |  |  |  |  |
| $(1)$ | -6.814717 | 4.915095 | -16.61278 | 2.983345 |
| $(2)$ | -5.715261 | 4.915095 | -15.51332 | 4.082801 |

The row labeled (1) is the test for $\mu_{1}=\mu_{2}$, the first specified contrast. The row labeled (2) is the test for $\mu_{1}=\mu_{3}$, the second specified contrast. The row labeled Joint is the overall test for the main effects of race.

Now, let's fit a model with two factors, race and age group:
. anova chol race\#\#agegrp

|  | $\begin{aligned} \text { Number of obs } & = \\ \text { Root MSE } & = \end{aligned}$ | $\begin{array}{r} 75 \\ 9.61785 \end{array}$ | R-squared |  | $\begin{aligned} & 0.7524 \\ & 0.6946 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | 16861.438 | 14 | 1204.3884 | 13.02 | 0.0000 |
| race | 669.27823 | 2 | 334.63912 | 3.62 | 0.0329 |
| agegrp | 14943.4 | 4 | 3735.8499 | 40.39 | 0.0000 |
| race\#agegrp | 1248.7601 | 8 | 156.09501 | 1.69 | 0.1201 |
| Residual | 5550.1814 | 60 | 92.503024 |  |  |
| Total | 22411.619 | 74 | 302.85972 |  |  |

The null hypothesis for the test of the main effects of race is now

$$
H_{0_{\text {race }}}: \mu_{1}=\mu_{2}=\mu_{3} .
$$

where $\mu_{i}$. is the marginal mean of chol when race is equal to its $i$ th level.
We can use the same syntax as above to perform this test by specifying contrasts on the marginal means of race:

```
. contrast {race -1 1 0} {race -1 0 1}
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | :---: | :---: | :---: |
| race |  |  |  |
| $(1)$ | 1 | 6.28 | 0.0150 |
| (2) | 1 | 4.41 | 0.0399 |
| Joint | 2 | 3.62 | 0.0329 |
| Denominator | 60 |  |  |


|  | Contrast | Std. err. | [95\% conf. interval] |  |
| :--- | ---: | :--- | ---: | ---: |
| race |  |  |  |  |
| $(1)$ | -6.814717 | 2.720339 | -12.2562 | -1.37323 |
| $(2)$ | -5.715261 | 2.720339 | -11.15675 | -.2737739 |

Custom contrasts may be specified on the cell means of interactions, too. Here we use margins to calculate the mean of chol for each cell in the interaction of race and agegrp:
. margins race\#agegrp
Adjusted predictions $\quad$ Number of obs $=75$
Expression: Linear prediction, predict()

|  | Delta-method |  | t | $P>\|t\|$ | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| race\#agegrp |  |  |  |  |  |  |
| Black\#10-19 | 179.2309 | 4.301233 | 41.67 | 0.000 | 170.6271 | 187.8346 |
| Black\#20-29 | 196.4777 | 4.301233 | 45.68 | 0.000 | 187.874 | 205.0814 |
| Black\#30-39 | 210.6694 | 4.301233 | 48.98 | 0.000 | 202.0656 | 219.2731 |
| Black\#40-59 | 214.097 | 4.301233 | 49.78 | 0.000 | 205.4933 | 222.7008 |
| Black\#60-79 | 221.6646 | 4.301233 | 51.54 | 0.000 | 213.0609 | 230.2684 |
| White\#10-19 | 186.0727 | 4.301233 | 43.26 | 0.000 | 177.469 | 194.6765 |
| White\#20-29 | 184.6714 | 4.301233 | 42.93 | 0.000 | 176.0676 | 193.2751 |
| White\#30-39 | 196.2633 | 4.301233 | 45.63 | 0.000 | 187.6595 | 204.867 |
| White\#40-59 | 209.9953 | 4.301233 | 48.82 | 0.000 | 201.3916 | 218.5991 |
| White\#60-79 | 211.0633 | 4.301233 | 49.07 | 0.000 | 202.4595 | 219.667 |
| Other\#10-19 | 176.2556 | 4.301233 | 40.98 | 0.000 | 167.6519 | 184.8594 |
| Other\#20-29 | 185.0209 | 4.301233 | 43.02 | 0.000 | 176.4172 | 193.6247 |
| Other\#30-39 | 199.2498 | 4.301233 | 46.32 | 0.000 | 190.646 | 207.8535 |
| Other\#40-59 | 207.9189 | 4.301233 | 48.34 | 0.000 | 199.3152 | 216.5227 |
| Other\#60-79 | 225.118 | 4.301233 | 52.34 | 0.000 | 216.5143 | 233.7218 |

Now, we are interested in testing the following linear combination of these cell means:

$$
\sum_{i=1}^{3} \sum_{j=1}^{5} c_{i j} \mu_{i j}
$$

We can specify this type of custom contrast using the following syntax:

$$
\left\{\text { race\#agegrp } c_{11} c_{12} \ldots c_{15} c_{21} c_{22} \ldots c_{25} c_{31} c_{32} \ldots c_{35}\right\}
$$

Because the marginal means of chol for each level of race are linear combinations of the cell means, we can compose the test for the main effects of race in terms of the cell means directly. The constraint that the marginal means for blacks and whites are equal, $\mu_{1} .=\mu_{2}$., translates to the following constraint on the cell means:

$$
\frac{1}{5}\left(\mu_{11}+\mu_{12}+\mu_{13}+\mu_{14}+\mu_{15}\right)=\frac{1}{5}\left(\mu_{21}+\mu_{22}+\mu_{23}+\mu_{24}+\mu_{25}\right)
$$

Ignoring the common factor, we can specify this contrast as

$$
\{\text { race\#agegrp -1 - } 1-1-1-1111111111000000\}
$$

contrast will fill in the trailing zeros for us if we neglect to specify them, so

$$
\{\text { race\#agegrp -1 -1 -1 -1 }-1
$$

is also allowed. The other constraint, $\mu_{1}=\mu_{3 .}$, translates to

$$
\frac{1}{5}\left(\mu_{11}+\mu_{12}+\mu_{13}+\mu_{14}+\mu_{15}\right)=\frac{1}{5}\left(\mu_{31}+\mu_{32}+\mu_{33}+\mu_{34}+\mu_{35}\right)
$$

This can be specified to contrast as

```
\{race\#agegrp - \(1-1-1-1-10000011111\}\)
```

The following call to contrast yields the same test results as above.

```
. contrast {race#agegrp -1 -1 -1 -1 -1 1 1 1 1 1 1}
> {race#agegrp -1 -1 -1 -1 -1 0 0 0 0 0 0 1 1 1 1 1 1}, noeffects
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| :---: | :---: | :---: | :---: |
| race\#agegrp |  |  |  |
| (1) (1) | 1 | 6.28 | 0.0150 |
| (2) (2) | 1 | 4.41 | 0.0399 |
| Joint | 2 | 3.62 | 0.0329 |
| Denominator | 60 |  |  |

The row labeled (1) (1) is the test for

$$
\mu_{11}+\mu_{12}+\mu_{13}+\mu_{14}+\mu_{15}=\mu_{21}+\mu_{22}+\mu_{23}+\mu_{24}+\mu_{25}
$$

It was the first specified contrast. The row labeled (2) (2) is the test for

$$
\mu_{11}+\mu_{12}+\mu_{13}+\mu_{14}+\mu_{15}=\mu_{31}+\mu_{32}+\mu_{33}+\mu_{34}+\mu_{35}
$$

It was the second specified contrast. The row labeled Joint tests (1) (1) and (2) (2) simultaneously.
We used the noeffects option above to suppress the table of contrasts. We can omit the $1 / 5$ from the equations for $\mu_{1}=\mu_{2}$. and $\mu_{1}=\mu_{3}$. and still obtain the appropriate tests. However, if we want to calculate the differences in the marginal means, we must include the $1 / 5=0.2$ on each of the contrast coefficients as follows:

```
. contrast {race#agegrp -0.2 -0.2 -0.2 -0.2 -0.2 ///
    0.2
    {race#agegrp -0.2 -0.2 -0.2 -0.2 -0.2 ///
    0}00000001/
    0.2
```

So far, we have reproduced the reference category contrasts by specifying user-defined contrasts on the marginal means and then on the cell means. For this test, it would have been easier to use the r. contrast operator:

```
. contrast r.race, noeffects
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | :---: | :---: | ---: |
| race |  |  |  |
| (White vs Black) | 1 | 6.28 | 0.0150 |
| (Other vs Black) | 1 | 4.41 | 0.0399 |
| Joint | 2 | 3.62 | 0.0329 |
| Denominator | 60 |  |  |

In most cases, we can use contrast operators to perform tests. However, if we want to compare, for instance, the second and third age groups with the fourth and fifth age groups with the test

$$
\frac{1}{2}\left(\mu_{\cdot 2}+\mu \cdot 3\right)=\frac{1}{2}\left(\mu_{\cdot 4}+\mu_{\cdot 5}\right)
$$

there is not a contrast operator that corresponds to this particular contrast. A custom contrast is necessary.
. contrast \{agegrp $0-0.5-0.50 .50 .5\}$
Contrasts of marginal linear predictions
Margins: asbalanced

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | ---: | ---: | ---: |
| agegrp | 1 | 62.19 | 0.0000 |
| Denominator | 60 |  |  |


|  | Contrast | Std. err. | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: |
| agegrp <br> $(1)$ | 19.58413 | 2.483318 | 14.61675 | 24.5515 |

## Empty cells

An empty cell is a combination of the levels of factor variables that is not observed in the estimation sample. In the previous examples, we have seen data with three levels of race, five levels of agegrp, and all level combinations of race and agegrp present. Suppose there are no observations for white individuals in the second age group (ages 20-29).

```
. use https://www.stata-press.com/data/r18/cholesterol2
(Artificial cholesterol data, empty cells)
. label list
race:
    1 Black
    2 White
    3 Other
ages:
    1 10-19
    2 20-29
    3 30-39
    4 40-59
    5 60-79
```

. regress chol race\#\#agegrp
note: 2.race\#2.agegrp identifies no observations in the sample.

| Source | SS | df | MS | Number of obs | = | 70 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | $\mathrm{F}(13,56)$ | = | 13.51 |
| Model | 15751.6113 | 13 | 1211.66241 | Prob > F | = | 0.0000 |
| Residual | 5022.71559 | 56 | 89.6913498 | R -squared | = | 0.7582 |
|  |  |  |  | Adj R-squared | = | 0.7021 |
| Total | 20774.3269 | 69 | 301.077201 | Root MSE | = | 9.4706 |


| chol | Coefficient | Std. err. | t | $P>\|t\|$ | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| race |  |  |  |  |  |  |
| White | 12.84185 | 5.989703 | 2.14 | 0.036 | . 8430383 | 24.84067 |
| Other | -. 167627 | 5.989703 | -0.03 | 0.978 | -12.16644 | 11.83119 |
| agegrp |  |  |  |  |  |  |
| 20-29 | 17.24681 | 5.989703 | 2.88 | 0.006 | 5.247991 | 29.24562 |
| 30-39 | 31.43847 | 5.989703 | 5.25 | 0.000 | 19.43966 | 43.43729 |
| 40-59 | 34.86613 | 5.989703 | 5.82 | 0.000 | 22.86732 | 46.86495 |
| 60-79 | 44.43374 | 5.989703 | 7.42 | 0.000 | 32.43492 | 56.43256 |
| race\#agegrp |  |  |  |  |  |  |
| White\#20-29 | 0 | (empty) |  |  |  |  |
| White\#30-39 | -22.83983 | 8.470719 | -2.70 | 0.009 | -39.80872 | -5.870939 |
| White\#40-59 | -14.67558 | 8.470719 | -1.73 | 0.089 | -31.64447 | 2.293306 |
| White\#60-79 | -10.51115 | 8.470719 | -1.24 | 0.220 | -27.48004 | 6.457735 |
| Other\#20-29 | -6.054425 | 8.470719 | -0.71 | 0.478 | -23.02331 | 10.91446 |
| Other\#30-39 | -11.48083 | 8.470719 | -1.36 | 0.181 | -28.44971 | 5.488063 |
| Other\#40-59 | -. 6796112 | 8.470719 | -0.08 | 0.936 | -17.6485 | 16.28928 |
| Other\#60-79 | -1.578052 | 8.470719 | -0.19 | 0.853 | -18.54694 | 15.39084 |
| _cons | 175.2309 | 4.235359 | 41.37 | 0.000 | 166.7464 | 183.7153 |

Now, let's use contrast to test the main effects of race:
. contrast race
Contrasts of marginal linear predictions
Margins: asbalanced

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | :---: | :---: | :---: |
| race | (not testable) |  |  |
| Denominator | 56 |  |  |

By "not testable", contrast means that it cannot form a test for the main effects of race based on estimable functions of the model coefficients. agegrp has five levels, so contrast constructs an estimate of the $i$ th margin for race as

$$
\widehat{\mu}_{i}=\frac{1}{5} \sum_{j=1}^{5} \widehat{\mu}_{i j}=\widehat{\mu}_{0}+\widehat{\alpha}_{i}+\frac{1}{5} \sum_{j=1}^{5}\left\{\widehat{\beta}_{j}+(\widehat{\alpha \beta})_{i j}\right\}
$$

but $(\widehat{\alpha \beta})_{22}$ was constrained to zero because of the empty cell, so $\widehat{\mu}_{2}$. is not an estimable function of the model coefficients.

See Estimable functions in Methods and formulas of [R] margins for a technical description of estimable functions. The emptycells (reweight) option causes contrast to estimate $\mu_{2}$. by

$$
\widehat{\mu}_{2 .}=\frac{\widehat{\mu}_{21}+\widehat{\mu}_{23}+\widehat{\mu}_{24}+\widehat{\mu}_{25}}{4}
$$

which is an estimable function of the model coefficients.

```
. contrast race, emptycells(reweight)
Contrasts of marginal linear predictions
Margins: asbalanced
Empty cells: reweight
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | ---: | ---: | ---: |
| race | 2 | 3.17 | 0.0498 |
| Denominator | 56 |  |  |

We can reconstruct the effect of the emptycells (reweight) option by using custom contrasts.

```
. contrast {race#agegrp -4 -4 -4 -4 -4 5 0 5 5 5}
> {race#agegrp -1 -1 -1 -1 -1 0 0 0 0 0 0 1 1 1 1 1 1}, noeffects
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | :---: | :---: | :---: |
| race\#agegrp |  |  |  |
| (1) (1) | 1 | 1.06 | 0.3080 |
| (2) (2) | 1 | 2.37 | 0.1291 |
| Joint | 2 | 3.17 | 0.0498 |
| Denominator | 56 |  |  |

The row labeled (1) (1) is the test for

$$
\frac{1}{5}\left(\mu_{11}+\mu_{12}+\mu_{13}+\mu_{14}+\mu_{15}\right)=\frac{1}{4}\left(\mu_{21}+\mu_{23}+\mu_{24}+\mu_{25}\right)
$$

It was the first specified contrast. The row labeled (2) (2) is the test for

$$
\mu_{11}+\mu_{12}+\mu_{13}+\mu_{14}+\mu_{15}=\mu_{31}+\mu_{32}+\mu_{33}+\mu_{34}+\mu_{35}
$$

It was the second specified contrast. The row labeled Joint is the overall test of the main effects of race.

## Empty cells, ANOVA style

Let's refit the linear model from the previous example with anova to compare with contrast's test for the main effects of race.

|  | $\begin{aligned} \text { Number of obs } & = \\ \text { Root MSE } & = \end{aligned}$ | $\begin{array}{r} 70 \\ 9.47055 \end{array}$ | R-squared = <br> Adj R-squared $=$ |  | $\begin{aligned} & 0.7582 \\ & 0.7021 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | 15751.611 | 13 | 1211.6624 | 13.51 | 0.0000 |
| race | 305.49046 | 2 | 152.74523 | 1.70 | 0.1914 |
| agegrp | 14387.856 | 4 | 3596.964 | 40.10 | 0.0000 |
| race\#agegrp | 795.80757 | 7 | 113.6868 | 1.27 | 0.2831 |
| Residual | 5022.7156 | 56 | 89.69135 |  |  |
| Total | 20774.327 | 69 | 301.0772 |  |  |

contrast and anova handled the empty cell differently; the $F$ statistic reported by contrast was 3.17 , but anova reported 1.70 . To see how they differ, consider the following table of the cell means and margins for our situation.

|  |  | agegrp |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  | 1 | 2 | 3 | 4 | 5 |  |
|  | 1 | $\mu_{11}$ | $\mu_{12}$ | $\mu_{13}$ | $\mu_{14}$ | $\mu_{15}$ | $\mu_{1}$. |
| race | 2 | $\mu_{21}$ |  | $\mu_{23}$ | $\mu_{24}$ | $\mu_{25}$ |  |
|  | 3 | $\mu_{31}$ | $\mu_{32}$ | $\mu_{33}$ | $\mu_{34}$ | $\mu_{35}$ | $\mu_{3}$. |
|  |  | $\mu_{\cdot 1}$ |  | $\mu_{\cdot 3}$ | $\mu_{\cdot 4}$ | $\mu_{\cdot 5}$ |  |

For testing the main effects of race, we know that we will be testing the equality of the marginal means for rows 1 and 3 , that is, $\mu_{1}=\mu_{3}$. This translates into the following constraint:

$$
\mu_{11}+\mu_{12}+\mu_{13}+\mu_{14}+\mu_{15}=\mu_{31}+\mu_{32}+\mu_{33}+\mu_{34}+\mu_{35}
$$

Because row 2 contains an empty cell in column 2, anova dropped column 2 and tested the equality of the marginal mean for row 2 with the average of the marginal means from rows 1 and 3 , using only the remaining cell means. This translates into the following constraint:

$$
\begin{equation*}
2\left(\mu_{21}+\mu_{23}+\mu_{24}+\mu_{25}\right)=\mu_{11}+\mu_{13}+\mu_{14}+\mu_{15}+\mu_{31}+\mu_{33}+\mu_{34}+\mu_{35} \tag{1}
\end{equation*}
$$

Now that we know the constraints that anova used to test for the main effects of race, we can use custom contrasts to reproduce the anova test result.

```
. contrast {race#agegrp -1 -1 -1 -1 -1 0 0 0 0 0 0 1 1 1 1 1 1}
> {race#agegrp 1 0 0 1 1 1 1 -2 0 -2 --2 -2 1 0 1 1 1 1}, noeffects
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | :---: | :---: | :---: |
| race\#agegrp |  |  |  |
| (1) (1) | 1 | 2.37 | 0.1291 |
| (2) (2) | 1 | 1.03 | 0.3138 |
| Joint | 2 | 1.70 | 0.1914 |
| Denominator | 56 |  |  |

The row labeled (1) (1) is the test for $\mu_{1}=\mu_{3}$.; it was the first specified contrast. The row labeled (2) (2) is the test for the constraint in (1); it was the second specified contrast. The row labeled Joint is an overall test for the main effects of race.

## Nested effects

contrast has the $\mid$ operator for computing simple effects when the levels of one factor are nested within the levels of another. Here is a fictional example where we are interested in the effect of five methods of teaching algebra on students' scores for the math portion of the SAT. Suppose three algebra classes are randomly sampled from classes using each of the five methods so that class is nested in method as demonstrated in the following tabulation.
. use https://www.stata-press.com/data/r18/sat
(Fictional SAT data)
• tabulate class method
Class ID

We will consider method as fixed and class nested in method as random. To use class nested in method as the error term for method, we can specify the following anova model:

|  | $\begin{array}{ll} \text { Number of obs } & = \\ \text { Root MSE } & = \end{array}$ | $\begin{array}{r} 75 \\ 71.8517 \end{array}$ | R-squared = <br> Adj R-squared = |  | $\begin{aligned} & 0.7599 \\ & 0.7039 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | 980312 | 14 | 70022.286 | 13.56 | 0.0000 |
| method | 905872 | 4 | 226468 | 30.42 | 0.0000 |
| class\|method | 74440 | 10 | 7444 |  |  |
| class/method | 74440 | 10 | 7444 | 1.44 | 0.1845 |
| Residual | 309760 | 60 | 5162.6667 |  |  |
| Total | 1290072 | 74 | 17433.405 |  |  |

Like anova, contrast allows the | operator, which specifies that one variable is nested in the levels of another. We can use contrast to test the main effects of method and the simple effects of class within method.
. contrast method class|method
Contrasts of marginal linear predictions
Margins: asbalanced

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | :---: | :---: | ---: |
| method | (not testable) |  |  |
| classlmethod |  |  |  |
| 1 |  | 2.80 | 0.0687 |
| 2 | 2 | 0.91 | 0.4089 |
| 3 | 2 | 1.10 | 0.3390 |
| 4 | 2 | 0.22 | 0.8025 |
| 5 | 2 | 2.18 | 0.1221 |
| Joint | 2 | 1.44 | 0.1845 |
|  | 10 |  |  |
| Denominator |  |  |  |

Although contrast was able to perform the individual tests for the simple effects of class within method, empty cells in the interaction between method and class prevented contrast from testing for a main effect of method. Here we add the emptycells (reweight) option so that contrast can take the empty cells into account when computing the marginal means for method.

```
. contrast method class|method, emptycells(reweight)
Contrasts of marginal linear predictions
Margins: asbalanced
Empty cells: reweight
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | ---: | ---: | ---: |
| method | 4 | 43.87 | 0.0000 |
| class\|method |  |  |  |
| 1 | 2 | 2.80 | 0.0687 |
| 2 | 2 | 0.91 | 0.4089 |
| 3 | 2 | 1.10 | 0.3390 |
| 4 | 2 | 0.22 | 0.8025 |
| 5 | 2 | 2.18 | 0.1221 |
| Joint | 10 | 1.44 | 0.1845 |
|  |  |  |  |
| Denominator | 60 |  |  |

Now, contrast does report a test for the main effects of method. However, if we compare this with the anova results, we will see that the results are different. They are different because contrast uses the residual error term to compute the $F$ test by default. Using notation similar to anova, we can use the / operator to specify a different error term for the test. Therefore, we can reproduce the test of main effects from our anova command by typing

```
. contrast method / class|method /, emptycells(reweight)
Contrasts of marginal linear predictions
Margins: asbalanced
Empty cells: reweight
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | ---: | :---: | ---: |
| method | 4 | 30.42 | 0.0000 |
| class\|method | 10 | (denominator) |  |
| class\|method |  |  |  |
| 1 | 2 | 2.80 | 0.0687 |
| 2 | 2 | 0.91 | 0.4089 |
| 3 | 2 | 1.10 | 0.3390 |
| 4 | 2 | 0.22 | 0.8025 |
| 5 | 2 | 2.18 | 0.1221 |
| Joint | 10 | 1.44 | 0.1845 |
|  |  |  |  |
| Denominator | 60 |  |  |

## Multiple comparisons

We have seen that contrast can report the individual linear combinations that make up the requested effects. Depending upon the specified option, contrast will report confidence intervals, $p$-values, or both in the effects table. By default, the reported confidence intervals and $p$-values are not adjusted for multiple comparisons. Use the mcompare () option to adjust the confidence intervals and $p$-values for multiple comparisons of the individual effects.

Let's compute the grand mean effects of race using the g . operator. We also specify the mcompare(bonferroni) option to compute $p$-values and confidence intervals using Bonferroni's adjustment.

```
- use https://www.stata-press.com/data/r18/cholesterol
(Artificial cholesterol data)
- anova chol race\#\#agegrp
    (output omitted)
. contrast g.race, mcompare(bonferroni)
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | Bonferroni |  |
| :---: | :---: | :---: | :---: | :---: |
|  |  |  | $\mathrm{P}>\mathrm{F}$ | $\mathrm{P}>\mathrm{F}$ |
| race |  |  |  |  |
| (Black vs mean) | 1 | 7.07 | 0.0100 | 0.0301 |
| (White vs mean) | 1 | 2.82 | 0.0982 | 0.2947 |
| (Other vs mean) | 1 | 0.96 | 0.3312 | 0.9936 |
| Joint | 2 | 3.62 | 0.0329 |  |
| Denominator | 60 |  |  |  |

Note: Bonferroni-adjusted p-values are reported for tests on individual contrasts only.

|  | Number of <br> comparisons |
| ---: | ---: |
| race | 3 |


|  |  |  | Bonferroni <br> [95\% conf. interval] |  |
| :--- | ---: | ---: | ---: | ---: |
|  | Contrast | Std. err. |  |  |
| race |  |  |  |  |
| (Black vs mean) | 4.17666 | 1.570588 | .3083743 | 8.044945 |
| (White vs mean) | -2.638058 | 1.570588 | -6.506343 | 1.230227 |
| (Other vs mean) | -1.538602 | 1.570588 | -5.406887 | 2.329684 |

The last table reports a Bonferroni-adjusted confidence interval for each individual contrast. (Use the effects option to add $p$-values to the last table.) The first table includes a Bonferroni-adjusted $p$-value for each test that is not a joint test.

Joint tests are never adjusted for multiple comparisons. For example,

```
. contrast race@agegrp, mcompare(bonferroni)
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| :---: | :---: | :---: | :---: |
| race@agegrp |  |  |  |
| $10-19$ | 2 | 1.37 | 0.2620 |
| $20-29$ | 2 | 2.44 | 0.0958 |
| $30-39$ | 2 | 3.12 | 0.0512 |
| $40-59$ | 2 | 0.53 | 0.5889 |
| 60-79 | 2 | 2.90 | 0.0628 |
| Joint | 10 | 2.07 | 0.0409 |
| Denominator | 60 |  |  |

Note: Bonferroni-adjusted p-values are reported for tests on individual contrasts only.

|  | Number of <br> comparisons |
| ---: | ---: |
| race@agegrp | 10 |


|  |  |  | Bonferroni <br> [95\% conf. interval] |  |
| :--- | ---: | ---: | ---: | ---: |
|  | Contrast | Std. err. |  |  |
| (White vs base) $10-19$ | 6.841855 | 6.082862 | -10.88697 | 24.57068 |
| (White vs base) 20-29 | -11.80631 | 6.082862 | -29.53513 | 5.922513 |
| (White vs base) 30-39 | -14.40607 | 6.082862 | -32.13489 | 3.322751 |
| (White vs base) 40-59 | -4.101691 | 6.082862 | -21.83051 | 13.62713 |
| (White vs base) 60-79 | -10.60137 | 6.082862 | -28.33019 | 7.127448 |
| (Other vs base) 10-19 | -2.975244 | 6.082862 | -20.70407 | 14.75358 |
| (Other vs base) 20-29 | -11.45679 | 6.082862 | -29.18561 | 6.272031 |
| (Other vs base) 30-39 | -11.41958 | 6.082862 | -29.1484 | 6.309244 |
| (Other vs base) 40-59 | -6.17807 | 6.082862 | -23.90689 | 11.55075 |
| (Other vs base) $60-79$ | 3.453375 | 6.082862 | -14.27545 | 21.1822 |

Here we have five tests of simple effects with two degrees of freedom each. No Bonferroni-adjusted $p$-values are available for these tests, but the confidence intervals for the individual contrasts are adjusted.

## Unbalanced data

By default, contrast treats all factors as balanced when computing marginal means. By balanced, we mean that contrast assumes an equal number of observations in each level of each factor and an equal number of observations in each cell of each interaction. If our data are balanced, there is no issue. If, however, our data are not balanced, we might prefer that contrast use the actual cell frequencies from our data in computing marginal means. We instruct contrast to use observed frequencies by adding the asobserved option.

Even if our data are unbalanced, we might still want contrast to compute balanced marginal means. It depends on what we want to test and what our data represent. If we have data from a designed
experiment that started with an equal number of males and females but the data became unbalanced because the data from a few males were unusable, we might still want our margins computed as though the data were balanced. If, however, we have a representative sample of individuals from Los Angeles with 40\% of European descent, 34\% African-American, 25\% Hispanic, and 1\% Australian, we probably want our margins computed using these representative frequencies. We do not want Australians receiving the same weight as Europeans.

The following examples will use an unbalanced version of our dataset.

```
. use https://www.stata-press.com/data/r18/cholesterol3
(Artificial cholesterol data, unbalanced)
```

. tab race agegrp

| Race | 10-19 | $20-29$ | Age group <br> $30-39$ | $40-59$ | $60-79$ | Total |
| :---: | ---: | ---: | ---: | ---: | ---: | ---: |
| Black | 1 | 5 | 5 | 4 | 3 | 18 |
| White | 4 | 5 | 7 | 4 | 4 | 24 |
| Other | 3 | 7 | 6 | 5 | 4 | 25 |
| Total | 8 | 17 | 18 | 13 | 11 | 67 |

The row labeled Total gives observed cell frequencies for age group. These can be obtained by summing frequencies from the cells in the corresponding column. In this respect, we can also refer to them as marginal frequencies. We use the terms marginal frequencies and cell frequencies interchangeably below.

We begin by fitting the two-factor model with an interaction.
. anova chol race\#\#agegrp


## Using observed cell frequencies

Recall that the marginal means are computed from the cell means. Treating the factors as balanced yields the following marginal means for race:

$$
\begin{aligned}
& \eta_{1 .}=\frac{1}{5}\left(\mu_{11}+\mu_{12}+\mu_{13}+\mu_{14}+\mu_{15}\right) \\
& \eta_{2 .}=\frac{1}{5}\left(\mu_{21}+\mu_{22}+\mu_{23}+\mu_{24}+\mu_{25}\right) \\
& \eta_{3 .}=\frac{1}{5}\left(\mu_{31}+\mu_{32}+\mu_{33}+\mu_{34}+\mu_{35}\right)
\end{aligned}
$$

If we have a fixed population and unbalanced cells, then the $\eta_{i}$. do not represent population means. If, however, our data are representative of the population, we can use the frequencies from our estimation sample to estimate the population marginal means, denoted $\mu_{i}$.

Here are the results of testing for a main effect of race, treating all the factors as balanced.

```
. contrast r.race
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | :---: | :---: | ---: |
| race |  |  |  |
| (White vs Black) | 1 | 3.28 | 0.0757 |
| (Other vs Black) | 1 | 1.50 | 0.2263 |
| Joint | 2 | 1.64 | 0.2029 |
| Denominator | 52 |  |  |


|  | Contrast | Std. err. | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: |
| race |  |  |  |  |
| (White vs Black) | -5.324254 | 2.93778 | -11.21934 | .5708338 |
| (Other vs Black) | -3.596867 | 2.93778 | -9.491955 | 2.298221 |

The row labeled (White vs Black) is the test for $\eta_{2} .=\eta_{1}$.. The row labeled (Other vs Black) is the test for $\eta_{3}=\eta_{1}$..

If the observed marginal frequencies are representative of the distribution of the levels of agegrp, we can use them to form the marginal means of chol for each of the levels of race from the cell means.

$$
\begin{aligned}
& \mu_{1 .}=\frac{1}{67}\left(8 \mu_{11}+17 \mu_{12}+18 \mu_{13}+13 \mu_{14}+11 \mu_{15}\right) \\
& \mu_{2 .}=\frac{1}{67}\left(8 \mu_{21}+17 \mu_{22}+18 \mu_{23}+13 \mu_{24}+11 \mu_{25}\right) \\
& \mu_{3 .}=\frac{1}{67}\left(8 \mu_{31}+17 \mu_{32}+18 \mu_{33}+13 \mu_{34}+11 \mu_{35}\right)
\end{aligned}
$$

Here are the results of testing for the main effects of race, using the observed marginal frequencies:

```
. contrast r.race, asobserved
Contrasts of marginal linear predictions
Margins: asobserved
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | :---: | :---: | :---: |
| race |  |  |  |
| (White vs Black) | 1 | 7.25 | 0.0095 |
| (Other vs Black) | 1 | 3.89 | 0.0538 |
| Joint | 2 | 3.74 | 0.0304 |
| Denominator | 52 |  |  |


|  | Contrast | Std. err. | [95\% conf. interval] |  |
| :--- | :--- | :--- | :--- | :--- |
| race |  |  |  |  |
| (White vs Black) | -7.232433 | 2.686089 | -12.62246 | -1.842402 |
| (Other vs Black) | -5.231198 | 2.651203 | -10.55123 | .0888295 |

The row labeled (White vs Black) is the test for $\mu_{2} .=\mu_{1}$.. The row labeled (Other vs Black) is the test for $\mu_{3}$. $=\mu_{1}$.. Both tests were insignificant when we tested the cell means resulting from balanced frequencies; however, when we tested the cell means from observed frequencies, the first test is significant beyond the $5 \%$ level (and the second test is nearly so).

Here we reproduce the results of the asobserved option with custom contrasts. Because we are modifying the way that the marginal means are constructed from the cell means, we will specify the contrasts on the predicted cell means. We use macro expansion, =exp, to evaluate the fractions instead of approximating them with decimals. Macro expansion guarantees that the contrast coefficients sum to zero. For more information, see Macro expansion operators and function in [P] macro.

```
. contrast {race#agegrp -'=8/67' - '=17/67' - '=18/67' -'=13/67' - '=11/67'
> '=8/67, '=17/67, '=18/67, '=13/67, '=11/67'}
> {race#agegrp -'{=8/67' -'=17/67' -'{=18/67' -'=13/67' -'=11/67'
> 0
> '=8/67' '=17/67' '=18/67' '=13/67' '=11/67'}
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | :---: | :---: | :---: |
| race\#agegrp |  |  |  |
| (1) (1) | 1 | 7.25 | 0.0095 |
| (2) (2) | 1 | 3.89 | 0.0538 |
| Joint | 2 | 3.74 | 0.0304 |
| Denominator | 52 |  |  |


|  | Contrast | Std. err. | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: |
| race\#agegrp |  |  |  |  |
| (1) (1) <br> (2) (2) | -7.232433 | 2.686089 | -12.62246 | -1.842402 |

## Weighted contrast operators

contrast provides observation-weighted versions of five of the contrast operators-gw., hw., jw., pw., and qw.. The first three of these operators perform comparisons of means across cells, and like the marginal means just discussed, these means can be computed in two ways: 1) as though the cell frequencies were equal or 2) using the observed cell frequencies from the estimation sample. The weighted operators provide versions of the standard (as balanced) operators that weight these means by their cell frequencies. The two orthogonal polynomial operators involve similar adjustments for weighting.

Let's examine what this means by using the gw. operator. The gw. operator is a weighted version of the g . operator. The gw. operator computes the grand mean using the cell frequencies for race obtained from the model fit.

Here we test the effects of race, comparing each level with the weighted grand mean but otherwise treating the factors as balanced in the marginal mean calculations.
. contrast gw.race
Contrasts of marginal linear predictions
Margins: asbalanced

|  | df | $F$ | $\mathrm{P}>\mathrm{F}$ |
| ---: | :---: | :---: | ---: |
| race |  |  |  |
| (Black vs mean) | 1 | 2.78 | 0.1014 |
| (White vs mean) | 1 | 2.06 | 0.1573 |
| (Other vs mean) | 1 | 0.06 | 0.8068 |
| Joint | 2 | 1.64 | 0.2029 |
| Denominator | 52 |  |  |


|  | Contrast | Std. err. | [95\% conf. interval] |  |
| :--- | ---: | ---: | ---: | ---: |
| race |  |  |  |  |
| (Black vs mean) | 3.24931 | 1.948468 | -.6605779 | 7.159198 |
| (White vs mean) | -2.074944 | 1.44618 | -4.976915 | .8270276 |
| (Other vs mean) | -.347557 | 1.414182 | -3.18532 | 2.490206 |

The observed marginal frequencies of race are 18,24 , and 25 . Thus, the row labeled (Black vs Mean) tests $\eta_{1} .=\left(18 \eta_{1} .+24 \eta_{2} .+25 \eta_{3 .}\right) / 67$; the row labeled (White vs Mean) tests $\eta_{2}=\left(18 \eta_{1} .+\right.$ $\left.24 \eta_{2} .+25 \eta_{3 .}\right) / 67$; and the row labeled (Other vs Mean) tests $\eta_{3 .}=\left(18 \eta_{1 .}+24 \eta_{2} .+25 \eta_{3 .}\right) / 67$.

Now, we reproduce the above results using custom contrasts. We are weighting the calculation of the grand mean from the marginal means for each of the races, but we are not weighting the calculation of the marginal means themselves. Therefore, we can specify the custom contrast on the marginal means for race instead of on the cell means.


|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| :---: | :---: | :---: | :---: |
| race |  |  |  |
| (1) | 1 | 2.78 | 0.1014 |
| (2) | 1 | 2.06 | 0.1573 |
| (3) | 1 | 0.06 | 0.8068 |
| Joint | 2 | 1.64 | 0.2029 |
| Denominator | 52 |  |  |


|  | Contrast | Std. err. | [95\% conf. interval] |  |
| :--- | ---: | ---: | ---: | ---: |
| race |  |  |  |  |
| $(1)$ | 3.24931 | 1.948468 | -.6605779 | 7.159198 |
| $(2)$ | -2.074944 | 1.44618 | -4.976915 | .8270276 |
| $(3)$ | -.347557 | 1.414182 | -3.18532 | 2.490206 |

Now, we will test for each race the difference between the marginal mean and the weighted grand mean, treating the factors as observed in the marginal mean calculations.
. contrast gw.race, asobserved wald ci
Contrasts of marginal linear predictions
Margins: asobserved

|  | df | $F$ | $P>F$ |
| :---: | :---: | :---: | :---: |
| race |  |  |  |
| (Black vs mean) | 1 | 6.81 | 0.0118 |
| (White vs mean) | 1 | 3.74 | 0.0587 |
| (Other vs mean) | 1 | 0.26 | 0.6099 |
| Joint | 2 | 3.74 | 0.0304 |
| Denominator | 52 |  |  |


|  | Contrast | Std. err. | [95\% conf. interval] |  |
| :--- | ---: | ---: | ---: | ---: |
| race |  |  |  |  |
| (Black vs mean) | 4.542662 | 1.740331 | 1.050432 | 8.034891 |
| (White vs mean) | -2.689771 | 1.39142 | -5.481859 | .1023172 |
| (Other vs mean) | -.6885363 | 1.341261 | -3.379973 | 2.002901 |

The row labeled (Black vs Mean) tests $\mu_{1}=\left(18 \mu_{1}+24 \mu_{2}+25 \mu_{3}.\right) / 67$; the row labeled (White vs Mean) tests $\mu_{2}=\left(18 \mu_{1}+24 \mu_{2}+25 \mu_{3}.\right) / 67$; and the row labeled (Other vs Mean) tests $\mu_{3}=\left(18 \mu_{1} .+24 \mu_{2}+25 \mu_{3}.\right) / 67$.

Here we use a custom contrast to reproduce the above result testing $\mu_{1}=\left(18 \mu_{1} .+24 \mu_{2} .+\right.$ $25 \mu_{3}$.) /67. Because both the calculation of the marginal means and the calculation of the grand mean are adjusted, we specify the custom contrast on the cell means.

| . contrast \{race\#agegrp | ${ }^{\prime}=49 / 67 * 8 / 67$ ' | $'=49 / 67 * 17 / 67$ ' | $'=49 / 67 * 18 / 67$ ' |
| :---: | :---: | :---: | :---: |
| > | ${ }^{\prime}=49 / 67 * 13 / 67$, | ${ }^{\prime}=49 / 67 * 11 / 67$ |  |
| > | - ' $=24 / 67 * 8 / 67$ ' | $-^{\prime}=24 / 67 * 17 / 67$ ' | = $24 / 67 * 18 / 67$, |
| > | $-^{\prime}=24 / 67 * 13 / 67$ ' | $-^{\prime}=24 / 67 * 11 / 67$ |  |
| $>$ | - ' $=25 / 67 * 8 / 67$ ' | $\mathrm{l}^{\prime}=25 / 67 * 17 / 67$ ' | = $25 / 67 * 18 / 67$, |
| > | $-^{\prime}=25 / 67 * 13 / 67$ ' | $-^{\prime}=25 / 67 * 11 / 67$ | , nowald |
| Contrasts of marginal linear predictions |  |  |  |
| Margins: asbalanced |  |  |  |


|  | Contrast | Std. err. | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: |
| race\#agegrp <br> (1) (1) | 4.542662 | 1.740331 | 1.050432 | 8.034891 |

The Helmert and reverse Helmert contrasts also involve calculating averages of the marginal means; therefore, weighted versions of these parameters are available as well. The hw. operator is a weighted version of the $h$. operator that computes the mean of the subsequent levels using the cell frequencies obtained from the model fit. The $j w$. operator is a weighted version of the $j$. operator that computes the mean of the previous levels using the cell frequencies obtained from the model fit.

For orthogonal polynomials, we can use the pw. and qw. operators, which are the weighted versions of the p . and q. operators. In this case, the cell frequencies from the model fit are used in the calculation of the orthogonal polynomial contrast coefficients.

## Testing factor effects on slopes

For linear models where the independent variables are all factor variables, the linear prediction at fixed levels of the factor variables turns out to be a cell mean. With these models, contrast computes and tests the effects of the factor variables on the expected mean of the dependent variable. When factor variables are interacted with continuous variables, contrast distinguishes factor effects on the intercept from factor effects on the slope.

Here we have 1980 census data including information on the birthrate (brate), the median age (medage), and the region of the country (region) for each of the 50 states. We can fit an ANCOVA model for brate using main effects of the factor variable region and the continuous variable medage.

```
. use https://www.stata-press.com/data/r18/census3
(1980 Census data by state)
. label list cenreg
cenreg:
    1 NE
    2 ~ N C e n t r a l ~
    3 South
    4 West
. anova brate i.region c.medage
\begin{tabular}{|c|c|c|c|c|c|}
\hline & \[
\begin{array}{ll}
\text { Number of obs } & = \\
\text { Root MSE } & =
\end{array}
\] & \[
\begin{array}{r}
50 \\
12.7575
\end{array}
\] & \multicolumn{2}{|l|}{R -squared} & \[
\begin{aligned}
& 0.8264 \\
& 0.8110
\end{aligned}
\] \\
\hline Source & Partial SS & df & MS & F & Prob>F \\
\hline Model & 34872.859 & 4 & 8718.2147 & 53.57 & 0.0000 \\
\hline region & 2197.7545 & 3 & 732.58484 & 4.50 & 0.0076 \\
\hline medage & 15327.423 & 1 & 15327.423 & 94.18 & 0.0000 \\
\hline Residual & 7323.9611 & 45 & 162.75469 & & \\
\hline Total & 42196.82 & 49 & 861.15959 & & \\
\hline
\end{tabular}
```

For those more comfortable with linear regression, this is equivalent to the regression model

```
. regress brate i.region c.medage
```

You may use either.
We can use contrast to compute reference category effects for region. These contrasts compare the adjusted means of NCentral, South, and West regions with the adjusted mean of the NE region.

```
. contrast r.region
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | :---: | ---: | ---: |
| region |  |  |  |
| (NCentral vs NE) | 1 | 2.24 | 0.1417 |
| (South vs NE) | 1 | 0.78 | 0.3805 |
| (West vs NE) | 1 | 10.33 | 0.0024 |
| Joint | 3 | 4.50 | 0.0076 |
| Denominator | 45 |  |  |


|  | Contrast | Std. err. | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: |
| region |  |  |  |  |
| (NCentral vs NE) | 9.061063 | 6.057484 | -3.139337 | 21.26146 |
| (South vs NE) | 5.06991 | 5.72396 | -6.458738 | 16.59856 |
| (West vs NE) | 21.71328 | 6.755616 | 8.106774 | 35.31979 |

Let's add the interaction between region and medage to the model.
. anova brate region\#\#c.medage

|  | $\begin{array}{ll} \text { Number of obs } & = \\ \text { Root MSE } & = \end{array}$ | $\begin{array}{r} 50 \\ 10.0244 \end{array}$ | R-squared = <br> Adj R-squared $=$ |  | $\begin{aligned} & 0.9000 \\ & 0.8833 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | 37976.315 | 7 | 5425.1878 | 53.99 | 0.0000 |
| region | 3405.0704 | 3 | 1135.0235 | 11.30 | 0.0000 |
| medage | 5279.7145 | 1 | 5279.7145 | 52.54 | 0.0000 |
| region\#medage | 3103.456 | 3 | 1034.4853 | 10.29 | 0.0000 |
| Residual | 4220.5051 | 42 | 100.48822 |  |  |
| Total | 42196.82 | 49 | 861.15959 |  |  |

The parameterization for the expected value of brate as a function of region and medage is given by

$$
E(\text { brate } \mid \text { region }=i, \text { medage })=\alpha_{0}+\alpha_{i}+\beta_{0} \text { medage }+\beta_{i} \text { medage }
$$

where $\alpha_{0}$ is the intercept and $\beta_{0}$ is the slope of medage. We are modeling the effects of region in two different ways. The $\alpha_{i}$ parameters measure the effect of region on the intercept, and the $\beta_{i}$ parameters measure the effect of region on the slope of medage.
contrast computes and tests effects on slopes separately from effects on intercepts. First, we will compute the reference category effects of region on the intercept:

```
. contrast r.region
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | ---: | ---: | ---: |
| region |  |  |  |
| (NCentral vs NE) | 1 | 0.09 | 0.7691 |
| (South vs NE) | 1 | 0.01 | 0.9389 |
| (West vs NE) | 1 | 8.50 | 0.0057 |
| Joint | 3 | 11.30 | 0.0000 |
| Denominator | 42 |  |  |


|  | Contrast | Std. err. | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: |
| region |  |  |  |  |
| (NCentral vs NE) | -49.38396 | 167.1281 | -386.6622 | 287.8942 |
| (South vs NE) | -9.058983 | 117.424 | -246.0302 | 227.9123 |
| (West vs NE) | 343.0024 | 117.6547 | 105.5656 | 580.4393 |

Now, we will compute the reference category effects of region on the slope of medage:

```
. contrast r.region#c.medage
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | :---: | ---: | ---: |
| region\#c.medage |  |  |  |
| (NCentral vs NE) | 1 | 0.16 | 0.6917 |
| (South vs NE) | 1 | 0.03 | 0.8558 |
| (West vs NE) | 1 | 8.18 | 0.0066 |
| Joint | 3 | 10.29 | 0.0000 |
| Denominator | 42 |  |  |


|  | Contrast | Std. err. | [95\% conf. interval] |  |
| ---: | ---: | :--- | ---: | ---: |
| region\#c.medage |  |  |  |  |
| (NCentral vs NE) | 2.208539 | 5.530981 | -8.953432 | 13.37051 |
| (South vs NE) | .6928008 | 3.788735 | -6.953175 | 8.338777 |
| (West vs NE) | -10.94649 | 3.827357 | -18.67041 | -3.22257 |

At the 5\% level, the slope of medage for the West region differs from that of the NE region, but at that level of significance, we cannot say that the slope for the NCentral or the South region differs from that of the NE region.

This model is simple enough that the reference category contrasts reproduce the coefficients for region and for the interactions in an equivalent model fit by regress.


This will not be the case for models that are more complicated.

## Chow tests

Now, let's suppose we are fitting a model for birthrates on median age and marriage rate. We are also interested in whether the regression coefficients differ for states in the east versus states in the west. We use census divisions to create a new variable, west, that indicates which states are in the western half of the United States.

```
. generate west = inlist(division, 4, 7, 8, 9)
```

We fit a model that includes a separate intercept for west as well as an interaction between west and each of the other variables in our model.
. regress brate i.west\#\#c.medage i.west\#\#c.mrgrate

| Source | SS | df | MS |
| ---: | ---: | ---: | :---: |
| Model | 38516.2172 | 5 | 7703.24344 |
| Residual | 3680.60281 | 44 | 83.6500639 |
| Total | 42196.82 | 49 | 861.159592 |


| Number of obs | $=$ | 50 |
| :--- | :--- | ---: |
| F(5, 44) | $=$ | 92.09 |
| Prob > F | $=$ | 0.0000 |
| R-squared | $=$ | 0.9128 |
| Adj R-squared | $=$ | 0.9029 |
| Root MSE | $=$ | 9.146 |


| brate | Coefficient | Std. err. | t | $P>\|t\|$ | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 1. west | 327.8733 | 58.71793 | 5.58 | 0.000 | 209.5351 | 446.2115 |
| medage | -7.532304 | 1.387624 | -5.43 | 0.000 | -10.32888 | -4.735731 |
| west\# <br> c.medage |  |  |  |  |  |  |
| 1 | -10.11443 | 1.849103 | -5.47 | 0.000 | -13.84105 | -6.387808 |
| mrgrate | 828.6813 | 643.3443 | 1.29 | 0.204 | -467.8939 | 2125.257 |
| $\begin{gathered} \text { west\# } \\ \text { c.mrgrate } \end{gathered}$ |  |  |  |  |  |  |
| 1 | -800.8036 | 645.488 | -1.24 | 0.221 | -2101.699 | 500.092 |
| _cons | 366.5325 | 47.08904 | 7.78 | 0.000 | 271.6308 | 461.4343 |

We can test the effects of west on the intercept and on the slopes of medage and mrgrate. We will specify all of these effects in a single contrast command and include the overall option to obtain a joint test of effects, that is, a test that the coefficients for eastern states and for western states are equal.

```
. contrast west west#c.medage west#c.mrgrate, overall
```

Contrasts of marginal linear predictions
Margins: asbalanced

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | :---: | :---: | ---: |
| west | 1 | 31.18 | 0.0000 |
| west\#c.medage | 1 | 29.92 | 0.0000 |
| west\#c.mrgrate | 1 | 1.54 | 0.2213 |
| Overall | 3 | 22.82 | 0.0000 |
| Denominator | 44 |  |  |

This overall test is referred to as a Chow test in econometrics (Chow 1960).

## Beyond linear models

contrast may be used after almost any estimation command, with the added benefit that contrast provides direct support for testing main and interaction effects that is not available in most estimation commands. To illustrate, we will use contrast with results from a logistic regression. Stata's logit command fits logistic regression models, reporting the fitted regression coefficients. The logistic command fits the same models but reports odds ratios. Although contrast can report odds ratios for the computed effects, the tests are all computed from linear combinations of the model coefficients regardless of which estimation command we used.

Suppose we have data on patient satisfaction for three hospitals in a city. Let's begin by fitting a model for satisfied, whether the patient was satisfied with his or her treatment, using the main effects of hospital:

| (Artificial hospital satisfac <br> . logit satisfied i.hospital |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Iteration 0: Log likelihood $=-393.72216$ |  |  |  |  |  |  |
| Iteration 1: Log likelihood $=-387.55736$ |  |  |  |  |  |  |
| Iteration 2: Log likelihood $=-387.4768$ |  |  |  |  |  |  |
| Iteration 3: Log likelihood $=-387.47679$ |  |  |  |  |  |  |
| Logistic regression |  |  |  |  | Number of obs | $=802$ |
|  |  |  |  |  | LR chi2(2) | $=12.49$ |
|  |  |  |  |  | Prob > chi2 | $=0.0019$ |
| Log likelihood $=-387.47679$ |  |  |  |  | Pseudo R2 | $=0.0159$ |
| satisfied | Coefficient | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. | interval] |
| hospital |  |  |  |  |  |  |
| 2 | . 5348129 | . 2136021 | 2.50 | 0.012 | . 1161604 | . 9534654 |
| 3 | .7354519 | . 2221929 | 3.31 | 0.001 | . 2999618 | 1.170942 |
| _cons | 1.034708 | . 1391469 | 7.44 | 0.000 | . 7619855 | 1.307431 |

Because there are no other independent variables in this model, the reference category effects of hospital computed by contrast will match the fitted model coefficients, assuming a common reference level.
. contrast r.hospital
Contrasts of marginal linear predictions
Margins: asbalanced

|  | df | chi2 | P>chi2 |
| :---: | :---: | ---: | ---: |
| hospital |  |  |  |
| $(2$ vs 1) | 1 | 6.27 | 0.0123 |
| $(3$ vs 1) | 1 | 10.96 | 0.0009 |
| Joint | 2 | 12.55 | 0.0019 |


|  | Contrast | Std. err. | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: |
| hospital |  |  |  |  |
| $(2$ vs 1) | .5348129 | .2136021 | .1161604 | .9534654 |
| $(3$ vs 1) | .7354519 | .2221929 | .2999618 | 1.170942 |

We see that the reference category effects are equal to the fitted coefficients. They also have the same interpretation, the difference in $\log$ odds from the reference category. The top table also provides a joint test of these effects, a test of the main effects of hospital.

We also have information on the condition for which each patient is being treated in the variable illness. Here we fit a logistic regression using a two-way crossed model of hospital and illness.

```
. label list illness
illness:
    1 \text { Heart attack}
    2 Stroke
    3 Pneumonia
    4 Lung disease
    5 Kidney failure
. logistic satisfied hospital##illness
\begin{tabular}{ll} 
Logistic regression & Number of obs \(=802\) \\
& LR chi2(14)
\end{tabular}
```

| satisfied | Odds ratio | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| hospital |  |  |  |  |  |  |
| 2 | 1.226496 | . 5492177 | 0.46 | 0.648 | . 509921 | 2.950049 |
| 3 | 1.711111 | . 8061016 | 1.14 | 0.254 | . 6796395 | 4.308021 |
| illness |  |  |  |  |  |  |
| Pneumonia | . 7993827 | . 3408305 | -0.53 | 0.599 | . 3466015 | 1.843653 |
| Lung dise.. | 1.231481 | . 5627958 | 0.46 | 0.649 | . 5028318 | 3.016012 |
| Kidney fa.. | 1.25 | . 5489438 | 0.51 | 0.611 | . 5285676 | 2.956102 |
| hospital\# illness 2\#Stroke |  |  | 1.22 |  |  |  |
|  | 2.434061 | 1.768427 | 1.22 | 0.221 | . 5860099 | 10.11016 |
| 2\#Pneumonia | 4.045805 | 2.868559 | 1.97 | 0.049 | 1.008058 | 16.23769 |
| Lung dise.. | . 54713 | . 3469342 | -0.95 | 0.342 | . 1578866 | 1.89599 |
| Kidney fa.. | 1.594425 | 1.081104 | 0.69 | 0.491 | . 4221288 | 6.022312 |
| 3\#Stroke | . 5416535 | . 3590089 | -0.93 | 0.355 | . 1477555 | 1.985635 |
| 3\#Pneumonia | 1.579502 | 1.042504 | 0.69 | 0.489 | . 4332209 | 5.758783 |
| Lung dise. 3 \# | 3.137388 | 2.595748 | 1.38 | 0.167 | . 6198955 | 15.87881 |
| Kidney fa. 3 \# | 1.672727 | 1.226149 | 0.70 | 0.483 | . 3976256 | 7.036812 |
| _cons | 2.571429 | . 8099239 | 3.00 | 0.003 | 1.386983 | 4.767358 |

Note: _cons estimates baseline odds.
Using contrast, we can obtain an ANOVA-style table of tests for the main effects and interaction effects of hospital and illness.

```
. contrast hospital##illness
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | chi2 | P>chi2 |
| ---: | :---: | :---: | :---: |
| hospital | 2 | 14.92 | 0.0006 |
| illness | 4 | 4.09 | 0.3937 |
| hospital\#illness | 8 | 20.45 | 0.0088 |

Our interaction effect is significant, so we decide to evaluate the simple reference category effects of hospital within illness. We are particularly interested in patient satisfaction when being treated for a heart attack or stroke, so we will use the i. operator to limit our output to simple effects within the first two illnesses.

```
. contrast r.hospital@i(1 2).illness, nowald
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | Contrast | Std. err. | [95\% conf | interval] |
| :---: | :---: | :---: | :---: | :---: |
| hospital@illness |  |  |  |  |
| (2 vs 1) Heart attack | . 2041611 | . 4477942 | -. 6734995 | 1.081822 |
| (2 vs 1) Stroke | 1.093722 | . 5721288 | -. 0276296 | 2.215074 |
| (3 vs 1) Heart attack | . 5371429 | . 4710983 | -. 3861928 | 1.460479 |
| (3 vs 1) Stroke | -. 0759859 | . 4662325 | -. 9897847 | . 8378129 |

The row labeled (2 vs 1) heart attack estimates simple effects on the log odds when comparing hospital 2 with hospital 1 for patients having heart attacks. These effects are differences in the cell means of the linear predictions.

We can add the or option to report an odds ratio for each of these simple effects:
. contrast r.hospital@i(1 2).illness, nowald or
Contrasts of marginal linear predictions
Margins: asbalanced

|  | Odds ratio | Std. err. | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: |
| hospital@illness |  |  |  |  |
| (2 vs 1) Heart attack | 1.226496 | . 5492177 | . 509921 | 2.950049 |
| (2 vs 1) Stroke | 2.985366 | 1.708014 | . 9727486 | 9.162089 |
| (3 vs 1) Heart attack | 1.711111 | . 8061016 | . 6796395 | 4.308021 |
| (3 vs 1) Stroke | . 9268293 | . 4321179 | . 3716567 | 2.311306 |

These odds ratios are just the exponentiated version of the contrasts in the previous table.
For contrasts of the margins of nonlinear predictions, such as predicted probabilities, see $[R]$ margins, contrast.

## Multiple equations

contrast works with models containing multiple equations. Commands such as intreg and gnbreg allow their ancillary parameters to be modeled as functions of independent variables, and contrast can compute and test effects within these equations. In addition, contrast allows a special pseudofactor for equation-called _eqns-when working with results from manova, mvreg, mlogit, and mprobit.

In example 4 of [MV] manova, we fit a two-way MANOVA model using data from Woodard (1931). Here we will fit this model using mvreg. The data represent patients with jaw fractures. y1 is the patient's age, y2 is blood lymphocytes, and y3 is blood polymorphonuclears. Two factor variables, gender and fracture, are used as independent variables.
. use https://www.stata-press.com/data/r18/jaw
(Table 4.6. Two-way unbalanced data for fractures of the jaw, Rencher (1998))
. mvreg y1 y2 y3 = gender\#\#fracture, vsquish nofvlabel

| Equation | Obs | Parms | RMSE | "R-sq" | F | P>F |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: |
| y1 | 27 | 6 | 10.21777 | 0.4086 | 2.902124 | 0.0382 |
| y2 | 27 | 6 | 5.268768 | 0.4743 | 3.78967 | 0.0133 |
| y3 | 27 | 6 | 4.993647 | 0.4518 | 3.460938 | 0.0195 |


contrast computes Wald tests using the coefficients from the first equation by default.
. contrast gender\#\#fracture
Contrasts of marginal linear predictions
Margins: asbalanced

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | :---: | :---: | ---: |
| y1 |  |  |  |
| fracture | 1 | 2.16 | 0.1569 |
| gender\#fracture | 2 | 2.74 | 0.0880 |
| Denominator | 2 | 1.69 | 0.2085 |

Here we use the equation() option to compute the Wald tests in the y2 equation:
. contrast gender\#\#fracture, equation(y2)
Contrasts of marginal linear predictions
Margins: asbalanced

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | :---: | :---: | ---: |
| y2 |  |  |  |
| fracture | 1 | 5.41 | 0.0301 |
| gender\#fracture | 2 | 7.97 | 0.0027 |
| Denominator | 2 | 5.97 | 0.0088 |

Here we use the equation index to compute the Wald tests in the third equation:

```
. contrast gender##fracture, equation(#3)
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | :---: | :---: | :---: |
| y3 |  |  |  |
| fracture | 1 | 2.23 | 0.1502 |
|  | 2 | 6.36 | 0.0069 |
| gender\#fracture | 2 | 6.66 | 0.0058 |
| $r$ Denominator | 21 |  |  |

Here we use the atequations option to compute Wald tests for each equation in the model. We also use the vsquish option to suppress the extra blank lines between terms.
. contrast gender\#\#fracture, atequations vsquish
Contrasts of marginal linear predictions
Margins: asbalanced

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| :---: | :---: | :---: | :---: |
| y1 |  |  |  |
| gender | 1 | 2.16 | 0.1569 |
| fracture | 2 | 2.74 | 0.0880 |
| gender\#fracture | 2 | 1.69 | 0.2085 |
| y2 |  |  |  |
| gender | 1 | 5.41 | 0.0301 |
| fracture | 2 | 7.97 | 0.0027 |
| gender\#fracture | 2 | 5.97 | 0.0088 |
| y3 |  |  |  |
| gender | 1 | 2.23 | 0.1502 |
| fracture | 2 | 6.36 | 0.0069 |
| gender\#fracture | 2 | 6.66 | 0.0058 |
| Denominator | 21 |  |  |

Because we are investigating the results from mvreg, we can use the special _eqns factor to test for a marginal effect on the means among the dependent variables:
. contrast _eqns
Contrasts of marginal linear predictions
Margins: asbalanced

|  | df | $F$ | $P>F$ |
| ---: | ---: | ---: | ---: |
| _eqns | 2 | 49.19 | 0.0000 |
| Denominator | 21 |  |  |

Here we test whether the main effects of gender differ among the dependent variables:

| . contrast gender\#_eqns |
| :--- |
| Contrasts of marginal linear predictions |
| Margins: asbalanced |

Although it is not terribly interesting in this case, we can even calculate contrasts across equations:

```
. contrast gender#r._eqns
Contrasts of marginal linear predictions
Margins: asbalanced
```

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| ---: | :---: | :---: | ---: |
| gender\#_eqns |  |  |  |
| (joint)(2 vs 1) | 1 | 5.82 | 0.0251 |
| (joint) (3 vs 1) | 1 | 0.40 | 0.5352 |
| Joint | 2 | 3.61 | 0.0448 |
| Denominator | 21 |  |  |

## Video example

Introduction to contrasts in Stata: One-way ANOVA

## Stored results

contrast stores the following in r() :
Scalars
$r\left(d f \_r\right)$
$r$ (k_terms)
$r$ (level)
Macros
$r$ (cmd)
$r$ (cmdline)
r(est_cmd)
r(est_cmdline)
$r$ (title)
r(overall)
$r$ (emptycells)
$r$ (mcmethod)
$r$ (mctitle)
r(mcadjustall)
$r$ (margin_method)
Matrices

```
        r(b)
    r(V)
    r(error)
    r(L)
    r(table)
    r(F)
    r(chi2)
    r(p)
    r(df)
    r(df2)
```

variance degrees of freedom
number of terms in termlist
confidence level of confidence intervals

## contrast

command as typed
$e(\mathrm{cmd})$ from original estimation results
$e$ (cmdline) from original estimation results
title in output
overall or empty
empspec from emptycells()
method from mcompare()
title for method from mcompare()
adjustall or empty
asbalanced or asobserved
contrast estimates
variance-covariance matrix of the contrast estimates
contrast estimability codes;
0 means estimable,
8 means not estimable
matrix of contrasts applied to the model coefficients matrix containing the contrasts with their standard errors, test statistics, $p$-values, and confidence intervals vector of $F$ statistics; $\mathbf{r}\left(\mathrm{df} \_\mathbf{r}\right)$ present vector of $\chi^{2}$ statistics; $r\left(d f \_r\right)$ not present vector of $p$-values corresponding to $r(F)$ or $r(c h i 2)$ vector of degrees of freedom corresponding to $r(p)$ vector of denominator degrees of freedom corresponding to $r(F)$
contrast with the post option stores the following in e() :
Scalars

> e(df_r)
> e(k_terms)
variance degrees of freedom
number of terms in termlist
Macros
e(cmd)
e(cmdline)
e(properties)
e(est_cmd)
e(est_cmdline)
e(title)
e(overall)
e(emptycells)
e(margin_method)
e(asbalanced)
e(asobserved)
Matrices
e(b)
e(V)
e(error)
e(L)
e(F)
e(chi2)
e(p)
e(df)
e(df2)

```
contrast
```

command as typed
b V
$e(\mathrm{cmd})$ from original estimation results
$e$ (cmdline) from original estimation results
title in output
overall or empty
empspec from emptycells()
asbalanced or asobserved
factor variables fvset as asbalanced
factor variables fvset as asobserved
contrast estimates
variance-covariance matrix of the contrast estimates
contrast estimability codes;
0 means estimable,
8 means not estimable
matrix of contrasts applied to the model coefficients
vector of unadjusted $F$ statistics; e(df_r) present
vector of $\chi^{2}$ statistics; e(df_r) not present
vector of unadjusted $p$-values corresponding to $e(F)$ or $e(c h i 2)$
vector of degrees of freedom corresponding to $e(p)$
vector of denominator degrees of freedom corresponding to e(F)

## Methods and formulas

Methods and formulas are presented under the following headings:

Marginal linear predictions<br>Contrast operators<br>Reference level contrasts<br>Adjacent contrasts<br>Grand mean contrasts<br>Helmert contrasts<br>Reverse Helmert contrasts<br>Orthogonal polynomial contrasts<br>Contrasts within interactions<br>Multiple comparisons

## Marginal linear predictions

contrast treats intercept effects separately from slope effects. To illustrate, consider the following parameterization for a quadratic regression of $y$ on $x$ that also models the effects of two factor variables $A$ and $B$, where the levels of $A$ are indexed by $i=1, \ldots, k_{a}$ and the levels of $B$ are indexed by $j=1, \ldots, k_{b}$.

$$
\begin{aligned}
E(y \mid A=i, B=j, x) & =\eta_{0 i j}+\eta_{1 i j} x+\eta_{2 i j} x^{2} \\
\eta_{0 i j} & =\eta_{0}+\alpha_{0 i}+\beta_{0 j}+(\alpha \beta)_{0 i j} \\
\eta_{1 i j} & =\eta_{1}+\alpha_{1 i}+\beta_{1 j}+(\alpha \beta)_{1 i j} \\
\eta_{2 i j} & =\eta_{2}+\alpha_{2 i}+\beta_{2 j}+(\alpha \beta)_{2 i j}
\end{aligned}
$$

We have partitioned the coefficients into three groups of parameters: $\eta_{0 i j}$ is a cell prediction for the intercept, $\eta_{1 i j}$ is a cell prediction for the slope on $x$, and $\eta_{2 i j}$ is a cell prediction for the slope on $x^{2}$. For the intercept parameters, $\eta_{0}$ is the intercept, $\alpha_{0 i}$ represents a main effect for factor $A$ at its $i$ th level, $\beta_{0 j}$ represents a main effect for factor $B$ at its $j$ th level, and $(\alpha \beta)_{0 i j}$ represents an effect for the interaction of $A$ and $B$ at the $i j$ th level. The individual coefficients in $\eta_{1 i j}$ and $\eta_{2 i j}$ have similar interpretations, but the effects are on the slopes of $x$ and $x^{2}$, respectively.

The marginal intercepts for $A$ are given by

$$
\eta_{0 i .}=\sum_{j=1}^{k_{b}} f_{i j} \eta_{0 i j}
$$

where $f_{i j}$ is a marginal relative frequency of the $j$ th level of $B$ and is controlled by the asobserved and emptycells (reweight) options according to

$$
f_{i j}= \begin{cases}1 / k_{b}, & \text { default } \\ w_{. j} / w_{. .}, & \text {asobserved } \\ 1 /\left(k_{b}-e_{i .}\right), & \text { emptycells(reweight) } \\ w_{i j} / w_{i .}, & \text { emptycells(reweight) and asobserved }\end{cases}
$$

Above, $w_{i j}$ is the number of individuals with $A$ at its $i$ th level and $B$ at its $j$ th,

$$
\begin{gathered}
w_{i .}=\sum_{j=1}^{k_{b}} w_{i j} \\
w_{. j}=\sum_{i=1}^{k_{a}} w_{i j} \\
w_{. .}=\sum_{i=1}^{k_{a}} \sum_{j=1}^{k_{b}} w_{i j}
\end{gathered}
$$

and $e_{i}$. is the number of empty cells where $A$ is at its $i$ th level. The marginal intercepts for $B$ and marginal slopes on $x$ and $x^{2}$ are similarly defined.

Estimates for the cell intercepts and slopes are computed using the corresponding linear combination of the coefficients from the fitted model. For example, the estimated cell intercepts are computed using

$$
\widehat{\eta}_{0 i j}=\widehat{\eta}_{0}+\widehat{\alpha}_{0 i}+\widehat{\beta}_{0 j}+(\widehat{\alpha \beta})_{0 i j}
$$

and the estimated marginal intercepts for $A$ are computed as

$$
\widehat{\eta}_{0 i .}=\sum_{j=1}^{k_{b}} f_{i j} \widehat{\eta}_{0 i j}
$$

## Contrast operators

contrast performs Wald tests using linear combinations of marginal linear predictions. For example, the following linear combination can be used to test for a specific effect of factor $A$ on the marginal intercepts.

$$
\sum_{i=1}^{k_{a}} c_{i} \eta_{0 i}
$$

If the $c_{i}$ elements sum to zero, the linear combination is called a contrast. If the factor $A$ is represented by a variable named A , then we specify this contrast using the following syntax:

$$
\left\{\mathrm{A} c_{1} c_{2} \ldots c_{k_{a}}\right\}
$$

Similarly, the following linear combination can be used to test for a specific interaction effect of factors $A$ and $B$ on the marginal slope of $x$.

$$
\sum_{i=1}^{k_{a}} \sum_{j=1}^{k_{b}} c_{i j} \eta_{1 i j}
$$

If the factor $B$ is represented by a variable named B , then we specify this contrast using the following syntax:

$$
\left\{\mathrm{A} \# \mathrm{~B} c_{11} c_{12} \ldots c_{1 k_{b}} c_{21} \ldots c_{k_{a} k_{b}}\right\}
$$

contrast has variable operators for several commonly used contrasts. Each contrast operator specifies a matrix of linear combinations that yield the requested set of contrasts to be applied to the marginal linear predictions associated with the attached factor variable.

## Reference level contrasts

The r. operator compares each level with a reference level. Let $\mathbf{R}$ be the corresponding contrast matrix for factor $A$, and then $\mathbf{R}$ is a $\left(k_{a}-1\right) \times k_{a}$ matrix with elements

$$
\mathbf{R}_{i j}=\left\{\begin{aligned}
-1, & \text { if } j \text { is the reference level } \\
1, & \text { if } i=j \text { and } j \text { is less than the reference level } \\
1, & \text { if } i+1=j \text { and } j \text { is greater than the reference level } \\
0, & \text { otherwise }
\end{aligned}\right.
$$

If $k_{a}=5$ and the reference level is the third level of $A$ (specified as $\left.\mathrm{rb}(\# 3) . \mathrm{A}\right)$, then

$$
\mathbf{R}=\left(\begin{array}{ccccc}
1 & 0 & -1 & 0 & 0 \\
0 & 1 & -1 & 0 & 0 \\
0 & 0 & -1 & 1 & 0 \\
0 & 0 & -1 & 0 & 1
\end{array}\right)
$$

## Adjacent contrasts

The a. operator compares each level with the next level. Let $\mathbf{A}$ be the corresponding contrast matrix for factor $A$, and then $\mathbf{A}$ is a $\left(k_{a}-1\right) \times k_{a}$ matrix with elements

$$
\mathbf{A}_{i j}=\left\{\begin{aligned}
1, & \text { if } i=j \\
-1, & \text { if } i+1=j \\
0, & \text { otherwise }
\end{aligned}\right.
$$

If $k_{a}=5$, then

$$
\mathbf{A}=\left(\begin{array}{rrrrr}
1 & -1 & 0 & 0 & 0 \\
0 & 1 & -1 & 0 & 0 \\
0 & 0 & 1 & -1 & 0 \\
0 & 0 & 0 & 1 & -1
\end{array}\right)
$$

The ar. operator compares each level with the previous level. If $\mathbf{A}$ is the contrast matrix for the a. operator, then $-\mathbf{A}$ is the corresponding contrast matrix for the ar. operator.

## Grand mean contrasts

The g . operator compares each level with the mean of all the levels. Let $\mathbf{G}$ be the corresponding contrast matrix for factor $A$, and then $\mathbf{G}$ is a $k_{a} \times k_{a}$ matrix with elements

$$
\mathbf{G}_{i j}=\left\{\begin{aligned}
1-1 / k_{a}, & \text { if } i=j \\
-1 / k_{a}, & \text { if } i \neq j
\end{aligned}\right.
$$

If $k_{a}=5$, then

$$
\mathbf{G}=\left(\begin{array}{rrrrr}
4 / 5 & -1 / 5 & -1 / 5 & -1 / 5 & -1 / 5 \\
-1 / 5 & 4 / 5 & -1 / 5 & -1 / 5 & -1 / 5 \\
-1 / 5 & -1 / 5 & 4 / 5 & -1 / 5 & -1 / 5 \\
-1 / 5 & -1 / 5 & -1 / 5 & 4 / 5 & -1 / 5 \\
-1 / 5 & -1 / 5 & -1 / 5 & -1 / 5 & 4 / 5
\end{array}\right)
$$

The gw. operator compares each level with the weighted mean of all the levels. The weights are taken from the observed weighted cell frequencies in the estimation sample of the fitted model. Let $\mathbf{G}_{w}$ be the corresponding contrast matrix for factor $A$, and then $\mathbf{G}_{w}$ is a $k_{a} \times k_{a}$ matrix with elements

$$
\mathbf{G}_{i j}=\left\{\begin{aligned}
1-w_{i} / w ., & \text { if } i=j \\
-w_{j} / w ., & \text { if } i \neq j
\end{aligned}\right.
$$

where $w_{i}$ is a marginal weight representing the number of individuals with $A$ at its $i$ th level and $w .=\sum_{i} w_{i}$.

## Helmert contrasts

The $h$. operator compares each level with the mean of the subsequent levels. Let $\mathbf{H}$ be the corresponding contrast matrix for factor $A$, and then $\mathbf{H}$ is a $\left(k_{a}-1\right) \times k_{a}$ matrix with elements

$$
\mathbf{H}_{i j}= \begin{cases}1, & \text { if } i=j \\ -1 /\left(k_{a}-i\right), & \text { if } i<j \\ 0, & \text { otherwise }\end{cases}
$$

If $k_{a}=5$, then

$$
\mathbf{H}=\left(\begin{array}{ccccl}
1 & -1 / 4 & -1 / 4 & -1 / 4 & -1 / 4 \\
0 & 1 & -1 / 3 & -1 / 3 & -1 / 3 \\
0 & 0 & 1 & -1 / 2 & -1 / 2 \\
0 & 0 & 0 & 1 & -1
\end{array}\right)
$$

The hw. operator compares each level with the weighted mean of the subsequent levels. Let $\mathbf{H}_{w}$ be the corresponding contrast matrix for factor $A$, and then $\mathbf{H}_{w}$ is a $\left(k_{a}-1\right) \times k_{a}$ matrix with elements

$$
\mathbf{H}_{w i j}= \begin{cases}1, & \text { if } i=j \\ -w_{j} / \sum_{l=j}^{k_{a}} w_{l}, & \text { if } i<j \\ 0, & \text { otherwise }\end{cases}
$$

## Reverse Helmert contrasts

The $\mathbf{j}$. operator compares each level with the mean of the previous levels. Let $\mathbf{J}$ be the corresponding contrast matrix for factor $A$, and then $\mathbf{J}$ is a $\left(k_{a}-1\right) \times k_{a}$ matrix with elements

$$
\mathbf{J}_{i j}=\left\{\begin{array}{cl}
1, & \text { if } i+1=j \\
-1 / i, & \text { if } j \leq i \\
0, & \text { otherwise }
\end{array}\right.
$$

If $k_{a}=5$, then

$$
\mathbf{H}=\left(\begin{array}{ccccc}
-1 & 1 & 0 & 0 & 0 \\
-1 / 2 & -1 / 2 & 1 & 0 & 0 \\
-1 / 3 & -1 / 3 & -1 / 3 & 1 & 0 \\
-1 / 4 & -1 / 4 & -1 / 4 & -1 / 4 & 1
\end{array}\right)
$$

The $j w$. operator compares each level with the weighted mean of the previous levels. Let $\mathbf{J}_{w}$ be the corresponding contrast matrix for factor $A$, and then $\mathbf{J}_{w}$ is a $\left(k_{a}-1\right) \times k_{a}$ matrix with elements

$$
\mathbf{J}_{w i j}= \begin{cases}1, & \text { if } i+1=j \\ -w_{j} / \sum_{l=1}^{i} w_{l}, & \text { if } i \leq j \\ 0, & \text { otherwise }\end{cases}
$$

## Orthogonal polynomial contrasts

The p. operator applies orthogonal polynomial contrasts using the level values of the attached factor variable. The q. operator applies orthogonal polynomial contrasts using the level indices of the attached factor variable. These two operators are equivalent when the level values of the attached factor are equally spaced. The pw. and qw. operators are weighted versions of p. and q., where the weights are taken from the observed weighted cell frequencies in the estimation sample of the fitted model. contrast uses the Christoffel-Darboux recurrence formula for computing orthogonal polynomial contrasts (Abramowitz and Stegun 1964). The elements of the contrasts are normalized such that

$$
\mathbf{Q}^{\prime} \mathbf{W Q}=\frac{1}{w .} \mathbf{I}
$$

where $\mathbf{W}$ is a diagonal matrix of the marginal cell weights $w_{1}, w_{2}, \ldots, w_{k}$ of the attached factor variable (all 1 for p . and q .), and $w$. is the sum of the weights (the number of levels $k$ for p . and q.).

## Contrasts within interactions

Contrast operators are allowed to be specified on factor variables participating in interactions. In such cases, contrast applies the proper matrix product of the contrast matrices to the cell margins of the interacted factor variables.

For example, consider the contrasts implied by specifying r.A\#h.B. Let $M$ be the matrix of estimated cell margins for the levels of $A$ and $B$, where the rows of $\mathbf{M}$ are indexed by the levels of $A$ and the columns are indexed by the levels of $B$. contrast puts the estimated cell margins in the following vector form:

$$
\mathbf{v}=\operatorname{vec}\left(\mathbf{M}^{\prime}\right)=\left(\begin{array}{c}
\mathbf{M}_{11} \\
\mathbf{M}_{12} \\
\vdots \\
\mathbf{M}_{1 k_{b}} \\
\mathbf{M}_{21} \\
\mathbf{M}_{22} \\
\vdots \\
\mathbf{M}_{2 k_{b}} \\
\vdots \\
\mathbf{M}_{k_{a} k_{b}}
\end{array}\right)
$$

The individual contrasts are then given by the elements of

$$
(\mathbf{R} \otimes \mathbf{H}) \mathbf{v}
$$

where $\otimes$ denotes the Kronecker direct product.

## Multiple comparisons

See $[R]$ pwcompare for details on the methods and formulas used to adjust $p$-values and confidence intervals for multiple comparisons. The formulas for Bonferroni's method and Šidák's method are presented with $m=k(k-1) / 2$, the number of pairwise comparisons for a factor term with $k$ levels. For contrasts, $m$ is instead the number of contrasts being performed on the factor term; often, $m=k-1$ for a term with $k$ levels.

## References

Abramowitz, M., and I. A. Stegun, ed. 1964. Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables. Washington, DC: National Bureau of Standards.
Baldwin, S. 2019. Psychological Statistics and Psychometrics Using Stata. College Station, TX: Stata Press.
Chow, G. C. 1960. Tests of equality between sets of coefficients in two linear regressions. Econometrica 28: 591-605. https://doi.org/10.2307/1910133.

Coster, D. 2005. Contrasts. In Vol. 2 of Encyclopedia of Biostatistics, ed. P. Armitage and T. Colton, 1153-1157. Chichester, UK: Wiley.
Freese, J., and S. Johfre. 2022. Binary contrasts for unordered polytomous regressors. Stata Journal 22: 125-133.
Milliken, G. A., and D. E. Johnson. 2009. Analysis of Messy Data, Volume 1: Designed Experiments. 2nd ed. Boca Raton, FL: CRC Press.
Mitchell, M. N. 2015. Stata for the Behavioral Sciences. College Station, TX: Stata Press.
——. 2021. Interpreting and Visualizing Regression Models Using Stata. 2nd ed. College Station, TX: Stata Press.
Rosenthal, R., R. L. Rosnow, and D. B. Rubin. 2000. Contrasts and Effect Sizes in Behavioral Research: A Correlational Approach. Cambridge: Cambridge University Press.
Searle, S. R. 1997. Linear Models for Unbalanced Data. New York: Wiley.
Searle, S. R., and M. H. J. Gruber. 2017. Linear Models. 2nd ed. Hoboken, NJ: Wiley.
Winer, B. J., D. R. Brown, and K. M. Michels. 1991. Statistical Principles in Experimental Design. 3rd ed. New York: McGraw-Hill.
Woodard, D. E. 1931. Healing time of fractures of the jaw in relation to delay before reduction, infection, syphilis and blood calcium and phosphorus content. Journal of the American Dental Association 18: 419-442. https://doi.org/10.14219/jada.archive.1931.0096.

## Also see

[R] contrast postestimation - Postestimation tools for contrast
[ $R$ ] lincom - Linear combinations of parameters
$[R]$ margins - Marginal means, predictive margins, and marginal effects
[R] margins, contrast - Contrasts of margins
[R] pwcompare - Pairwise comparisons
[R] test - Test linear hypotheses after estimation
[U] 20 Estimation and postestimation commands

## Title

contrast postestimation - Postestimation tools for contrast
Postestimation commands Remarks and examples Also see

## Postestimation commands

The following postestimation commands are available after contrast, post:
Command Description
estat vce variance-covariance matrix of the estimators (VCE)
estat (svy) postestimation statistics for survey data
estimates cataloging estimation results
etable table of estimation results
lincom point estimates, standard errors, testing, and inference for linear combinations of coefficients
nlcom point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients
test Wald tests of simple and composite linear hypotheses
testnl Wald tests of nonlinear hypotheses

## Remarks and examples

In Orthogonal polynomial contrasts in $[\mathrm{R}]$ contrast, we used the p . operator to test the orthogonal polynomial effects of age group.

```
. contrast p.agegrp, noeffects
```

We then used a second contrast command,
. contrast p(2 3 4).agegrp, noeffects
selecting levels to test whether the quadratic, cubic, and quartic contrasts were jointly significant.
We can perform the same joint test by using the test command after specifying the post option with our first contrast command.

```
. use https://www.stata-press.com/data/r18/cholesterol
(Artificial cholesterol data)
```

- anova chol agegrp
(output omitted)
. contrast p.agegrp, noeffects post
Contrasts of marginal linear predictions
Margins: asbalanced

|  | df | F | $\mathrm{P}>\mathrm{F}$ |
| :---: | :---: | ---: | ---: |
| agegrp |  |  |  |
| (linear) | 1 | 139.11 | 0.0000 |
| (quadratic) | 1 | 0.15 | 0.6962 |
| (cubic) | 1 | 0.37 | 0.5448 |
| (quartic) | 1 | 0.43 | 0.5153 |
| Joint | 4 | 35.02 | 0.0000 |
| Denominator | 70 |  |  |

. test p2.agegrp p3.agegrp p4.agegrp
( 1) p2.agegrp $=0$
( 2) p3.agegrp $=0$
( 3) p4.agegrp $=0$
F ( 3, 70) $=0.32$
Prob > F = 0.8129

## Also see

[R] contrast - Contrasts and linear hypothesis tests after estimation
[U] 20 Estimation and postestimation commands

## Title

copyright - Display copyright information

## Description

copyright presents copyright notifications concerning tools, libraries, etc., used in the construction of Stata.

## Syntax

```
copyright
```


## Remarks and examples

The correct form for a copyright notice is
Copyright dates by author/owner
The word "Copyright" is spelled out. You can use the © symbol, but "(C)" has never been given legal recognition. The phrase "All Rights Reserved" was historically required but is no longer needed.

Currently, most works are copyrighted from the moment they are written, and no copyright notice is required. Copyright concerns the protection of the expression and structure of facts and ideas, not the facts and ideas themselves. Copyright concerns the ownership of the expression and not the name given to the expression, which is covered under trademark law.

Copyright law as it exists today began in England in 1710 with the Statute of Anne, An Act for the Encouragement of Learning, by Vesting the Copies of Printed Books in the Authors or Purchases of Such Copies, during the Times therein mentioned. In 1672, Massachusetts introduced the first copyright law in what was to become the United States. After the Revolutionary War, copyright was introduced into the US Constitution in 1787 and went into effect on May 31, 1790. On June 9, 1790, the first copyright in the United States was registered for The Philadelphia Spelling Book by John Barry.

There are significant differences in the understanding of copyright in the English- and non-Englishspeaking world. The Napoleonic or Civil Code, the dominant legal system in the non-English-speaking world, splits the rights into two classes: the author's economic rights and the author's moral rights. Moral rights are available only to "natural persons". Legal persons (corporations) have economic rights but not moral rights.

## Also see <br> Copyright page of this book

## Title

Copyright Apache - Apache copyright notification

## Description Also see

## Description

Stata uses portions of the Apache Commons Java components library, Apache log4j Java library, the docx4j Java library, the FlatLaf Java library, the JSON.simple Java library, and Apache Batik SVG Toolkit with the express permission of the authors under the Apache License, version 2.0, pursuant to the following notice:

Apache License
Version 2.0, January 2004
http://www.apache.org/licenses/

## TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

## 1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent $(50 \%)$ or more of the outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royaltyfree, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may
add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License. You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

## END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work
To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]
Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

## Also see

[R] copyright - Display copyright information

## Title

Copyright autolink — autolink copyright notification
Description Also see

## Description

Stata uses portions of autolink with the express permission of the authors pursuant to the following notice:

Copyright (c) 2015 Robin Stocker
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## Also see

[R] copyright - Display copyright information

## Title

Copyright Boost - Boost copyright notification
Description Also see

## Description

Stata uses portions of Boost with the express permission of the authors pursuant to the following notice:

Boost Software License - Version 1.0 - August 17, 2003
Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## Also see <br> [R] copyright - Display copyright information

## Title

Copyright flexmark - flexmark copyright notification

Description Also see

## Description

Stata uses portions of flexmark with the express permission of the authors pursuant to the following notice:

Copyright (c) 2015-2016, Atlassian Pty Ltd
All rights reserved.
Copyright (c) 2016, Vladimir Schneider, All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

[^14]
## Title

Copyright Hamcrest - Hamcrest copyright notification

## Description Also see

## Description

Stata uses portions of Hamcrest with the express permission of the authors pursuant to the following notice:

Copyright © 2000-2015 www.hamcrest.org All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Hamcrest nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

## Also see

[R] copyright - Display copyright information

## Title

Copyright H2O — H2O copyright notification

## Description Also see

## Description

Stata uses portions of the H2O Java library with the express permission of the authors under the Apache License, version 2.0, pursuant to the following notice:

## Apache License

Version 2.0, January 2004
http://www.apache.org/licenses/

## TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.
"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.
"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50\%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.
"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.
"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).
"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."
"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royaltyfree, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
(b) You must cause any modified files to carry prominent notices stating that You changed the files; and
(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may
add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

## END OF TERMS AND CONDITIONS

## APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets " $\}$ " replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description
of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2014-2021 H2O.ai, Inc.
Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

## Also see

[R] copyright - Display copyright information

## Title

Copyright ICD-10 - ICD-10 copyright notification

## Description Also see

## Description

Stata uses the ICD-10 codes in [D] icd10 with the express permission of the World Health Organization (WHO) pursuant to the following notice:

ICD-10 codes used by permission of WHO, from: International Statistical Classification of Diseases and Related Health Problems, Tenth Revision (ICD-10) 2010 Edition. Vols 1-3. Geneva, World Health Organization, 2011.

The use of ICD-10 in this Product does not imply any endorsement by WHO of any specific product.
The ICD-10 codes shall not be amended, abridged, translated, deleted or in any other way changed without the consent of WHO.

The ICD-10 codes are for the internal use of the end user. They are not to be reproduced, transmitted or distributed outside of the user's organization in any form or by any means except in summary results of analyses.

ICD-10 is distributed without warranty of any kind, either express or implied. In no event shall the World Health Organization be liable for damages, including any general, special, incidental, or consequential damages, arising out of the use of ICD-10.

## Also see

[ R ] copyright - Display copyright information

## Title

Copyright ICU — ICU copyright notification

Description Also see

## Description

Stata uses portions of ICU with the express permission of the authors pursuant to the following notice:

## COPYRIGHT AND PERMISSION NOTICE

Copyright © 1995-2014 International Business Machines Corporation and others All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.
THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.

## Also see <br> [R] copyright - Display copyright information

## Title

Copyright JAXB — JAXB copyright notification

## Description Source code Also see

## Description

Stata uses portions of JAXB with the express permission of the authors, pursuant to the terms of the Common Development and Distribution License (CDDL) version 1.1.

## COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.1

1. Definitions.
1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.
1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.
1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.
1.4. "Executable" means the Covered Software in any form other than Source Code.
1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.
1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.
1.7. "License" means this document.
1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.
1.9. "Modifications" means the Source Code and Executable form of any of the following:
A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
B. Any new file that contains any part of the Original Software or previous Modification; or
C. Any new file that is contributed or otherwise made available under the terms of this License.
1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.
1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.
1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.
1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50\%) of the outstanding shares or beneficial ownership of such entity.

## 2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).
(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.
(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

### 2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:
(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and
(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).
(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.
(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.
3. Distribution Obligations.
3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

### 3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

### 3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.
3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.
3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of
this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.
3.6. Larger Works.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.
4. Versions of the License.

### 4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

### 4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.
4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

## 5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGING. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF

THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.
6. TERMINATION.
6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.
6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.
6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.
6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

## 7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

## 8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. §252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.
9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

## 10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

## Source code

Per Section 3.1 above, the source code for JAXB is publicly available via https://javaee.github.io/ jaxb-v2/.

[^15]
## Title

Copyright JGoodies Common - JGoodies Common copyright notification

Description Also see

## Description

Stata uses portions of JGoodies Common with the express permission of the authors pursuant to the following notice:

The BSD License for the JGoodies Common

Copyright © 2009-2014 JGoodies Software GmbH. All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of JGoodies Software GmbH nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

## Also see <br> [R] copyright - Display copyright information

## Title

Copyright JGoodies Forms - JGoodies Forms copyright notification

Description Also see

## Description

Stata uses portions of JGoodies Forms with the express permission of the authors pursuant to the following notice:

The BSD License for the JGoodies Forms

Copyright © 2002-2014 JGoodies Software GmbH. All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of JGoodies Software GmbH nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

[^16]
## Title

Copyright JSON - JSON for Modern C++ copyright notification
Description Also see

## Description

Stata uses portions of JSON for Modern C++ with the express permission of the author pursuant to the following notice:
(c) 2013-2021, Niels Lohmann < mail@nlohmann.me>

All Rights Reserved
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## Also see

[ R ] copyright - Display copyright information

## Title

Copyright jsoup - jsoup copyright notification
Description Also see

## Description

Stata uses portions of jsoup with the express permission of the authors pursuant to the following notice:
(c) 2009-2017, Jonathan Hedley [jonathan@hedley.net](mailto:jonathan@hedley.net)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## Also see

[R] copyright - Display copyright information

## Title

Copyright LAPACK — LAPACK copyright notification

Description Also see

## Description

Stata uses portions of LAPACK, a linear algebra package, with the express permission of the authors pursuant to the following notice:

Copyright (c) 1992-2008 The University of Tennessee. All rights reserved.

- Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the following disclaimer, listed in this license in the documentation or other materials provided with the distribution or both.
- Neither the names of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

## Also see

[R] copyright - Display copyright information

## Title

Copyright libHaru - HARU copyright notification

## Description Also see

## Description

Stata uses portions of HARU with the express permission of the author pursuant to the following notice:

Copyright (C) 1999-2006 Takeshi Kanno
This software is provided 'as-is', without any express or implied warranty.
In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

## Also see

[R] copyright - Display copyright information

## Title

Copyright libpng — libpng copyright notification

## Description Also see

## Description

Stata uses portions of libpng with the express permission of the authors.
For the purposes of this acknowledgment, "Contributing Authors" is as defined by the copyright notice below.

StataCorp thanks and acknowledges the Contributing Authors of libpng and Group 42, Inc. for producing libpng and allowing its use in Stata and other software.

For more information about libpng, visit http://www.libpng.org/.
The full libpng copyright notice is

## COPYRIGHT NOTICE, DISCLAIMER, and LICENSE:

If you modify libpng you may insert additional notices immediately following this sentence.

This code is released under the libpng license.
libpng versions 1.2.6, August 15, 2004, through 1.6.16, December 22, 2014, are Copyright (c) 2004, 2006-2014 Glenn Randers-Pehrson, and are distributed according to the same disclaimer and license as libpng-1.2.5 with the following individual added to the list of Contributing Authors

Cosmin Truta
libpng versions 1.0.7, July 1, 2000, through 1.2.5 - October 3, 2002, are Copyright (C) 2000-2002 Glenn Randers-Pehrson, and are distributed according to the same disclaimer and license as libpng-1.0.6 with the following individuals added to the list of Contributing Authors

Simon-Pierre Cadieux
Eric S. Raymond
Gilles Vollant
and with the following additions to the disclaimer:
There is no warranty against interference with your enjoyment of the library or against infringement. There is no warranty that our efforts or the library will fulfill any of your particular purposes or needs. This library is provided with all faults, and the entire risk of satisfactory quality, performance, accuracy, and effort is with the user.
libpng versions 0.97 , January 1998, through 1.0.6, March 20, 2000, are Copyright (c) 1998, 1999 Glenn Randers-Pehrson, and are distributed according to the same disclaimer and license as libpng-0.96, with the following individuals added to the list of Contributing Authors:

Tom Lane
Glenn Randers-Pehrson
Willem van Schaik
libpng versions 0.89, June 1996, through 0.96, May 1997, are Copyright © 1996, 1997 Andreas Dilger Distributed according to the same disclaimer and license as libpng- 0.88 , with the following individuals added to the list of Contributing Authors:

John Bowler
Kevin Bracey
Sam Bushell
Magnus Holmgren
Greg Roelofs
Tom Tanner
libpng versions 0.5, May 1995, through 0.88, January 1996, are Copyright © 1995, 1996 Guy Eric Schalnat, Group 42, Inc.

For the purposes of this copyright and license, "Contributing Authors" is defined as the following set of individuals:

Andreas Dilger

Dave Martindale
Guy Eric Schalnat
Paul Schmidt
Tim Wegner
The PNG Reference Library is supplied "AS IS". The Contributing Authors and Group 42, Inc. disclaim all warranties, expressed or implied, including, without limitation, the warranties of merchantability and of fitness for any purpose. The Contributing Authors and Group 42, Inc. assume no liability for direct, indirect, incidental, special, exemplary, or consequential damages, which may result from the use of the PNG Reference Library, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this source code, or portions hereof, for any purpose, without fee, subject to the following restrictions:

1. The origin of this source code must not be misrepresented.
2. Altered versions must be plainly marked as such and must not be misrepresented as being the original source.
3. This Copyright notice may not be removed or altered from any source or altered source distribution.

The Contributing Authors and Group 42, Inc. specifically permit, without fee, and encourage the use of this source code as a component to supporting the PNG file format in commercial products. If you use this source code in a product, acknowledgment is not required but would be appreciated.

## Also see <br> [R] copyright - Display copyright information

## Title

Copyright Mersenne Twister - Mersenne Twister copyright notification

Description Also see

## Description

Stata uses portions of Mersenne Twister with the express permission of the author, pursuant to the following notice:

Commercial Use of Mersenne Twister
2001/4/6
Until 2001/4/6, MT had been distributed under GNU Public License, but after 2001/4/6, we decided to let MT be used for any purpose, including commercial use. 2002-versions mt 19937 ar.c, mt19937ar-cok.c are considered to be usable freely.
Copyright © 2004, Makoto Matsumoto and Takuji Nishimura, All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The names of its contributors may not be used to endorse or promote products derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

## Also see

[R] copyright - Display copyright information

## Title

Copyright MiG Layout - MiG Layout copyright notification

Description Also see

## Description

Stata uses portions of MiG Layout with the express permission of the author, pursuant to the following notice:

Copyright (c) 2004, Mikael Grev, MiG InfoCom AB. (miglayout (at) miginfocom (dot) com) All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met: Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution. Neither the name of the MiG InfoCom AB nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

## Also see

[R] copyright - Display copyright information

## Title

Copyright Parsington - Parsington copyright notification

Description Also see

## Description

Stata uses portions of Parsington with the express permission of the authors pursuant to the following notice:

Copyright © 2015-2019, Board of Regents of the University of Wisconsin-Madison. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

## Also see

[R] copyright - Display copyright information

## Title

Copyright PolyHook — PolyHook copyright notification

## Description Also see

## Description

Stata uses portions of the PolyHook library with the express permission of the author under the MIT License pursuant to the following notice:

## MIT License

Copyright (c) 2018 Stephen Eckels
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:
The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## Also see

[ R ] copyright - Display copyright information

## Title

Copyright ReadStat — ReadStat copyright notification

## Description Also see

## Description

Stata uses portions of ReadStat with the express permission of the author, pursuant to the following notice:

Copyright (c) 2013-2016 Evan Miller (except where otherwise noted)
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## Also see

[R] copyright - Display copyright information

## Title

Copyright Scintilla - Scintilla copyright notification

## Description Also see

## Description

Stata uses portions of Scintilla with the express permission of the author, pursuant to the following notice:

Copyright © 1998-2002 by Neil Hodgson [neilh@scintilla.org](mailto:neilh@scintilla.org)
All Rights Reserved
Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.
NEIL HODGSON DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE, INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS, IN NO EVENT SHALL NEIL HODGSON BE LIABLE FOR ANY SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

## Also see

[R] copyright - Display copyright information

## Title

Copyright slf4j — slf4j copyright notification

## Description Also see

## Description

Stata uses portions of slf4j with the express permission of the authors pursuant to the following notice:

Copyright (c) 2004-2008 QOS.ch All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## Also see

[R] copyright - Display copyright information

## Title

Copyright ttf2pt1 - ttf2pt1 copyright notification

## Description Also see

## Description

Stata uses portions of ttf2pt1 to convert TrueType fonts to PostScript fonts, with express permission of the authors, pursuant to the following notice:

```
Copyright © 1997-2003 by the AUTHORS:
    Andrew Weeks <ccsaw@bath.ac.uk>
    Frank M. Siegert <fms@this.net>
    Mark Heath <mheath@netspace.net.au>
    Thomas Henlich <thenlich@rcs.urz.tu-dresden.de>
    Sergey Babkin <babkin@users.sourceforge.net>,<sab123@hotmail.com>
    Turgut Uyar <uyar@cs.itu.edu.tr>
    Rihardas Hepas <rch@WriteMe.Com>
    Szalay Tamas <tomek@elender.hu>
    Johan Vromans <jvromans@ squirrel.nl>
    Petr Titera < P.Titera@sh.cvut.cz>
    Lei Wang <lwang@amath8.amt.ac.cn>
    Chen Xiangyang <chenxy@sun.ihep.ac.cn>
    Zvezdan Petkovic <z.petkovic@computer.org>
    Rigel <rigel863@yahoo.com>
```

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. All advertising materials mentioning features or use of this software must display the following acknowledgment: This product includes software developed by the TTF2PT1 Project and its contributors.

THIS SOFTWARE IS PROVIDED BY THE AUTHORS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHORS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

## Also see

[R] copyright - Display copyright information

## Title

Copyright Win32 Dark Mode - Win32 Dark Mode copyright notification
Description Also see

## Description

Stata uses portions of the Win32 Dark Mode library with the express permission of the author under the MIT License pursuant to the following notice:

## MIT License

Copyright (c) 2019 Richard Yu
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## Also see

[ R ] copyright - Display copyright information

## Title

Copyright zlib — zlib copyright notification

## Description Also see

## Description

Stata uses portions of zlib with the express permission of the authors.
StataCorp thanks and acknowledges the authors of zlib, Jean-loup Gailly and Mark Adler, for producing zlib and allowing its use in Stata and other software.

For more information about zlib, visit http://www.zlib.net/.
The full zlib copyright notice is

Copyright © 1995-2013 Jean-loup Gailly and Mark Adler
This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.
Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly
Mark Adler

## Also see

[R] copyright - Display copyright information

## Title

correlate - Correlations of variables

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options for correlate | Options for pwcorr | Remarks and examples | Stored results |
| Methods and formulas | References | Also see |  |

## Description

The correlate command displays the correlation matrix or covariance matrix for a group of variables. If varlist is not specified, the matrix is displayed for all variables in the dataset.
pwcorr displays all the pairwise correlation coefficients between the variables in varlist or, if varlist is not specified, all the variables in the dataset.

## Quick start

Correlation matrix for variables v1, v2, and v3
correlate v1 v2 v3
Same as above, but display covariances instead of correlations
correlate v1 v2 v3, covariance
Pairwise correlation coefficients between v1, v2, and v3
pwcorr v1 v2 v3
Also print significance level of each correlation coefficient
pwcorr v1 v2 v3, sig
Same as above, but star correlation coefficients significant at the 5\% level
pwcorr v1 v2 v3, sig star(.05)
Same as above, but use Bonferroni-adjusted significance levels
pwcorr v1 v2 v3, sig star(.05) bonferroni

## Menu

## correlate

Statistics $>$ Summaries, tables, and tests $>$ Summary and descriptive statistics $>$ Correlations and covariances

## pwcorr

Statistics $>$ Summaries, tables, and tests $>$ Summary and descriptive statistics $>$ Pairwise correlations

## Syntax

Display correlation matrix or covariance matrix
correlate $[$ varlist $][$ if $][$ in $][$ weight $][$, correlate_options $]$

Display all pairwise correlation coefficients

```
pwcorr [varlist] [if] [in] [weight] [, pwcorr_options]
```

correlate_options Description
Options
means display means, standard deviations, minimums, and maximums with matrix
noformat ignore display format associated with variables
covariance
display covariances
wrap
allow wide matrices to wrap
pwcorr_options Description
Main
obs
print number of observations for each entry
sig
listwise
casewise
print(\#)
star(\#)
bonferroni
sidak
print significance level for each entry
use listwise deletion to handle missing values
synonym for listwise
significance level for displaying coefficients
significance level for displaying with a star
use Bonferroni-adjusted significance level
use Šidák-adjusted significance level
varlist may contain time-series operators; see [U] 11.4.4 Time-series varlists. by and collect are allowed with correlate and pwcorr; see [U] 11.1.10 Prefix commands. aweights and fweights are allowed; see [U] 11.1.6 weight.

## Options for correlate

$\square$ Options
means displays summary statistics (means, standard deviations, minimums, and maximums) with the matrix.
noformat displays the summary statistics requested by the means option in g format, regardless of the display formats associated with the variables.
covariance displays the covariances rather than the correlation coefficients.
wrap requests that no action be taken on wide correlation matrices to make them readable. It prevents Stata from breaking wide matrices into pieces to enhance readability. You might want to specify this option if you are displaying results in a window wider than 80 characters. Then you may need to set linesize to however many characters you can display across a line; see $[R] \log$.

## Options for pwcorr

$\qquad$ Main
obs adds a line to each row of the matrix reporting the number of observations used to calculate the correlation coefficient.
sig adds a line to each row of the matrix reporting the significance level of each correlation coefficient.
listwise handles missing values through listwise deletion, meaning that the entire observation is omitted from the estimation sample if any of the variables in varlist is missing for that observation. By default, pwcorr handles missing values by pairwise deletion; all available observations are used to calculate each pairwise correlation without regard to whether variables outside that pair are missing.
correlate uses listwise deletion. Thus, listwise allows users of pwcorr to mimic correlate's treatment of missing values while retaining access to pwcorr's features.
casewise is a synonym for listwise.
print(\#) specifies the significance level of correlation coefficients to be printed. Correlation coefficients with larger significance levels are left blank in the matrix. Typing pwcorr, print (.10) would list only correlation coefficients significant at the $10 \%$ level or better.
star(\#) specifies the significance level of correlation coefficients to be starred. Typing pwcorr, star (.05) would star all correlation coefficients significant at the $5 \%$ level or better.
bonferroni makes the Bonferroni adjustment to calculated significance levels. This option affects printed significance levels and the print() and star() options. Thus, pwcorr, print(.05) bonferroni prints coefficients with Bonferroni-adjusted significance levels of 0.05 or less.
sidak makes the Šidák adjustment to calculated significance levels. This option affects printed significance levels and the print() and star() options. Thus, pwcorr, print(.05) sidak prints coefficients with Šidák-adjusted significance levels of 0.05 or less.

## Remarks and examples

Remarks are presented under the following headings:

```
correlate
pwcorr
Video example
```


## correlate

Typing correlate by itself produces a correlation matrix for all variables in the dataset. If you specify the varlist, a correlation matrix for just those variables is displayed.

## > Example 1

We have state data on demographic characteristics of the population. To obtain a correlation matrix, we type

|  | state | brate | pop | medage | division | region | mrgrate |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| state | 1.0000 |  |  |  |  |  |  |
| brate | 0.0208 | 1.0000 |  |  |  |  |  |
| pop | -0.0540 | -0.2830 | 1.0000 |  |  |  |  |
| medage | -0.0624 | -0.8800 | 0.3294 | 1.0000 |  |  |  |
| division | -0.1345 | 0.6356 | -0.1081 | -0.5207 | 1.0000 |  |  |
| region | -0.1339 | 0.6086 | -0.1515 | -0.5292 | 0.9688 | 1.0000 |  |
| mrgrate | 0.0509 | 0.0677 | -0.1502 | -0.0177 | 0.2280 | 0.2490 | 1.0000 |
| dvcrate | -0.0655 | 0.3508 | -0.2064 | -0.2229 | 0.5522 | 0.5682 | 0.7700 |
| medagesq | -0.0621 | -0.8609 | 0.3324 | 0.9984 | -0.5162 | -0.5239 | -0.0202 |
| dvcrate medagesq |  |  |  |  |  |  |  |
| dvcrate | 1.0000 |  |  |  |  |  |  |
| medagesq | -0.2192 | 1.0000 |  |  |  |  |  |

Because we did not specify the wrap option, Stata did its best to make the result readable by breaking the table into two parts.

To obtain the correlations between mrgrate, dvcrate, and medage, we type

```
. correlate mrgrate dvcrate medage
(obs=50)
```

|  | mrgrate | dvcrate | medage |
| ---: | ---: | ---: | ---: |
| mrgrate | 1.0000 |  |  |
| dvcrate | 0.7700 | 1.0000 |  |
| medage | -0.0177 | -0.2229 | 1.0000 |

## Example 2

The pop variable in example 1 represents the total population of the state. Thus, to obtain population-weighted correlations among mrgrate, dvcrate, and medage, we type

```
. correlate mrgrate dvcrate medage [w=pop]
(analytic weights assumed)
(sum of wgt is 225,907,472)
(obs=50)
\begin{tabular}{r|rrr} 
& mrgrate & dvcrate & medage \\
\hline mrgrate & 1.0000 & & \\
dvcrate & 0.5854 & 1.0000 & \\
medage & -0.1316 & -0.2833 & 1.0000
\end{tabular}
```

With the covariance option, correlate can be used to obtain covariance matrices, as well as correlation matrices, for both weighted and unweighted data.

## > Example 3

To obtain the matrix of covariances between mrgrate, dvcrate, and medage, we type correlate mrgrate dvcrate medage, covariance:

```
. correlate mrgrate dvcrate medage, covariance
(obs=50)
\begin{tabular}{r|rrr} 
& mrgrate & dvcrate & medage \\
\hline mrgrate & .000662 & & \\
dvcrate & .000063 & \(1.0 \mathrm{e}-05\) & \\
medage & -.000769 & -.001191 & 2.86775
\end{tabular}
```

We could have obtained the pop-weighted covariance matrix by typing correlate mrgrate dvcrate medage [w=pop], covariance.

## pwcorr

correlate calculates correlation coefficients by using casewise deletion; when you request correlations of variables $x_{1}, x_{2}, \ldots, x_{k}$, any observation for which any of $x_{1}, x_{2}, \ldots, x_{k}$ is missing is not used. Thus if $x_{3}$ and $x_{4}$ have no missing values, but $x_{2}$ is missing for half the data, the correlation between $x_{3}$ and $x_{4}$ is calculated using only the half of the data for which $x_{2}$ is not missing. Of course, you can obtain the correlation between $x_{3}$ and $x_{4}$ by using all the data by typing correlate $x_{3} x_{4}$. pwcorr makes obtaining such pairwise correlation coefficients easier.

## > Example 4

Using auto.dta, we investigate the correlation between several of the variables.


|  | mpg | price | headroom r | rear_s ${ }^{\sim}$ t | trunk | rep78 | foreign |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| mpg | 1.0000 |  |  |  |  |  |  |
| price | -0.4594* | 1.0000 |  |  |  |  |  |
| headroom | -0.4220* |  | 1.0000 |  |  |  |  |
| rear_seat | -0.5213* | 0.4194* | 0.5238* | 1.0000 |  |  |  |
| trunk | -0.5703* | 0.3143* | 0.6620* | 0.6480* | 1.0000 |  |  |
| rep78 | 0.3739* |  |  |  |  | 1.0000 |  |
| foreign | 0.3613* |  | -0.2939 | -0.2409 | -0.3594* | 0.5922* | 1.0000 |
| pwcorr mpg price headroom rear_seat trunk rep78 foreign, print(.05) bon mpg price headroom rear_s $\sim$ t trunk rep78 foreign |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| mpg | 1.0000 |  |  |  |  |  |  |
| price | -0.4594 | 1.0000 |  |  |  |  |  |
| headroom | -0.4220 | 1.0000 |  |  |  |  |  |
| rear_seat | -0.5213 | 0.4194 | 0.5238 | 1.0000 |  |  |  |
| trunk | -0.5703 |  | 0.6620 | 0.6480 | 1.0000 |  |  |
| rep78 | 0.3739 |  |  |  |  | 1.0000 |  |
| foreign | 0.3613 |  |  |  | -0.3594 | 0.5922 | 1.0000 |

## Technical note

The correlate command will report the correlation matrix of the data, but there are occasions when you need the matrix stored as a Stata matrix so that you can further manipulate it. You can obtain the matrix by typing
. matrix accum $\mathrm{R}=$ varlist, noconstant deviations
. matrix $R=\operatorname{corr}(R)$
The first line places the cross-product matrix of the data in matrix R . The second line converts that to a correlation matrix. Also see $[\mathrm{P}]$ matrix define and $[\mathrm{P}]$ matrix accum.

## Video example

Pearson's correlation coefficient in Stata

## Stored results

correlate stores the following in r() :
Scalars
$r(N) \quad$ number of observations
$r$ (rho) $\quad \rho$ (first and second variables)
r(cov_12) covariance (covariance only)
$r$ (Var_1) variance of first variable (covariance only)
$r$ (Var_2) variance of second variable (covariance only)
r(sum_w) sum of weights
Matrices
$r$ (C) correlation or covariance matrix
pwcorr stores the following in $r()$ :
Scalars
$r(N) \quad$ number of observations (first and second variables)
$r$ (rho) $\quad \rho$ (first and second variables)
Matrices
$r$ (Nobs) number of observations for each correlation coefficient
r(C) pairwise correlation matrix
r(sig) significance level of each correlation coefficient

## Methods and formulas

For a discussion of correlation, see, for instance, Snedecor and Cochran (1989, 177-195); for an introductory explanation using Stata examples, see Acock (2023, 205-210).

According to Snedecor and Cochran (1989, 180), the term "co-relation" was first proposed by Galton (1888). The product-moment correlation coefficient is often called the Pearson product-moment correlation coefficient because Pearson (1896) and Pearson and Filon (1898) were partially responsible for popularizing its use. See Stigler (1986) for information on the history of correlation.

The estimate of the product-moment correlation coefficient, $\rho$, is

$$
\widehat{\rho}=\frac{\sum_{i=1}^{n} w_{i}\left(x_{i}-\bar{x}\right)\left(y_{i}-\bar{y}\right)}{\sqrt{\sum_{i=1}^{n} w_{i}\left(x_{i}-\bar{x}\right)^{2}} \sqrt{\sum_{i=1}^{n} w_{i}\left(y_{i}-\bar{y}\right)^{2}}}
$$

where $w_{i}$ are the weights, if specified, or $w_{i}=1$ if weights are not specified. $\bar{x}=\left(\sum w_{i} x_{i}\right) /\left(\sum w_{i}\right)$ is the mean of $x$, and $\bar{y}$ is similarly defined.

The unadjusted significance level is calculated by pwcorr as

$$
p=2 * \operatorname{ttail}\left(n-2,|\widehat{\rho}| \sqrt{n-2} / \sqrt{1-\widehat{\rho}^{2}}\right)
$$

Let $v$ be the number of variables specified so that $k=v(v-1) / 2$ correlation coefficients are to be estimated. If bonferroni is specified, the adjusted significance level is $p^{\prime}=\min (1, k p)$. If sidak is specified, $p^{\prime}=\min \left\{1,1-(1-p)^{k}\right\}$. In both cases, see Methods and formulas in [R] oneway for a more complete description of the logic behind these adjustments.

Carlo Emilio Bonferroni (1892-1960) was born in Bergamo, Italy, in 1892. Before enrolling in the mathematics department at the University of Turin, he studied conducting and the piano at the Music Conservatory of Turin. Bonferroni, like many men of his generation, fought in World War I, during which he was part of The Engineer Corps of the Italian Army.

After the war, Bonferroni was appointed as an assistant professor at the Polytechnic University of Turin. He taught geometry, mechanics, and analysis. In 1923, he moved to Bari and began teaching at the Economics Institute, where he got to teach financial mathematics, a topic that was of great interest to him. In 1933, Bonferroni moved to Florence, where he was chair of the mathematics department until his death in 1960.
Bonferroni's interests had a large breadth. He published on actuarial mathematics, probability, statistics, analysis, geometry, and mechanics. His work on probability inequalities has been applied to simultaneous statistical inference. However, the application of Bonferroni's theory to the construction of confidence intervals is the work of Olive Jean Dunn.

Olive Jean Dunn (1915-2008) was born in the United States in 1915. She obtained her bachelor's degree in 1936, her master's degree in 1951, and her PhD in 1956, all from the University of California in Los Angeles (UCLA). After spending one year as an assistant professor at Iowa State College, she returned to UCLA to serve in the biostatistics and preventive medicine and health departments. She remained at UCLA for the rest of her career. Dunn died in 2008.

Dunn is best known for her application of Bonferroni's inequalities to construct corrections to confidence intervals for multiple comparisons. Although the literature refers to it as the Bonferroni correction, it is Dunn who developed the application we use today.

Dunn is also well known for her textbooks Basic Statistics: A Primer for the Biomedical Sciences, written in 1977 with later editions coauthored with Virginia A. Clark, and Applied Statistics: An Analysis of Variance and Regression, which was also coauthored with Clark.

In 1968, Dunn became a Fellow of the American Statistical Association. She also was a fellow of the American Public Health Association and the American Association for the Advancement of Science. In 1974, she was awarded the honor of UCLA Woman of Science.

Florence Nightingale David (1909-1993) was born in Ivington, England, to parents who were friends with Florence Nightingale, David's namesake. She began her studies in statistics under the direction of Karl Pearson at University College London and continued her studies under the direction of Jerzy Neyman. After receiving her doctorate in statistics in 1938, David became a senior statistician for various departments within the British military. She developed statistical models to forecast the toll on life and infrastructure that would occur if a large city were bombed. In 1938, she also published her book Tables of the Correlation Coefficient, dealing with the distributions of correlation coefficients. After the war, she returned to University College London, serving as a lecturer until her promotion to professor in 1962. In 1967, David joined the University of California-Riverside, eventually becoming chair of the Department of Statistics. One of her most well-known works is the book Games, Gods and Gambling: The Origins and History of Probability and Statistical Ideas from the Earliest Times to the Newtonian Era, a history of statistics. David published over 100 papers on topics including combinatorics, symmetric functions, the history of statistics, and applications of statistics, including ecological diversity. She published under the name F. N. David to avoid revealing her gender in a male-dominated profession.

Karl Pearson (1857-1936) studied mathematics at Cambridge. He was professor of applied mathematics (1884-1911) and eugenics (1911-1933) at University College London. His publications include literary, historical, philosophical, and religious topics. Statistics became his main interest in the early 1890s after he learned about its application to biological problems. His work centered on distribution theory, the method of moments, correlation, and regression. Pearson introduced the $\chi^{2}$ test and the terms coefficient of variation, contingency table, heteroskedastic, histogram, homoskedastic, kurtosis, mode, random sampling, random walk, skewness, standard deviation, and truncation. Despite many strong qualities, he also fell into prolonged disagreements with others, most notably, William Bateson and R. A. Fisher.

Zbyněk Šidák (1933-1999) was a notable Czech statistician and probabilist. He worked on Markov chains, rank tests, multivariate distribution theory and multiple-comparison methods, and he served as the chief editor of Applications of Mathematics.

## References

Acock, A. C. 2023. A Gentle Introduction to Stata. Rev. 6th ed. College Station, TX: Stata Press.
Dewey, M. E., and E. Seneta. 2001. Carlo Emilio Bonferroni. In Statisticians of the Centuries, ed. C. C. Heyde and E. Seneta, 411-414. New York: Springer.

Eisenhart, C. 1974. Pearson, Karl. In Vol. 10 of Dictionary of Scientific Biography, ed. C. C. Gillispie, 447-473. New York: Charles Scribner's Sons.
Galton, F. 1888. Co-relations and their measurement, chiefly from anthropometric data. Proceedings of the Royal Society of London 45: 135-145.

Pearson, K. 1896. Mathematical contributions to the theory of evolution-III. Regression, heredity, and panmixia. Philosophical Transactions of the Royal Society of London, Series A 187: 253-318.
Pearson, K., and L. N. G. Filon. 1898. Mathematical contributions to the theory of evolution. IV. On the probable errors of frequency constants and on the influence of random selection on variation and correlation. Philosophical Transactions of the Royal Society of London, Series A 191: 229-311.

Porter, T. M. 2004. Karl Pearson: The Scientific Life in a Statistical Age. Princeton, NJ: Princeton University Press.
Rodgers, J. L., and W. A. Nicewander. 1988. Thirteen ways to look at the correlation coefficient. American Statistician 42: 59-66. https://doi.org/10.1080/00031305.1988.10475524.

Rovine, M. J., and A. von Eye. 1997. A 14th way to look at a correlation coefficient: Correlation as the proportion of matches. American Statistician 51: 42-46. https://doi.org/10.1080/00031305.1997.10473586.
Seidler, J., J. Vondráček, and I. Saxl. 2000. The life and work of Zbyněk Šidák (1933-1999). Applications of Mathematics 45: 321-336. https://doi.org/10.1023/A:1022238410461.
Snedecor, G. W., and W. G. Cochran. 1989. Statistical Methods. 8th ed. Ames, IA: Iowa State University Press.
Stigler, S. M. 1986. The History of Statistics: The Measurement of Uncertainty before 1900. Cambridge, MA: Belknap Press.
Verardi, V., and C. Dehon. 2010. Multivariate outlier detection in Stata. Stata Journal 10: 259-266.
Weber, S. 2010. bacon: An effective way to detect outliers in multivariate data using Stata (and Mata). Stata Journal 10: 331-338.

## Also see

[R] esize - Effect size based on mean comparison
$[R]$ estat vce - Display covariance matrix estimates
$[\mathrm{R}]$ icc - Intraclass correlation coefficients
[R] pcorr - Partial and semipartial correlation coefficients
[R] spearman - Spearman's and Kendall's correlations
[R] summarize - Summary statistics
[R] tetrachoric - Tetrachoric correlations for binary variables

## Title

cpoisson - Censored Poisson regression

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

cpoisson fits a Poisson model of a count dependent variable with some censored values. The command can be used when the dependent variable is left-censored (has a lower limit), is right-censored (has an upper limit), or is interval-censored (has a lower and an upper limit).

## Quick start

Censored Poisson regression of y on x without options 11 () and ul (), equivalent to Poisson regression cpoisson y x

Add categorical variable a using factor-variable syntax, and specify censoring at an upper limit of 4 cpoisson y x i.a, ul(4)

Also specify a lower-censoring limit that varies across observations by using the variable lower cpoisson y x i.a, ul(4) ll(lower)

Add offset variable v , and report results as incidence-rate ratios
cpoisson y x i.a, ul(4) ll(lower) offset(v) irr
Constrain the coefficient for x to 2
constraint define $1 \mathrm{x}=2$
cpoisson y x i.a, ul(4) constraints(1)

## Menu

Statistics $>$ Count outcomes $>$ Censored Poisson regression

## Syntax

cpoisson depvar [indepvars] [if] [in] [weight] [, options]
options
Model
noconstant
$11[$ varname |\#) $]$
ul $[$ (varname |\#) $]$
exposure (varname $_{e}$ )
offset (varname ${ }_{o}$ )
constraints (constraints)
SE/Robust
vce (vcetype)

Reporting
level(\#)
irr
nocnsreport
display_options

Description
suppress constant term
left-censoring variable or limit
right-censoring variable or limit
include $\ln \left(\right.$ varname $\left._{e}\right)$ in model with coefficient constrained to 1 include varname ${ }_{o}$ in model with coefficient constrained to 1 apply specified linear constraints
vcetype may be oim, robust, cluster clustvar, bootstrap, or jackknife
set confidence level; default is level (95)
report incidence-rate ratios
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
control the maximization process; seldom used
keep collinear variables
display legend instead of statistics

Maximization
maximize_options
collinear
coeflegend
indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
depvar and indepvars may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bootstrap, by, collect, fp, jackknife, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands. Weights are not allowed with the bootstrap prefix; see $[R]$ bootstrap.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

$\qquad$ Model

## noconstant; see [R] Estimation options.

$11[($ varname $\mid \#)]$ and $u l[$ (varname $\mid \#)]$ indicate the lower and upper limits for censoring, respectively. Observations with depvar $\leq 11$ () are left-censored; observations with depvar $\geq \mathrm{ul}$ () are right-censored; and remaining observations are not censored. You do not have to specify the censoring values. If you specify 11 , the lower limit is the minimum of depvar. If you specify $u l$, the upper limit is the maximum of depvar.
exposure ( varname $_{e}$ ), offset ( varname $_{o}$ ), constraints (constraints); see [R] Estimation options.

## SE/Robust

vce(vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.

## Reporting

level(\#); see [R] Estimation options.
irr reports estimated coefficients transformed to incidence-rate ratios, that is, $e^{\beta_{i}}$ rather than $\beta_{i}$. Standard errors and confidence intervals are similarly transformed. This option affects how results are displayed, not how they are estimated. irr may be specified at estimation or when replaying previously estimated results.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fmt), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate (\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from (init_specs); see [R] Maximize. These options are seldom used.

The following options are available with cpoisson but are not shown in the dialog box:
collinear, coeflegend; see [R] Estimation options.

## Remarks and examples

Censored Poisson regression is a method for analyzing censored count data. One of the most common sources of censored count data is top coding, data that record only the value $x$ when $x$ or greater is observed. Not observing subjects for a sufficient period of time is another common cause.

Censored count data models have been studied by Terza (1985) and Brännäs (1992), among others. For an introduction to censored Poisson regression, see Cameron and Trivedi $(2005,2013)$ and Winkelmann (2008). Raciborski (2011) discusses a command for right-censored Poisson regression and presented Monte Carlo results indicating that the estimator performs well in finite samples. See Creel and Loomis (1990) and Gurmu and Trivedi (1996) for some examples of survey applications with top coding.

Censored data can be right-censored, left-censored, or interval-censored. Right-censoring occurs when we observe the covariates but only observe that the dependent variable is greater than or equal to an upper limit. Left-censoring occurs when we observe the covariates but only observe that the dependent variable is less than or equal to a lower limit.

When the dependent variable is censored, we must use estimation methods that account for this limitation. If we do not account for censoring when our data are censored, our estimates will not converge to the true values. More formally, failure to control for censoring when it exists leads to inconsistent parameter estimation.

Censored Poisson regression provides an alternative to standard Poisson regression that produces consistent estimates when the dependent variable is censored. If the dependent variable is not censored, standard Poisson regression may be more appropriate; see $[R]$ poisson.

Censoring differs from truncation. For censored observations, we observe complete covariate information but only a censored value of the dependent variable. When the data are truncated, we do not observe either the dependent variable or the covariates. Different research designs can give rise to censored data or truncated data.

For example, consider a study about the use of national parks. We could ask a random sample of people in the population how many national parks each has visited in the past year. Suppose we decide to record three for those individuals who visited three or more parks. In this case, individuals who visit four or more parks will have observations that are right-censored at three visits. Now suppose that instead of sending out surveys to a random sample from the population, we ask questions only of individuals who come to parks. We will have no information about individuals who do not visit at least one park, and the data will be truncated at zero visits.

Censoring and truncation are different statistical phenomenon and require different analytic methods. See [R] tpoisson for information on truncated Poisson regression.

## > Example 1: Poisson model with top-coded data

Imagine that we have collected survey data about how many times a household has visited the ABC amusement park from a random sample of households in the state in which ABC is located. Respondents were asked about the number of visits to the park in the last year (trips), their income (income), and the number of children in the household (children). The number of trips recorded in trips was top coded at "three or more" visits.

We model right-censored trips as a function of income and children.

| (Visits to the ABC amusement park) <br> . cpoisson trips income children, ul(3) |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Initial: Log likelihood $=-620.68749$ |  |  |  |  |  |  |
| Rescale: Log likelihood $=-620.68749$ |  |  |  |  |  |  |
| Iteration 0: Log likelihood $=-620.68749$ |  |  |  |  |  |  |
| Iteration 1: Log likelihood $=-600.96763$ |  |  |  |  |  |  |
| Iteration 2: Log likelihood $=-600.78416$ |  |  |  |  |  |  |
| Iteration 3: Log likelihood = -600.78415 |  |  |  |  |  |  |
| Censored Poisson regression |  |  |  |  | r of obs | $=500$ |
|  |  |  |  |  | Uncensored | $=278$ |
| Limits: Lower $=0$ |  |  |  |  | Left-censored | $=0$ |
| Upper $=3$ |  |  |  |  | ght-censored | 222 |
|  |  |  |  |  | i2(2) | $=49.29$ |
| Log likelihood = -600.78415 |  |  |  |  | > chi2 | $=0.0000$ |
| trips | Coefficient | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. | interval] |
| income | . 0740477 | . 0137653 | 5.38 | 0.000 | . 0470683 | . 1010272 |
| children | . 1346922 | . 028617 | 4.71 | 0.000 | . 078604 | . 1907805 |
| _cons | . 0033918 | . 1455473 | 0.02 | 0.981 | -. 2818756 | . 2886592 |

Both income and the number of children have positive effects on the expected number of trips to the amusement park. The estimated parameters provide the sign, but not the magnitude of the effect, because the model is nonlinear; see $[R]$ cpoisson postestimation.

## Stored results

cpoisson stores the following in e():
Scalars

```
e(N)
e(N_unc)
e(N_lc)
e(N_rc)
e(k)
e(k_eq)
e(k_eq_model)
e(k_dv)
e(df_m)
e(11)
e(11_0)
e(N_clust)
e(chi2)
e(p)
e(rank)
e(ic)
e(rc)
e(converged)
```

Macros
e(cmd)
e(cmdline)
e(depvar)
e(llopt)
e(ulopt)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(offset)
e(chi2type)
e(vce)
e(vcetype)
e(opt)
$e$ (which)
e(ml_method)
e(user)
e(technique)
e(properties)
e(predict)
e(marginsok)
$e$ (marginsnotok)
$e$ (asbalanced)
e(asobserved)
Matrices
e(b)
e(Cns)
e(ilog)
e(gradient)
e(V)
e(V_modelbased)

Functions
e(sample)
number of observations
number of uncensored observations
number of left-censored observations
number of right-censored observations
number of parameters
number of equations in $e(b)$
number of equations in overall model test
number of dependent variables
model degrees of freedom
log likelihood
log likelihood, constant-only model
number of clusters
$\chi^{2}$
$p$-value for model test
rank of e(V)
number of iterations
return code
1 if converged, 0 otherwise

## cpoisson

command as typed
name of dependent variable
contents of 11() , if specified
contents of ul(), if specified
weight type
weight expression
title in estimation output
name of cluster variable
linear offset variable
Wald or LR; type of model $\chi^{2}$ test
vcetype specified in vce()
title used to label Std. err.
type of optimization
max or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement predict
predictions allowed by margins
predictions disallowed by margins
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
constraints matrix
iteration $\log$ (up to 20 iterations)
gradient vector
variance-covariance matrix of the estimators
model-based variance
marks estimation sample

In addition to the above, the following is stored in r() :
Matrices
$r$ (table) matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

We let $y_{j}$ be the observed, interval-censored dependent variable for observation $j$ and let $y_{j}^{*}$ be the uncensored, latent dependent variable. When $y_{j}$ is not censored, it is the same as $y_{j}^{*}$. When $y_{j}$ is censored, only the censoring point is observed. Letting $L_{j}$ denote the left-censoring point (lower limit) and $U_{j}$ denote the right-censoring point (upper limit), we see that

$$
y_{j}= \begin{cases}L_{j} & \text { if } y_{j}^{*} \leq L_{j} \\ y_{j}^{*} & \text { if } L_{j}<y_{j}^{*}<U_{j} \\ U_{j} & \text { if } y_{j}^{*} \geq U_{j}\end{cases}
$$

Note that $L_{j}$ and $U_{j}$ may vary over the observations so that individuals may have different left- and right-censoring points.

Although cpoisson may be used with data that are left-censored, right-censored, or censored from both sides (which is known as interval-censored), we present the formulas for the interval-censored case because it applies to all three cases.

Let $f\left(y_{j} \mid \mathbf{x}_{j}\right)$ denote the probability mass function of the Poisson distribution. Defining $\xi_{j}=\mathbf{x}_{j} \boldsymbol{\beta}+$ offset $_{j}$ implies that the conditional mean of the uncensored variable is given by $E\left(y_{j}^{*} \mid \mathbf{x}_{j}\right)=\exp \left(\xi_{j}\right)$. The log likelihood for observation $j$ is given by

$$
l_{j}=w_{j}\left[d_{j}\left\{-\exp \left(\xi_{j}\right)+y_{j} \xi_{j}-\ln \left(y_{j}!\right)\right\}+\left(1-d_{j}\right) \ln \left\{1-\sum_{k=0}^{U_{j}-1} f\left(k \mid \mathbf{x}_{j}\right)+\sum_{k=0}^{L_{j}} f\left(k \mid \mathbf{x}_{j}\right)\right\}\right]
$$

where $d_{j}$ equals 1 when $L_{j}<y_{j}^{*}<U_{j}$ and equals 0 when $y_{j}^{*} \leq L_{j}$ or $y_{j}^{*} \geq U_{j}$. The log likelihood is thus

$$
\ln L=\sum_{j=1}^{N} l_{j}
$$

This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce (robust) and vce(cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.
cpoisson also supports estimation with survey data. For details on variance-covariance estimates with survey data, see [SVY] Variance estimation.

## References

Brännäs, K. 1992. Limited dependent Poisson regression. Journal of the Royal Statistical Society, Series D 41: 413-423. https://doi.org/10.2307/2349006.
Cameron, A. C., and P. K. Trivedi. 2005. Microeconometrics: Methods and Applications. New York: Cambridge University Press.
——. 2013. Regression Analysis of Count Data. 2nd ed. New York: Cambridge University Press.
Creel, M. D., and J. B. Loomis. 1990. Theoretical and empirical advantages of truncated count data estimators for analysis of deer hunting in California. American Journal of Agricultural Economics 72: 434-441. https://doi.org/10.2307/1242345.
Farbmacher, H. 2011. Estimation of hurdle models for overdispersed count data. Stata Journal 11: 82-94.
Gurmu, S., and P. K. Trivedi. 1996. Excess zeros in count models for recreational trips. Journal of Business and Economic Statistics 14: 469-477. https://doi.org/10.2307/1392255.
Raciborski, R. 2011. Right-censored Poisson regression model. Stata Journal 11: 95-105.
Terza, J. V. 1985. A tobit-type estimator for the censored Poisson regression model. Economics Letters 18: 361-365. https://doi.org/10.1016/0165-1765(85)90053-9.

Winkelmann, R. 2008. Econometric Analysis of Count Data. 5th ed. Berlin: Springer.

## Also see

[R] cpoisson postestimation - Postestimation tools for cpoisson
[R] nbreg - Negative binomial regression
[R] poisson - Poisson regression
[R] tnbreg - Truncated negative binomial regression
[R] tpoisson - Truncated Poisson regression
[R] zinb - Zero-inflated negative binomial regression
[R] zip — Zero-inflated Poisson regression
[SVY] svy estimation - Estimation commands for survey data
[XT] xtpoisson - Fixed-effects, random-effects, and population-averaged Poisson models
[U] 20 Estimation and postestimation commands

## Title

cpoisson postestimation - Postestimation tools for cpoisson

| Postestimation commands | predict | margins |
| :--- | :--- | :--- | Remarks and examples

## Postestimation commands

The following postestimation commands are available after cpoisson:
\(\left.$$
\begin{array}{ll}\text { Command } & \text { Description } \\
\hline \text { contrast } & \begin{array}{l}\text { contrasts and ANOVA-style joint tests of estimates } \\
\text { estat ic }\end{array}
$$ <br>
Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian in- <br>

formation criteria (AIC, CAIC, AICc, and BIC)\end{array}\right]\)| estat vummarize | summary statistics for the estimation sample |
| :--- | :--- |
| estat (svy) | variance-covariance matrix of the estimators (VCE) |
| postestimation statistics for survey data |  |
| estable | cataloging estimation results |
| * table of estimation results |  |

[^17]
## predict

## Description for predict

predict creates a new variable containing predictions such as numbers of events, incidence rates, conditional means, probabilities, conditional probabilities, linear predictions, standard errors, and the equation-level score.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

predict $[$ type $]$ newvar $[i f][i n][$, statistic nooffset $]$
statistic Description

| Main |  |
| :---: | :---: |
| n | number of events; the default |
| ir | incidence rate |
| cm | conditional mean, $E\left(y_{j} \mid y_{j}>L_{j}\right), E\left(y_{j} \mid y_{j}<U_{j}\right)$, or $E\left(y_{j} \mid L_{j}<y_{j}<U_{j}\right)$ |
| $\mathrm{pr}(n)$ | probability $\operatorname{Pr}\left(y_{j}=n\right)$ |
| $\mathrm{pr}(a, b)$ | probability $\operatorname{Pr}\left(a \leq y_{j} \leq b\right)$ |
| $\operatorname{cpr}(n)$ | conditional probability $\operatorname{Pr}\left(y_{j}=n \mid y_{j}>L_{j}\right), \operatorname{Pr}\left(y_{j}=n \mid y_{j}<U_{j}\right)$, or $\operatorname{Pr}\left(y_{j}=n \mid L_{j}<y_{j}<U_{j}\right)$ |
| $\operatorname{cpr}(a, b)$ | $\begin{aligned} & \text { conditional probability } \operatorname{Pr}\left(a \leq y_{j} \leq b \mid y_{j}>L_{j}\right), \operatorname{Pr}\left(a \leq y_{j} \leq b \mid y_{j}<U_{j}\right), \\ & \quad \text { or } \operatorname{Pr}\left(a \leq y_{j} \leq b \mid L_{j}<y_{j}<U_{j}\right) \end{aligned}$ |
| xb | linear prediction |
| stdp | standard error of the linear prediction |
| score | first derivative of the log likelihood with respect to $\mathbf{x}_{j} \boldsymbol{\beta}$ |

These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

## Options for predict

$\qquad$ Main
n , the default, calculates the predicted number of events, which is $\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}\right)$ if neither offset () nor exposure() was specified when the model was fit; $\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}+\right.$ offset $\left._{j}\right)$ if offset () was specified; or $\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}\right) \times$ exposure $_{j}$ if exposure() was specified.
ir calculates the incidence rate $\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}\right)$, which is the predicted number of events when exposure is 1 . This is equivalent to specifying both the n and the nooffset options.
cm calculates the conditional mean,

$$
E\left(y_{j} \mid \Omega_{j}\right)=\frac{E\left(y_{j}\right)}{\operatorname{Pr}\left(\Omega_{j}\right)}
$$

where $\Omega_{j}$ represents $y_{j}>L_{j}$ for a left-censored model, $y_{j}<U_{j}$ for a right-censored model, and $L_{j}<y_{j}<U_{j}$ for an interval-censored model. $L_{j}$ is the left-censoring point found in e(llopt), and $U_{j}$ is the right-censoring point found in e(ulopt).
$\operatorname{pr}(n)$ calculates the probability $\operatorname{Pr}\left(y_{j}=n\right)$, where $n$ is a nonnegative integer that may be specified as a number or a variable.
$\operatorname{pr}(a, b)$ calculates the probability $\operatorname{Pr}\left(a \leq y_{j} \leq b\right)$, where $a$ and $b$ are nonnegative integers that may be specified as numbers or variables;
$b$ missing ( $b \geq$.) means $+\infty$;
$\operatorname{pr}(20,$.$) calculates \operatorname{Pr}\left(y_{j} \geq 20\right)$;
$\operatorname{pr}(20, b)$ calculates $\operatorname{Pr}\left(y_{j} \geq 20\right)$ in observations for which $b \geq$. and calculates
$\operatorname{Pr}\left(20 \leq y_{j} \leq b\right)$ elsewhere.
$\operatorname{pr}(., b)$ produces a syntax error. A missing value in an observation of the variable $a$ causes a missing value in that observation for $\operatorname{pr}(a, b)$.
$\mathrm{cpr}(n)$ calculates the conditional probability $\operatorname{Pr}\left(y_{j}=n \mid \Omega_{j}\right)$, where $\Omega_{j}$ represents $y_{j}>L_{j}$ for a left-censored model, $y_{j}<U_{j}$ for a right-censored model, and $L_{j}<y_{j}<U_{j}$ for an intervalcensored model. $L_{j}$ is the left-censoring point found in e(llopt), and $U_{j}$ is the right-censoring point found in e(ulopt). $n$ is an integer in the noncensored range.
$\operatorname{cpr}(a, b)$ calculates the conditional probability $\operatorname{Pr}\left(a \leq y_{j} \leq b \mid \Omega_{j}\right)$, where $\Omega_{j}$ represents $y_{j}>L_{j}$ for a left-censored model, $y_{j}<U_{j}$ for a right-censored model, and $L_{j}<y_{j}<U_{j}$ for an intervalcensored model. $L_{j}$ is the left-censoring point found in e(llopt), and $U_{j}$ is the right-censoring point found in e (ulopt). $a$ and $b$ must fall in the noncensored range if they are not missing. A missing value in an observation of the variable $a$ causes a missing value in that observation for $\operatorname{cpr}(a, b)$.
xb calculates the linear prediction, which is $\mathbf{x}_{j} \boldsymbol{\beta}$ if neither offset () nor exposure() was specified when the model was fit; $\mathbf{x}_{j} \boldsymbol{\beta}+$ offset $_{j}$ if offset() was specified; or $\mathbf{x}_{j} \boldsymbol{\beta}+\ln \left(\operatorname{exposure}_{j}\right)$ if exposure() was specified; see nooffset below.
stdp calculates the standard error of the linear prediction.
score calculates the equation-level score, $\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)$.
nooffset is relevant only if you specified offset() or exposure() when you fit the model. It modifies the calculations made by predict so that they ignore the offset or exposure variable; the linear prediction is treated as $\mathbf{x}_{j} \boldsymbol{\beta}$ rather than as $\mathbf{x}_{j} \boldsymbol{\beta}+$ offset $_{j}$ or $\mathbf{x}_{j} \boldsymbol{\beta}+\ln \left(\right.$ exposure $\left._{j}\right)$. Specifying predict..., nooffset is equivalent to specifying predict..., ir.

## margins

## Description for margins

margins estimates margins of response for numbers of events, incidence rates, conditional means, probabilities, and linear predictions.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins


statistic
Description

| n | number of events; the default <br> ir <br> incidence rate |
| :--- | :--- |
| cm | conditional mean, $E\left(y_{j} \mid y_{j}>L_{j}\right), E\left(y_{j} \mid y_{j}<U_{j}\right)$, <br> or $E\left(y_{j} \mid L_{j}<y_{j}<U_{j}\right)$ |
| $\operatorname{pr}(n)$ | probability $\operatorname{Pr}\left(y_{j}=n\right)$ <br> $\operatorname{pr}(a, b)$ <br> $\operatorname{probability~} \operatorname{Pr}\left(a \leq y_{j} \leq b\right)$ <br> conditional probability $\operatorname{Pr}\left(y_{j}=n \mid y_{j}>L_{j}\right), \operatorname{Pr}\left(y_{j}=n \mid y_{j}<U_{j}\right)$, <br> $\quad$ or $\operatorname{Pr}\left(y_{j}=n \mid L_{j}<y_{j}<U_{j}\right)$ |
| $\operatorname{cpr}(a, b)$ | conditional probability $\operatorname{Pr}\left(a \leq y_{j} \leq b \mid y_{j}>L_{j}\right), \operatorname{Pr}\left(a \leq y_{j} \leq b \mid y_{j}<U_{j}\right)$, <br> or $\operatorname{Pr}\left(a \leq y_{j} \leq b \mid L_{j}<y_{j}<U_{j}\right)$ |
| xb | linear prediction <br> stdp |
| not allowed with margins |  |
| score | not allowed with margins |

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

## Remarks and examples

> Example 1: Obtaining marginal effects
Continuing with example 1 of $[\mathrm{R}]$ cpoisson, we estimate the effect of having another child on the uncensored number of trips to amusement parks. We use margins to estimate the average number of trips when each household has its actual number of children and when each household has one additional child. We include the post option so that we can use the results in subsequent commands.

```
. use https://www.stata-press.com/data/r18/trips
(Visits to the ABC amusement park)
. cpoisson trips income children, ul(3)
    (output omitted)
```

```
. margins, at(children = generate(children))
> at(children = generate(children+1)) post
Predictive margins Number of obs = 500
Model VCE: OIM
Expression: Predicted number of events, predict()
1._at: children = children
2._at: children = children+1
```

|  | Delta-method <br> std. err. |  |  |  | z | $\mathrm{P}>\|\mathrm{z}\|$ |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Margin | [95\% conf. interval] |  |  |  |  |  |
| _at |  |  |  |  |  |  |
| 1 | 2.525517 | .0836237 | 30.20 | 0.000 | 2.361618 | 2.689417 |
| 2 | 2.889658 | .1337997 | 21.60 | 0.000 | 2.627416 | 3.151901 |

An average of 2.53 trips are taken when each household has its observed number of children. If each household has one additional child, then the average number of trips increases to 2.89 .

We now use contrast to compute the effect of having an additional child. The Wald test, in this case, is superfluous, so we suppress it with the nowald option.

```
. contrast r._at, nowald
Contrasts of predictive margins Number of obs = 500
Model VCE: OIM
Expression: Predicted number of events, predict()
1._at: children = children
2._at: children = children+1
```

|  | Delta-method <br> std. err. |  |  |  |
| :---: | :---: | :---: | :---: | :---: |
| Contrast | [95\% conf. interval] |  |  |  |
| (2 vs 1) at | .3641407 | .0849951 | .1975535 | .530728 |

Adding one child to each household increases the average by 0.36 trips.

## Methods and formulas

Using the notation under Methods and formulas of $[R]$ cpoisson, we see that the equation-level score is given by

$$
\begin{aligned}
\operatorname{score}(\mathbf{x} \boldsymbol{\beta})_{j}= & d_{j}\left\{-\exp \left(\xi_{j}\right)+y_{j}\right\} \\
& +\left(1-d_{j}\right) \frac{\Psi_{1}\left(L_{j} \mid \mathbf{x}_{j}\right)-\Psi_{1}\left(U_{j}-1 \mid \mathbf{x}_{j}\right)}{1-F\left(U_{j}-1 \mid \mathbf{x}_{j}\right)+F\left(L_{j} \mid \mathbf{x}_{j}\right)}
\end{aligned}
$$

where $\Psi_{1}(C)=\sum_{k=0}^{C} f\left(k \mid \mathbf{x}_{j}\right)\left\{k-\exp \left(\xi_{j}\right)\right\} ; f\left(y_{j} \mid \mathbf{x}_{j}\right)$ and $F\left(y_{j} \mid \mathbf{x}_{j}\right)$ denote the probability mass function and the cumulative distribution function of the Poisson, respectively. $L_{j}$ is the left-censoring point found in e(llopt), and $U_{j}$ is the right-censoring point found in e(ulopt).

## Also see

[R] cpoisson - Censored Poisson regression
[U] 20 Estimation and postestimation commands

## Title

cumul - Cumulative distribution

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Acknowledgment | References |
| Also see |  |  |  |

## Description

cumul creates newvar, defined as the empirical cumulative distribution function of varname.

## Quick start

Create new variable ecd containing the empirical cumulative distribution of v cumul v, gen(ecd)

Use frequency as the unit for $v$ to generate ecdf
cumul v, gen(ecdf) freq
Give equal values of $v$ the same value in generated ecde
cumul v, gen(ecde) equal
Graph the empirical cumulative distribution of $v$
line ecd v, sort
Graph the distributions of variables v1 and v2
cumul v1, gen(ecd1) equal
cumul v2, gen(ecd2) equal
stack ecd1 v1 ecd2 v2, into(ecd v) wide clear
line ecd1 ecd2 v, sort

## Menu

Statistics $>$ Summaries, tables, and tests $>$ Distributional plots and tests $>$ Generate cumulative distribution

## Syntax

cumul varname $[$ if $][$ in $]$ [weight], generate(newvar) [options]
options
Description
Main

| * generate (newvar) | create variable newvar |
| :--- | :--- |
| freq | use frequency units for cumulative |
| equal | generate equal cumulatives for tied values |

* generate (newvar) is required.
by is allowed; see [D] by.
fweights and aweights are allowed; see [U] 11.1.6 weight.


## Options

$\qquad$ Main
generate (newvar) is required. It specifies the name of the new variable to be created.
freq specifies that the cumulative be in frequency units; otherwise, it is normalized so that newvar
is 1 for the largest value of varname.
equal requests that observations with equal values in varname get the same cumulative value in newvar.

## Remarks and examples

## Example 1

cumul is most often used with graph to graph the empirical cumulative distribution. For instance, we have data on the median family income of 957 US cities:

```
. use https://www.stata-press.com/data/r18/hsng
(1980 Census housing data)
. cumul faminc, gen(cum)
. sort cum
. line cum faminc, ytitle("") xlabel(, format(%6.Of))
> title("Cumulative of median family income")
> subtitle("1980 Census, }957\mathrm{ US cities")
```

Cumulative of median family income
1980 Census, 957 US cities


It would have been enough to type line cum faminc, but we wanted to make the graph look better; see [G-2] graph twoway line.

If we had wanted a weighted cumulative, we would have typed cumul faminc [w=pop] at the first step.

## > Example 2

To graph two (or more) cumulatives on the same graph, use cumul and stack; see [D] stack. For instance, we have data on the average January and July temperatures of 956 US cities:

```
. use https://www.stata-press.com/data/r18/citytemp, clear
(City temperature data)
. cumul tempjan, gen(cjan)
. cumul tempjuly, gen(cjuly)
. stack cjan tempjan cjuly tempjuly, into(c temp) wide clear
. line cjan cjuly temp, sort ytitle("") xtitle("Temperature (F)")
> title("Cumulatives:" "Average January and July temperatures")
> subtitle("956 US cities") legend(label(1 January) label(2 July))
```

Cumulatives:
Average January and July temperatures
956 US cities


As before, it would have been enough to type line cjan cjuly temp, sort. See [D] stack for an explanation of how the stack command works.

## - Technical note

According to Beniger and Robyn (1978), Fourier (1821) published the first graph of a cumulative frequency distribution, which was later given the name "ogive" by Galton (1875).

Jean Baptiste Joseph Fourier (1768-1830) was born in Auxerre in France. As a young man, Fourier became entangled in the complications of the French Revolution. As a result, he was arrested and put into prison, where he feared he might meet his end at the guillotine. When he was not in prison, he was studying, researching, and teaching mathematics. Later, he served Napolean's army in Egypt as a scientific adviser. Upon his return to France in 1801, he was appointed Prefect of the Department of Isère. While prefect, Fourier worked on the mathematical basis of the theory of heat, which is based on what are now called Fourier series. This work was published in 1822, despite the skepticism of Lagrange, Laplace, Legendre, and others-who found the work lacking in generality and even rigor-and disagreements of both priority and substance with Biot and Poisson.

## Acknowledgment

The equal option was added by Nicholas J. Cox of the Department of Geography at Durham University, UK, who is coeditor of the Stata Journal and author of Speaking Stata Graphics.

## References

Beniger, J. R., and D. L. Robyn. 1978. Quantitative graphics in statistics: A brief history. American Statistician 32: 1-11. https://doi.org/10.2307/2683467.
Fourier, J. B. J. 1821. Notions générales, sur la population. Recherches Statistiques sur la Ville de Paris et le Département de la Seine 1: 1-70.
Galton, F. 1875. Statistics by intercomparison, with remarks on the law of frequency of error. Philosophical Magazine 49: 33-46. https://doi.org/10.1080/14786447508641172.

Wilk, M. B., and R. Gnanadesikan. 1968. Probability plotting methods for the analysis of data. Biometrika 55: 1-17. https://doi.org/10.2307/2334448.

## Also see

[R] Diagnostic plots - Distributional diagnostic plots
[R] kdensity - Univariate kernel density estimation
[D] stack - Stack data

## Title

cusum - Cusum plots and tests for binary variables

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Acknowledgment |
| Reference | Also see |  |  |

## Description

cusum graphs the cumulative sum (cusum) of a binary (0/1) variable, yvar, against a (usually) continuous variable, xvar.

## Quick start

Cusum statistics for binary variable y and graph of cumulative sum against values of x cusum y x

Also generate cs to store the cumulative sum

```
cusum y x, generate(cs)
```

Set the seed first for reproducible results
set seed 87534690
cusum $y$ x, generate(cs)
Cumulative sum of y against a variable containing fitted values yhat
cusum y x, yfit(yhat)

## Menu

Statistics $>$ Other $>$ Quality control $>$ Cusum plots and tests for binary variables

## Syntax

```
cusum yvar xvar [if] [in] [, options]
```

options
Description
Main
generate (newvar) save cumulative sum in newvar
yfit (fitvar) calculate cumulative sum against fitvar
nograph suppress the plot
nocalc
suppress cusum test statistics
Cusum plot
connect_options
affect the rendition of the plotted line
Add plots
addplot (plot) add plots to the generated graph
Y axis, X axis, Titles, Legend, Overall
twoway_options any options other than by() documented in [G-3] twoway_options
collect is allowed; see [U] 11.1.10 Prefix commands.

## Options

## Main

generate (newvar) saves the cusum in newvar.
yfit (fitvar) calculates a cusum against fitvar, that is, the running sums of the "residuals" fitvar minus yvar. Typically, fitvar is the predicted probability of a positive outcome obtained from a logistic regression analysis.
nograph suppresses the plot.
nocalc suppresses calculation of the cusum test statistics.
$\qquad$ Cusum plot
connect_options affect the rendition of the plotted line; see [G-3] connect_options.

```
            Add plots
    addplot (plot) provides a way to add other plots to the generated graph. See [G-3] addplot_option.
```

$\qquad$ Y axis, X axis, Titles, Legend, Overall
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

## Remarks and examples

The cusum is the running sum of the proportion of ones in the sample, a constant number, minus yvar,

$$
c_{j}=\sum_{k=1}^{j} f-y \operatorname{var}_{(k)}, \quad 1 \leq j \leq N
$$

where $f=\left(\sum y v a r\right) / N$ and $y v a r_{(k)}$ refers to the corresponding value of $y v a r$ when $x v a r$ is placed in ascending order: $\operatorname{xvar}_{(k+1)} \geq \operatorname{xvar}_{(k)}$. Tied values of xvar are broken at random. If you want them broken the same way in two runs, you must set the random-number seed to the same value before giving the cusum command; see [R] set seed.

A U-shaped or inverted U-shaped cusum indicates, respectively, a negative or a positive trend of $y v a r$ with xvar. A sinusoidal shape is evidence of a nonmonotonic (for example, quadratic) trend. cusum displays the maximum absolute cusum for monotonic and nonmonotonic trends of yvar on xvar. These are nonparametric tests of departure from randomness of yvar with respect to xvar. Approximate values for the tests are given.

## > Example 1

For the automobile dataset, auto.dta, we wish to investigate the relationship between foreign ( $0=$ domestic, $1=$ foreign) and car weight as follows:
. use https://www.stata-press.com/data/r18/auto

| (1978 automobile data) |  |  |  |  |  |  |  |  |
| :--- | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| . cusum foreign weight |  |  |  |  |  |  |  |  |
| Variable | Obs | $\operatorname{Pr}(1)$ | CusumL | zL | Pr>zL | CusumQ | zQ | Pr>zQ |
| foreign | 74 | 0.2973 | 10.30 | 3.963 | 0.000 | 2.92 | 0.064 | 0.475 |



The resulting plot, which is U-shaped, suggests a negative monotonic relationship. The trend is confirmed by a highly significant linear cusum statistic, labeled CusumL in the output above.

Some $29.73 \%$ of the cars are foreign (coded 1). The proportion of foreign cars diminishes with increasing weight. The domestic cars are crudely heavier than the foreign ones. We could have discovered that by typing table foreign, statistics(mean weight), but such an approach does not give the full picture of the relationship. The quadratic cusum (CusumQ) is not significant, so we do not suspect any tendency for the very heavy cars to be foreign rather than domestic. A slightly
enhanced version of the plot shows the preponderance of domestic (coded 0) cars at the heavy end of the weight axis:

| Clabel values foreign |
| :--- |
| . cusum foreign weight, |
| V (none) |
| Variable |
|  |
| foreign |



The example is, of course, artificial, because we would not really try to model the probability of a car being foreign given its weight.

## Stored results

cusum stores the following in $r()$ :
Scalars

| $r(N)$ | number of observations | $r\left(P \_z l\right)$ | $p$-value for test (linear) |
| :--- | :--- | :--- | :--- |
| $r(p r o p 1)$ | proportion of positive outcomes | $r(c u s u m q)$ | quadratic cusum |
| $r($ cusuml $)$ | cusum | $r(z q)$ | test (quadratic) |
| $r(z l)$ | test (linear) | $r\left(P_{-z q)}\right.$ | $p$-value for test (quadratic) |

## Acknowledgment

cusum was written by Patrick Royston of the MRC Clinical Trials Unit, London, and coauthor of the Stata Press book Flexible Parametric Survival Analysis Using Stata: Beyond the Cox Model.

## Reference

Royston, P. 1992. The use of cusums and other techniques in modelling continuous covariates in logistic regression. Statistics in Medicine 11: 1115-1129. https://doi.org/10.1002/sim. 4780110813.

## Also see

[R] logistic - Logistic regression, reporting odds ratios
[R] logit - Logistic regression, reporting coefficients
[R] probit - Probit regression

## Title

db — Launch dialog

> Description Syntax Options Remarks and examples Also see

## Description

db opens the dialog box for the specified command. Programmers who wish to allow the launching of dialogs from a help file, see [P] smcl for information on the dialog SMCL directive.
set maxdb sets the maximum number of dialog boxes whose contents are remembered from one invocation to the next during a session. The default value of maxdb is 50 .

## Syntax

Syntax for $d b$
db commandname

For programmers
db commandname [, message(string) debug dryrun]
Set system parameter

```
set maxdb # [, permanently]
```

where \# must be between 5 and 1,000 .

## Options

message (string) specifies that string be passed to the dialog box, where it can be referred to from the __MESSAGE STRING property.
debug specifies that the underlying dialog box be loaded with debug messaging turned on.
dryrun specifies that, rather than launching the dialog, db show the commands it would issue to launch the dialog.
permanently specifies that, in addition to making the change right now, the maxdb setting be remembered and become the default setting when you invoke Stata.

## Remarks and examples

The usual way to launch a dialog is to open the Data, Graphics, or Statistics menu and to make your selection from there. When you know the name of the command that you want to run, however, db provides a way to invoke the dialog from the command line.
db follows the same abbreviation rules that Stata's command-line interface follows. So, to launch the dialog for regress, you can type

[^18]or
. db reg
Say that you use the dialog box for regress, either by selecting

## Statistics > Linear models and related > Linear regression

or by typing
. db regress
You fit a regression.
Much later during the session, you return to the regress dialog box. It will have the contents as you left them if 1) you have not typed clear all between the first and second invocations; 2) you have not typed discard between the two invocations; and 3) you have not used more than 50 different dialog boxes-regardless of how many times you have used each-between the first and second invocations of regress. If you use 51 or more, the contents of the regress dialog box will be forgotten.
set maxdb determines how many different dialog boxes are remembered. A dialog box takes, on average, about 20 KB of memory, so the 50 default corresponds to allowing dialog boxes to consume about 1 MB of memory.

## Also see

[R] query - Display system parameters

## Title

demandsys - Estimation of flexible demand systems

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| Acknowledgment | References | Also see |  |

## Description

demandsys fits demand systems, sets of equations derived from economic theory that describe consumers' purchases of various goods or services. demandsys allows you to fit eight different demand systems, including the Cobb-Douglas system, the almost ideal demand system (AIDS) of Deaton and Muellbauer (1980b), the translog indirect utility demand system of Christensen, Jorgenson, and Lau (1975), and variants of the latter two. You can also include demographic variables that affect a consumer's or household's demands.

## Quick start

Fit an AIDS demand system with four goods with expenditure shares w1, w2, w3, and w4; prices, p1, $\mathrm{p} 2, \mathrm{p} 3$, and p 4 ; and total expenditure, totexp

```
demandsys aids w1 w2 w3 w4, prices(p1 p2 p3 p4) ///
expenditure(totexp)
```

Same as above, reporting Marshallian (uncompensated) elasticities rather than coefficients

```
demandsys aids w1 w2 w3 w4, prices(p1 p2 p3 p4) ///
expenditure(totexp) elasticities(uncompensated) ///
nocoeftable
```

Same as above, labeling goods in output for easier interpretation

```
demandsys aids w1 w2 w3 w4, prices(p1 p2 p3 p4) ///
expenditure(totexp) elasticities(uncompensated) ///
nocoeftable labels("apples bananas carrots dates")
```

Fit a quadratic AIDS model with four goods, controlling for x 1 and x 2 using demographic translation

```
demandsys quaids w1 w2 w3 w4, prices(p1 p2 p3 p4)
                                    ///
        expenditure(totexp) demographics(x1 x2)
```

Same as above, but use demographic scaling rather than translating

```
demandsys quaids w1 w2 w3 w4, prices(p1 p2 p3 p4)///expenditure(totexp) demographics(x1 \(x 2\), scaling)
```


## Menu

Statistics $>$ Linear models and related $>$ Multiple-equation models $>$ Demand system

## Syntax

```
demandsys model varlists \([\) if \(][\) in \(][\) weight \(], ~ p r i c e s ~_{\text {(varlist }}^{p}\) )
expenditure(varname) [options]
```

model
Description
cdouglas
les
translog
gtranslog
aids
gaids
quaids
gquaids

Cobb-Douglas demand system
linear expenditure system
basic translog demand system
generalized translog demand system
almost ideal demand system (AIDS)
generalized AIDS
quadratic AIDS
generalized quadratic AIDS
varlist $t_{s}$ indicates the list of $G$ variables containing the expenditure shares of the $G$ goods in the model.
options
Description

## Main

*prices (varlist ${ }_{p}$ )

* expenditure (varname)
demographics(varlist ${ }_{d}[$, scaling])
specify the variables containing prices
specify the variable containing total expenditure
specify other variables affecting a consumer's demand for the goods; use demographic scaling instead of translating
piconstant(\#)
specify constant term in transcendental logarithmic price index

SE/Robust
vce (vcetype)
vcetype may be gnr, robust, cluster clustvar, bootstrap, or jackknife

Reporting elasticities (e_type)
labels (string)
level(\#)
noheader
nocoeftable
display_options
Optimization
optimization_options
coeflegend
report elasticities; e_type may be expenditure, compensated, or uncompensated
specify labels for goods
set confidence level; default is level (95)
suppress the summary header at the top of the output
do not display the table of estimated coefficients control columns and column formats and line width
control the optimization process; seldom used
display legend instead of statistics
*prices() and expenditure() are required.
collect is allowed; see [U] 11.1.10 Prefix commands.
aweights, fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
coeflegend does not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

Main
prices (varlist ${ }_{p}$ ) specifies a list of $G$ variables corresponding to the prices of the $G$ goods faced by each consumer. The number of variables specified here must match the number specified in varlist , and the price variables must be specified in the same order as the share variables. All the price variables must be strictly positive for all the demand systems implemented. prices() is required.
expenditure (varname) specifies the variable corresponding to the total expenditure on all goods within the system by each consumer. This variable must be strictly positive. expenditure() is required.
demographics (varlist ${ }_{d}[$, scaling]) specifies one or more demographic variables that affect each consumer's demand for the goods in the system. Suboption scaling, available only with models aids and quaids, requests that demographics be incorporated via demographic scaling rather than demographic translation, the default.
piconstant(\#) specifies the value of the constant term in the transcendental logarithmic price index; by default, this is set to the logarithm of the minimum expenditure observed in the sample. This option is relevant only for models aids and quaids.

## SE/Robust

vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (gnr), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
vce (gnr), the default, uses the conventionally derived variance estimator for nonlinear models fit using Gauss-Newton regression.

Reporting
elasticities (e_type) requests that elasticities be reported instead of, or in addition to, the parameter estimates. e_type may be expenditure, compensated, or uncompensated. expenditure computes demand elasticity to changes in expenditure. compensated computes demand elasticity to changes in prices, ignoring income effects. These elasticities are also known as Hicksian price elasticities. uncompensated computes demand elasticity to changes in prices. These elasticities are also known as Marshallian price elasticities. The elasticities are computed at the estimation sample means of the prices, expenditures, and any demographic variables specified. For more flexibility in obtaining elasticities, use the postestimation command estat elasticities.
labels(string) specifies a set of names with which the goods are to be labeled. By default, if you specify a four-good demand system, then the goods will be labeled generically: "Good 1", "Good 2", "Good 3", and "Good 4".

If you specify labels("shelter fuel food other"), then the four goods will be labeled "shelter", "fuel", "food", and "other" in the output. If you specify a demand system with $G$ goods, then you must supply $G$ labels separated by spaces.

## level(\#); see [R] Estimation options.

noheader requests that the header summarizing the model, estimation sample, and other statistics not be shown in the output.
nocoeftable requests that the table containing the parameter estimates, their standard errors, and so on not be displayed. Typically, you would use this option if you specify the elasticities() option.
display_options: noci, nopvalues, cformat (\%fmt), pformat (\%fmt), sformat (\%fmt), and nolstretch; see [R] Estimation options.

Optimization
optimization_options: iterate (\#), [no]log, trace, eps (\#), ifgnlsiterate(\#), ifgnlseps(\#), and delta(\#).
iterate(\#) specifies the maximum number of iterations to use for nonlinear least squares at each round of feasible generalized nonlinear least-squares (FGNLS) estimation. The default is the number set using set maxiter, which is 300 by default.
$\log$ and nolog specify whether to display the iteration $\log$. The iteration $\log$ is displayed by default unless you used set iterlog off to suppress it; see set iterlog in [R] set iter.
trace specifies that the iteration $\log$ should include the current parameter vector.
eps (\#) specifies the convergence criterion for successive parameter estimates and for the residual sum of squares. The default is eps (1e-5) (0.00001). eps () also specifies the convergence criterion for successive parameter estimates between rounds of iterative FGNLS.
ifgnlsiterate(\#) specifies the maximum number of FGNLS iterations to perform. The default is the number set using set maxiter, which is 300 by default.
ifgnlseps (\#) specifies the convergence criterion for successive estimates of the error covariance matrix during iterative FGNLS estimation. The default is ifgnlseps (1e-10).
delta(\#) specifies the relative change in a parameter, $\delta$, to be used in computing the numeric derivatives. The derivative for parameter beta $a_{i}$ is computed as

$$
\left\{f_{i}\left(\mathbf{x}_{i}, \beta_{1}, \beta_{2}, \ldots, \beta_{i}+d, \beta_{i+1}, \ldots\right)-f_{i}\left(\mathbf{x}_{i}, \beta_{1}, \beta_{2}, \ldots, \beta_{i}, \beta_{i+1}, \ldots\right)\right\} / d
$$

where $d=\delta\left(\left|\beta_{i}\right|+\delta\right)$. The default is delta(4e-7).
The following option is available with demandsys but is not shown in the dialog box:
coeflegend; see [R] Estimation options.

## Remarks and examples

Remarks are presented under the following headings:
Introduction
Some notation
Cobb-Douglas
Linear expenditure system (LES)
Translog
Basic translog
Generalized translog
AIDS
QUAIDS
Controlling for demographic factors
Demographic translation
Demographic scaling
Epilogue

## Introduction

demandsys fits demand systems, sets of equations derived from economic theory that describe consumers' purchases of various goods or services. Typically, you will have a large cross-sectional survey containing consumers' data on their expenditures on various goods and services along with the prices paid for them. We often refer to the items as "goods" for brevity, but of course some of the items may be services. Whether they are physical goods or intangible services, in the context of utility-maximizing consumers, they are both goods in the sense that "more is better". We also use the term "purchase" somewhat loosely; in some cases, what we are interested in is not the actual purchase of goods but rather their consumption over a fixed time period.

To fit a demand system, you must first decide on what is in the consumption basket or bundle and the set of goods whose demands you wish to model. Using the parameters from the model, you may then obtain the elasticities-the effects of changes in prices or changes in expenditure on demand of goods. You may also perform welfare analysis by contrasting changes in demand or utility that occur at different price or expenditure levels. We assume that you have decided which consumption bundle to model already or else that you have several alternative baskets to model and compare.

The left-hand-side variables you specify with demandsys are expenditure shares, the shares of total expenditure spent on goods or services. Given $p_{g}$ (the price of good $g$ ), $q_{g}$ (the quantity of good $g$ purchased), and $m$ (the consumer's total expenditure on all goods within the demand system), the expenditure share for good $g$ is defined as

$$
w_{g}=\frac{p_{g} q_{g}}{m}
$$

demandsys requires that you have the prices of all the goods and that you have the total expenditure across all the goods. demandsys does not need the quantities of the goods purchased, though you may need them to calculate the expenditure shares.

Consumer theory in microeconomics presents demand models in terms of quantities. In demand system analysis, we model expenditure shares, not quantities. By our definition of total expenditure, $0 \leq w_{g} \leq 1$ for all $g$, and $\sum_{g} w_{g}=1$. For all but the simplest utility or cost functions, the algebra to obtain expenditure shares is arguably easier than that to obtain quantities. demandsys checks your expenditure shares for all goods and will exit with an error message if an expenditure share is found outside that range or if the sum is not equal to one (allowing for small rounding errors).

You may also specify additional variables that may affect a consumer's purchase decisions. These are often demographic variables, such as the number of children and adults in a household, or a set of indicator variables to denote the region of a country in which a consumer is located. If you have data that were collected over the course of different months or years, you may wish to include indicators to control for the time at which the consumer's purchases were observed. We call all of these variables demographics, even if some of these variables are not really demographic characteristics.

Based on the theoretical considerations in Lewbel (2001), we strongly encourage you to include demographic variables in your model. He shows that econometrically estimated demand functions will not satisfy rationality conditions unless other variables that affect demand and are correlated with expenditures are included. Conversely, if we include all such variables in our model, then the estimated demands will satisfy rationality even if preferences vary among households. On the other hand, you should also bear in mind that including too many demographic variables can greatly increase the number of parameters in your model, especially if your demand system includes many goods.
demandsys offers just a small sampling of demand systems that have been proposed in the literature. As a practical matter, which one should you use? We have included the Cobb-Douglas model mainly for pedagogical purposes because most students are familiar with Cobb-Douglas utility, though soon thereafter they learn about its severe restrictions on consumer behavior. Stone's (1954) linear expenditure system (LES) is historically important as an early demand system but also places somewhat strong restrictions on consumer behavior. Christensen, Jorgenson, and Lau's (1975) translog model relaxes some of the LES's restrictions. It is not as widely used as some of the newer models, though a very similar translog production function those authors proposed continues to see extensive use. Shortly after the translog model arrived, Deaton and Muellbauer (1980b) proposed their more flexible AIDS, which has been a workhorse model of demand system analysis ever since its arrival. Banks, Blundell, and Lewbel (1997) provided a quadratic extension to AIDS that is also popular.

Banks, Blundell, and Lewbel (1997) suggest using QUAIDS because of its flexibility. Moreover, QUAIDS satisfies certain theoretical properties developed in Gorman (1981); and it allows goods to be either luxuries or necessities depending on a consumer's income, which Banks, Blundell, and Lewbel (1997) show to be important in the data they consider.

With the AIDS and QUAIDS models, you can include demographics via two different methods: demographic translation and demographic scaling; see Demographic translation and Demographic scaling. If you believe that subsistence or committed quantities are appropriate for the goods you are modeling, you can use what we call the generalized QUAIDS model. If you do that, you should check to see that the subsistence quantity interpretation holds; we do that in example 2 of $[R]$ demandsys postestimation. You can also do a simple likelihood-ratio test to see whether the quadratic terms of these QUAIDS models increase their explanatory power over Deaton and Muellbauer's AIDS and its generalized variant; see example 5 below.

Throughout the rest of the discussion, we assume that you are familiar with a few basic microeconomic concepts, including utility maximization, the expenditure function, and the indirect utility function. The presentations in standard texts like Varian (1992, chap. 7-10) and Mas-Colell, Whinston, and Green (1995, chap. 3) provide good introductions to these concepts. The classic monograph by Deaton and Muellbauer (1980a) and the book by Pollak and Wales (1992) describe demand system analysis in much greater detail than space permits us here. More recent survey papers include Holt and Goodwin (2009) and Barnett and Serletis (2008). Fisher, Fleissig, and Serletis (2001) compare many flexible functional forms for demand system analysis, including several that are implemented by demandsys.

## Some notation

It will be useful to establish at the outset some standard notation we will use throughout this discussion. We use subscript $i=1, \ldots, N$ to index observations, $g=1, \ldots, G$ to index goods, and $d=1, \ldots, D$ to index demographic variables. In formulas where we must use double summations across goods or else have used subscript $g$ elsewhere, we also use $h=1, \ldots, G$ and $j=1, \ldots, G$ to index goods. When no confusion could arise, we omit the observation subscript to reduce the number of subscripts.

Let $w_{g}$ be the expenditure share for good $g$ defined as $w_{g} \equiv\left(p_{g} q_{g}\right) / m$, where $p_{g}$ is the price of good $g, q_{g}$ is the quantity of good $g$ consumed, and $m$ denotes total expenditure on all the goods in the system being modeled. We use the notation $\mathbf{p}$ to refer to the $G$-length vector of all $G$ prices. When we refer to $w_{g}$, we are referring to the observed expenditure share for good $g$ for consumer $i$. When we refer to, say, $w_{g}(\mathbf{p}, m ; \boldsymbol{\alpha}, \boldsymbol{\beta})$, we are referring to an expenditure-share equation or function for good $g$ that depends on the prices of all the goods and total expenditure as well as parameter vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. We include the parameter vectors and matrices in the arguments for expenditure shares as well as some other functions to emphasize that they are estimated and will appear in the output from demandsys and that statistics available via predict or estat elasticities after estimation depend on those parameters.

The observed $w_{g}$ is assumed to be related to the expenditure-share function $w_{g}(\mathbf{p}, m ; \boldsymbol{\alpha}, \boldsymbol{\beta})$ as

$$
w_{g}=w_{g}(\mathbf{p}, m ; \boldsymbol{\alpha}, \boldsymbol{\beta})+\epsilon_{g}
$$

where $\epsilon_{g}$ is a zero-mean error term that we discuss in more detail in Methods and formulas.

## Cobb-Douglas

Although widely used in many economic models, the Cobb-Douglas utility function is arguably too simple for serious demand system analysis because of its severe restrictions on the expenditure-share equations and elasticities. For instance, it restricts expenditure elasticities to be identically equal to 1 for all goods. Nevertheless, we include it because it serves as a good starting point for our discussion and allows us to present various aspects of demandsys. For three goods, consumers maximize their Cobb-Douglas utility function subject to the constraint that total expenditure does not exceed the allocated budget,

$$
\begin{aligned}
& \max _{q_{1}, q_{2}, q_{3}} q_{1}^{\alpha_{1}} q_{2}^{\alpha_{2}} q_{3}^{\left(1-\alpha_{1}-\alpha_{2}\right)} \\
& \quad \text { subject to } p_{1} q_{1}+p_{2} q_{2}+p_{3} q_{3} \leq m
\end{aligned}
$$

where, without loss of generality, we have made the sum of the exponents of the Cobb-Douglas utility function sum to 1 . It is easy to show that the optimal quantities $q_{1}^{*}, q_{2}^{*}$, and $q_{3}^{*}$ are

$$
q_{1}^{*}=\alpha_{1} \frac{m}{p_{1}} \quad q_{2}^{*}=\alpha_{2} \frac{m}{p_{2}} \quad q_{3}^{*}=\left(1-\alpha_{1}-\alpha_{2}\right) \frac{m}{p_{3}}
$$

To obtain the expenditure-share function of, say, good 1 , we have

$$
w_{1}(\mathbf{p}, m ; \boldsymbol{\alpha})=\frac{p_{1} q_{1}}{m}=\alpha_{1} \frac{m}{p_{1}} \frac{p_{1}}{m}=\alpha_{1}
$$

and likewise for goods 2 and 3. Notice that the expenditure-share function for good $g$ is simply its coefficient $\alpha_{g}$ in the Cobb-Douglas utility function, or, equivalently, the coefficients in the CobbDouglas utility function are simply the shares of total expenditure allocated to each good. Most importantly, the expenditure shares for the Cobb-Douglas model are not functions of prices, income, or any other variables.

We made the sum of the exponents in the Cobb-Douglas utility function sum to 1 , and it is clear that $w_{1}+w_{2}+w_{3}=1$. Without the constraint on the sum, we would have to carry around the divisor $\alpha_{1}+\alpha_{2}+\alpha_{3}$ in our expenditure-share equations, but we would gain absolutely no additional insight. Moreover, with the constraint we need estimate only two parameters rather than three.

More generally, for a Cobb-Douglas utility function of the form

$$
u(\mathbf{q} ; \boldsymbol{\alpha})=\prod_{g=1}^{G} q_{g}^{\alpha_{g}} \quad \text { with } \quad \sum_{g=1}^{G} \alpha_{g}=1
$$

the expenditure-share function for the $g$ th good is simply $w_{g}(\mathbf{p}, m ; \boldsymbol{\alpha})=\alpha_{g}$.

## $>$ Example 1

We first describe the dataset we will use in this example and all others in this manual entry:
. use https://www.stata-press.com/data/r18/food_consumption
(Food consumption)

| . describe |  |  |
| :--- | :--- | :--- |
| Contains data from https://www.stata-press.com/data/r18/food_consumption.dta |  |  |
| Observations: | 4,160 | Food consumption |
| Variables: | 13 | 17 Jul 2022 16:03 |
|  |  | (_dta has notes) |


| Variable name | Storage type | Display format | Value <br> label | Variable label |
| :---: | :---: | :---: | :---: | :---: |
| w_dairy | float | \%10.6f |  | Expenditure share on dairy |
| w_proteins | float | \%10.6f |  | Expenditure share on meats and proteins |
| w_fruitveg | float | \%10.6f |  | Expenditure share on fruits and vegetables |
| w_flours | float | \%10.6f |  | Expenditure share on flours, breads, pasta, and cereals |
| w_misc | float | $\% 10.6 \mathrm{f}$ |  | Expenditure share on misc. food items |
| p_dairy | float | \%10.6f |  | Price of dairy |
| p_proteins | float | \%10.6f |  | Price of meats and proteins |
| p_fruitveg | float | \%10.6f |  | Price of fruits and vegetables |
| p_flours | float | \%10.6f |  | Price of flours, breads, pasta, and cereals |
| p_misc | float | \%10.6f |  | Price of misc. food items |
| expfd | float | \%10.6f |  | Total expenditure on all food categories |
| n_adults | byte | \%8.0g |  | \# adults in household |
| n_kids | byte | \%8.0g |  | \# kids in household |

Sorted by:
The observation level of this dataset is the household, and we have data on 4,160 households. The data include five categories of food: dairy products; proteins, including meats and fish; fruits and vegetables; flour-based products, including breads, pastas, and cereals; and a catchall category. For each of the five categories, we have the expenditure shares and prices, and we have the household's total expenditure on all five categories in the week that the household was surveyed. We also have demographic data for the household, including the numbers of children and adults.

Although we constructed the expenditure shares ourselves and know they sum to one, it is a good idea to verify that you have created your expenditure-share variables properly:
. egen wsum = rowtotal(w_dairy w_proteins w_fruitveg w_flours w_misc)
. summarize wsum

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| wsum | 4,160 | 1 | $4.40 \mathrm{e}-08$ | .9999999 | 1 |

demandsys will check that for you and exit with an error message if it is not the case, but doing so during your data management tasks may make debugging easier. demandsys will also check that all your price variables and expenditure variable are strictly positive, though again you might want to check those facts earlier in your data pipeline.

With a dataset in hand, we are ready to fit our first Cobb-Douglas demand system.

```
. demandsys cdouglas w_dairy w_proteins w_fruitveg w_flours w_misc,
> prices(p_dairy p_proteins p_fruitveg p_flours p_misc) expenditure(expfd)
Calculating NLS estimates:
Iteration 0: Residual SS = 180.2506
Iteration 1: Residual SS = 180.2506
Calculating FGNLS estimates:
Iteration 0: Scaled RSS = 16640
FGNLS iteration 2:
Iteration 0: Scaled RSS = 16640
Parameter change = 0.00e+00
Covariance matrix change = 0.00e+00
Cobb-Douglas model Number of obs = 4,160
Expenditure variable: expfd Number of goods = 5
Uncentered R2 of model for
    Good 1 = 0.7519
    Good 2 = 0.8851
    Good 3 = 0.8329
    Good 4 = 0.7467
    Good 5 = 0.6836
```

|  |  | Estimate | Std. err. | $z$ | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| alpha |  |  |  |  |  |  |  |
|  | Good |  |  |  |  |  |  |
|  | 1 | .1505844 | .0013411 | 112.29 | 0.000 | .147956 | .1532129 |
| 2 | .3986526 | .0022265 | 179.05 | 0.000 | .3942887 | .4030166 |  |
| 3 | .2406533 | .0016711 | 144.01 | 0.000 | .237378 | .2439286 |  |
|  | 4 | .1031129 | .0009312 | 110.73 | 0.000 | .1012877 | .104938 |

Note: alpha estimates are expenditure shares.
Normalized parameter

|  |  | Estimate | Std. err. | $z$ | $\mathrm{P}>\|\mathrm{z}\|$ | $[95 \%$ conf. interval] |  |
| :--- | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| alpha |  |  |  |  |  |  |  |
|  | Good <br> 5 | .1069968 | .0011286 | 94.80 | 0.000 | .1047847 | .1092088 |

[^19]We specified the command name, demandsys, followed by the model we wish to fit, cdouglas. We then specified the five expenditure-share variables, w_dairy through w_misc. The price variables, p_dairy through p_misc, go into the prices() option, and the expenditure variable, expfd, goes into the expenditure() option. demandsys uses a nonlinear multiple-equation estimator, so the top of the output includes an iteration log showing the model's convergence. In fact, the Cobb-Douglas model is linear, but that is the only model fit by demandsys that is. Stata uses the same nonlinear estimator for cdouglas as we use for the other demand systems. The header of the output includes $R^{2}$ values to get some idea of how well the model fits the data.

The table of coefficients is a bit different from most other estimators in Stata, which report coefficients organized by equations. The Cobb-Douglas model is unique in that each equation has its own parameter. However, that is not the case for any other demand system that demandsys fits. For most demand systems, many of the parameters appear in multiple equations, and there is no way to link parameters and equations. Hence, we cannot provide a coefficient table organized by equations as other multiple-equation estimators such as sureg provide. Instead, demandsys organizes estimates by parameter type.

We mentioned having a five-good demand system, so why do we see only four estimated parameters in the main output table? Recall that we used the normalization that the sum of the parameters in the Cobb-Douglas utility function is one. The upshot is that we really have only four free parameters because the fifth parameter must equal one minus the sum of the other four parameters. We provide the estimate of the fifth parameter in a separate table. It is computed separately using the parameter estimates of our fitted demand system and their standard errors. Given how the fifth parameter was computed, it does not share a covariance matrix with the other parameters and cannot be used for testing. We provide it here for reference.
demandsys shows only unconstrained parameters in the main estimation output because, with many goods and more complicated models, the output becomes long. Moreover, the estimated parameters themselves are of less interest than elasticities and other statistics derived from the fitted model.

## - Technical note

In the header of the output in the previous example, we see that the equation-level $R^{2}$ values are labeled as uncentered. Recall that the standard $R^{2}$ in regression analysis measures the ability of the model to predict the regressand compared with a model that contains just a constant term. For the Cobb-Douglas demand system, the expenditure shares are simply constant terms to be estimated. Hence, the traditional $R^{2}$ is by definition zero. We therefore report the uncentered $R^{2}$, which is an alternative that measures the model's explanatory power as a fraction of the sum of squares of an equation's left-hand-side variable. A higher uncentered $R^{2}$ is presumably better than a lower uncentered $R^{2}$, but you cannot compare the uncentered $R^{2}$ of a Cobb-Douglas share equation with the centered $R^{2}$ of a share equation from a different demand system.

The command estat elasticities, which is available after demandsys, allows you to calculate expenditure and price elasticities after fitting your demand system, and estat elasticities provides options to specify how those elasticities are calculated and for which observations. The full syntax for that command is listed in $[R]$ demandsys postestimation, but the basic command is easy to pick up as we work through examples below.

Here we use estat elasticities to obtain the expenditure elasticities for the estimation sample.

```
. estat elasticities, expenditure
Expenditure elasticities Number of obs = 4,160
```

| Expenditure | Elasticity |
| ---: | :---: |
| Good |  |
| 1 | 1 |
| 2 | 1 |
| 3 | 1 |
| 4 | 1 |
| 5 | 1 |

Note: No standard errors are displayed because all elasticities are identically equal to one.

The expenditure elasticities for all five goods are identically equal to one, and so the standard errors, test statistics, and confidence interval are not displayed because there is no sampling variance. The Cobb-Douglas function is an example of a utility function with "homothetic preferences", meaning that the ratio of two goods demanded by a consumer with such preferences depends only on the goods' relative prices and not on income. Homothetic preferences also imply that, regardless of changes in income, the shares consumed of each good remain the same; the expansion path of consumption is linear in income. The Cobb-Douglas case is a particularly extreme example, where the expenditure elasticities are one, the uncompensated own-price elasticities are minus one, and uncompensated cross-price elasticities are zero, which you can verify by typing

```
estat elasticities, uncompensated
```

Particularly with more complex demand systems, the coefficients themselves can be more difficult to interpret. Therefore, demandsys allows you to request that expenditure, uncompensated price, or compensated price elasticities be displayed in addition to, or in place of, the table of estimated coefficients. The elasticities reported directly by demandsys are calculated at the estimation sample means of prices, expenditures, and any demographics that you specify. To obtain, say, the uncompensated price elasticities instead of the coefficient table from a Cobb-Douglas demand system, you would type

```
demandsys cdouglas ..., prices(...) expenditure(...) ///
    elasticities(uncompensated) nocoeftable
```

estat elasticities gives you much more flexibility in terms of the sample used, and you can even specify particular prices or expenditures at which you want the elasticities to be calculated.

## Linear expenditure system (LES)

Nobel laureate Sir Richard Stone's estimation of the LES, developed based on theory by Nobel laureates Paul Samuelson and Lawrence Klein, together with Herman Rubin, represents the genesis of flexible demand system estimation (Stone 1954). LES provides for more flexible consumption patterns than the Cobb-Douglas model. The LES begins with the utility function

$$
u(\mathbf{q} ; \boldsymbol{\beta}, \boldsymbol{\mu})=\prod_{g}\left(q_{g}-\mu_{g}\right)^{\beta_{g}}
$$

where $\mu_{g}$ is the $g$ th element of $G \times 1$ vector $\mu$, which is to be estimated. This utility function results from translating or shifting the origin of the Cobb-Douglas utility function; utility does not accrue from good $g$ unless its consumption exceeds an amount $\mu_{g}$. That parameter is known as a subsistence or committed quantity that a consumer must purchase for survival. $\beta_{g}$ is the $g$ th element of $G \times 1$ vector $\boldsymbol{\beta} . \beta_{g}$ measures the share of supernumerary expenditure, namely, the share of remaining expenditure once all subsistence quantities have been purchased, on good $g$. As in the Cobb-Douglas case, we normalize the $G \times 1$ parameter vector $\boldsymbol{\beta}$ so that $\sum_{g} \beta_{g}=1$. Thus, the model has $2 G-1$ parameters we must estimate.

Solving

$$
\max _{\mathbf{q}} \prod_{g}\left(q_{g}-\mu_{g}\right)^{\beta_{g}} \quad \text { s. t. } \quad \mathbf{p}^{\prime} \mathbf{q} \leq m
$$

yields the expenditure-share functions

$$
w_{g}(\mathbf{p}, m ; \boldsymbol{\beta}, \boldsymbol{\mu})=\frac{p_{g} \mu_{g}}{m}+\beta_{g}\left(1-\sum_{h} \frac{p_{h} \mu_{h}}{m}\right)
$$

The term to the left of the plus sign is the fraction of expenditure that must be spent on good $g$ to maintain subsistence. To the right, the term bound in parentheses is the fraction of expenditure left after the subsistence level for all other goods has been spent. Of this amount, a share $\beta_{g}$ is spent on good $g$. The normalization that $\sum_{g} \beta_{g}=1$ ensures that $\sum_{g} w_{g}(\mathbf{p}, m)=1$, as must be true of a demand system.

While we have interpreted the $\mu_{g}$ as minimum required amounts of each good, there is no requirement that $\mu_{g}>0$ for all $g$, nor do we impose any such constraints during estimation. Moreover, for this interpretation to hold we must have that for each household $i, \sum_{g} p_{g i} \mu_{g i} \leq m_{i}$.

## > Example 2

Here we fit an LES to the same data as in example 1.

| ```(Food consumption) . demandsys les w_dairy w_proteins w_fruitveg w_flours w_misc, > prices(p_dairy p_proteins p_fruitveg p_flours p_misc) > expenditure(expfd) nolog``` |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Calculating NLS estimates ... |  |  |  |  |  |  |
| Calculating FGNLS estimates ... |  |  |  |  |  |  |
| FGNLS iteration $2 . .$. |  |  |  |  |  |  |
| FGNLS iteration 3 ... |  |  |  |  |  |  |
| FGNLS iteration 4 ... |  |  |  |  |  |  |
| FGNLS iteration 5 ... |  |  |  |  |  |  |
| Linear expenditure system ${ }_{\text {Expenditure variable: expfd }}^{\text {d }}$ |  |  |  |  | $\begin{aligned} & \text { Number of obs }=4,160 \\ & \text { Number of goods }=5 \end{aligned}$ |  |
|  |  |  |  |  |  |  |
| Centered R2 of model for |  |  |  |  |  |  |
| Good $1=-0.0508$ |  |  |  |  |  |  |
| Good $2=0.0494$ |  |  |  |  |  |  |
| Good $3=0.0366$ |  |  |  |  |  |  |
| Good $4=0.0829$ |  |  |  |  |  |  |
| Good $5=0.0603$ |  |  |  |  |  |  |
|  | Estimate | Std. err. | z | $P>\|z\|$ | [95\% conf | interval] |
| beta |  |  |  |  |  |  |
| Good |  |  |  |  |  |  |
| 1 | . 1650894 | . 0018783 | 87.89 | 0.000 | . 161408 | . 1687708 |
| 2 | . 4108231 | . 0033135 | 123.98 | 0.000 | . 4043288 | . 4173175 |
| 3 | . 2262407 | . 0024899 | 90.86 | 0.000 | . 2213607 | . 2311208 |
| 4 | . 0949082 | . 0013152 | 72.16 | 0.000 | . 0923304 | . 097486 |
| mu |  |  |  |  |  |  |
| Good |  |  |  |  |  |  |
| 1 | . 1794068 | . 0383858 | 4.67 | 0.000 | . 1041721 | . 2546416 |
| 2 | . 6975436 | . 0800327 | 8.72 | 0.000 | . 5406824 | . 8544048 |
| 3 | 2.368491 | . 1351367 | 17.53 | 0.000 | 2.103628 | 2.633354 |
| 4 | . 4215745 | . 0202499 | 20.82 | 0.000 | . 3818854 | . 4612636 |
| 5 | . 3742182 | . 0243646 | 15.36 | 0.000 | . 3264644 | . 421972 |

Note: beta estimates measure how expenditure shares respond to increases in supernumerary income.
Note: mu estimates are subsistence levels of consumption for each good.
Normalized parameter

|  |  | Estimate | Std. err. | $\mathbf{z}$ | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| :--- | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| beta | Good |  |  |  |  |  |  |
|  | 5 | .1029385 | .0015833 | 65.02 | 0.000 | .0998354 | .1060416 |

```
Note: beta estimates sum to 1.
Shares: w_dairy w_proteins w_fruitveg w_flours w_misc
Prices: p_dairy p_proteins p_fruitveg p_flours p_misc
Expenditure: expfd
```

We specified the nolog option to suppress the detailed iteration log. We first noticed that the $R^{2}$ for the model for Good 1 is actually negative. Because the equations here are nonlinear, the value of $R^{2}$ is not constrained to be between 0 and 1 ; that is only true for linear regression. The footer of the table with normalized parameters reemphasizes the fact that we have made $\sum_{g} \beta_{g}=1$.

All the $\mu$ parameters are greater than zero. The $\mu$ parameter for Good 3, fruits and vegetables, is largest, implying the physical quantity of them required is much higher than the other four goods. However, we should also look at the average prices of the goods:

| . summarize p_* |  |  |  |  |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| Variable | Obs | Mean | Std. dev. | Min | Max |
| p_dairy | 4,160 | .4387958 | .4238236 | .1041262 | 18 |
| p_proteins | 4,160 | 1.729284 | .69554 | .3333333 | 9.258823 |
| p_fruitveg | 4,160 | .5472199 | .1945325 | .1073369 | 2.672269 |
| p_flours | 4,160 | 1.49421 | .6816424 | .0805687 | 7.202127 |
| p_misc | 4,160 | 1.47115 | .8588024 | .1428571 | 15.74629 |

Based on the average price of each good, the dollar amount of protein required for survival is only 9 cents less than the dollar amount of fruits and vegetables required $(2.368 \times 0.547=\$ 1.30$ versus $0.698 \times 1.729=\$ 1.21$ ).

As we mentioned in the introductory remarks, if you have demographic variables, you should incorporate them into your model. demandsys incorporates demographics into models by applying Pollak and Wales's (1978) "demographic translation" to the expenditure-share equations. Just as the linear expenditure system is a translated version of the Cobb-Douglas system, Pollak and Wales incorporate demographics by translating the consumer's available level of expenditures.

Suppose for each consumer we have a $D \times 1$ vector of demographic characteristics $\mathbf{d}$. We have $G$ expenditure-share equations, so associated with $\mathbf{d}$ is a $G \times D$ matrix of parameters $\mathbf{N}$ with typical row $\boldsymbol{\nu}_{g}$. Let $\mathbf{c}=\mathbf{N d}$ with typical element $c_{g}$. In demographic translation, we write each translated demand function as

$$
\bar{x}_{g}(\mathbf{p}, m, \mathbf{c})=c_{g}+x_{g}\left(\mathbf{p}, m-\sum_{h} p_{h} c_{h}\right)
$$

For the LES, we have

$$
\bar{x}_{g}(\mathbf{p}, m, \mathbf{d} ; \boldsymbol{\beta}, \boldsymbol{\mu}, \mathbf{N})=\left(\mu_{g}+c_{g}\right)+\frac{\beta_{g}}{p_{g}}\left(m-\sum_{h} p_{h} \mu_{h}-\sum_{h} p_{h} c_{h}\right)
$$

so that

$$
\bar{w}_{g}(\mathbf{p}, m, \mathbf{d} ; \boldsymbol{\beta}, \boldsymbol{\mu}, \mathbf{N})=\frac{p_{g}\left(\mu_{g}+\boldsymbol{\nu}_{g} \mathbf{d}\right)}{m}+\beta_{g}\left\{1-\frac{\sum_{h} p_{h}\left(\mu_{h}+\boldsymbol{\nu}_{h} \mathbf{d}\right)}{m}\right\}
$$

For the LES, the effect of the demographic variables is to adjust the $\mu$ parameters, though we must be cautious in interpreting the term $\mu_{g}+\nu_{g} \mathbf{d}$ as a minimum required quantity because there is nothing preventing it from being negative.

## > Example 3

The dataset described in example 1 also includes two demographic characteristics, the numbers of children and adults in each household. Here we refit the LES to these data, controlling for these two demographics.

```
. use https://www.stata-press.com/data/r18/food_consumption
(Food consumption)
. demandsys les w_dairy w_proteins w_fruitveg w_flours w_misc,
> prices(p_dairy p_proteins p_fruitveg p_flours p_misc)
> expenditure(expfd) demographics(n_kids n_adults) nolog
Calculating NLS estimates ...
Calculating FGNLS estimates ...
FGNLS iteration 2 ...
FGNLS iteration 3 ...
FGNLS iteration 4 ...
FGNLS iteration 5 ...
\begin{tabular}{llr} 
Linear expenditure system & Number of obs & \(=4,160\) \\
Expenditure variable: expfd & Number of goods & \(=\) \\
Demographic method: Translating & Number of demographics & \(=\) \\
\hline
\end{tabular}
Centered R2 of model for
    Good 1 = -0.0444
    Good 2 = 0.0665
    Good 3 = 0.0584
    Good 4 = 0.1576
    Good 5 = 0.0922
```



Note: beta estimates measure how expenditure shares respond to increases in supernumerary income.
Note: mu estimates are subsistence levels of consumption for each good.
Note: Nu estimates measure the effect of demographic variables on subsistence levels of consumption.
Normalized parameter

|  |  | Estimate | Std. err. | $z$ | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| :--- | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| beta |  |  |  |  |  |  |  |
|  | Good <br> 5 | .0937826 | .0019264 | 48.68 | 0.000 | .090007 | .0975582 |

Note: beta estimates sum to 1.
Shares: w_dairy w_proteins w_fruitveg w_flours w_misc
Prices: p_dairy p_proteins p_fruitveg p_flours p_misc
Expenditure: expfd
Demographics: n_kids n_adults
The estimated parameter matrix $\mathbf{N}$ is shown in the last block of the coefficient table, and demandsys labels the coefficients using factor-variable notation. Yet you should not think of the terms as you would of interactions in a regression. Here the notation means something different. The first five rows of the output for $\mathbf{N}$ correspond to the demographic variable $n \_k i d s$ and how it affects the consumption of each good. The remaining five correspond to n_adults. For instance, all the coefficients for n_adults are positive, as we would expect, meaning that having more adults in the family increases consumption of all goods. The coefficient for $n \_k i d s$ on the consumption of Good 3, fruits and vegetables, is negative, meaning consumption decreases as the number of children increases.

Directly interpreting the coefficients on demographic variables is not difficult for the LES, but for more complicated models, it can be. One easy way to see the practical effect of that negative coefficient is to use the margins command to see how our predicted quantity for Good 3 changes as the number of children changes.

```
. margins, predict(quantities equation(#3)) at(n_kids=1 n_kids=2 n_kids=3)
Predictive margins Number of obs = 4,160
Model VCE: GNR
Expression: Predicted quantity of good 3, predict(quantities equation(#3))
1._at: n_kids = 1
2._at: n_kids = 2
3._at: n_kids = 3
```

|  | Delta-method <br> std. err. |  |  |  | z | $\mathrm{P}>\|\mathrm{z}\|$ |
| :---: | ---: | ---: | ---: | ---: | ---: | ---: |
| Margin | [95\% conf. interval] |  |  |  |  |  |
| at |  |  |  |  |  |  |
| 2 | 22.57831 | .1734791 | 130.15 | 0.000 | 22.2383 | 22.91833 |
| 3 | 20.23569 | .323032 | 62.64 | 0.000 | 19.60256 | 20.86882 |

We asked margins to produce predictions of quantities of Good 3, fruits and vegetables, by specifying the equation(\#3) option, and we asked it to do so when the number of children is equal to one, again when the number of children is equal to two, and finally when the number of children is equal to three. margins first sets $n \_k i d s$ equal to one for all 4,160 observations in the estimation sample and obtains the predicted quantities; the mean of those predictions is 22.6 . When $\mathrm{n} \_\mathrm{kids}$ is equal to two for all observations, the mean is 21.4, and when $n \_k i d s$ is equal to three, the mean is 20.2 .

Kids do not always eat their fruits and vegetables, but finding that the quantity of fruits and vegetables actually declines as the household grows is rather surprising. Perhaps we have omitted
other relevant demographic variables from our model, biasing our estimates, or perhaps the model itself is simply too rigid.

Finally, we use estat elasticities to obtain uncompensated price elasticities. We will obtain two sets of results, one for households with fewer than three kids and one for households with three or more kids.

| Uncompensated (Marshallian) price elasticities |  |  |  |  | Number of obs $=3,742$ |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Price | Elasticity | Std. err. | z | $P>\|z\|$ | [95\% conf. | interval] |
| Good 1 |  |  |  |  |  |  |
| Good |  |  |  |  |  |  |
| 1 | -. 9537446 | . 0083361 | -114.41 | 0.000 | -. 970083 | -. 9374061 |
| 2 | -. 00852 | . 0015032 | -5.67 | 0.000 | -. 0114662 | -. 0055738 |
| 3 | -. 0083598 | . 0014808 | -5.65 | 0.000 | -. 0112621 | -. 0054574 |
| 4 | -. 0062373 | . 0011228 | -5.56 | 0.000 | -. 0084379 | -. 0040367 |
| 5 | -. 0071346 | . 0012983 | -5.50 | 0.000 | -. 0096793 | -. 00459 |
| Good 2 |  |  |  |  |  |  |
| Good |  |  |  |  |  |  |
| 1 | -. 0621117 | . 012096 | -5.13 | 0.000 | -. 0858194 | -. 038404 |
| 2 | -. 9289446 | . 0085912 | -108.13 | 0.000 | -. 945783 | -. 9121063 |
| 3 | -. 0499231 | . 0062304 | -8.01 | 0.000 | -. 0621346 | -. 0377117 |
| 4 | -. 0390871 | . 0048967 | -7.98 | 0.000 | -. 0486844 | -. 0294897 |
| 5 | -. 0445883 | . 0056323 | -7.92 | 0.000 | -. 0556274 | -. 0335492 |
| Good 3 |  |  |  |  |  |  |
| Good |  |  |  |  |  |  |
| 1 | -. 0503597 | . 0118715 | -4.24 | 0.000 | -. 0736274 | -. 0270921 |
| 2 | -. 0410463 | . 0036302 | -11.31 | 0.000 | -. 0481614 | -. 0339312 |
| 3 | -. 886701 | . 0100534 | -88.20 | 0.000 | -. 9064052 | -. 8669968 |
| 4 | -. 030438 | . 0027264 | -11.16 | 0.000 | -. 0357816 | -. 0250944 |
| 5 | -. 0348829 | . 0032109 | -10.86 | 0.000 | -. 0411761 | -. 0285897 |
| Good 4 |  |  |  |  |  |  |
| Good |  |  |  |  |  |  |
| 1 | -. 0428307 | . 0038014 | -11.27 | 0.000 | -. 0502814 | -. 0353801 |
| 2 | -. 0373442 | . 0014421 | -25.90 | 0.000 | -. 0401707 | -. 0345178 |
| 3 | -. 0357288 | . 0013809 | -25.87 | 0.000 | -. 0384354 | -. 0330223 |
| 4 | -. 7107582 | . 0101357 | -70.12 | 0.000 | -. 7306238 | -. 6908925 |
| 5 | -. 0306015 | . 0012718 | -24.06 | 0.000 | -. 0330941 | -. 0281089 |
| Good 5 |  |  |  |  |  |  |
| Good |  |  |  |  |  |  |
| 1 | -. 0347514 | . 0047892 | -7.26 | 0.000 | -. 0441381 | -. 0253647 |
| 2 | -. 0296409 | . 0016063 | -18.45 | 0.000 | -. 0327893 | -. 0264926 |
| 3 | -. 0285217 | . 0015907 | -17.93 | 0.000 | -. 0316394 | -. 025404 |
| 4 | -. 0212131 | . 0011859 | -17.89 | 0.000 | -. 0235375 | -. 0188887 |
| 5 | -. 7824229 | . 0109448 | -71.49 | 0.000 | -. 8038744 | -. 7609714 |

. estat elasticities if n_kids >= 3, uncompensated
Uncompensated (Marshallian) price elasticities Number of obs = 418

| Price | Elasticity | Std. err. | z | $P>\|z\|$ | [95\% conf | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Good 1 |  |  |  |  |  |  |
| Good |  |  |  |  |  |  |
| 1 | -. 9395485 | . 0146154 | -64.28 | 0.000 | -. 9681941 | -. 9109029 |
| 2 | -. 0126359 | . 0032625 | -3.87 | 0.000 | -. 0190303 | -. 0062416 |
| 3 | -. 0146746 | . 004299 | -3.41 | 0.001 | -. 0231005 | -. 0062486 |
| 4 | -. 0082076 | . 002108 | -3.89 | 0.000 | -. 0123393 | -. 0040759 |
| 5 | -. 0093949 | . 0024408 | -3.85 | 0.000 | -. 0141788 | -. 0046109 |
| Good 2 |  |  |  |  |  |  |
| Good |  |  |  |  |  |  |
| 1 | -. 023357 | . 0100783 | -2.32 | 0.020 | -. 04311 | -. 0036039 |
| 2 | -. 9674982 | . 0135281 | -71.52 | 0.000 | -. 9940128 | -. 9409836 |
| 3 | -. 028745 | . 0152303 | -1.89 | 0.059 | -. 0585959 | . 0011058 |
| 4 | -. 0152942 | . 0064352 | -2.38 | 0.017 | -. 0279071 | -. 0026813 |
| 5 | -. 0178107 | . 007578 | -2.35 | 0.019 | -. 0326633 | -. 0029581 |
| Good 3 |  |  |  |  |  |  |
| Good |  |  |  |  |  |  |
| 1 | -. 0000379 | . 0055043 | -0.01 | 0.995 | -. 0108261 | . 0107504 |
| 2 | . 0001398 | . 0056976 | 0.02 | 0.980 | -. 0110273 | . 0113068 |
| 3 | -1.000682 | . 0201954 | -49.55 | 0.000 | -1.040264 | -. 9610998 |
| 4 | -. 0000842 | . 0035771 | -0.02 | 0.981 | -. 0070951 | . 0069268 |
| 5 | -. 0000569 | . 0041395 | -0.01 | 0.989 | -. 0081702 | . 0080565 |
| Good 4 |  |  |  |  |  |  |
| Good |  |  |  |  |  |  |
| 1 | -. 0451601 | . 0025582 | -17.65 | 0.000 | -. 0501741 | -. 0401461 |
| 2 | -. 0463532 | . 0024489 | -18.93 | 0.000 | -. 051153 | -. 0415534 |
| 3 | -. 052944 | . 0045994 | -11.51 | 0.000 | -. 0619587 | -. 0439293 |
| 4 | -. 6843304 | . 0131677 | -51.97 | 0.000 | -. 7101385 | -. 6585222 |
| 5 | -. 0334225 | . 0018152 | -18.41 | 0.000 | -. 0369803 | -. 0298648 |
| Good 5 |  |  |  |  |  |  |
| Good |  |  |  |  |  |  |
| 1 | -. 0356486 | . 002797 | -12.75 | 0.000 | -. 0411305 | -. 0301666 |
| 2 | -. 0365794 | . 0027339 | -13.38 | 0.000 | -. 0419377 | -. 0312212 |
| 3 | -. 0401475 | . 0035616 | -11.27 | 0.000 | -. 0471281 | -. 033167 |
| 4 | -. 0234098 | . 0017471 | -13.40 | 0.000 | -. 0268341 | -. 0199856 |
| 5 | -. 7542306 | . 0156676 | -48.14 | 0.000 | $-.7849385$ | -. 7235226 |

For example, in the table for households with 3 or more children, the price elasticity for Good 5 with respect to Good 1 is -0.0356486 . Among these households, if the price of Good 5 (miscellaneous items) increases by $1 \%$, then the quantity of Good 1 (dairy products) purchased will decrease by an average of $0.036 \%$.

We summarize the own-price elasticities in table 1. We again focus on the demand for fruits and vegetables. Among households with two children or fewer, it is inelastic, while it is (barely) elastic for households with three or more children. Again, we have no explanation for why that would be the case, though we suspect the model is simply too rigid to accurately model the consumption patterns seen in our data.

Table 1: Own-price elasticities based on number of children

|  | \# of Children |  |
| :--- | :---: | :---: |
| Good | $\leq 2$ | $\geq 3$ |
| Dairy | -0.954 | -0.940 |
| Proteins | -0.939 | -0.967 |
| Fruits \& vegetables | -0.887 | -1.001 |
| Flours | -0.711 | -0.684 |
| Miscellany | -0.782 | -0.754 |

## Translog

Pollak and Wales (1992, sec. 3.1) discuss an entire class of demand systems that they describe as the transcendental logarithmic or "translog" family. We implement the version they call the "basic translog" function, which is probably the most well known, because it corresponds to the translog indirect utility function proposed by Christensen, Jorgenson, and Lau (1975). We also implement the version Pollak and Wales call the "generalized translog" function, which is a more flexible version of the basic translog that has been translated in a way analogous to how the LES is a shifted version of the Cobb-Douglas utility function.

## Basic translog

Christensen, Jorgenson, and Lau (1975) proposed both direct and indirect utility functions with the "translog" functional form. For their direct utility function, they obtain expenditure shares that are a function of expenditure and quantities, which is not the focus of demandsys. We instead implement the demand system based on their translog indirect utility function. The starting point is the indirect utility function

$$
\begin{equation*}
\ln v(\mathbf{p}, m ; \boldsymbol{\alpha}, \boldsymbol{\Gamma})=-\sum_{g} \alpha_{g} \ln \frac{p_{g}}{m}-\frac{1}{2} \sum_{g} \sum_{h} \gamma_{g h} \ln \frac{p_{g}}{m} \ln \frac{p_{h}}{m} \tag{1}
\end{equation*}
$$

Equation (1) is twice differentiable in prices, and based on Young's (1909) theorem, we must have $\gamma_{g h}=\gamma_{h g}$ for all $g$ and $h$. (Young's theorem states that cross-partial derivatives are equal.)

Roy (1943) introduced what would be known as Roy's identity, which equates a consumer's Marshallian demand function for a good with a function of their indirect utility function. Applying Roy's identity, we have the expenditure-share functions

$$
w_{g}(\mathbf{p}, m ; \boldsymbol{\alpha}, \boldsymbol{\Gamma})=\frac{\alpha_{g}+\sum_{h} \gamma_{g h} \ln \frac{p_{h}}{m}}{1+\sum_{j} \sum_{h} \gamma_{j h} \ln \frac{p_{h}}{m}}
$$

where we have imposed the normalization $\sum_{g} \alpha_{g}=1$. This ensures that the expenditure shares sum to 1 , a property known as "additivity".

As we remarked in the introduction, if one has relevant demographic variables, one should include them in the demand system, so we will not present an example of the translog model without demographic variables. Moreover, it is in fact easier to describe how we incorporate demographic variables into the translog system if we first describe the generalized translog model.

## Generalized translog

Another way to perform translation and introduce subsistence or committed quantities into a demand system is to translate the indirect utility function, so that consumers' effective expenditures are their expenditures after the subsistence quantities of all goods are purchased. Pollak and Wales $(1992,56)$ do this to Christensen, Jorgenson, and Lau's (1975) indirect utility function to obtain

$$
\begin{aligned}
\ln v(\mathbf{p}, m ; \boldsymbol{\alpha}, \boldsymbol{\Gamma}, \boldsymbol{\mu}) & =-\sum_{g} \alpha_{g} \ln \frac{p_{g}}{\bar{m}}-\frac{1}{2} \sum_{g} \sum_{h} \gamma_{g h} \ln \frac{p_{g}}{\bar{m}} \ln \frac{p_{h}}{\bar{m}} \\
\bar{m} & =m-\sum_{h} p_{h} \mu_{h} \quad \sum_{h} \alpha_{h}=1 \quad \gamma_{g h}=\gamma_{h g}
\end{aligned}
$$

where again $\boldsymbol{\mu}$ is a $G \times 1$ vector of subsistence quantities with typical element $\mu_{g}$. A tedious but straightforward application of Roy's identity gives the expenditure-share functions as

$$
\begin{aligned}
w_{g}(\mathbf{p}, m ; \boldsymbol{\alpha}, \boldsymbol{\Gamma}, \boldsymbol{\mu}) & =\frac{p_{g} \mu_{g}}{m}+\frac{\bar{m}}{m}\left(\frac{\alpha_{g}+\sum_{h} \gamma_{g h} \ln \frac{p_{h}}{\bar{m}}}{1+\sum_{j} \sum_{h} \gamma_{j h} \ln \frac{p_{h}}{\bar{m}}}\right) \\
\bar{m} & =m-\sum_{h} p_{h} \mu_{h} \quad \sum_{h} \alpha_{h}=1 \quad \gamma_{h j}=\gamma_{j h}
\end{aligned}
$$

It is apparent that the sum of $w_{g}(\mathbf{p}, m ; \boldsymbol{\alpha}, \boldsymbol{\Gamma}, \boldsymbol{\mu})$ across all $g$ is equal to one, as must be true of a demand system. Again, suppose for each consumer that we have a $D \times 1$ vector of demographics $\mathbf{d}$ and an associated $G \times D$ matrix $\mathbf{N}$, and again, let $c_{g}$ denote the $g$ th element of the vector $\mathbf{c}=\mathbf{N d}$.

To apply both demographic characteristics and committed quantities, we can translate the translog indirect utility function, this time with the value $\left(\mu_{g}+c_{g}\right)$ replacing $\mu_{g}$ in the developments above. Thus, the generalized translog system with demographic variables has expenditure-share functions of the form

$$
\begin{aligned}
w_{g}(\mathbf{p}, m, \mathbf{d} ; \boldsymbol{\alpha}, \boldsymbol{\Gamma}, \boldsymbol{\mu}, \mathbf{N}) & =\frac{p_{g}\left(\mu_{g}+\boldsymbol{\nu}_{g} \mathbf{d}\right)}{m}+\frac{\bar{m}}{m}\left(\frac{\alpha_{g}+\sum_{h} \gamma_{g h} \ln \frac{p_{h}}{\bar{m}}}{1+\sum_{j} \sum_{h} \gamma_{j h} \ln \frac{p_{h}}{\bar{m}}}\right) \\
\bar{m} & =m-\sum_{h} p_{h}\left(\mu_{h}+\boldsymbol{\nu}_{h} \mathbf{d}\right) \quad \sum_{h} \alpha_{h}=1 \quad \gamma_{h j}=\gamma_{j h}
\end{aligned}
$$

The basic translog model results if we set $\boldsymbol{\mu}=\mathbf{0}$ and $\mathbf{N}=\mathbf{0}$, and the basic translog model with demographics results if we set just $\boldsymbol{\mu}=\mathbf{0}$. The generalized translog model results if we set just $\mathbf{N}=\mathbf{0}$. The generalized translog model is essentially a translog model with a constant term included among the demographic variables.

## - Example 4

We first fit a basic translog model to our food consumption data, controlling for the number of children and adults in each household. We will also store these estimation results so we can use them later.

```
. use https://www.stata-press.com/data/r18/food_consumption
(Food consumption)
. demandsys translog w_dairy w_proteins w_fruitveg w_flours w_misc,
> prices(p_dairy p_proteins p_fruitveg p_flours p_misc)
> demographics(n_kids n_adults)
> labels("dairy proteins fruitveg flours misc")
> expenditure(expfd) elasticities(expenditure) nolog nocoeftable
Calculating NLS estimates ...
Calculating FGNLS estimates ...
FGNLS iteration 2 ...
FGNLS iteration 3 ...
FGNLS iteration 4 ...
Basic translog model Number of obs = 4,160
Expenditure variable: expfd Number of goods = N
Demographic method: Translating Number of demographics = N
Centered R2 of model for
    dairy = 0.0448
    proteins = 0.1593
    fruitveg = 0.0976
    flours = 0.1881
    misc = 0.1678
Calculating expenditure elasticities ...
Expenditure elasticities
```

| Expenditure | Elasticity | Std. err. | $\mathbf{z}$ | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | :---: | ---: |
| dairy | .90727 | .0157948 | 57.44 | 0.000 | .8763129 | .9382272 |
| proteins | 1.0912 | .0102476 | 106.48 | 0.000 | 1.071115 | 1.111285 |
| fruitveg | .9629558 | .0128444 | 74.97 | 0.000 | .9377813 | .9881303 |
| flours | .8728076 | .0152504 | 57.23 | 0.000 | .8429174 | .9026979 |
| misc | 1.000333 | .0177894 | 56.23 | 0.000 | .9654664 | 1.0352 |

Note: Elasticities are calculated at prices', demographic variables', and expenditure means.
. estimates store translog
We specified the nocoeftable option to suppress the coefficient table and the elasticities (expenditure) option to obtain expenditure elasticities. Finally, we used the labels() option to name our five goods. Had we not done that, then the output would simply have numbered the goods from one to five.

For comparison's sake, we also fit the equivalent generalized translog model. We note that despite the long commands, the following one differs from the previous by just one character:

```
. demandsys gtranslog w_dairy w_proteins w_fruitveg w_flours w_misc,
> prices(p_dairy p_proteins p_fruitveg p_flours p_misc)
> demographics(n_kids n_adults)
> labels("dairy proteins fruitveg flours misc")
> expenditure(expfd) elasticities(expenditure) nolog nocoeftable
Calculating NLS estimates ...
Calculating FGNLS estimates ...
FGNLS iteration 2 ...
FGNLS iteration 3 ...
FGNLS iteration 4 ...
Generalized translog model Number of obs = 4,160
Expenditure variable: expfd Number of goods = 5
Demographic method: Translating Number of demographics = 2
Centered R2 of model for
    dairy = 0.0478
    proteins = 0.1615
    fruitveg = 0.0988
    flours = 0.1903
    misc = 0.1732
Calculating expenditure elasticities ...
Expenditure elasticities
```

| Expenditure | Elasticity | Std. err. | $z$ | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | ---: | ---: |
| dairy | .906851 | .0174393 | 52.00 | 0.000 | .8726707 | .9410313 |
| proteins | 1.074059 | .01144 | 93.89 | 0.000 | 1.051637 | 1.096481 |
| fruitveg | .9716098 | .0144251 | 67.36 | 0.000 | .943337 | .9998826 |
| flours | .892461 | .0170431 | 52.36 | 0.000 | .8590571 | .9258649 |
| misc | 1.023297 | .0194942 | 52.49 | 0.000 | .9850886 | 1.061504 |

Note: Elasticities are calculated at prices', demographic variables', and expenditure means.

```
. estimates store gtranslog
```

The $R^{2}$ statistics from the generalized translog model are trivially higher than for the basic translog model. Because demandsys is based on maximum likelihood estimation and because the translog model is nested within the generalized translog model, we can use a likelihood-ratio test to see whether the inclusion of committed quantities is warranted here.

```
. lrtest gtranslog translog
Likelihood-ratio test
Assumption: translog nested within gtranslog
    LR chi2(5) = 59.00
Prob > chi2 = 0.0000
```

The generalized translog model adds additional $G$ parameters over the basic translog model, so the $\chi^{2}$ statistic has five degrees of freedom in this example. The result indicates that the use of the generalized version of the translog model is warranted. Of course, if you do this test for your own models, you must use the same demographic specification in both.

As we cautioned with the LES, interpreting the $\mu$ parameters of the generalized translog model as subsistence quantities requires care. We do not force the estimates to be positive, nor does demandsys check that $\sum_{g} p_{g i}\left(\mu_{g}+\boldsymbol{\nu}_{g}^{\prime} \mathbf{d}_{i}\right) \leq m_{i}$ for each observation $i$ in the dataset or that $\mu_{g}+\nu_{g}^{\prime} \mathbf{d}_{i} \geq 0$ for each observation and each good, both of which must be true if the subsistence argument is to be credible. We have noted that the generalized translog model can produce negative estimates of $\mu$ 's quite frequently and changing the demographic specification is often enough to produce negative estimates.

For example, if we refit the previous example controlling for just the number of children but not the number of adults, we obtain negative estimates for some of the $\mu$ parameters:

```
. demandsys gtranslog w_dairy w_proteins w_fruitveg w_flours w_misc,
> prices(p_dairy p_proteins p_fruitveg p_flours p_misc)
> demographics(n_kids) expenditure(expfd) nolog
Calculating NLS estimates ...
Calculating FGNLS estimates ...
FGNLS iteration 2 ...
FGNLS iteration 3 ...
FGNLS iteration 4 ...
\begin{tabular}{lll} 
Generalized translog model & Number of obs & \(=4,160\) \\
Expenditure variable: expfd & Number of goods & \(=\) \\
Demographic method: Translating & Number of demographics & \(=\) \\
\hline
\end{tabular}
Centered R2 of model for
    Good 1 = 0.0421
    Good 2 = 0.1600
    Good 3 = 0.0929
    Good 4 = 0.1825
    Good 5 = 0.1706
```

|  | Estimate | Std. err. | z | $P>\|z\|$ | [95\% con | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| alpha |  |  |  |  |  |  |
| Good |  |  |  |  |  |  |
| 1 | . 1745891 | . 0097353 | 17.93 | 0.000 | . 1555082 | . 1936699 |
| 2 | . 3416641 | . 0188744 | 18.10 | 0.000 | . 304671 | . 3786572 |
| 3 | . 3634044 | . 0192262 | 18.90 | 0.000 | . 3257218 | . 401087 |
| 4 | . 0598082 | . 0066154 | 9.04 | 0.000 | . 0468423 | . 0727741 |
| Gamma |  |  |  |  |  |  |
| Good_g\# |  |  |  |  |  |  |
| Good_h |  |  |  |  |  |  |
| 1\#1 | . 0232919 | . 002496 | 9.33 | 0.000 | . 0183998 | . 0281841 |
| 1\#2 | -. 0146955 | . 0032932 | -4.46 | 0.000 | -. 02115 | -. 0082411 |
| 1\#3 | . 0005704 | . 0018905 | 0.30 | 0.763 | -. 0031349 | . 0042758 |
| 1\#4 | . 001351 | . 0009723 | 1.39 | 0.165 | -. 0005546 | . 0032566 |
| 1\#5 | . 0026193 | . 0010776 | 2.43 | 0.015 | . 0005073 | . 0047312 |
| 2\#2 | . 120652 | . 0085591 | 14.10 | 0.000 | . 1038764 | . 1374275 |
| 2\#3 | -. 0301533 | . 0055644 | -5.42 | 0.000 | -. 0410593 | -. 0192474 |
| 2\#4 | -. 0184411 | . 0027052 | -6.82 | 0.000 | -. 0237432 | -. 013139 |
| 2\#5 | -. 0228452 | . 0032034 | -7.13 | 0.000 | -. 0291238 | -. 0165667 |
| 3\#3 | . 0698867 | . 0067428 | 10.36 | 0.000 | . 056671 | . 0831023 |
| 3\#4 | . 0006331 | . 0013394 | 0.47 | 0.636 | -. 0019921 | . 0032583 |
| 3\#5 | -. 0065477 | . 0017672 | -3.71 | 0.000 | -. 0100113 | -. 003084 |
| 4\#4 | . 0233853 | . 0023855 | 9.80 | 0.000 | . 0187097 | . 0280608 |
| 4\#5 | -. 0074747 | . 0011361 | -6.58 | 0.000 | -. 0097014 | -. 005248 |
| 5\#5 | . 0323539 | . 0030878 | 10.48 | 0.000 | . 0263019 | . 0384059 |


| mu |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Good |  |  |  |  |  |  |
| 1 | -. 1560689 | . 0474664 | -3.29 | 0.001 | -. 2491014 | -. 0630364 |
| 2 | -. 811357 | . 2053695 | -3.95 | 0.000 | -1.213874 | -. 4088403 |
| 3 | -. 7661874 | . 3576453 | -2.14 | 0.032 | -1.467159 | -. 0652155 |
| 4 | . 2121302 | . 0367754 | 5.77 | 0.000 | . 1400517 | . 2842088 |
| 5 | . 0887401 | . 0446207 | 1.99 | 0.047 | . 0012853 | . 176195 |
| Nu |  |  |  |  |  |  |
| Good\# |  |  |  |  |  |  |
| c.n_kids |  |  |  |  |  |  |
| 1 | -. 5508243 | . 1957429 | -2.81 | 0.005 | -. 9344733 | -. 1671753 |
| 2 | -1.350384 | . 1934558 | -6.98 | 0.000 | -1.72955 | -. 9712174 |
| 3 | -2.749218 | . 303765 | -9.05 | 0.000 | -3.344586 | -2.153849 |
| 4 | . 0446037 | . 0442943 | 1.01 | 0.314 | -. 0422117 | . 131419 |
| 5 | -. 1036959 | . 0559887 | -1.85 | 0.064 | -. 2134318 | . 0060399 |

Note: alpha estimates are constant expenditures for each good.
Note: Gamma estimates measure the effect of price on expenditures shares across goods.
Note: mu estimates are subsistence levels of consumption for each good.
Note: Nu estimates measure the effect of demographic variables on shares and subsistence levels of consumption.
Normalized parameter

|  |  | Estimate | Std. err. | $z$ | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| :--- | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| alpha | Good |  |  |  |  |  |  |
|  | 5 | .0605342 | .007785 | 7.78 | 0.000 | .045276 | .0757925 |

Note: alpha estimates sum to 1.
Shares: w_dairy w_proteins w_fruitveg w_flours w_misc
Prices: p_dairy p_proteins p_fruitveg p_flours p_misc
Expenditure: expfd
Demographics: n_kids
Here, again, we see factor-variable notation for the parameter matrix $\mathbf{N}$ but also for $\boldsymbol{\Gamma}$. As we mentioned before, you should not interpret the factor-variable notation as you would in a regression output. We see the estimate labeled $1 \# 2$ under Good_g\#Good_h. This is the estimate of $\gamma_{12}$, the coefficient associated with $\ln p_{2} / \bar{m}$ in the equation for Good 1.

## AIDS

One of the most commonly used demand systems is the celebrated AIDS model of Deaton and Muellbauer (1980b). Among other benefits, the AIDS model can be viewed as a first-order approximation to any demand system. Historically, the AIDS model was also favored because it is "almost linear" in the sense that replacing the price index with an approximation results in share equations that are linear. Given the speed with which computers can fit nonlinear systems of equations, we do not implement that approximation.

An additional nice property of the AIDS model is that the functional form for its budget shares is a member of the "price-independent generalized log-linear" and the broader "price-independent generalized linear" families. An advantage to price-independent generalized linear expenditure shares is that they satisfy certain conditions required for the existence of a representative consumer and the related theory of aggregation. In short, expenditure shares of this form can be fit to data aggregated across consumers. The implied market demand curves are consistent with a single utility-maximizing
representative agent and satisfy the same conditions like Slutsky symmetry that hold for demand curves implied by a single consumer's utility-maximizing behavior. For more on price-independent generalized linear demands, see, for example, Muellbauer (1975), Lewbel (1989), and Pollak and Wales (1992, chap. 2).

The starting point for the AIDS model is the expenditure function

$$
\begin{equation*}
\ln e\left(\mathbf{p}, u ; \alpha_{0}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\Gamma}\right)=(1-\ln u) \ln a(\mathbf{p})+\ln u\{\ln a(\mathbf{p})+b(\mathbf{p})\} \tag{2}
\end{equation*}
$$

where $u$ is utility, the price index $\ln a(\mathbf{p})$ is defined as

$$
\ln a(\mathbf{p}) \equiv \alpha_{0}+\sum_{g} \alpha_{g} \ln p_{g}+\frac{1}{2} \sum_{g} \sum_{h} \gamma_{g h} \ln p_{g} \ln p_{h}
$$

where $p_{g}$ is the price of good $g$, and the price aggregator $b(\mathbf{p})$ is defined as

$$
b(\mathbf{p}) \equiv \beta_{0} \prod_{g} p_{g}^{\beta_{g}}
$$

Parameter $\beta_{0}$ cannot be distinguished from $u$, so we take $\beta_{0}=1$ in the following. In their original formulation, Deaton and Muellbauer (1980b) wrote their expenditure function slightly differently, with $u$ in place of $\ln u$ in (2). We have chosen to use $\ln u$ to draw some comparisons with the QUAIDS model later, and because utility is an ordinal concept, it will not affect the expenditure-share equations.

Suppose that all prices are equal to one. Then (2) with $\beta_{0}=1$ implies that $\ln e(p, u)=\alpha_{0}$. Thus, $\exp \left(\alpha_{0}\right)$ can be interpreted as the level of expenditure needed for minimal subsistence when all prices are equal to one. In practice, estimating $\alpha_{0}$ is difficult, particularly when prices are correlated, as they often are. Therefore, by default, we set $\alpha_{0}$ equal to the natural logarithm of the minimum level of expenditure in the estimation sample. You can override the default by specifying the piconstant() option. Of course, prices are generally not all equal to one, so you will want to try various values for $\alpha_{0}$ to see how sensitive elasticities and other calculations you may perform are to its value.

Expenditure functions consistent with rational consumer choice must be homogeneous of degree one in prices and expenditure, which imply that we must have $\sum_{g} \alpha_{g}=1, \sum_{g} \beta_{g}=0$, and $\sum_{g} \gamma_{g h}=\sum_{h} \gamma_{g h}=0$. Slutsky symmetry further requires that $\gamma_{g h}=\gamma_{h g}$ for all $g$ and $h$. Note that demandsys imposes these restrictions on the model being fit. An alternative is to fit the model with and without one or more of those restrictions and then test whether they hold. However, such unrestricted models can often fail to converge because of the increased number of parameters or because the models are "too flexible" and contain unidentifiable parameters.

Shephard's (1970) lemma equates the Hicksian demand functions with the partial derivatives of the expenditure function with respect to the price of the goods. Using Shephard's lemma, we can write the expenditure-share functions as

$$
\begin{equation*}
w_{g}\left(\mathbf{p}, m ; \alpha_{0}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\Gamma}\right)=\alpha_{g}+\sum_{h} \gamma_{g h} \ln p_{h}+\beta_{g} \ln \left\{\frac{m}{a(\mathbf{p})}\right\} \tag{3}
\end{equation*}
$$

Notice the simplicity of these share equations. Good $g$ 's expenditure share depends linearly on the $\log$ prices of all the goods as well as the $\log$ of an expenditure term. The expenditure term $m / a(\mathbf{p})$ essentially adjusts each household's expenditure to control the cost of living faced by that household. This term is sometimes referred to as deflated expenditure. Equation (3) also makes clear that luxury goods have $\beta_{g}>1$ while necessities have $\beta_{g}<1$; inferior goods have $\beta_{g}<0$.

Setting $e(\mathbf{p}, u)=m$ and manipulating (2) yields the indirect utility function

$$
\begin{equation*}
\ln v\left(\mathbf{p}, m ; \alpha_{0}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\Gamma}\right)=\frac{\ln m-\ln a(\mathbf{p})}{b(\mathbf{p})} \tag{4}
\end{equation*}
$$

We will come back to this equation below. We show how the indirect utility function plays a role in computing welfare measures in $[R]$ demandsys postestimation.

## QUAIDS

The Engel curves for the AIDS model have the so-called "Working-Leser" form after Working (1943) and Leser (1963),

$$
w_{g}=\omega_{g}+\psi_{g} \ln m
$$

so that the share of expenditures spent on good $g$ is a linear function of the logarithm of total expenditures. Banks, Blundell, and Lewbel (1997) provided evidence that in fact linear Engel curves provide a poor fit for many goods. They therefore consider Engel curves with an additional expenditure term,

$$
\begin{equation*}
w_{g}=A_{g}(\mathbf{p})+B_{g}(\mathbf{p}) \ln \left\{\frac{m}{a(\mathbf{p})}\right\}+C_{g}(\mathbf{p}) f\left\{\frac{m}{a(\mathbf{p})}\right\} \tag{5}
\end{equation*}
$$

where $A_{g}(\mathbf{p}), B_{g}(\mathbf{p})$, and $C_{g}(\mathbf{p})$ are differentiable functions of prices and $f\{m / a(\mathbf{p})\}$ is a differentiable function of real expenditures. $a(\mathbf{p})$ is the same translog price index as in the AIDS model.

Lewbel (1991), building on the work of Gorman (1981), showed that any exactly aggregable demand system must have the form of (5) and that additional terms containing functions of $m / a(\mathbf{p})$ are not theoretically possible if exact aggregability is to hold. Lewbel (1991) termed demand systems of the form (5) as having rank three because there are three linearly independent terms; Gorman (1981) showed that the maximum rank of demand systems is three if exact aggregability is to hold. In contrast, the AIDS model has rank two.

Banks, Blundell, and Lewbel (1997) further showed that the only indirect utility functions consistent with rank-three expenditure-share equations like (5) have the form

$$
\ln v(\mathbf{p}, m)=\left[\left\{\frac{\ln m-\ln a(\mathbf{p})}{b(\mathbf{p})}\right\}^{-1}+\lambda(\mathbf{p})\right]^{-1}
$$

where in their application they use the same definitions for $a(\mathbf{p})$ and $b(\mathbf{p})$ as in the AIDS model and they take $\lambda(\mathbf{p})=\sum_{g} \lambda_{g} \ln p_{g}$ with $\sum_{g} \lambda_{g}=0$. As with the restrictions implied by economic theory for the AIDS model, demandsys imposes the restriction that the $\lambda$ 's sum to zero. Also notice that if we restrict $\lambda_{g}=0$ for all $g$, we are left with the indirect utility function for the AIDS model shown in (4).

Solving for $\ln m$ yields the cost function

$$
\ln e(\mathbf{p}, u)=\frac{b(\mathbf{p})}{\frac{1}{\ln u}-\lambda(\mathbf{p})}+\ln a(\mathbf{p})
$$

Applying Shephard's lemma, we have the expenditure-share functions

$$
w_{g}(\mathbf{p}, m ; \boldsymbol{\alpha}, \boldsymbol{\Gamma}, \boldsymbol{\Lambda})=\alpha_{g}+\sum_{h} \gamma_{g h} \ln p_{h}+\beta_{g} \ln \left\{\frac{m}{a(\mathbf{p})}\right\}+\frac{\lambda_{g}}{b(\mathbf{p})}\left[\ln \left\{\frac{m}{a(\mathbf{p})}\right\}\right]^{2}
$$

This equation also makes clear our earlier claim that if $\lambda_{g}=0$ for all $g$ then the QUAIDS model reduces to the AIDS model; see (3).

## Controlling for demographic factors

demandsys allows you to choose between two different methods of incorporating demographic characteristics into AIDS and QUAIDS. The first method is the same as we have used with demand systems we have already covered: Pollak and Wales's (1978) demographic translation. An added benefit of translation is that it allows us to introduce constant terms that might be interpreted as subsistence or committed quantities. A possible downside to demographic translation is that the intuition behind how demographic variables affect expenditure shares may not strike all users as entirely lucid. Essentially, demographic variables alter the level of subsistence for each household or observation. However, as we cautioned above in our discussion of the generalized translog model, there are no restrictions to force $\sum_{g} p_{g i}\left(\mu_{g}+\boldsymbol{\nu}_{g}^{\prime} \mathbf{d}_{i}\right) \leq m_{i}$, nor are there any restrictions to force $\left(\mu_{g}+\nu_{g}^{\prime} \mathbf{d}_{i}\right) \geq 0$.

The second method of incorporating demographics that demandsys implements is based on Ray (1983) and Poi (2002) and is known as demographic scaling. This method explicitly allows for demographic variables to have "scale" and "composition" effects on expenditures as we explain below. There are two possible downsides to demographic scaling. First, it does not allow one to incorporate subsistence quantities into the demand system. While one may be tempted to include a constant term in the list of demographic variables because that is essentially what demographic translation does to incorporate subsistence quantities, the resulting parameter estimates do not have such a clearcut interpretation. Second, in our experience, models that incorporate demographic scaling sometimes take many iterations to converge. Specifying fewer demographic characteristics may help achieve convergence. We think this method is more intuitive than demographic translation, but either method should prove adequate for most applications.

## Demographic translation

We have already discussed demographic translation in the context of the LES and the translog and generalized translog demand systems, so we will not show the details here. In Methods and formulas, we show the formula for the expenditure shares for the generalized QUAIDS model, and we discuss the parameter restrictions that give rise to the generalized AIDS model and their nongeneralized variants.
demandsys provides for four models in the AIDS family that can incorporate demographic variables via demographic translation: aids, quaids, gaids, and gquaids. The first two models do not allow for committed quantities, while the latter two represent generalized variants that do. The arguments in Pollak and Wales $(1992,75)$ suggest that the generalized variants are to be preferred because doing so ensures that a constant term is included in each demand equation (compare with share equation). Whether the constants allow for a subsistence interpretation is another matter, and in [R] demandsys postestimation, we provide an example of how to see whether that interpretation holds.

## Demographic scaling

demandsys also implements the demographic scaling method used by Ray (1983) for the AIDS model and extended by Poi (2002) to the QUAIDS model. For each consumer, we again have a $D$-length vector of demographic characteristics $\mathbf{d}$. Suppose $\mathbf{d}$ is the number of children in a household and we are modeling expenditures on household goods, rent, utilities, and food. Then the household's expenditures will presumably increase by virtue of there being more members in the household; call this the scale effect. Second, the consumption pattern of the household may shift as more money is spent on items consumed by children versus adults; think of that as a composition effect. Ray (1983) does this by writing the expenditure function as

$$
e(\mathbf{p}, \mathbf{d}, u)=e^{R}(\mathbf{p}, u) \bar{m}_{0}(\mathbf{d}) \phi(\mathbf{p}, \mathbf{d}, u)
$$

Here $e^{R}(\mathbf{p}, u)$ is the expenditure function for a reference household, where $\mathbf{d}=0 . \bar{m}_{0}(\mathbf{d})$ increases total expenditure of the household and accounts for the scale effect. $\phi(\mathbf{p}, \mathbf{d}, u)$ controls for composition effects. For AIDS, Ray (1983) controlled for both of these effects by taking

$$
\bar{m}_{0}(\mathbf{d})=1+\rho^{\prime} \mathbf{d}
$$

for $G \times 1$ parameter vector $\rho$ and

$$
\phi(\mathbf{p}, \mathbf{d}, u)=\exp \left\{u \prod_{i} p_{g}^{\beta_{g}}\left(\prod_{g} p_{g}^{\boldsymbol{\eta}_{g} \mathbf{d}}-1\right)\right\}
$$

where $\boldsymbol{\eta}_{g}$ is the $g$ th row of $G \times D$ parameter matrix $\mathbf{H}$. In Methods and formulas, we show that the expenditure-share functions for the AIDS model with demographic scaling are

$$
w_{g}\left(\mathbf{p}, m, \mathbf{d} ; \alpha_{0}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\Gamma}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{H}\right)=\alpha_{g}+\sum_{h} \gamma_{g h} \ln p_{h}+\left(\beta_{g}+\boldsymbol{\eta}_{i} \mathbf{d}\right) \ln \left\{\frac{m}{\bar{m}_{0}(\mathbf{d} ; \boldsymbol{\rho}) a(\mathbf{p})}\right\}
$$

Basically, the function $\bar{m}_{0}(\mathbf{d} ; \boldsymbol{\rho})$ acts to reduce the effective amount of money available for spending. If $\mathbf{d}$ represents a set of indicator variables for regions of a country, then $\bar{m}_{0}(\mathbf{d} ; \boldsymbol{\rho})$ is controlling for the cost of living in different regions of the country. The function $\phi(\mathbf{p}, \mathbf{d}, u)$ is to make each good's expenditure expansion path a function of the household's demographics. Rather than all households increasing their expenditure share of good $g$ by the same $\beta_{g}$ in response to a change in $m$, now their responses can vary by household composition.

For the QUAIDS model, we show in Methods and formulas that an appropriate choice for $\phi(\mathbf{p}, \mathbf{d}, u)$ leads to expenditure share equations

$$
\begin{aligned}
w_{g}\left(\mathbf{p}, m, \mathbf{d} ; \alpha_{0}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\Gamma}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{H}\right)=\alpha_{g} & +\sum_{h} \gamma_{g h} \ln p_{h}+\left(\beta_{g}+\boldsymbol{\eta}_{i} \mathbf{d}\right) \ln \left\{\frac{m}{\bar{m}_{0}(\mathbf{d} ; \boldsymbol{\rho}) a(\mathbf{p})}\right\} \\
& +\frac{\lambda_{g}}{b(\mathbf{p}) c(\mathbf{p}, \mathbf{d})}\left[\ln \left\{\frac{m}{\bar{m}_{0}(\mathbf{d} ; \boldsymbol{\rho}) a(\mathbf{p})}\right\}\right]^{2}
\end{aligned}
$$

The models aids and quaids allow for demographic scaling. To request demographic scaling, rather than demographic translation, we specify the scaling suboption of the demographics() option, as the next example shows.

## > Example 5

Here we fit a QUAIDS model to our food consumption data, controlling for the numbers of children and adults in each household. We request that demographic scaling be used rather than the default demographic translation.

```
. use https://www.stata-press.com/data/r18/food_consumption
(Food consumption)
. demandsys quaids w_dairy w_proteins w_fruitveg w_flours w_misc,
> prices(p_dairy p_proteins p_fruitveg p_flours p_misc)
> demographics(n_kids n_adults, scaling)
> labels("dairy proteins fruitveg flours misc")
> expenditure(expfd) elasticities(expenditure) nolog nocoeftable
Calculating NLS estimates ...
Calculating FGNLS estimates ...
FGNLS iteration 2 ...
FGNLS iteration 3 ...
FGNLS iteration 4 ...
Quadratic AIDS model Number of obs = 4,160
Expenditure variable: expfd Number of goods = 5
Price index constant = 1.615
Demographic method: Scaling Number of demographics = 2
Centered R2 of model for
    dairy = 0.0453
    proteins = 0.1530
    fruitveg = 0.0949
    flours = 0.1813
    misc = 0.1726
Calculating expenditure elasticities ...
Expenditure elasticities
```

| Expenditure | Elasticity | Std. err. | $z$ | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | ---: | ---: |
| dairy | .8918188 | .0176252 | 50.60 | 0.000 | .857274 | .9263636 |
| proteins | 1.077283 | .0117431 | 91.74 | 0.000 | 1.054267 | 1.100299 |
| fruitveg | 1.043209 | .014428 | 72.30 | 0.000 | 1.01493 | 1.071487 |
| flours | .8409016 | .0162384 | 51.78 | 0.000 | .8090749 | .8727284 |
| misc | .9460643 | .0189172 | 50.01 | 0.000 | .9089873 | .9831414 |

Note: Elasticities are calculated at prices', demographic variables', and expenditure means.
. estimates store quaids_s
Because the QUAIDS model with demographic scaling for a 5-good system with 2 demographic variables has 32 estimated parameters, we instructed demandsys to report expenditure elasticities rather than the estimated parameters. We draw your attention to this part of the command:

```
demographics(n_kids n_adults, scaling)
```

Notice that we specified the suboption scaling. Had we not included this suboption, then demandsys would have used demographic translation instead. Finally, we asked Stata to store these estimation results as quaids_s.

The header of the output shows that demandsys fit a QUAIDS model as requested and that our demographic variables were incorporated using demographic scaling. Take note of the line that reads

```
Price index constant = 1.615
```

In our discussion of the basic AIDS model, we discussed the price index $a(\mathbf{p})$ that includes a constant term $\alpha_{0}$. We discussed that when all prices are equal to one, then $\exp \left(\alpha_{0}\right)$ can be interpreted as the
level of expenditure needed for minimal subsistence. demandsys looked at our expenditure variable expfd and found that the minimum value within the estimation sample is $\$ 5.03$. It therefore set $\alpha_{0}=\ln 5.03=1.615$. You can specify your own value for $\alpha_{0}$ by using the piconstant () option, which stands for price index constant.

The QUAIDS model nests the AIDS model, so we can easily use a likelihood-ratio test to see whether the quadratic terms in the expenditure-share equations contribute to the explanatory power of the model. We include quietly to fit the corresponding aids model without displaying the results and store the estimation results:

```
. quietly demandsys aids w_dairy w_proteins w_fruitveg w_flours w_misc,
> prices(p_dairy p_proteins p_fruitveg p_flours p_misc)
> demographics(n_kids n_adults, scaling)
> labels("dairy proteins fruitveg flours misc") expenditure(expfd)
. estimates store aids_s
```

Because we specified quietly, we did not request expenditure elasticities. Doing so would have required Stata to do more computations that we would not have seen.

We now perform the likelihood-ratio test:

```
. lrtest quaids_s aids_s
Likelihood-ratio test
Assumption: aids_s nested within quaids_s
    LR chi2(4) = 36.58
Prob > chi2 = 0.0000
```

In this example, we reject the null hypothesis that the four free $\lambda$ parameters are jointly equal to zero. Hence, the use of the QUAIDS model instead of the linear AIDS model is justified here. (Recall that to satisfy the adding-up constraint, we impose $\sum_{g} \lambda_{g}=0$ so that in our example the fifth parameter is determined by the first four and is not estimated.)

When one conducts tests of one model versus another using the likelihood-ratio principle, the more restrictive model must be nested within the more general model. For example, each of the AIDS models implemented here is nested within the corresponding QUAIDS models, and the AIDS model with demographic translation is nested within the generalized AIDS model. But it would make no sense to fit a generalized AIDS model and compare it with a generalized translog model because one is not a restricted variant of the other. In Methods and formulas, we describe the most flexible variant within each class of demand system, and we describe the parameter restrictions that give rise to less flexible variants; using lrtest to choose among members within the same class is valid.

Also, for a likelihood-ratio test to be valid, the model specifications-apart from the component we are testing-must be identical. That is, the two models must include the same estimation sample, the same goods, and the same demographic specification, including the method by which demographic variables are introduced for AIDS and QUAIDS models. Moreover, for AIDS and QUAIDS models, you must use the same value for the price index constant $\alpha_{0}$. If you do not specify the piconstant () option, demandsys will use the same value, assuming you use the same estimation sample.

## > Example 6

In this example, we will fit a generalized QUAIDS model, again controlling for the number of children and adults. Because we are fitting a generalized model, the demographics will be incorporated via demographic translation. To demonstrate the use of the piconstant() option, we will specify $\alpha_{0}=3$.
. use https://www.stata-press.com/data/r18/food_consumption
(Food consumption)
. demandsys gquaids w_dairy w_proteins w_fruitveg w_flours w_misc,
> prices(p_dairy p_proteins p_fruitveg p_flours p_misc)
> demographics(n_kids n_adults) piconstant(3) expenditure (expfd) nolog
Calculating NLS estimates ...
Calculating FGNLS estimates ...
FGNLS iteration 2 ...
FGNLS iteration 3 ...
FGNLS iteration 4 ...
Generalized quadratic AIDS model

| Number of obs | $=$ | 4,160 |
| :--- | :--- | ---: |
| Number of goods | $=$ | 5 |
| Price index constant | $=$ | 3 |
| Number of demographics | $=$ | 2 |

Demographic method: Translating
Number of demographics $=2$
Centered R2 of model for
Good $1=0.0486$
Good $2=0.1596$
Good $3=0.0996$
Good $4=0.1893$
Good $5=0.1715$


| mu |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Good |  |  |  |  |  |  |
| 1 | 1.06405 | . 2097966 | 5.07 | 0.000 | . 6528563 | 1.475244 |
| 2 | . 6058134 | . 452923 | 1.34 | 0.181 | -. 2818993 | 1.493526 |
| 3 | 2.146931 | . 6621706 | 3.24 | 0.001 | . 8491008 | 3.444762 |
| 4 | . 3204353 | . 0685528 | 4.67 | 0.000 | . 1860742 | . 4547964 |
| 5 | . 3588502 | . 0854053 | 4.20 | 0.000 | . 1914589 | . 5262414 |
| Nu |  |  |  |  |  |  |
| $\begin{gathered} \text { Good\# } \\ \text { c.n_kids } \end{gathered}$ |  |  |  |  |  |  |
| 1 | -. 2399127 | . 1953618 | -1.23 | 0.219 | -. 6228148 | . 1429893 |
| 2 | -1.176653 | . 1938486 | -6.07 | 0.000 | -1.556589 | -. 7967166 |
| 3 | -2.604984 | . 3121795 | -8.34 | 0.000 | -3.216844 | -1.993123 |
| 4 | . 1120979 | . 0416122 | 2.69 | 0.007 | . 0305396 | . 1936563 |
| 5 | -. 0296108 | . 0541832 | -0.55 | 0.585 | -. 1358079 | . 0765863 |
| $\begin{aligned} & \text { Good\# } \\ & \text { c.n_adults } \end{aligned}$ |  |  |  |  |  |  |
| 1 | -1.271932 | . 2071674 | -6.14 | 0.000 | -1.677972 | -. 8658909 |
| 2 | -1.278973 | . 2249348 | -5.69 | 0.000 | -1.719837 | -. 8381089 |
| 3 | -2.698807 | . 3809636 | -7.08 | 0.000 | -3.445482 | -1.952132 |
| 4 | -. 056265 | . 0483113 | -1.16 | 0.244 | -. 1509535 | . 0384234 |
| 5 | -. 269388 | . 0569314 | -4.73 | 0.000 | -. 3809716 | -. 1578045 |

Note: alpha estimates are constant terms in expenditure-share equations and also appear in the price index.
Note: beta estimates measure sensitivity of expenditure shares to changes in deflated expenditure and also appear in the price aggregator function.
Note: Gamma estimates measure the effect of price on expenditures shares across goods.
Note: lambda estimates measure the sensitivity of expenditure shares to changes in deflated expenditure.
Note: Nu estimates measure the effect of demographic variables on expenditure shares of each good.
Normalized parameters

|  | Estimate | Std. err. | z | $P>\|z\|$ | [95\% con | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| alpha | . 0849402 | . 0049286 | 17.23 | 0.000 | . 0752803 | . 0946001 |
| $\begin{gathered} \text { Good } \\ 5 \end{gathered}$ |  |  |  |  |  |  |
| beta | . 0081246 | . 0050562 | 1.61 | 0.108 | -. 0017854 | . 0180345 |
| Good 5 |  |  |  |  |  |  |
| Gamma |  |  |  |  |  |  |
| Good_g\# <br> Good_h |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| 1\#5 | . 0030308 | . 0014661 | 2.07 | 0.039 | . 0001572 | . 0059044 |
| 2\#5 | -. 0318352 | . 0023155 | -13.75 | 0.000 | -. 0363734 | -. 0272969 |
| 3\#5 | -. 0120502 | . 0019986 | -6.03 | 0.000 | -. 0159674 | -. 008133 |
| 4\#5 | -. 0093209 | . 0011512 | -8.10 | 0.000 | -. 0115772 | -. 0070646 |
| lambda |  |  |  |  |  |  |
| Good |  |  |  |  |  |  |
| 5 | . 0020771 | . 0019631 | 1.06 | 0.290 | -. 0017705 | . 0059247 |

```
Note: alpha estimates sum to 1.
Note: beta estimates sum to 0.
Note: Gamma estimates sum to O over goods.
Note: lambda estimates sum to O over goods.
Shares: w_dairy w_proteins w_fruitveg w_flours w_misc
Prices: p_dairy p_proteins p_fruitveg p_flours p_misc
Expenditure: expfd
Demographics: n_kids n_adults
```

We see in the header of the output that demandsys did set the price index constant $\alpha_{0}=3$ as per our request.

The $\mu$ parameter estimates are all greater than zero, but it is difficult to interpret them in isolation. Because we control for the number of children and the number of adults in each household, the $\mu$ parameters themselves would represent subsistence expenditure shares for a hypothetical household with neither any children nor any adults! In a somewhat technical example in $[\mathrm{R}]$ demandsys postestimation, we continue this example and show how to recover the estimated $\boldsymbol{\mu}$ and $\mathbf{N}$ parameters and then use those estimates to calculate the actual subsistence shares for each household. We then compare the calculated subsistence expenditures with actual expenditures to see whether the subsistence interpretation even holds.

## Epilogue

demandsys provides easy access to some of the most commonly used demand systems and their extensions. You can control for demographic characteristics. Options and postestimation commands make obtaining elasticities trivial.

Despite what may appear to be a thorough implementation, we have only scratched the surface. demandsys is designed for large cross-sectional datasets. The Rotterdam model of Theil (1965) and Barten (1966) and its extensions are widely used in the analysis of time-series data.

A frequent concern is that datasets often have expenditure shares that are zero for some households. Deaton and Irish (1984) is an early contribution to the literature on how to deal with zero shares. Heien and Wessells (1990) proposed a two-step Heckman estimation procedure, but it has been shown to lead to inconsistent estimates (Vermeulen 2001). Tauchmann (2010) provides an alternative Heckmantype estimator that is consistent. Shonkwiler and Yen (1999) proposed an alternative estimator for zero shares. See also Yen and Lin (2006) for yet another approach and Meyerhoefer, Ranney, and Sahn (2005), who develop an estimator that controls for censoring with panel data. More recently, Caro et al. (2021) have developed the community-contributed Stata command quaidsce, which provides an implementation of Schonkwiler and Yen's estimator.

All of our estimators treat price as an exogenous variable. There are multiple reasons why one may want to allow for endogenous prices. In the classical supply and demand setting, which may occur for example in small villages, there are a small number of buyers and sellers, so the decisions that buyers make will influence prices, causing them to be endogenous in expenditure-share equations. A more pertinent concern is that prices may be measured with error. Consumption is often collected by survey data, and consumers are often not able to recall the exact price paid for an item. Unobserved quality differences also cause prices to be measured with error, especially when the goods being modeled are aggregates of individual items. For example, you may purchase filet mignon every week, while I purchase economy-grade ground beef. Both items are lumped together as "meat" even though, arguably, the two products are not even close substitutes; Nelson (1991) considers this type of problem. Spatial patterns may also make prices endogenous; see Case (1991) as an example.

Lecocq and Robin (2015) provide a community-contributed command called aidsills that allows one to fit an AIDS model with endogenous variables. Their command, however, provides the AIDS and QUAIDS models with only one method of including demographic variables that is similar to the demographic translation approach used here.

We have presented the QUAIDS model as somewhat of an endpoint because it has rank 3, which Gorman (1981) and Lewbel (1991) showed is the maximum rank of an exactly aggregable demand system. Lewbel and Pendakur (2009) develop what they call a theory of "implicit Marshallian demands" that are not within the class of demand equations considered by Gorman and Lewbel and hence can have any rank. Lewbel and Pendakur propose a demand system they call the exact affine Stone index that allows for interactions between prices and expenditures and allows for much more flexible Engel curves. Their empirical application shows that observed demands often deviate from those implied by demands that are linear or quadratic in income. Moreover, their model can be fit using the generalized method of moments, so the possibility of controlling for price endogeneity exists.

## Stored results

demandsys stores the following in e():
Scalars
e(N)
e(k)
e(k_eq)
e(n_demos)
$e$ (n_goods)
e(mss_\#)
e(rss_\#)
e(tss_\#)
e(r2_\#)
e(11)
e(N_clust)
e(piconstant)
e(rank)
e(ic)
e(converged)
Macros
e(cmd)
e(cmdline)
$e$ (wtype)
e(wexp)
e(model)
e(model_eval)
e(title)
e(clustvar)
e(vce)
e(vcetype)
e(demo_type)
e(demos)
e(has_demos)
$e$ (expenditures)
e(prices)
e(shares)
e(r2_type)
$e$ (p_index)
e(properties)
e(estat_cmd)
e(predict)
e(marginsok)
number of observations
number of parameters
number of equations in $e(b)$
number of demographic factors
number of goods
model sum of squares for the \#th equation
residual sum of squares for the \#th equation
total sum of squares for the \#th equation
$R^{2}$ for the \#th equation
Gaussian $\log$ likelihood
number of clusters
number of constant term
rank of e(v)
number of iterations
1 if converged, 0 otherwise
demandsys
command as typed
weight type
weight expression
demand model
demand evaluator
title in estimation output
name of cluster variable
vcetype specified in vce()
title used to label Std. err.
demographic variable specification
demographic variables
1 if demographic variables are used, 0 otherwise
expenditures variable
price variables
expenditure-share variables
$R^{2}$ type computed
whether model has price index
b V
program used to implement estat
program used to implement predict
predictions allowed by margins

```
Matrices
    e(b) coefficient vector
    e(Sigma) error covariance matrix (\widehat{\boldsymbol{\Sigma}})
    e(V) variance-covariance matrix of the estimators
    e(b_normalized) coefficient vector of the normalized parameters
    e(V_normalized) variance-covariance matrix of the normalized parameters
Functions
    e(sample) marks estimation sample
```

In addition to the above, the following is stored in r() :

Matrices
r(table)
matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r -class command is run after the estimation command.

## Methods and formulas

Methods and formulas are presented under the following headings:
Introduction
LES
Generalized translog
QUAIDS with demographic translation
QUAIDS with demographic scaling
Estimation

## Introduction

As in the text in Remarks and examples, we continue to use subscript $i=1, \ldots, N$ to index observations, $g=1, \ldots, G$ to index goods, and $d=1, \ldots, D$ to index demographic variables. In formulas where we must use double summations across goods, we use $h=1, \ldots, G$ and $j=1, \ldots, G$ to index goods. We omit the observation subscript for most of this discussion; all variables and expressions without $i$ subscripts implicitly refer to the $i$ th observation. Only when we sum across observations do we make the $i$ subscript explicit.

Let $w_{g}$ be the expenditure share for good $g$ defined as $w_{g} \equiv\left(p_{g} q_{g}\right) / m$, where $p_{g}$ is the price of good $g, q_{g}$ is the quantity of good $g$ consumed, and $m$ denotes total expenditure on all the goods in the system being modeled. We use $\mathbf{w}, \mathbf{p}$, and $\mathbf{q}$ to represent the $G \times 1$ vectors of expenditure shares, prices, and quantities for observation $i$. Let $\mathbf{d}$ represent the $D \times 1$ vector of demographic characteristics.

We use the notation $w_{g}$ to denote the observed expenditure share for good $g$ for observation $i$. We use the notation $w_{g}(\mathbf{p}, m ; \boldsymbol{\theta})$ to represent an expenditure-share functions where we will replace generic parameter vector $\boldsymbol{\theta}$ with each model's parameter vectors and matrices. We adorn direct and indirect utility functions and expenditure functions analogously.

We use the notation $E_{h}^{g}$ to denote the uncompensated (Marshallian) elasticity of the quantity of good $g$ with respect to the price of good $h$. We use the notation $E^{g}$ to note the expenditure elasticity of good $g$. Then, given the definition of $w_{g}$, one can easily verify the following facts that are useful in obtaining the formulas for the elasticities:

$$
\begin{aligned}
& E^{g}=\frac{1}{w_{g}} \frac{\partial w_{g}}{\partial \ln m}+1=\frac{m}{w_{g}} \frac{\partial w_{g}}{\partial m}+1 \\
& E_{g}^{g}=\frac{1}{w_{g}} \frac{\partial w_{g}}{\partial \ln p_{g}}-1=\frac{p_{g}}{w_{g}} \frac{\partial w_{g}}{\partial p_{g}}-1 \\
& E_{h}^{g}=\frac{1}{w_{g}} \frac{\partial w_{g}}{\partial \ln p_{h}}=\frac{p_{h}}{w_{g}} \frac{\partial w_{g}}{\partial p_{h}}
\end{aligned}
$$

The Slutsky equation can be written in elasticity form to obtain the compensated (Hicksian) elasticity of the quantity of good $g$ with respect to the price of good $h: \widetilde{E}_{h}^{g}=E^{g} \widehat{w}_{h}+E_{h}^{g}$, where $\widehat{w}_{h}$ is the predicted expenditure share for good $h$. We discuss elasticities in more detail, including why we use $\widehat{w}_{h}$ rather than $w_{h}$, in [ R$]$ demandsys postestimation.

For each demand system, we provide the equations for the $g$ th expenditure share, the direct utility function (if available), the indirect utility function, the expenditure function (if available), and the elasticities of the quantity of good $g$ with respect to expenditure and the price of good $h$. Predicted values of the first four items are available via predict; the elasticities are available via estat elasticities. When one demand system is nested within a more general demand system, we provide the equations for only the more general demand system and note what parameter restrictions would result in the less general demand system. Moreover, we include demographic variables in our exposition; versions of models without demographic variables result when the corresponding parameter vectors and matrices are set to zero. We also note what parameter restrictions are imposed at estimation to ensure adding up, homogeneity, and Slutsky symmetry.

## LES

Let $c_{g}=\mu_{g}+\boldsymbol{\nu}_{g} \mathbf{d}$, where $\boldsymbol{\nu}_{g}$ is the $g$ th row of $G \times D$ parameter matrix $\mathbf{N}$. The LES begins with the utility function

$$
u(\mathbf{q}, \mathbf{d} ; \boldsymbol{\beta}, \boldsymbol{\mu}, \mathbf{N})=\prod_{g}\left(q_{g}-c_{g}\right)^{\beta_{g}}
$$

Utility maximization yields the expenditure-share equations

$$
w_{g}(\mathbf{p}, m, \mathbf{d} ; \boldsymbol{\beta}, \boldsymbol{\mu}, \mathbf{N})=\frac{p_{g} c_{g}}{m}+\beta_{g}\left(1-\sum_{h} \frac{p_{h} c_{h}}{m}\right)
$$

To enforce adding up, we impose the constraint $\sum_{g} \beta_{g}=1$. Slutsky symmetry and homogeneity are implied by the functional form of the LES model. Straightforward algebra shows the indirect utility function is

$$
v(\mathbf{p}, m, \mathbf{d} ; \boldsymbol{\beta}, \boldsymbol{\mu}, \mathbf{N})=\frac{\prod_{g} \beta_{g}^{\beta_{g}}}{\prod_{g} p_{g}^{\beta_{g}}}\left(m-\sum_{g} p_{g} c_{g}\right)
$$

and inverting provides the expenditure function

$$
e(\mathbf{p}, u, \mathbf{d} ; \boldsymbol{\beta}, \boldsymbol{\mu}, \mathbf{N})=u \frac{\prod_{g} p_{g}^{\beta_{g}}}{\prod_{g} \beta_{g}^{\beta_{g}}}+\sum_{g} p_{g} c_{g}
$$

The expenditure and uncompensated price elasticities are given by

$$
\begin{aligned}
& E^{g}=\frac{\beta_{g} m}{p_{g} c_{g}+\beta_{g}\left(m-\sum_{h} p_{h} c_{h}\right)} \\
& E_{h}^{g}= \begin{cases}\frac{p_{g} c_{g}\left(1-\beta_{g}\right)}{p_{g} c_{g}+\beta_{g}\left(m-\sum_{h} p_{h} c_{h}\right)}-1 & g=h \\
\frac{-\beta_{g} p_{h} c_{h}}{p_{g} c_{g}+\beta_{g}\left(m-\sum_{h} p_{h} c_{h}\right)} & g \neq h\end{cases}
\end{aligned}
$$

The Cobb-Douglas demand system results if we set $\boldsymbol{\mu}=\mathbf{0}$. Models without demographics result if we set $\mathbf{N}=\mathbf{0}$.

## Generalized translog

Continuing to use $c_{g}=\mu_{g}+\nu_{g} \mathbf{d}$, we can obtain the generalized translog expenditure-share equations with demographics by first obtaining the regular translog expenditure-share equations and then applying translation with both a constant term and demographic characteristics to those expenditure-share equations to obtain

$$
\begin{aligned}
w_{g}(\mathbf{p}, m, \mathbf{d} ; \boldsymbol{\alpha}, \boldsymbol{\Gamma}, \boldsymbol{\mu}, \mathbf{N}) & =\frac{p_{g} c_{g}}{m}+\frac{\bar{m}}{m} \frac{\alpha_{g}+\sum_{h} \gamma_{g h} \ln \left(\frac{p_{h}}{\bar{m}}\right)}{1+\sum_{h} \sum_{j} \gamma_{h j} \ln \left(\frac{p_{j}}{\bar{m}}\right)} \\
\bar{m} & =m-\sum_{j} p_{j} c_{j}
\end{aligned}
$$

The normalization $\sum_{g} \alpha_{g}=1$ ensures the expenditure shares sum to 1 , and Slutsky symmetry requires that $\gamma_{j h}=\gamma_{h j}$. Homogeneity is implied by the functional form of the expenditure-share equation.

Noting that the translated indirect utility function has the same form as the untranslated version except with $\bar{m}$ replacing $m$, we have the indirect utility function

$$
\ln v(\mathbf{p}, m, \mathbf{d} ; \boldsymbol{\alpha}, \boldsymbol{\Gamma}, \boldsymbol{\mu}, \mathbf{N})=-\sum_{g} \alpha_{g} \ln \left(\frac{p_{g}}{\bar{m}}\right)-\frac{1}{2} \sum_{g} \sum_{h} \gamma_{g h} \ln \left(\frac{p_{g}}{\bar{m}}\right) \ln \left(\frac{p_{h}}{\bar{m}}\right)
$$

A limitation of the generalized translog model is that we cannot solve the indirect utility function for a closed-form expression for the cost function, nor does a closed-form expression for the direct utility function exist. (The direct utility function described in Christensen, Jorgenson, and Lau [1975] is not the utility function that is implied by their indirect utility function.)

Demand systems with translated demographics and committed quantities tend to produce elasticity formulas that are rather long. Omitting function arguments for clarity, we begin by writing the expenditure-share equation for good $g$ as

$$
w_{g}=A_{g}+B_{g} \times \frac{N_{g}}{D_{g}}
$$

where

$$
\begin{aligned}
A_{g} & =\frac{p_{g} c_{g}}{m} \\
B_{g} & =\frac{\bar{m}}{m}=1-\frac{\sum_{h} p_{h} c_{h}}{m} \\
N_{g} & =\alpha_{g}+\sum_{h} \gamma_{g h} \ln p_{h}-\ln \left(m-\sum_{h} p_{h} c_{h}\right) \sum_{h} \gamma_{g h} \\
D_{g} & =1+\sum_{j} \sum_{h} \gamma_{j h} \ln p_{h}-\ln \left(m-\sum_{h} p_{h} c_{h}\right) \sum_{j} \sum_{h} \gamma_{j h}
\end{aligned}
$$

Applying the chain and quotient rules of elementary calculus, we have

$$
\frac{\partial w_{g}}{\partial m}=\frac{\partial A_{g}}{\partial m}+B_{g}\left(\frac{D_{g} \frac{\partial N_{g}}{\partial m}-N_{g} \frac{\partial D_{g}}{\partial m}}{D_{g}^{2}}\right)+\frac{N_{g}}{D_{g}} \frac{\partial B_{g}}{\partial m}
$$

where

$$
\begin{aligned}
\frac{\partial A_{g}}{\partial m} & =-\frac{p_{g} c_{g}}{m^{2}} & \frac{\partial B_{g}}{\partial m} & =\frac{\sum_{h} p_{h} c_{h}}{m^{2}} \\
\frac{\partial N_{g}}{\partial m} & =-\frac{1}{m-\sum_{h} p_{h} c_{h}} \sum_{h} \gamma_{g h} & \frac{\partial D_{g}}{\partial m} & =-\frac{1}{m-\sum_{h} p_{h} c_{h}} \sum_{j} \sum_{h} \gamma_{j h}
\end{aligned}
$$

Given $\partial w_{g} / \partial m$, the expenditure elasticity of good $g$ is calculated as $E^{g}=\left(m / \widehat{w}_{g}\right)\left(\partial w_{g} / \partial m\right)+1$. We also have

$$
\frac{\partial w_{g}}{\partial p_{h}}=\frac{\partial A_{g}}{\partial p_{h}}+B_{g}\left(\frac{D_{g} \frac{\partial N_{g}}{\partial p_{h}}-N_{g} \frac{\partial D_{g}}{\partial p_{h}}}{D_{g}^{2}}\right)+\frac{N_{g}}{D_{g}} \frac{\partial B_{g}}{\partial p_{h}}
$$

where

$$
\begin{aligned}
& \frac{\partial A_{g}}{\partial p_{h}}= \begin{cases}c_{g} / m & h=g \\
0 & h \neq g\end{cases} \\
& \frac{\partial B_{g}}{\partial p_{h}}=-\frac{c_{g}}{m} \\
& \frac{\partial N_{g}}{\partial p_{h}}=\frac{\gamma_{g h}}{c_{h}}+\frac{c_{h}}{m-\sum_{j} p_{j} c_{j}} \sum_{j} \gamma_{g j} \\
& \frac{\partial D_{g}}{\partial p_{h}}=\frac{1}{p_{h}} \sum_{j} \gamma_{j h}+\frac{c_{h}}{m-\sum_{j} p_{j} c_{j}} \sum_{j k} \gamma_{j k}
\end{aligned}
$$

Given $\partial w_{g} / \partial p_{h}$, the uncompensated price elasticity of good $g$ with respect to price $h$ is calculated as $E_{h}^{g}=\left(p_{h} / w_{g}\right)\left(\partial w_{g} / \partial p_{h}\right)+\mathbb{I}_{h=g}$, where $\mathbb{I}_{x}$ is the indicator function that takes on the value one if $x$ is true and zero otherwise.

The basic translog results if we set $\boldsymbol{\mu}=\mathbf{0}$, and models without demographics have $\mathbf{N}=\mathbf{0}$.

## QUAIDS with demographic translation

Translating Banks, Blundell, and Lewbel's (1997) indirect utility function, we have

$$
\begin{aligned}
\ln V\left(\mathbf{p}, m, \mathbf{d} ; \alpha_{0}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\Gamma}, \boldsymbol{\lambda}, \boldsymbol{\mu}, \mathbf{N}\right) & =\left[\left\{\frac{\ln \bar{m}-\ln a(\mathbf{p})}{b(\mathbf{p})}\right\}^{-1}+\lambda(\mathbf{p})\right]^{-1} \\
\ln (\mathbf{p}) & =\alpha_{0}+\sum_{g} \alpha_{g} \ln p_{g}+\frac{1}{2} \sum_{g} \sum_{h} \gamma_{g h} \ln p_{g} \ln p_{h} \\
b(\mathbf{p}) & =\prod_{g} p_{g}^{\beta_{g}} \\
\lambda(\mathbf{p}) & =\sum_{g} \lambda_{g} \ln p_{g}
\end{aligned}
$$

where $\alpha_{0}$ is the value specified in the piconstant() option. Solving for $\bar{m}$ and recalling that $m=\bar{m}+\sum_{g} p_{g} c_{g}$ for $c_{g}=\mu_{g}+\nu_{g} d$, we have the expenditure function

$$
\begin{equation*}
e\left(\mathbf{p}, u, \mathbf{d} ; \alpha_{0}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\Gamma}, \boldsymbol{\lambda}, \boldsymbol{\mu}, \mathbf{N}\right)=\exp \left\{\frac{b(\mathbf{p}) \ln u}{1-\lambda(\mathbf{p}) \ln u}+\ln a(\mathbf{p})\right\}+\sum_{g} p_{g} c_{g} \tag{6}
\end{equation*}
$$

Applying Shepherd's lemma yields the expenditure-share equations

$$
\begin{align*}
& w_{g}\left(\mathbf{p}, m, \mathbf{d} ; \alpha_{0}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\Gamma}, \boldsymbol{\lambda}, \boldsymbol{\mu}, \mathbf{N}\right)= \\
& \frac{p_{g} c_{g}}{m}+\frac{\bar{m}}{m}\left(\alpha_{g}+\sum_{h} \gamma_{g h} \ln p_{h}+\beta_{g} \ln \left\{\frac{\bar{m}}{a(\mathbf{p})}\right\}+\frac{\lambda_{g}}{b(\mathbf{p})}\left[\ln \left\{\frac{\bar{m}}{a(\mathbf{p})}\right\}\right]^{2}\right) \tag{7}
\end{align*}
$$

To obtain the expenditure elasticities, we first write (7) as

$$
\begin{equation*}
w_{g}=R_{g}+S_{g} \times T_{g} \tag{8}
\end{equation*}
$$

where

$$
\begin{aligned}
& R_{g}=\frac{p_{g} c_{g}}{m} \quad S_{g}=\frac{\bar{m}}{m} \\
& T_{g}=\alpha_{g}+\sum_{h} \gamma_{g h} \ln p_{h}+\beta_{g} \ln \left\{\frac{\bar{m}}{a(\mathbf{p})}\right\}+\frac{\lambda_{g}}{b(\mathbf{p})}\left[\ln \left\{\frac{\bar{m}}{a(\mathbf{p})}\right\}\right]^{2}
\end{aligned}
$$

Taking the derivative with respect to $\ln m$, we have

$$
\frac{\partial w_{g}}{\partial \ln m}=\frac{\partial R_{g}}{\partial \ln m}+S_{g} \frac{\partial T_{g}}{\partial \ln m}+T_{g} \frac{\partial S_{g}}{\partial \ln m}
$$

where

$$
\begin{aligned}
\frac{\partial R_{g}}{\partial \ln m} & =-\frac{p_{g} c_{g}}{m} \quad \frac{\partial S_{g}}{\partial \ln m}=1-S_{g} \\
\frac{\partial T_{g}}{\partial \ln m} & =\frac{1}{S_{g}}\left[\beta_{g}+2 \frac{\lambda_{g}}{b(\mathbf{p})} \ln \left\{\frac{\bar{m}}{a(\mathbf{p})}\right\}\right]
\end{aligned}
$$

Then the expenditure elasticity for good $g$ is given by $E^{g}=\left(1 / w_{g}\right)\left(\partial w_{g} / \partial \ln m\right)+1$.

To obtain the uncompensated price elasticity of good $g$ with respect to price $h$, we again use (8). Taking the partial derivative with respect to $\ln p_{h}$, we have

$$
\frac{\partial w_{g}}{\partial \ln p_{h}}=\frac{\partial R_{g}}{\partial \ln p_{h}}+S_{g} \frac{\partial T_{g}}{\partial \ln p_{h}}+T_{g} \frac{\partial S_{g}}{\partial \ln p_{h}}
$$

The required partials are

$$
\begin{aligned}
\frac{\partial R_{g}}{\partial \ln p_{h}} & = \begin{cases}\frac{p_{h} c_{h}}{m} & h=g \\
0 & h \neq g\end{cases} \\
\frac{\partial S_{g}}{\partial \ln p_{h}} & =-\frac{p_{h} c_{h}}{m} \\
\frac{\partial T_{g}}{\partial \ln p_{h}} & =\gamma_{g h}+\beta_{g} \frac{\partial}{\partial \ln p_{h}}\left[\ln \left\{\frac{\bar{m}}{a(\mathbf{p})}\right\}\right]+\frac{\partial}{\partial \ln p_{h}}\left(\frac{\lambda_{g}}{b(\mathbf{p})}\left[\ln \left\{\frac{\bar{m}}{a(\mathbf{p})}\right\}\right]^{2}\right)
\end{aligned}
$$

where

$$
\begin{aligned}
\frac{\partial}{\partial \ln p_{h}}\left[\ln \left\{\frac{\bar{m}}{a(\mathbf{p})}\right\}\right]= & -\left(\frac{p_{h} c_{h}}{m-\sum_{j} p_{j} c_{j}}+\alpha_{h}+\sum_{j} \gamma_{h j} \ln p_{j}\right) \\
\frac{\partial}{\partial \ln p_{h}}\left(\frac{\lambda_{g}}{b(\mathbf{p})}\left[\ln \left\{\frac{\bar{m}}{a(\mathbf{p})}\right\}\right]^{2}\right)= & 2 \frac{\lambda_{g}}{b(\mathbf{p})} \ln \left\{\frac{\bar{m}}{a(\mathbf{p})}\right\} \frac{\partial}{\partial \ln p_{h}}\left[\ln \left\{\frac{\bar{m}}{a(\mathbf{p})}\right\}\right] \\
& -\frac{\lambda_{g} \beta_{h}}{b(\mathbf{p})}\left[\ln \left\{\frac{\bar{m}}{a(\mathbf{p})}\right\}^{2}\right]
\end{aligned}
$$

The uncompensated price elasticity of good $g$ with respect to price $h$ is $E_{h}^{g}=\left(p_{h} / w_{g}\right)\left(\partial w_{g} / \partial p_{h}\right)+$ $\mathbb{I}_{h=g}$.

The AIDS model results if $\boldsymbol{\lambda}=\mathbf{0}$. The nongeneralized variants result if $\boldsymbol{\mu}=\mathbf{0}$. The variants without demographics result if $\mathbf{N}=\mathbf{0}$.

## QUAIDS with demographic scaling

As can be gleaned from (6), the expenditure function for a QUAIDS model without any form of translation or scaling is

$$
\begin{aligned}
e\left(\mathbf{p}, u ; \alpha_{0}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\Gamma}, \boldsymbol{\lambda}\right) & =\exp \left\{\frac{b(\mathbf{p}) \ln u}{1-\lambda(\mathbf{p}) \ln u}+\ln a(\mathbf{p})\right\}=\exp \left\{\frac{b(\mathbf{p})}{\frac{1}{\ln u}-\lambda(\mathbf{p})}+\ln a(\mathbf{p})\right\} \\
a(\mathbf{p}) & =\alpha_{0}+\sum_{j} \alpha_{j} \ln p_{j}+\frac{1}{2} \sum_{j} \sum_{k} \gamma_{j k} \ln p_{j} \ln p_{k} \\
b(\mathbf{p}) & =\prod_{j} p_{j}^{\beta_{j}} \\
\lambda(\mathbf{p}) & =\sum_{j} \lambda_{j} \ln p_{j}
\end{aligned}
$$

The AIDS case results when $\boldsymbol{\lambda}=\mathbf{0}$, and we obtain the expenditure function given by (2).

To implement Ray's (1983) demographic scaling as extended to the QUAIDS case in Poi (2002), we require a function that can be split into two parts, one part that depends on prices and utility and one part that does not:

$$
m_{0}(\mathbf{p}, u, \mathbf{d})=\bar{m}_{0}(\mathbf{d}) \phi(\mathbf{p}, \mathbf{d}, u)
$$

For $\bar{m}_{0}(\mathbf{d})$, we use the same function as in Ray,

$$
\bar{m}_{0}(\mathbf{d} ; \boldsymbol{\rho})=1+\boldsymbol{\rho}^{\prime} \mathbf{d}
$$

where $\boldsymbol{\rho}$ is a $D \times 1$ parameter vector. For $\phi(\mathbf{p}, \boldsymbol{\delta}, u)$, we follow Poi (2002) and use

$$
\phi(\mathbf{p}, \mathbf{d}, u ; \boldsymbol{\beta}, \boldsymbol{\lambda}, \mathbf{H})=\exp \left\{\frac{b(\mathbf{p})\left(\prod_{j} p_{j}^{\boldsymbol{\eta}_{j}{ }^{\mathbf{d}}}-1\right)}{\frac{1}{\ln u}-\lambda(\mathbf{p})}\right\}
$$

where $\boldsymbol{\eta}_{j}$ is the $j$ th row of $G \times D$ parameter matrix $\mathbf{H}$. For AIDS when $\boldsymbol{\lambda}=0$, we are left with the function used by Ray (1983). The expenditure function for the QUAIDS model with demographic scaling is then

$$
e\left(\mathbf{p}, u, \mathbf{d} ; \alpha_{0}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\Gamma}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{H}\right)=e\left(\mathbf{p}, u ; \alpha_{0}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\Gamma}, \boldsymbol{\lambda}\right) \times \bar{m}_{0}(\mathbf{d} ; \boldsymbol{\rho}) \times \phi(\mathbf{p}, \mathbf{d}, u ; \boldsymbol{\beta}, \mathbf{H})
$$

Applying Shepherd's lemma, we have the share equations

$$
\begin{aligned}
& w_{g}\left(\mathbf{p}, m, \mathbf{d} ; \alpha_{0}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\Gamma}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{H}\right)= \\
& \alpha_{g}+\sum_{h} \gamma_{g h} \ln p_{h}+\left(\beta_{g}+\boldsymbol{\eta}_{g} \mathbf{d}\right) \ln \left[\frac{m}{\bar{m}_{0}(\mathbf{d} ; \boldsymbol{\rho}) a(\mathbf{p})}\right]+\frac{\lambda_{g}}{b(\mathbf{p}) c(\mathbf{p}, \mathbf{d})}\left[\ln \left\{\frac{m}{\bar{m}_{0}(\mathbf{d} ; \boldsymbol{\rho}) a(\mathbf{p})}\right\}\right]^{2}
\end{aligned}
$$

where the demographic-adjusted price aggregator is defined as $c(\mathbf{p}, \mathbf{d}) \equiv \prod_{j} p_{j}^{\boldsymbol{\eta}_{j} \mathbf{d}}$. The indirect utility function is

$$
\ln V\left(\mathbf{p}, m, \mathbf{d} ; \alpha_{0}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \mathbf{H}\right)=\left[\frac{b(\mathbf{p}) c(\mathbf{p}, \mathbf{d})}{\ln \left\{m-\bar{m}_{0}(\mathbf{d} ; \boldsymbol{\rho})-a(\mathbf{p})\right\}}+\lambda(\mathbf{p})\right]^{-1}
$$

Because we are not dealing with demographic translation here, the partial derivatives required for elasticities are not as involved as they are for the QUAIDS model with demographic translation or the generalized translog model. We have

$$
\frac{\partial w_{g}}{\partial \ln m}=\beta_{g}+\boldsymbol{\eta}_{g} \mathbf{d}+\frac{2 \lambda_{g}}{b(\mathbf{p}) c(\mathbf{p}, \mathbf{d})} \ln \left\{\frac{m}{\left(1+\boldsymbol{\rho}^{\prime} \mathbf{d}\right) a(\mathbf{p})}\right\}
$$

and

$$
\begin{aligned}
\frac{\partial w_{g}}{\partial \ln p_{h}}=\gamma_{g h} & -\left[\beta_{g}+\boldsymbol{\eta}_{g} \mathbf{d}+\frac{2 \lambda_{g}}{b(\mathbf{p}) c(\mathbf{p}, \mathbf{d})} \ln \left\{\frac{m}{\left(1+\boldsymbol{\rho}^{\prime} \mathbf{d}\right) a(\mathbf{p})}\right\}\right] \times\left(\alpha_{h}+\sum_{k} \gamma_{h k} \ln p_{k}\right) \\
& +\frac{\left(\beta_{h}+\boldsymbol{\eta}_{h} \mathbf{d}\right) \lambda_{g}}{b(\mathbf{p}) c(\mathbf{p}, \mathbf{d})}\left[\ln \left\{\frac{m}{\left(1+\boldsymbol{\rho}^{\prime} \mathbf{d}\right) a(\mathbf{p})}\right\}\right]^{2}
\end{aligned}
$$

The AIDS model with demographic scaling results if $\boldsymbol{\lambda}=0$, and models without demographic scaling result if $\boldsymbol{\rho}=\mathbf{0}$ and $\mathbf{H}=\mathbf{0}$.

## Estimation

The expenditure shares of a demand system represent a set of nonlinear seemingly unrelated regression (SUR) equations. Hence, estimation is performed using nlsur with just one complication. Our system of equations for observation $i$ and generic parameter vector $\boldsymbol{\theta}$ can be written as

$$
\begin{aligned}
w_{1 i} & =w_{1}\left(\mathbf{p}_{i}, m_{i}, \mathbf{d}_{i} ; \boldsymbol{\theta}\right)+\epsilon_{1 i} \\
w_{2 i} & =w_{2}\left(\mathbf{p}_{i}, m_{i}, \mathbf{d}_{i} ; \boldsymbol{\theta}\right)+\epsilon_{2 i} \\
\vdots & =\quad \vdots \\
w_{G i} & =w_{G}\left(\mathbf{p}_{i}, m_{i}, \mathbf{d}_{i} ; \boldsymbol{\theta}\right)+\epsilon_{G i}
\end{aligned}
$$

where $\boldsymbol{\epsilon}_{i}=\left(\epsilon_{1 i}, \epsilon_{2 i}, \ldots, \epsilon_{G i}\right)$ is a vector of zero-mean disturbances. Because $\sum_{g} w_{g i}=1$ and by the construction of our demand systems we also have $\sum_{g} w_{g}\left(\mathbf{p}_{i}, m_{i}, \mathbf{d}_{i} ; \boldsymbol{\theta}\right)=1$, it must be the case that $\sum_{g} \epsilon_{g i}=0$. Therefore, $\operatorname{det}(\boldsymbol{\Sigma})=\operatorname{det}\left\{E\left(\boldsymbol{\epsilon} \boldsymbol{\epsilon}^{\prime}\right)\right\}=\mathbf{0}$, and neither the (quasi)maximum likelihood estimator nor the feasible generalized nonlinear SUR estimator is defined.

Barten (1969) showed that to obtain a well-defined likelihood function, we can drop any one of the expenditure-share equations and fit the system containing the remaining $G-1$ equations. Parameter restrictions that force the demand system's equations to sum to 1 can then be used to recover the parameters of the dropped equation. demandsys drops the final equation from the demand system, but which equation is dropped should not concern you: Barten (1969) showed that you obtain the same likelihood function regardless of which one is dropped.

Moreover, the nonlinear SUR model satisfies the so-called Oberhofer-Kmenta (1974) conditions that allow us to cycle between estimating the parameters $\boldsymbol{\theta}$ and the error covariance matrix $\boldsymbol{\Sigma}$ as nlsur does when we specify option ifgnls. The upshot is that we fit our expenditure-share equations by calling nlsur with option ifgnls. The resulting parameter estimate $\widehat{\boldsymbol{\theta}}$ is the (quasi)maximum likelihood estimate of the parameters of our demand system. The $(G-1) \times(G-1)$ estimated error covariance matrix is returned in matrix e(Sigma).

Estimation weights, variance-covariance matrix options, and options to control the optimization process available with demandsys are simply passed to nlsur. See Methods and formulas in [R] nlsur.

When you type predict ..., shares eq(\#), you obtain the predicted shares for the \#th equation, based on the expenditure-share equation for the demand system estimated. By construction, the sum of predicted shares for each observation is equal to one.

When you type predict ..., residuals eq(\#), you obtain the residuals for the \#th equation defined as $r_{\# i}=w_{\# i}-\widehat{w_{\# i}}$, where $\widehat{w_{\# i}}$ is the predicted share for equation \# for observation $i$.

When you type predict..., quantities eq(\#), predict first computes the predicted shares $\widehat{w_{\# i}}$ and then computes $\widehat{q_{\# i}}=m_{i} \widehat{w_{\# i}} / p_{\# i}$.

When you type predict ..., iuf, predict evaluates the indirect utility function based on the formulas given above.

When you type predict ..., ef utilities $(u)$, predict evaluates the expenditure function based on the formulas given above at the levels of utility specified in variable $u$.

## Acknowledgment

We thank Brian Poi of Poi Consulting LLC, East Fallowfield, PA, for writing the popular communitycontributed quaids command, which motivated us to implement demandsys. We also thank Dr. Poi for his advice on and review of the demandsys command.

## References

Banks, J., R. W. Blundell, and A. Lewbel. 1997. Quadratic Engel curves and consumer demand. Review of Economics and Statistics 79: 527-539. https://doi.org/10.1162/003465397557015.

Barnett, W. A., and A. Serletis. 2008. Consumer preferences and demand systems. Journal of Econometrics 147: 210-224. https://doi.org/10.1016/j.jeconom.2008.09.009.
Barten, A. P. 1966. Theorie en empirie van een volledig stelsel van vraagvergelijkingen. PhD thesis, University of Rotterdam.
-_ 1969. Maximum likelihood estimation of a complete system of demand equations. European Economic Review 1: 7-73. https://doi.org/10.1016/0014-2921(69)90017-8.

Caro, J. C., G. Melo, J. A. Molina, and J. C. Salgado. 2021. Censored demand system estimation with quaidsce. Presented at the Stata Conference 2021, August 6, 2021, virtual. https://www.stata.com/meeting/us21/slides/US21_Melo.pdf.
Case, A. C. 1991. Spatial patterns in household demand. Econometrica 59: 953-965. https://doi.org/10.2307/2938168.
Christensen, L. R., D. W. Jorgenson, and L. J. Lau. 1975. Transcendental logarithmic utility functions. American Economic Review 65: 367-383. https://doi.org/10.2307/1927992.
Deaton, A. S., and M. Irish. 1984. Statistical models for zero expenditures in household budgets. Journal of Public Economics 23: 59-80. https://doi.org/10.1016/0047-2727(84)90067-7.
Deaton, A. S., and J. Muellbauer. 1980a. Economics and Consumer Behavior. Cambridge: Cambridge University Press.
——. 1980b. An almost ideal demand system. American Economic Review 70: 312-326.
Fisher, D., A. R. Fleissig, and A. Serletis. 2001. An empirical comparison of flexible demand system functional forms. Journal of Applied Econometrics 16: 59-80. https://doi.org/10.1002/jae.585.

Gorman, W. M. 1981. Some Engle curves. In Essays in the Theory and Measurement of Consumer Behavior: In Honour of Sir Richard Stone, ed. A. Deaton, 7-30. Cambridge: Cambridge University Press. https://doi.org/10.1017/CBO9780511984082.003.
Heien, D., and C. R. Wessells. 1990. Demand systems estimation with microdata: A censored regression approach. Journal of Business and Economic Statistics 8: 365-371. https://doi.org/10.2307/1391973.

Holt, M. T., and B. K. Goodwin. 2009. The almost ideal and translog demand systems. MPRA Paper 15092, University Library, Ludwig Maximilians University, München, Germany. https://mpra.ub.uni-muenchen.de/15092/.
Lecocq, S., and J.-M. Robin. 2015. Estimating almost-ideal demand systems with endogenous regressors. Stata Journal 15: 554-573.

Leser, C. E. V. 1963. Forms of Engel functions. Econometrica 31: 694-703. https://doi.org/10.2307/1909167.
Lewbel, A. 1989. Household equivalence scales and welfare comparisons. Journal of Public Economics 39: 377-391. https://doi.org/10.1016/0047-2727(89)90035-2.
-_. 1991. The rank of demand systems: Theory and nonparametric estimation. Econometrica 59: 711-730. https://doi.org/10.2307/2938225.
——. 2001. Demand systems with and without errors. American Economic Review 91: 611-618. https://doi.org/10.1257/aer.91.3.611.
Lewbel, A., and K. Pendakur. 2009. Tricks with Hicks: The EASI demand system. American Economic Review 99: 827-863. https://doi.org/10.1257/aer.99.3.827.
Mas-Colell, A., M. D. Whinston, and J. R. Green. 1995. Microeconomic Theory. New York: Oxford University Press.
Meyerhoefer, C. D., C. K. Ranney, and D. E. Sahn. 2005. Consistent estimation of censored demand systems using panel data. American Journal of Agricultural Economics 87: 660-672. https://doi.org/10.1111/j.1467-8276.2005.00754.x.
Muellbauer, J. 1975. Aggregation, income distribution, and consumer demand. Review of Economic Studies 42: 525-543. https://doi.org/10.2307/2296792.
Nelson, D. B. 1991. Conditional heteroskedasticity in asset returns: A new approach. Econometrica 59: 347-370. https://doi.org/10.2307/2938260.

Oberhofer, W., and J. Kmenta. 1974. A general procedure for obtaining maximum likelihood estimates in generalized regression models. Econometrica 42: 579-590. https://doi.org/10.2307/1911792.
Poi, B. P. 2002. From the help desk: Demand system estimation. Stata Journal 2: 403-410.
2012. Easy demand-system estimation with quaids. Stata Journal 12: 433-446.

Pollak, R. A., and T. J. Wales. 1978. Estimation of complete demand systems from household budget data: The linear and quadratic expenditure systems. American Economic Review 68: 348-359.
——. 1992. Demand System Specification and Estimation. New York: Oxford University Press.
Ray, R. 1983. Measuring the costs of children: An alternative approach. Journal of Public Economics 22: 89-102. https://doi.org/10.1016/0047-2727(83)90058-0.

Roy, P. R. 1943. La hiérarchie des besoins et la notion de groupes dans l'économie de choix. Econometrica 11: 13-24. https://doi.org/10.2307/1905715.
Shephard, R. W. 1970. Theory of Cost and Production Functions. Princeton, NJ: Princeton University Press.
Shonkwiler, J. S., and S. T. Yen. 1999. Two-step estimation of a censored system of equations. American Journal of Agricultural Economics 81: 972-982. https://doi.org/10.2307/1244339.
Stone, R. 1954. Linear expenditure systems and demand analysis: An application to the pattern of British demand. Economic Journal 64: 511-527. https://doi.org/10.2307/2227743.

Tauchmann, H. 2010. Consistency of Heckman-type two-step estimators for the multivariate sample-selection model. Applied Economics 42: 3895-3902. https://doi.org/10.1080/00036840802360179.
Theil, H. 1965. The information approach to demand analysis. Econometrica 33: 67-87. https://doi.org/10.2307/1911889.
Varian, H. R. 1992. Microeconomic Analysis. 3rd ed. New York: W. W. Norton.
Vermeulen, F. 2001. A note on Heckman-type corrections in models for zero expenditures. Applied Economics 33: 1089-1092. https://doi.org/10.1080/00036840010004004.
Working, H. 1943. Statistical laws of family expenditure. Journal of the American Statistical Association 38: 43-56. https://doi.org/10.2307/2279311.

Yen, S. T., and B.-H. Lin. 2006. A sample selection approach to censored demand systems. American Journal of Agricultural Economics 88: 742-749. https://doi.org/10.1111/j.1467-8276.2006.00892.x.
Young, W. H. 1909. On the conditions for the reversibility of the order of partial differentiation. Proceedings of the Royal Society of Edinburgh, Series B: Biological Sciences 29: 136-164. https://doi.org/10.1017/S0370164600008865.

## Also see

[R] demandsys postestimation - Postestimation tools for demandsys
[R] nlsur - Estimation of nonlinear systems of equations
[U] 20 Estimation and postestimation commands

## Title

demandsys postestimation - Postestimation tools for demandsys

| Postestimation commands | predict | margins | estat |
| :--- | :--- | :--- | :--- |
| Remarks and examples | References | Also see |  |

## Postestimation commands

The following postestimation commands are of special interest after demandsys:

| Command | Description |
| :--- | :--- |
| estat elasticities | price and expenditure elasticities |
| estat parameters | estimated parameter vectors and matrices |

The following standard postestimation commands are also available:

| Command | Description |
| :--- | :--- |
| estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian <br> information criteria (AIC, CAIC, AICc, and BIC) |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) <br> cataloging estimation results |
| estimates | point estimates, standard errors, testing, and inference for linear combinations <br> of coefficients |
| lincom | likelihood-ratio test <br> marginal means, predictive margins, marginal effects, and average marginal <br> effects |
| margins | graph the results from margins (profile plots, interaction plots, etc.) <br> point estimates, standard errors, testing, and inference for nonlinear combi- <br> nations of coefficients |
| marginsplot | predicted shares, residuals, quantities, and functions <br> point estimates, standard errors, testing, and inference for generalized pre- <br> nlcomict |
| predictnl | Wald tests of simple and composite linear hypotheses <br> Wald tests of nonlinear hypotheses |
| test |  |

## predict

## Description for predict

predict creates a new variable containing statistics such as predicted shares and residuals based on the demandsys estimation results currently active.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

```
predict [type] newvar [if] [in] [, statistic options]
```

statistic Description

Main

## shares

quantities
residuals
iuf
ef
options
Main
equation(\#eqno)
prices(varlist ${ }_{p}$ )
expenditures(varname)
utilities(varname)
demographics(varlist)
predicted shares; the default
predicted quantities
residuals
indirect utility function
expenditure function

Description
display only specified goods; available only with shares, quantities, and residuals
specify set of counterfactual price variables
specify a counterfactual total expenditure variable
specify the variable containing the utilities at which to compute the expenditure function; available only with and required with ef
specify a set of counterfactual demographic variables

## Options for predict

$\qquad$ Main
shares, the default, calculates the predicted shares.
quantities calculates the predicted quantities given actual prices, actual expenditures, and predicted shares.
residuals calculates the residuals.
iuf calculates the indirect utility function $v(\mathbf{p}, m)$.
ef calculates the expenditure function $e(\mathbf{p}, u)$. You must also specify option utilities() with this option. This option is not available after translog and generalized translog models.
equation(\#eqno) specifies to which good you are referring. equation(\#1) would mean the calculation is to be made for the first good, equation(\#2) would mean the second, and so on. If you do not specify equation(), results are the same as if you specified equation(\#1).
prices ( varlist $_{p}$ ) specifies a set of counterfactual price variables at which to compute the requested statistic. You must specify $G$ variables, where $G$ is the number of goods in the demand system. By default, predict uses the same price variables that were used to fit the demand system. This option aids in the computation of welfare and other policy measures by allowing you to contrast predicted values at different price levels.
expenditures (varname) specifies a counterfactual total expenditure variable at which to compute the requested statistic. By default, predict uses the same expenditure variable that was used to fit the demand system. This option aids in the computation of welfare and other policy measures by allowing you to contrast predicted values at different expenditure levels.
utilities (varname) specifies the variable containing the utilities at which the expenditure function is to be computed. Almost surely, varname is a variable that was created by calling predict ..., iuf .... This option aids in the computation of welfare and other policy measures by allowing you to contrast predicted values at different levels of utility.
demographics (varlist) specifies a set of counterfactual demographic variables at which to compute the requested statistic. If you specified $D$ demographic variables at estimation, then you must specify precisely $D$ variables here. By default, predict uses the same demographic variables (if any) that were used to fit the demand system.

## margins

## Description for margins

margins estimates margins for the predicted share of the first good.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

margins [, options]
margins, predict(statistic ...) [options]
statistic Description
shares
quantities
residuals
iuf indirect utility function
*ef expenditure function
*You must specify the predict option utilities() with statistic ef.
Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

## estat

## Description for estat

estat elasticities computes expenditure and price elasticities after fitting a demand system using demandsys. Elasticities are calculated at the price and expenditure levels in the data. However, you may compute elasticities at the means of prices and expenditures or use other counterfactual prices or expenditures.
estat parameters returns the estimated parameter vectors and matrices based on the currently active demandsys estimation results. Results are stored in $r$ () and vary depending on the demand model fitted. See [U] 18.8 Accessing results calculated by other programs.

## Menu for estat

Statistics $>$ Postestimation

## Syntax for estat

Compute price and expenditure elasticities

```
estat elasticities [if] [in], { expenditure|\underline{compensated|uncompensated }}
    [[atmeans|at(atspec)] generate(stub)]
```

Report estimated parameter vectors and matrices
estat parameters

## Options for estat elasticities

$\qquad$ Main
expenditure, compensated, and uncompensated indicate the type of elasticities to be reported. You must specify one of these.
expenditure requests that the $G$ expenditure elasticities be calculated, where $G$ is the number of goods in the system.
compensated requests that compensated price elasticities be calculated. There is a total of $G^{2}$ such elasticities because there are $G$ goods, each of whose quantity consumed can be affected by any one of the $G$ prices changing.
uncompensated requests that the $G^{2}$ uncompensated price elasticities be calculated.
atmeans requests that elasticities be calculated at the means of the price, expenditure, and demographic
variables. By default, elasticities are computed for each observation and then averaged.
at (atspec) specifies the prices, expenditures, and demographic variables' values at which elasticities are to be calculated. If you do not specify this option, elasticities are calculated at the observed values unless you specify option atmeans, in which case they are calculated at the means of the variables. See example 1 below. You may not specify option at () if you request compensated price elasticities.
generate (stub) requests that new variables be generated holding the observation-level elasticities. For expenditure elasticities, $G$ new variables $s t u b \_1$, stub_2, ..., stub_ $G$ will be created, where $G$ is the number of goods. For price elasticities, $G^{2}$ new variables will be created of the form stub_g_h for $g=1, \ldots, G$ and $h=1, \ldots, G$. Variable stub_ $g \_h$ will contain the percentage change in the quantity of good $h$ with respect to the price of good $g$.

## Remarks and examples

Remarks are presented under the following headings:

Introduction<br>Elasticities<br>Evaluating elasticities<br>Compensating and equivalent variation

## Introduction

estat elasticities allows you to compute expenditure and price elasticities after you fit a demand model and is much more flexible than the options available with demandsys to display elasticities in lieu of coefficient estimates. With estat elasticities, you can calculate elasticities at the same prices, expenditures, and demographic variables' levels as was used at estimation time, or you can specify your own values. For example, you could calculate price elasticities for every observation in your dataset assuming each household had two children, even if the number of children varies by household. Example 1 below shows you how that can be done.
predict allows you to obtain values of the indirect utility function and of the expenditure function underlying the demand system you fit. Example 3 leverages that ability and shows you how to compute compensated and equivalent variation in response to a price change.
estat parameters allows you to obtain the estimated parameter vectors and matrices for the demand system based on the coefficient vector stored in e(b). This may be useful if you wish to conduct further policy experiments or calculate your own welfare measures. For instance, in example 2 below, we show how estat parameters with just a bit of coding can be used to check whether a fitted generalized translog model can be interpreted as having subsistence quantities for each good.

## Elasticities

Although one can have demandsys report expenditure or price elasticities for the estimation sample, the estat elasticities command gives you much more flexibility. We illustrate features of estat elasticities in the next example. We often request expenditure elasticities to keep the output shorter, but all our commands would work regardless of the type of elasticity requested.

## > Example 1

Before we can obtain elasticities, we first need to fit a demand system to our data. Here we include the quietly prefix command to omit the results of fitting a quadratic almost ideal demand system (QUAIDS) model with demographic translation, and we use the labels() option to provide short names for each of our goods. We type

```
. use https://www.stata-press.com/data/r18/food_consumption
(Food consumption)
. quietly demandsys quaids w_dairy w_proteins w_fruitveg w_flours w_misc,
> prices(p_dairy p_proteins p_fruitveg p_flours p_misc)
> expenditures(expfd)
> demographics(n_kids n_adults)
> labels(Dairy Meats FruitVeg Flours Misc)
```

Suppose we want the average expenditure elasticity among households with two children. We type
. estat elasticities if n_kids == 2, expenditure

| Expenditure elasticities |  |  |  |  | Number of obs $=751$ |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Expenditure | Elasticity | Std. err. | z | $P>\|z\|$ | [95\% conf | interval] |
| Dairy | . 9028921 | . 0240869 | 37.48 | 0.000 | . 8556826 | . 9501016 |
| Meats | 1.087763 | . 0156852 | 69.35 | 0.000 | 1.05702 | 1.118505 |
| FruitVeg | . 9978958 | . 0215996 | 46.20 | 0.000 | . 9555614 | 1.04023 |
| Flours | . 8523383 | . 0220511 | 38.65 | 0.000 | . 8091189 | . 8955576 |
| Misc | . 9971274 | . 0265993 | 37.49 | 0.000 | . 9449938 | 1.049261 |

The header of the output reports that there are 751 households in the dataset with two children. Had we not specified the labels() option when we fit our model, estat elasticities would have simply numbered the goods from one through five.

What is displayed in the table of output is an average elasticity. Behind the scenes, estat elasticities calculated the five expenditure elasticities for each of the 751 households in this sample. What is reported as the expenditure elasticity for dairy products is the average over the 751 individual expenditure elasticities calculated. The column labeled Std. err. contains the standard error of the estimated mean. Looking to the right, we see the $95 \%$ confidence interval for the mean dairy elasticity is [0.856, 0.950]. More importantly, the column labeled Std. err . is not the sample standard deviation of the 751 individual elasticities, nor does the $95 \%$ confidence interval report something resembling the central $95 \%$ of the individual elasticities.

To save the individual elasticities for the 751 observations, you can use the generate() option like so:

```
. quietly estat elasticities if n_kids == 2, expenditure generate(ee)
```

. summarize ee*

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| ee_1 | 751 | .9028921 | .0753773 | .5095888 | 1.44884 |
| ee_2 | 751 | 1.087763 | .0664746 | .8448506 | 1.525114 |
| ee_3 | 751 | .9978958 | .1755427 | .7566877 | 5.07766 |
| ee_4 | 751 | .8523383 | .1053903 | .3864381 | 1.110392 |
| ee_5 | 751 | .9971274 | .082875 | .3257591 | 1.296199 |

The means of these variables match the point estimates in the previous estat elasticities output. The standard deviations reported here measure the spread of the individual observations' elasticities rather than the standard errors of the means as in the previous output.

A second way to compute elasticities is to first compute the means of the price variables, the expenditure variable, and any demographics and then compute the elasticity for this hypothetical household. To have estat elasticities do that, you use the atmeans option. Here we obtain the expenditure elasticities at the means for the subset of households with two children:
. estat elasticities if n_kids == 2, expenditure atmeans
Expenditure elasticities Number of obs $=751$

| Expenditure | Elasticity | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | $[95 \%$ conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | :---: | ---: |
| Dairy | .9155493 | .0232877 | 39.31 | 0.000 | .8699063 | .9611923 |
| Meats | 1.078329 | .0149179 | 72.28 | 0.000 | 1.04909 | 1.107567 |
| FruitVeg | .9804481 | .0201581 | 48.64 | 0.000 | .9409389 | 1.019957 |
| Flours | .8695964 | .0212882 | 40.85 | 0.000 | .8278722 | .9113206 |
| Misc | 1.004098 | .0246939 | 40.66 | 0.000 | .9556994 | 1.052498 |

Note: Elasticities are calculated at prices', demographic variables', and expenditure means.

With the atmeans option, we calculate the elasticities for a hypothetical household that may have, for example, 0.87 children and 1.93 adults. Without the atmeans option, we are calculating elasticities for households that do exist in the data, though taking the average of household elasticities poses its own conceptual issues.
Consider the following table:
Table 1. Average expenditure elasticities

| Consumer | $m$ | $E^{m}$ | $Q$ | $m^{\prime}$ | $Q^{\prime}$ |
| :--- | :--- | :--- | :--- | :--- | :--- |
| $A$ | 100 | 2 | 20 | 110 | 24 |
| $B$ | 200 | 1 | 30 | 220 | 33 |
| $C$ | 400 | 0.5 | 30 | 440 | 31.5 |
| Sum | 700 |  | 80 | 770 | 88.5 |
| Mean |  | 1.17 |  |  |  |

Column $m$ represents each consumer's initial total expenditure, $E^{m}$ is the consumer's expenditure elasticity for the good in question, and $Q$ is the quantity consumed. Column $m^{\prime}$ represents a $10 \%$ increase in each consumer's expenditure, and $Q^{\prime}$ represents the level of consumption based on the increases in total expenditure and the expenditure elasticities. The mean elasticity, analogous to what estat elasticities reports without the atmeans option, is 1.17 . However, based on the aggregate data, the expenditure elasticity works out to 1.06 .

So far in this example, we computed the expenditure elasticities for the subset of households with two children. We can also calculate the average elasticity for all households in our dataset under the counterfactual that all of them have two children. One way to do that is to type

```
. generate n_kids_backup = n_kids
. replace n_kids = 2
(3,409 real changes made)
. estat elasticities, expenditure
```

Expenditure elasticities $\quad$ Number of obs $=4,160$

| Expenditure | Elasticity | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | :---: | :---: |
| Dairy | .8464408 | .0266003 | 31.82 | 0.000 | .7943051 | .8985764 |
| Meats | 1.20188 | .0252998 | 47.51 | 0.000 | 1.152294 | 1.251467 |
| FruitVeg | 1.130297 | .0323216 | 34.97 | 0.000 | 1.066948 | 1.193646 |
| Flours | .7604338 | .0229378 | 33.15 | 0.000 | .7154766 | .8053911 |
| Misc | .9082412 | .0287833 | 31.55 | 0.000 | .8518269 | .9646554 |

```
. replace n_kids = n_kids_backup
(3,409 real changes made)
```

Here we first created a backup variable holding the true values of $n \_k i d s$, and then we set $n \_k i d s$ equal to two for all observations. After calling estat elasticities, we restored the true values.

## - Technical note

In principle, a second way to obtain the expenditure elasticities under the counterfactual that all households have two children would be to type

```
estat elasticities, expenditure at(n_kids=2)
```

The key is the at () option, which allows us to request that variables be set to the values we specify before calculating the elasticities. For more information about the at() option, see [R] margins, especially Syntax of at().

However, in this example, if you issue the previous command, you would obtain the error message

```
. estat elasticities, expenditure at(n_kids=2)
Option at() specified; using margins.
could not calculate numerical derivatives -- discontinuous region with missing
values encountered
r(459);
```

The issue is that for most demand systems the predicted quantities are nonlinear functions of prices. As the output indicates, when you specify the at() option, estat elasticities calls margins to obtain the desired results. Depending on the exact demand system and demographic specification, margins may not be able to obtain all the required derivatives numerically.

When you do not use the at() option, estat elasticities uses analytics formulas for the expenditure and price elasticities; Methods and formulas of [R] demandsys includes the elasticity formulas for all the demand systems implemented. Evaluating analytic formulas is quick, and most of the time is spent computing the variance-covariance matrix of the estimated mean elasticities, because that does use numerical derivatives. Because those derivatives are for a sample mean, rather than individual observations, you are unlikely to obtain the same message as margins issued in the previous command.

In contrast, margins uses numerical derivatives for all its calculations, and, moreover, it must incur the added expense of having to call predict to evaluate predicted quantities. One case where you would need to use margins is to obtain, say, a quantity elasticity with respect to a change in a demographic variable, because those elasticities are not programmed into demandsys.

In the text after example 4 and in example 6 of $[R]$ demandsys, we cautioned that after fitting a generalized model that allows for subsistence or committed quantities, you should check whether that interpretation actually holds. If for some household $\sum_{g} p_{g}\left(\mu_{g}+\boldsymbol{\nu}_{g}^{\prime} \mathbf{d}\right)>m$, then either the household is not consuming enough to meet what we think is a minimum subsistence or our assumption that this quantity represents a minimum subsistence is wrong. Moreover, if $\mu_{g}+\boldsymbol{\nu}_{g}^{\prime} \mathbf{d}<0$, then we are saying that the minimum required level of $g$ is less than 0 , which implies that good $g$ is in fact not needed for subsistence.

## > Example 2

This example is a continuation of example 6 in [R] demandsys. In that example, we fit a generalized QUAIDS model, which we refit here:

```
. use https://www.stata-press.com/data/r18/food_consumption, clear
(Food consumption)
. demandsys gquaids w_dairy w_proteins w_fruitveg w_flours w_misc,
> prices(p_dairy p_proteins p_fruitveg p_flours p_misc)
> demographics(n_kids n_adults) piconstant(3) expenditure(expfd) nolog
    (output omitted)
```

Next, we compute variables containing each household's implied minimum quantity of each good conditional on that household's demographics. Given those variables, we can then calculate the expenditure needed to acquire the minimum quantities given the prices each household faces. We type

```
. estat parameters
. return list
macros:
                r(names) : "alpha beta Gamma lambda mu Nu"
matrices:
            r(Nu) : 5 x 2
            r(mu) : 5 x 1
            r(lambda) : 5 x 1
                        r(Gamma) : 5 x 5
                        r(beta) : 5 x 1
                            r(alpha) : 5 x 1
. matrix mu = r(mu)
. matrix Nu = r(Nu)
. forvalues g = 1/5 {
    2. generate minq_'g' = mu['g', 1] + Nu['g',1]*n_kids + Nu['g',2]*n_adults
    3. }
. generate sub_exp = p_dairy*minq_1 + p_proteins*minq_2
> + p_fruitveg*minq_3 + p_flour*minq_4 + p_misc*minq_5
```

We first issued the command estat parameters to obtain the estimated parameter matrices from our demand system. The following command shows that six vectors and matrices were stored, following the layout of the coefficient table from demandsys. To compute minimum quantities, we need matrices $\boldsymbol{\mu}$ and $\mathbf{N}$, so we transferred those $r()$ results to Stata matrices.

Because we specified n_kids first in the demographics() option of demandsys, that variable's coefficients are stored in the first column of our matrix $\mathrm{r}(\mathrm{Nu})$, which we converted to Nu . The second column corresponds to coefficients associated with n_adults. We looped over the five goods in our system and created variables named minq- $g$ containing the minimum quantities. The final generate statement calculates the minimum expenditure for each household given the prices it faced and our calculated minimum quantities.

Now we can look at the results.

| . summarize minq_* |  |  |  |  |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| Variable | Obs | Mean | Std. dev. | Min | Max |
| minq_1 | 4,160 | -1.604524 | .9955551 | -7.287278 | -.2078816 |
| minq_2 | 4,160 | -2.893996 | 1.743902 | -13.71866 | -.6731596 |
| minq_3 | 4,160 | -5.344703 | 3.802853 | -29.30052 | -.5518757 |
| minq_4 | 4,160 | .3094949 | .1354879 | -.0171549 | 1.328885 |
| minq_5 | 4,160 | -.1878336 | .2030023 | -1.34631 | .0894621 |

For the first three goods in our system, the calculated minimum quantity is negative for every single household in our sample. That is hardly encouraging if we wish to take the subsistence interpretation seriously, because it implies three of the five categories of food we have are not required for subsistence. Moreover, on average, good 5 is not required either. Even for good 4, for some households, the minimum quantity is negative. Finally, we look at the expenditure needed to purchase the minimum quantities of the five goods:

| • summarize sub_exp |  |  |  |  |  |
| ---: | :---: | :---: | :---: | :---: | ---: |
| Variable | Obs | Mean | Std. dev. | Min | Max |
| sub_exp | 4,160 | -8.10118 | 5.480523 | -43.8327 | .2678578 |

Not surprisingly given the previous summary statistics, the minimum required expenditure for most households is less than zero.

What are we to make of the results in the previous example? We think there are at least four explanations:

1. There are minimum quantities of each good consumers must purchase for subsistence, but our demand system is misspecified. If you believe this to be true, you might try changing the demographic variables for which you control. Alternatively, try a different model; for example, use a generalized QUAIDS model rather than a generalized translog or AIDS model.
2. The data are simply not compatible with a minimum subsistence interpretation. That does not imply the results of the fitted demand system must be ignored. As Pollak and Wales (1992, 75) write, any demand system that includes demographic translation should include a constant term in the demand equations-which is the key characteristic of the models we refer to as "generalized". For example, rather than using the translog model with demographic translation, you should instead use the generalized translog model with demographic translation. In other words, the $\mu$ parameters do belong in the model, but they should not be construed as minimum quantities required for subsistence.
3. Items for which the calculated minimum quantities are positive for all observations must be purchased to achieve subsistence, but items for which the minimum quantities are negative need not be consumed for subsistence. Under this interpretation, in our example, households must consume breads, cereals, and pastas but need not consume any other foods for minimal subsistence. That is not particularly palatable to us.
4. The procedure we used is useful as a diagnostic tool, but it is not a formal statistical test. Our estimated parameters are subject to sampling variation, and therefore our calculated minimum quantities are also subject to sampling variation. We have not accounted for this. If we were to conduct a test of the hypothesis that all minimum quantities are zero or more versus the alternative that some are negative, we may not be able to reject the null hypothesis.

## Evaluating elasticities

The elasticities that are reported with estat elasticities refer to the slope of the fitted demand curve. Let us explain what we mean. Look at figure 1 . We have plotted a fitted demand curve $D^{\mathrm{F}}$ as a straight line for simplicity, and the large dots represent observed (price, quantity) pairs for two observations. The dotted lines represent each observation's residual, and the plus signs represent the predicted quantities given the two observations' prices. While quantity is a function of price here, following the influential English economist Alfred Marshall, we plot price on the ordinate; hence, the lines connecting observations to the regression line are horizontal, not vertical.


Figure 1. Actual and predicted quantities
For the AIDS model without demographic variables included, we can show that the expenditure elasticity for the $g$ th good is given by

$$
E^{g}=1+\frac{\beta_{g}}{w_{g}}
$$

We do not observe the true value $\beta_{g}$, so to compute the elasticity for household $i$, we have no choice but to use the estimated $\widehat{\beta}_{g}$. However, we observe both household $i$ 's true expenditure share $w_{g i}$, and we can compute its predicted expenditure share $\widehat{w_{g i}}$.

In terms of figure 1, the issue is whether we evaluate the expenditure or any other elasticity formula at the red and blue dots or at their corresponding fitted values shown by the pluses on the demand curve. We believe the correct procedure is to evaluate elasticity formulas at their fitted values. That is, estat elasticities evaluates the expenditure share elasticity for household $i$ as

$$
\widehat{E_{i}^{g}}=1+\frac{\widehat{\beta_{g}}}{\widehat{w_{g i}}}
$$

After all, for policy or other analyses, what we want to know is essentially the slope of the fitted demand curve. We need to be able to predict the effect of a policy on a household, and in general to do that, we predict the household's response with the policy imposed and compare it with the predicted response in the absence of the policy. In other words, our elasticity computations measure the slope of the fitted demand curve, not the slope of an imaginary line that passes through one of our dots that may or may not be parallel to the fitted demand curve.

In contrast, the postestimation tools accompanying the community-contributed quaids command of Poi (2012) use the actual values $w_{g}$ to calculate elasticities. Thus, users of that command will see
minor differences when refitting models using demandsys and estat elasticities. We believe, however, that evaluating the formulas at the predicted expenditure shares, or equivalently, evaluating the elasticity formulas along the demand curve rather than at points in space that need not lie on the demand curve, is conceptually more sound. Moreover, the elasticity results reported after quaids cannot be replicated using the margins command, but the results reported by estat elasticities are identical to those one would obtain using margins.

## Compensating and equivalent variation

Although students in introductory economics courses learn of consumer's surplus as a way to measure whether consumers are better or worse off in response to a policy such as a tax, it has limitations. Perhaps most importantly, consumer's surplus is not well defined in most cases; see Auerbach (1985). Also see Harberger (1964), who develops a method for calculating consumer welfare when multiple prices change but assumes the timing of the price changes is scattered rather than occurring simultaneously. Hines (1999) provides a historical overview of consumer's welfare measurement.

The workhorse metrics used to evaluate a policy are compensating and equivalent variation (CV and EV, respectively). These metrics assign a monetary amount to a household's change in welfare and can accommodate policies that cause multiple prices to change simultaneously. Because predict after demandsys allows us to obtain the predicted values of the expenditure and indirect utility functions, these metrics are easy to obtain.

CV measures how much a social planner would have to compensate a consumer to offset the effect of, say, a tax that causes prices to rise. Consider figure 2, which illustrates CV for two goods and would be familiar to those who have studied intermediate microeconomics.


Figure 2. CV if the price of good 1 rises
Initially, the price of good 1 is $p_{0}$, and the line segment $\overline{A \cdot p_{0} / m}$ represents the budget constraint. The consumer purchases $q_{0}$ units of good 1 and enjoys utility level $U_{0}$. Now suppose the price of good 1 rises to $p_{1}$ : consumption of good 1 falls to $q_{1}$ and utility falls to $U_{1}$; line segment $\overline{A \cdot p_{1} / m}$ represents the new budget constraint.

The dotted line segment $\overline{B \cdot p_{1}^{\prime} m}$ represents a budget constraint based on price $p_{1}$ but shifted outward so that the consumer can still achieve utility level $U_{0}$ when facing price $p_{1}$, by consuming $q_{1}^{\prime}$. The line segment $\overline{B \cdot A}$ represents the CV . It tells us, in the face of a higher price for good 1 , how much the social planner would have to compensate this consumer so that he or she can enjoy
the same level of utility as before the price increase. In this case, CV is negative because the social planner must pay the consumer.

Notice the close similarity to the concept of the income and substitution effects that cause demand curves to slope downward. CV, and EV as we will see shortly, is a measure of the income effect.

Mathematically, CV is defined as

$$
\begin{aligned}
\mathbf{C V} & =e\left(\mathbf{p}^{1}, u^{1}\right)-e\left(\mathbf{p}^{1}, u^{0}\right) \\
& =e\left(\mathbf{p}^{0}, u^{0}\right)-e\left(\mathbf{p}^{1}, u^{0}\right) \\
& =m-e\left(\mathbf{p}^{1}, u^{0}\right)
\end{aligned}
$$

where $\mathbf{p}^{0}$ and $\mathbf{p}^{1}$ are the price vectors before and after the policy is implemented and $u^{0}$ is the household's initial level of utility. Where do we get that level of utility, considering that utility is unobserved? The answer is the indirect utility function, which expresses utility as a function of observed prices and expenditure; we can evaluate the indirect utility function by calling predict with the iuf option. With the indirect utility variable in hand, we then call predict with the ef option and set the price variables to $\mathbf{p}^{1}$.

## > Example 3

We compute household CV in response to a $10 \%$ increase in the price of dairy products, holding all other prices constant. See Poi (2002) for similar computations for an expanded look at the welfare implications of lower dairy and other prices. First, we fit a generalized QUAIDS model controlling for the number of children and adults in each household. We type

```
. use https://www.stata-press.com/data/r18/food_consumption, clear
(Food consumption)
. quietly demandsys gquaids w_dairy w_proteins w_fruitveg w_flours w_misc,
> prices(p_dairy p_proteins p_fruitveg p_flours p_misc)
> expenditures(expfd) demographics(n_kids n_adults) nolog
. estimates store gquaids
```

We stored our estimation results for later use by specifying the estimates store command. Next, we obtain each household's initial utility $u^{0}$ by evaluating the indirect utility function:

```
. predict u0, iuf
```

Now, we create a new dairy price variable containing the price each household will face after the imposition of the tax. Given that price variable and initial utility $u^{0}$, we can calculate each household's new level of expenditure. Finally, we compute the CV. We type

```
. generate ptax_dairy = p_dairy * 1.10
. predict ep1u0, ef utilities(u0)
> prices(ptax_dairy p_proteins p_fruitveg p_flours p_misc)
. generate cv = expfd - ep1u0
```

Here we used the prices () option of predict to specify the new prices $\mathbf{p}^{1}$ at which we want the expenditure function. Had we not specified that option, predict would have used the prices used at estimation, $\mathbf{p}^{0}$. Because we wanted to evaluate the function at the level of expenditures $m$, the same as used at estimation, we did not need to specify the expenditures () option of predict.

We summarize our results by looking at the average CV by household size:
. mean cv, over(n_kids)
Mean estimation

|  |  |  | Number of obs $=4,160$ |  |
| :---: | ---: | ---: | ---: | ---: |
| c.cv@n_kids |  |  |  |  |
| 0 | -.5142129 | .0066562 | -.5272626 | -.5011631 |
| 1 | -.7762653 | .013486 | -.802705 | -.7498257 |
| 2 | -.9060989 | .014986 | -.9354795 | -.8767183 |
| 3 | -1.067159 | .0268302 | -1.11976 | -1.014557 |
| 4 | -1.22238 | .0506472 | -1.321675 | -1.123084 |
| 5 | -1.550712 | .158197 | -1.860863 | -1.240561 |
| 6 | -1.304841 | .2223387 | -1.740744 | -.8689384 |
| 7 | -1.902748 | .1828158 | -2.261165 | -1.544331 |
| 8 | -2.923615 | . | . | . |
| 10 | -1.743889 | . | . | . |

The standard errors of the means are missing for households with 8 and 10 children because only one household of each size exists in our data. For the most part, CV becomes more negative with the number of children, as we expected, because larger households are likely to consume more dairy products by virtue of there being more people within the household. To reiterate, CV is negative here, indicating that the social planner must transfer money to the consumer. For a price decrease, CV would be positive.

EV is similar to CV, except that we use the postpolicy level of utility and the initial price vector rather than the initial level of utility and postpolicy price vector. We have

$$
\begin{aligned}
\mathrm{EV} & =e\left(\mathbf{p}^{0}, u^{1}\right)-e\left(\mathbf{p}^{0}, u^{0}\right) \\
& =e\left(\mathbf{p}^{0}, u^{1}\right)-e\left(\mathbf{p}^{1}, u^{1}\right) \\
& =e\left(\mathbf{p}^{0}, u^{1}\right)-m
\end{aligned}
$$

To aid in understanding EV, you may wish to sketch a diagram similar to figure 2. EV tells us how much consumers would be willing to pay to trade at the original price vector rather than the postpolicy one. In other words, EV tells us the change in expenditure that would yield $u^{1}$ if total expenditure were to change rather than the price vector. A negative value indicates that consumers are worse off.

## > Example 4

Continuing the previous example, to calculate EV, we first need to calculate the new level of utility $u^{1}$ each household will enjoy based on the postpolicy dairy price. Because the preferences underlying all the demand systems implemented by demandsys are locally nonsatiated, each household will exhaust its full budget. Thus, the level of expenditure we supply as an argument to the indirect utility function is simply $m$, the value we used during estimation. To get $u^{1}$, we can therefore call predict, iuf at postpolicy prices.

```
. estimates restore gquaids
(results gquaids are active now)
. predict u1, iuf prices(ptax_dairy p_proteins p_fruitveg p_flours p_misc)
```

We had to restore our estimation results because the means command we used previously overwrites results in e ().

Now we evaluate the expenditure function at original prices and our computed level of utility $u^{1}$. Because the original prices were used at estimation, we do not need to supply alternative prices via the prices() option. We type

```
. predict epOu1, ef utilities(u1)
```

We are now ready to compute EV and summarize the results:

```
. generate ev = ep0u1 - expfd
. mean ev, over(n_kids)
```

Mean estimation $\quad$ Number of obs $=4,160$

|  | Mean | Std. err. | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: |
| c.ev@n_kids |  |  |  |  |
| 0 | -.5073723 | .0065622 | -.5202378 | -.4945068 |
| 1 | -.7660214 | .013292 | -.7920807 | -.739962 |
| 2 | -.8940226 | .014762 | -.922964 | -.8650811 |
| 3 | -1.052524 | .0264206 | -1.104322 | -1.000725 |
| 4 | -1.205508 | .0498826 | -1.303304 | -1.107711 |
| 5 | -1.528722 | .1554489 | -1.833485 | -1.223959 |
| 6 | -1.285955 | .218652 | -1.71463 | -.8572807 |
| 7 | -1.871663 | .178062 | -2.22076 | -1.522567 |
| 8 | -2.87738 | . | . | . |
| 10 | -1.716629 | . | . | . |

For small price changes, CV and EV will be similar; however, they will not match because CV is based on the original level of utility and postpolicy prices and EV is based on the postpolicy level of utility and original prices. Which one is more appropriate in any situation depends on the policy and goals. If the government is implementing, say, a carbon tax to reduce consumption of goods and services that emit carbon dioxide, CV may be a better choice because it tells how much the government must compensate consumers so that their overall utilities do not change. If one is evaluating the introduction of a new product competing with an existing product into the market, EV may be a better metric because it could be interpreted as a measure of how much better off consumers would be thanks to lower prices.

As with our analysis of estimated minimum quantities in example 2, here we have not attempted to obtain measures of the sampling variation associated with the CV or EV. One alternative to obtain, say, the sampling variation of the mean CV across households would be to use the bootstrap, being sure to bootstrap the entire procedure, including refitting the model at each bootstrap replication as well as obtaining the desired statistics.

## References

Auerbach, A. J. 1985. The theory of excess burden and optimal taxation. In Vol. 1 of Handbook of Public Economics, ed. A. J. Auerbach and M. Feldstein, 61-127. Amsterdam: Elsevier. https://doi.org/10.1016/S1573-4420(85)80005-7.

Harberger, A. C. 1964. The measurement of waste. American Economic Review 54: 58-76.
Hines, J. R. 1999. Three sides of Harberger triangles. Journal of Economic Perspectives 13: 167-188. https://doi.org/10.1257/jep.13.2.167.
Poi, B. P. 2002. From the help desk: Demand system estimation. Stata Journal 2: 403-410.
——. 2012. Easy demand-system estimation with quaids. Stata Journal 12: 433-446.
Pollak, R. A., and T. J. Wales. 1992. Demand System Specification and Estimation. New York: Oxford University Press.

Varian, H. R. 1992. Microeconomic Analysis. 3rd ed. New York: W. W. Norton.

## Also see

[R] demandsys - Estimation of flexible demand systems
[U] 20 Estimation and postestimation commands

## Title

Diagnostic plots - Distributional diagnostic plots

## Description

Menu
Options for symplot, quantile, and qqplot
Options for qchi and pchi
Methods and formulas
References

Quick start
Syntax
Options for qnorm and pnorm
Remarks and examples
Acknowledgments
Also see

## Description

symplot graphs a symmetry plot of varname.
quantile plots the ordered values of varname against the quantiles of a uniform distribution.
qqplot plots the quantiles of varname $_{1}$ against the quantiles of varname ${ }_{2}$ (Q-Q plot).
qnorm plots the quantiles of varname against the quantiles of the normal distribution ( $\mathrm{Q}-\mathrm{Q}$ plot).
pnorm graphs a standardized normal probability plot ( $\mathrm{P}-\mathrm{P}$ plot).
qchi plots the quantiles of varname against the quantiles of a $\chi^{2}$ distribution ( $\mathrm{Q}-\mathrm{Q}$ plot).
pchi graphs a $\chi^{2}$ probability plot ( $\mathrm{P}-\mathrm{P}$ plot).
See $[R]$ regress postestimation diagnostic plots for regression diagnostic plots and $[R]$ logistic postestimation for logistic regression diagnostic plots.

## Quick start

Symmetry plot for v1
symplot v1
Change marker color and size
symplot v1, mcolor(red) msize(large)
Plot ordered values of v1 against quantiles of the uniform distribution
quantile v 1
Same as above, but only for observations with v2 greater than 5
quantile v1 if v2 > 5
Plot quantiles of v 1 against quantiles of v 2
qqplot v1 v2
Change thickness of the reference line
qqplot v1 v2, rlopts(lwidth(thick))
Plot quantiles of v1 against quantiles of the normal distribution
qnorm v 1
Add grid lines
qnorm v1, grid

Standardized normal probability plot for v1
pnorm v1
Change labels on the $x$ and $y$ axes
pnorm v1, xlabel(0(0.1)1) ylabel(0(0.1)1)
Plot quantiles of v1 against quantiles of the $\chi_{1}^{2}$ distribution
qchi v1
Same as above, but comparing with quantiles of the $\chi_{2}^{2}$ distribution qchi v1, df(2)

## $\chi^{2}$ probability plot for v 1 pchi v1

Add " $\chi^{2}$ (1) P-P plot" to graph pchi v1, title("\{\&chi\}\{sup:2\}(1) P-P plot")

## Menu

## symplot

Statistics $>$ Summaries, tables, and tests $>$ Distributional plots and tests $>$ Symmetry plot

## quantile

Statistics $>$ Summaries, tables, and tests $>$ Distributional plots and tests $>$ Quantiles plot

## qqplot

Statistics $>$ Summaries, tables, and tests $>$ Distributional plots and tests $>$ Quantile-quantile plot

## qnorm

Statistics $>$ Summaries, tables, and tests $>$ Distributional plots and tests $>$ Normal quantile plot

## pnorm

Statistics $>$ Summaries, tables, and tests $>$ Distributional plots and tests $>$ Normal probability plot, standardized

## qchi

Statistics $>$ Summaries, tables, and tests $>$ Distributional plots and tests $>$ Chi-squared quantile plot

## pchi

Statistics $>$ Summaries, tables, and tests $>$ Distributional plots and tests $>$ Chi-squared probability plot

## Syntax

Symmetry plot
symplot varname [if][in][, options $\left.{ }_{1}\right]$

Ordered values of varname against quantiles of uniform distribution quantile varname [if][in] [, options ${ }_{1}$ ]

Quantiles of varname ${ }_{1}$ against quantiles of varname ${ }_{2}$ qqplot varname $_{1}$ varname $_{2}[$ if $][$ in $]\left[\right.$, options $\left.{ }_{1}\right]$

Quantiles of varname against quantiles of normal distribution qnorm varname [if] [in] [, options 2 ]

Standardized normal probability plot
pnorm varname [if] [in] [, options 2 ]

Quantiles of varname against quantiles of $\chi^{2}$ distribution qchi varname [if] [in] [, options 3 ]
$\chi^{2}$ probability plot
pchi varname [if] [in] [, options 3 ]
options $_{1}$
Description
Plot
marker_options
marker_label_options
Reference line
rlopts (cline_options) affect rendition of the reference line
Add plots
addplot (plot) add other plots to the generated graph
Y axis, X axis, Titles, Legend, Overall
twoway_options any options other than by () documented in [G-3] twoway_options


## Options for symplot, quantile, and qqplot

$\qquad$ Plot
marker_options affect the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G-3] marker_options.
marker_label_options specify if and how the markers are to be labeled; see [G-3] marker_label_options.
$\qquad$ Reference line
rlopts(cline_options) affect the rendition of the reference line; see [G-3] cline_options.
addplot (plot) provides a way to add other plots to the generated graph; see [G-3] addplot_option.

Y axis, X axis, Titles, Legend, Overall
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

## Options for qnorm and pnorm

$\qquad$ Main
grid adds grid lines at the $0.05,0.10,0.25,0.50,0.75,0.90$, and 0.95 quantiles when specified with qnorm. With pnorm, grid is equivalent to yline(.25,.5,.75) xline(.25, .5, .75).

## $\sqrt{\text { Plot }}$

marker_options affect the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G-3] marker_options.
marker_label_options specify if and how the markers are to be labeled; see [G-3] marker_label_options.

rlopts (cline_options) affect the rendition of the reference line; see [G-3] cline_options.

```
            Add plots
    addplot (plot) provides a way to add other plots to the generated graph; see [G-3] addplot_option.
```

$\longleftarrow \sqrt{\mathrm{Y} \text { axis, } \mathrm{X} \text { axis, Titles, Legend, Overall }}$
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

## Options for qchi and pchi

$\qquad$ Main
grid adds grid lines at the $0.05,0.10,0.25,0.50,0.75,0.90$, and .95 quantiles when specified with qchi. With pchi, grid is equivalent to yline (. $25, .5, .75$ ) xline (. $25, .5, .75$ ). $\mathrm{df}(\#)$ specifies the degrees of freedom of the $\chi^{2}$ distribution. The default is $\mathrm{df}(1)$.
$\qquad$ $\sqrt{\text { Plot }}$
marker_options affect the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G-3] marker_options.
marker_label_options specify if and how the markers are to be labeled; see [G-3] marker_label_options.

Reference line
rlopts(cline_options) affect the rendition of the reference line; see [G-3] cline_options.

Add plots
addplot (plot) provides a way to add other plots to the generated graph; see [G-3] addplot_option.

Y axis, X axis, Titles, Legend, Overall
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

## Remarks and examples

Remarks are presented under the following headings:

```
symplot
quantile
qqplot
qnorm
pnorm
qchi
pchi
```


## symplot

$\rangle$ Example 1
We have data on 74 automobiles. To make a symmetry plot of the variable price, we type

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. symplot price
```



All points would lie along the reference line (defined as $y=x$ ) if car prices were symmetrically distributed. The points in this plot lie above the reference line, indicating that the distribution of car prices is skewed to the right - the most expensive cars are far more expensive than the least expensive cars are inexpensive.

The logic works as follows: a variable, $z$, is distributed symmetrically if

$$
\text { median }-z_{(i)}=z_{(N+1-i)}-\text { median }
$$

where $z_{(i)}$ indicates the $i$ th-order statistic of $z$. symplot graphs $y_{i}=$ median $-z_{(i)}$ versus $x_{i}=$ $z_{(N+1-i)}$ - median.

For instance, consider the largest and smallest values of price in the example above. The most expensive car costs $\$ 15,906$ and the least expensive, $\$ 3,291$. Let's compare these two cars with the typical car in the data and see how much more it costs to buy the most expensive car, and compare that with how much less it costs to buy the least expensive car. If the automobile price distribution is symmetric, the price differences would be the same.

Before we can make this comparison, we must agree on a definition for the word "typical". Let's agree that "typical" means median. The price of the median car is $\$ 5,006.50$, so the most expensive car costs $\$ 10,899.50$ more than the median car, and the least expensive car costs $\$ 1,715.50$ less than the median car. We now have one piece of evidence that the car price distribution is not symmetric. We can repeat the experiment for the second-most-expensive car and the second-least-expensive car. We find that the second-most-expensive car costs $\$ 9,494.50$ more than the median car, and the second-least-expensive car costs $\$ 1,707.50$ less than the median car. We now have more evidence. We can continue doing this with the third most expensive and the third least expensive, and so on.

Once we have all of these numbers, we want to compare each pair and ask how similar, on average, they are. The easiest way to do that is to plot all the pairs.

## quantile

> Example 2
We have data on the prices of 74 automobiles. To make a quantile plot of price, we type

```
. use https://www.stata-press.com/data/r18/auto, clear
(1978 automobile data)
. quantile price, rlopts(clpattern(dash)) ytitle(Quantiles of price)
```



We changed the pattern of the reference line by specifying rlopts(clpattern(dash)).
In a quantile plot, each value of the variable is plotted against the fraction of the data that have values less than that fraction. The diagonal line is a reference line. If automobile prices were rectangularly distributed, all the data would be plotted along the line. Because all the points are below the reference line, we know that the price distribution is skewed right.

## qqplot

> Example 3
We have data on the weight and country of manufacture of 74 automobiles. We wish to compare the distributions of weights for domestic and foreign automobiles:

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. generate weightd=weight if !foreign
(22 missing values generated)
. generate weightf=weight if foreign
(52 missing values generated)
. qqplot weightd weightf
```



## qnorm

## > Example 4

Continuing with our price data on 74 automobiles, we now wish to compare the distribution of price with the normal distribution:
. qnorm price, grid


Grid lines are $5,10,25,50,75,90$, and 95 percentiles.
The result shows that the distributions are different.

## - Technical note

The idea behind qnorm is recommended strongly by Miller (1997): he calls it probit plotting. His recommendations from much practical experience should interest many users. "My recommendation for detecting nonnormality is probit plotting" (Miller 1997, 10). "If a deviation from normality cannot be spotted by eye on probit paper, it is not worth worrying about. I never use the Kolmogorov-Smirnov test (or one of its cousins) or the $\chi^{2}$ test as a preliminary test of normality. They do not tell you how the sample is differing from normality, and I have a feeling they are more likely to detect irregularities in the middle of the distribution than in the tails" (Miller 1997, 13-14).

## pnorm

## Example 5

Quantile-normal plots emphasize the tails of the distribution. Normal probability plots put the focus on the center of the distribution:
. pnorm price, grid


## qchi

## > Example 6

Suppose that we want to examine the distribution of the sum of squares of price and mpg, standardized for their variances.
. egen $c 1=$ std(price)
. egen $c 2=\operatorname{std}(m p g)$
. generate ch = c1^2 + c2~2
. qchi ch, df(2) grid ylabel(, axis(2) labsize(*.8) format(\%4.2f))
> xlabel(, axis(2) format(\%4.2f))


The quadratic form is clearly not $\chi^{2}$ with 2 degrees of freedom.

## pchi

## > Example 7

We can focus on the center of the distribution by doing a probability plot:
. pchi ch, df(2) grid


## Methods and formulas

Let $x_{(1)}, x_{(2)}, \ldots, x_{(N)}$ be the data sorted in ascending order.
If a continuous variable, $x$, has a cumulative distribution function $F(x)=P(X \leq x)=p$, the quantiles $x_{p_{i}}$ are such that $F\left(x_{p_{i}}\right)=p_{i}$. For example, if $p_{i}=0.5$, then $x_{0.5}$ is the median. When we plot data, the probabilities, $p_{i}$, are often referred to as plotting positions. There are many different conventions for choice of plotting positions, given $x_{(1)} \leq \cdots \leq x_{(N)}$. Most belong to the family $(i-a) /(N-2 a+1) . a=0.5$ (suggested by Hazen) and $a=0$ (suggested by Weibull) are popular choices.

For a wider discussion of the calculation of plotting positions, see Cox (2002).
symplot plots median $-x_{(i)}$ versus $x_{(N+1-i)}$ - median.
quantile plots $x_{(i)}$ versus $(i-0.5) / N$ (the Hazen position).
qnorm plots $x_{(i)}$ against $q_{i} \times \widehat{\sigma}+\widehat{\mu}$, where $q_{i}=\Phi^{-1}\left(p_{i}\right), \Phi$ is the cumulative normal distribution, $p_{i}=i /(N+1)$ (the Weibull position), $\widehat{\sigma}$ is the standard deviation, and $\widehat{\mu}$ is the mean of the data.
pnorm plots $\Phi\left\{\left(x_{i}-\widehat{\mu}\right) / \widehat{\sigma}\right\}$ versus $p_{i}=i /(N+1)$, where $\widehat{\mu}$ is the mean of the data and $\widehat{\sigma}$ is the standard deviation.
qchi and pchi are similar to qnorm and pnorm; the cumulative $\chi^{2}$ distribution is used in place of the cumulative normal distribution.
qqplot is just a two-way scatterplot of one variable against the other after both variables have been sorted into ascending order, and both variables have the same number of nonmissing observations. If the variables have unequal numbers of nonmissing observations, interpolated values of the variable with more data are plotted against the variable with fewer data.

Ramanathan Gnanadesikan (1932-2015) was born in Madras. He obtained degrees from the Universities of Madras and North Carolina. He worked in industry at Procter \& Gamble, Bell Labs, and Bellcore, as well as in universities, retiring from Rutgers in 1998. Among many contributions to statistics, he is especially well known for work on probability plotting, robustness, outlier detection, clustering, classification, and pattern recognition.
Martin Bradbury Wilk (1922-2013) was born in Montreal. He obtained degrees in chemical engineering and statistics from McGill and Iowa State Universities. After holding several statisticsrelated posts in industry and at universities (including periods at Princeton, Bell Labs, and Rutgers), Wilk was appointed Chief Statistician at Statistics Canada (1980-1986). He is especially well known for his work with Gnanadesikan on probability plotting and with Shapiro on tests for normality.

## Acknowledgments

We thank Peter A. Lachenbruch (1937-2021), Emeritus Appointment, Biostatistics, College of Public Health and Human Sciences, Oregon State University for writing the original versions of qchi and pchi. Patrick Royston of the MrC Clinical Trials Unit, London, and coauthor of the Stata Press book Flexible Parametric Survival Analysis Using Stata: Beyond the Cox Model also published a similar command in Royston (1996).

## References

Chambers, J. M., W. S. Cleveland, B. Kleiner, and P. A. Tukey. 1983. Graphical Methods for Data Analysis. Belmont, CA: Wadsworth.

Cox, N. J. 2002. Speaking Stata: On getting functions to do the work. Stata Journal 2: 411-427.
-_. 2004a. Speaking Stata: Graphing distributions. Stata Journal 4: 66-88.
-_. 2004b. gr42_2: Software update: Quantile plots, generalized. Stata Journal 4: 97.
——. 2005a. Speaking Stata: Density probability plots. Stata Journal 5: 259-273.
-_. 2005b. Speaking Stata: The protean quantile plot. Stata Journal 5: 442-460.
-_. 2005c. Speaking Stata: Smoothing in various directions. Stata Journal 5: 574-593.
-_. 2007. Stata tip 47: Quantile-quantile plots without programming. Stata Journal 7: 275-279.

- 2012. Speaking Stata: Axis practice, or what goes where on a graph. Stata Journal 12: 549-561.

Daniel, C., and F. S. Wood. 1980. Fitting Equations to Data: Computer Analysis of Multifactor Data. 2nd ed. New York: Wiley.
Gan, F. F., K. J. Koehler, and J. C. Thompson. 1991. Probability plots and distribution curves for assessing the fit of probability models. American Statistician 45: 14-21. https://doi.org/10.2307/2685233.

Genest, C., and G. J. Brackstone. 2013. Obituary: Martin B. Wilk, 1922-2013. IMS Bulletin 42(4): 7-8.
Hoaglin, D. C. 1985. Using quantiles to study shape. In Exploring Data Tables, Trends, and Shapes, ed. D. C. Hoaglin, C. F. Mosteller, and J. W. Tukey, 417-460. New York: Wiley.
Kettenring, J. R. 2001. A conversation with Ramanathan Gnanadesikan. Statistical Science 16: 295-309. https://doi.org/10.1214/ss/1009213730.

Miller, R. G., Jr. 1997. Beyond ANOVA: Basics of Applied Statistics. London: Chapman and Hall.
Nolan, D., and T. Speed. 2000. Stat Labs: Mathematical Statistics Through Applications. New York: Springer.
Royston, P. 1996. sg47: A plot and a test for the $\chi^{2}$ distribution. Stata Technical Bulletin 29: 26-27. Reprinted in Stata Technical Bulletin Reprints, vol. 5, pp. 142-144. College Station, TX: Stata Press.

Wilk, M. B., and R. Gnanadesikan. 1968. Probability plotting methods for the analysis of data. Biometrika 55: 1-17. https://doi.org/10.2307/2334448.

## Also see

[ R ] cumul - Cumulative distribution
$[\mathrm{R}]$ kdensity - Univariate kernel density estimation
[R] logistic postestimation - Postestimation tools for logistic
[R] lv - Letter-value displays
[R] regress postestimation diagnostic plots - Postestimation plots for regress

## Title

display - Substitute for a hand calculator

Description Quick start Syntax Remarks and examples Also see

## Description

display displays strings and values of scalar expressions.
display really has many more features and a more complex syntax diagram, but the diagram shown above is adequate for interactive use. For a full discussion of display's capabilities, see [P] display.

## Quick start

Perform calculations interactively
display $100 * 100$
Same as above, but include comma in the result
display \%6.Ofc 100*100
Verify choice of datetime function
display \%tm monthly("January 1983", "MY")
View formatted mean after summarize
display $\% 5.2 \mathrm{fr}$ (mean)
Add the variance with a different format on its own line

```
display "mean = " %5.2f r(mean) _newline "variance = " %10.4f r(Var)
```


## Syntax

display exp

## Remarks and examples

display can be used as a substitute for a hand calculator.

## $>$ Example 1

display $2+2$ produces the output 4. Stata variables may also appear in the expression, such as in display myvar/2. Because display works only with scalars, the resulting calculation is performed only for the first observation. You could type display myvar[10]/2 to display the calculation for the 10th observation. Here are more examples:

```
. display sqrt(2)/2
.70710678
. display normal(-1.1)
. }1356660
. di (57.2-3)/(12-2)
5.42
    display myvar/10
7
. display myvar[10]/2
3.5
```


## Also see

[P] display - Display strings and values of scalar expressions
[U] 13 Functions and expressions

## Title

do - Execute commands from a file

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Option | Reference | Also see |  |

## Description

do and run cause Stata to execute the commands stored in filename just as if they were entered from the keyboard. do echoes the commands as it executes them, whereas run is silent.

## Quick start

Execute commands stored in myfile.do
do myfile

Same as above, passing first arg in local macro 1 and arg2 in local macro 2 for use by myfile. do do myfile "first arg" arg2

Execute commands stored in myfile.do, continuing execution even if an error occurs in one or more commands
do myfile, nostop
Execute commands stored in myfile.do silently
run myfile

## Menu

File > Do...

## Syntax

$\{\mathrm{do} \mid \underline{\text { run }}\}$ filename $[$ arguments $][$, nostop $]$
filename (called a do-file) can be created using Stata's Do-file Editor; see [R] doedit. This file will be a standard text file. filename can also be created by using a non-Stata text editor; see [D] shell for a way to invoke your favorite editor from inside Stata. Ensure that you save the file in ASCII or UTF-8 format.

If filename is specified without an extension, .do is assumed.
If the path or filename contains spaces, it should be enclosed in double quotes.
A complete discussion of do-files, including information on arguments, can be found in [U] 16 Do-files.

## Option

nostop allows the do-file to continue executing even if an error occurs. Normally, Stata stops executing the do-file when it detects an error (nonzero return code).

## Reference

Jenkins, S. P. 2006. Stata tip 32: Do not stop. Stata Journal 6: 281.

## Also see

[R] doedit - Edit do-files and other text files
[P] include - Include commands from file
[GSM] 13 Using the Do-file Editor-automating Stata
[GSU] 13 Using the Do-file Editor-automating Stata
[GSW] 13 Using the Do-file Editor-automating Stata
[U] 15 Saving and printing output-log files
[U] 16 Do-files

## Title

doedit - Edit do-files and other text files
Description Quick start Menu Syntax Remarks and examples
Also see

## Description

doedit opens the Do-file Editor. This text editor lets you create and edit do-files, which typically contain a series of Stata commands. If you specify filename, doedit will open a text file, such as a do-file or an ado-file, saved to disk.

## Quick start

Open a new untitled do-file in the Do-file Editor
doedit
Open new or existing do-file myfile.do in the Do-file Editor
doedit myfile

## Menu

Window > Do-file Editor

## Syntax

doedit [filename]

## Remarks and examples

Clicking on the Do-file Editor button is equivalent to typing doedit.
doedit, typed by itself, invokes the Editor with an empty document. If you specify filename, that file is displayed in the Editor.

You may have more than one Do-file Editor open at once. Each time you submit the doedit command, a new window will be opened.

A tutorial discussion of doedit can be found in the Getting Started with Stata manual. Read [U] 16 Do-files for an explanation of do-files, and then read [GSW] 13 Using the Do-file Editorautomating Stata to learn how to use the Do-file Editor to create and execute do-files.

## Also see

[R] do - Execute commands from a file
[GSM] 13 Using the Do-file Editor-automating Stata
[GSU] 13 Using the Do-file Editor-automating Stata
[GSW] 13 Using the Do-file Editor-automating Stata
[U] 16 Do-files

## Title

dotplot - Comparative distribution dotplots

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Acknowledgments |
| Reference |  |  |  |

## Description

A dotplot is a scatterplot with values grouped together vertically ("binning", as in a histogram) and with plotted points separated horizontally. The aim is to display all the data for several variables or groups in one compact graphic.

## Quick start

Dotplot of v1

```
dotplot v1
```

Columns with separate dotplots of v 1 for each level of categorical variable a dotplot v1, over(a)

Same as above, but with the dots centered in each column
dotplot v1, over(a) center
Dotplots for v1, v2, and v3 in separate columns
dotplot v1 v2 v3
Add a horizontal line of pluses at the mean of each variable dotplot v1 v2 v3, mean

Add pluses for the medians and dashed lines for the upper and lower quartiles dotplot v1 v2 v3, median bar

## Menu

Graphics $>$ Distributional graphs $>$ Distribution dotplot

## Syntax

Dotplot of varname, with one column per value of groupvar
dotplot varname [if] [in] [, options]

Dotplot for each variable in varlist, with one column per variable

```
dotplot varlist [if] [in] [, options]
```

options

## Options

over(groupvar)
nx (\#)
ny (\#)
incr (\#)
mean median
bounded
bar
nogroup
center
Plot
marker_options
marker_label_options
Y axis, X axis, Titles, Legend, Overall
twoway_options
display one columnar dotplot for each value of groupvar
horizontal dot density; default is $\mathrm{nx}(0)$
vertical dot density; default is ny (35)
label every \# group; default is incr (1)
plot a horizontal line of pluses at the mean or median use minimum and maximum as boundaries
plot horizontal dashed lines at shoulders of each group
use the actual values of yvar
center the dot for each column
change look of markers (color, size, etc.)
add marker labels; change look or position
any options other than by () documented in [G-3] twoway_options
collect is allowed; see [U] 11.1.10 Prefix commands.

## Options

Options
over (groupvar) identifies the variable for which dotplot will display one columnar dotplot for each value of groupvar. over() may not be specified in the second syntax.
nx (\#) sets the horizontal dot density. A larger value of \# will increase the dot density, reducing the horizontal separation between dots. This option will increase the separation between columns if two or more groups or variables are used.
ny (\#) sets the vertical dot density (number of "bins" on the $y$ axis). A larger value of \# will result in more bins and a plot that is less spread out horizontally. \# should be determined in conjunction with nx() to give the most pleasing appearance.
incr(\#) specifies how the $x$ axis is to be labeled. incr (1), the default, labels all groups. incr (2) labels every second group.
[mean $\mid$ median] plots a horizontal line of pluses at the mean or median of each group.
bounded forces the minimum and maximum of the variable to be used as boundaries of the smallest and largest bins. It should be used with one variable whose support is not the whole of the real line and whose density does not tend to zero at the ends of its support, for example, a uniform random variable or an exponential random variable.
bar plots horizontal dashed lines at the "shoulders" of each group. The shoulders are taken to be the upper and lower quartiles unless mean has been specified; here they will be the mean plus or minus the standard deviation.
nogroup uses the actual values of yvar rather than grouping them (the default). This option may be useful if yvar takes on only a few values.
center centers the dots for each column on a hidden vertical line.
$\qquad$ Plot
marker_options affect the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G-3] marker_options.
marker_label_options specify if and how the markers are to be labeled; see [G-3] marker_label_options. marker_label_options are not allowed if varlist is specified.

Y axis, X axis, Titles, Legend, Overall
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

## Remarks and examples

dotplot produces a figure that has elements of a boxplot, a histogram, and a scatterplot. Like a boxplot, it is most useful for comparing the distributions of several variables or the distribution of 1 variable in several groups. Like a histogram, the figure provides a crude estimate of the density, and, as with a scatterplot, each symbol (dot) represents 1 observation.

## > Example 1

dotplot may be used as an alternative to Stata's histogram graph for displaying the distribution of one variable.
. set seed 123456789
. set obs 1000
. generate norm = rnormal()
. dotplot norm, title("Normal distribution, sample size 1000")
Normal distribution, sample size 1000


## Example 2

The over () option lets us use dotplot to compare the distribution of one variable within different levels of a grouping variable. The center, median, and bar options create a graph that may be compared with Stata's boxplot; see [G-2] graph box. The next graph illustrates this option with Stata's automobile dataset.

```
. use https://www.stata-press.com/data/r18/auto, clear
(1978 automobile data)
. dotplot mpg, over(foreign) nx(25) ny(10) center median bar
```



## > Example 3

The second version of dotplot lets us compare the distribution of several variables. In the next graph, all 10 variables contain measurements on tumor volume.

```
. use https://www.stata-press.com/data/r18/dotgr
. dotplot g1r1-g1r10, ytitle("Tumor volume, cu mm")
```



## Example 4

When using the first form with the over () option, we can encode a third dimension in a dotplot by using a different plotting symbol for different groups. The third dimension cannot be encoded with a varlist. The example is of a hypothetical matched case-control study. The next graph shows the exposure of each individual in each matched stratum. Cases are marked by the letter ' $x$ ', and controls are marked by the letter ' $o$ '.

```
. use https://www.stata-press.com/data/r18/dotdose
. label define symbol 0 "o" 1 "x"
. label values case symbol
```

```
dotplot dose, over(strata) m(none) mlabel(case) mlabp(0) center
```


> Example 5
dotplot can also be used with two virtually continuous variables as an alternative to jittering the data to distinguish ties. We must use the xlabel() option, because otherwise dotplot will attempt to label too many points on the $x$ axis. It is often useful in such instances to use a value of nx that is smaller than the default. That was not necessary in this example, partly because of our choice of symbols.

```
. use https://www.stata-press.com/data/r18/auto, clear
(1978 automobile data)
. generate byte hi_price = (price>10000) if price < .
. label define symbol 0 "|" 1 "o"
. label values hi_price symbol
. dotplot weight, over(gear_ratio) m(none) mlabel(hi_price) mlabp(0) center
> xlabel(#5)
```



## > Example 6

The following figure is included mostly for aesthetic reasons. It also demonstrates dotplot's ability to cope with even very large datasets. The sample size for each variable is 10,000 , so it may take a long time to print.

```
. clear all
. set seed 123456789
. set obs }1000
. generate norm0 = rnormal()
. generate norm1 = rnormal() + 1
. generate norm2 = rnormal() + 2
. label variable norm0 "N(0,1)"
. label variable norm1 "N(1,1)"
. label variable norm2 "N(2,1)"
. dotplot norm0 norm1 norm2
```



## Stored results

dotplot stores the following in $r()$ :
Scalars
$r(n x) \quad$ horizontal dot density
$r(n y) \quad$ vertical dot density

## Acknowledgments

dotplot was written by Peter Sasieni of the King's Clinical Trials Unit at King's College London, and Patrick Royston of the MrC Clinical Trials Unit, London, and coauthor of the Stata Press book Flexible Parametric Survival Analysis Using Stata: Beyond the Cox Model.

## Reference

Sasieni, P. D., and P. Royston. 1996. Dotplots. Journal of the Royal Statistical Society, Series C 45: 219-234. https://doi.org/10.2307/2986156.

## Title

dstdize - Direct and indirect standardization

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options for dstdize | Options for istdize | Remarks and examples | Stored results |
| Methods and formulas | Acknowledgments | References | Also see |

## Description

dstdize produces standardized rates, a weighted average of the stratum-specific rates.
istdize produces indirectly standardized rates that are appropriate when the stratum-specific rates for the population being studied are either unavailable or unreliable.
istdize also calculates a point estimate and exact confidence interval for the study population's standardized mortality ratio (SMR) or the standardized incidence ratio (SIR).

## Quick start

## Direct standardization

Use reference population saved in mypop.dta to standardize v 1 with stratum identifier svar and stratum size v2 for catvar dstdize v1 v2 svar, by (catvar) using(mypop)

Same as above, but with reference population in memory where catvar $=1$ dstdize v1 v2 svar, by (catvar) base(1)

Same as above, but with reference population in memory where catvar = "nation"
dstdize v1 v2 svar, by(catvar) base("nation")

## Indirect standardization

Use population cases and size saved in cases and pop to standardize study cases v3 and stratum size v4 at each level of svar
istdize v3 v4 svar using mypop.dta, popvars(cases pop)
Same as above, but standardize subpopulations identified by levels of catvar
isdize v3 v4 svar using mypop.dta, popvars (cases pop) by (catvar)
Same as above, but standardize by population stratum-specific and crude rates saved in srate and crate and display summary of standard population
isdize v3 v4 svar using mypop.dta, rate(srate crate) by (catvar) print
Same as above, but indicate that the crude population rate is 0.01
isdize v3 v4 svar using mypop.dta, rate(srate .01) by (catvar) print

## Menu

dstdize
Statistics $>$ Epidemiology and related $>$ Other $>$ Direct standardization
istdize
Statistics $>$ Epidemiology and related $>$ Other $>$ Indirect standardization

## Syntax

Direct standardization dstdize charvar popvar stratavars [if] [in], by (groupvars) [dstdize_options]

Indirect standardization
istdize casevar popvar $_{s}$ stratavars [if] [in] using filename,
$\left\{\right.$ popvars $^{\left.\left(\text {casevar }_{p} \text { popvar }_{p}\right) \mid \operatorname{rate}\left(\text { ratevar }_{p}\left\{\# \mid \text { crudevar }_{p}\right\}\right)\right\}}$
[istdize_options]
charvar is the characteristic to be standardized across different subpopulations identified by groupvars. popvar defines the weights used in standardization.
stratavars defines the strata across which the weights are to be averaged in dstdize. For istdize, stratavars defines the strata for which casevar $_{s}$ is measured.
casevar $_{s}$ is the variable name for the study population's number of cases. If by (groupvars) is specified, casevar ${ }_{s}$ must be constant or missing within each group defined by combinations of groupvars.
popvar ${ }_{s}$ identifies the number of subjects in each strata in the study population.
filename must be a Stata dataset and contain popvar and stratavars.
dstdize_options

## Main

* by (groupvars)
using (filename)
base(\#| string)
level(\#)
Options
saving (filename)
format ( $\%$ fmt)
print
nores

Description
study populations
use standard population from Stata dataset use standard population from a value of grouping variable set confidence level; default is level(95)
save computed standard population distribution as a Stata dataset final summary table display format; default is $\% 10.0 \mathrm{~g}$ include table summary of standard population in output suppress storing results in $r()$
*by (groupvars) is required.
istdize_options Description
Main

* popvars (casevar popvar $_{p}$ )
for standard population, casevar $_{p}$ is number of cases and popvar ${ }_{p}$ is number of individuals
${ }^{*}$ rate ( ratevar $_{p}\left\{\# \mid\right.$ crudevar $\left._{p}\right\}$ ) ratevar $_{p}$ is stratum-specific rates and \# or crudevar ${ }_{p}$ is the crude case rate value or variable
level(\#)
set confidence level; default is level (95)
Options
by (groupvars)
variables identifying study populations
format ( $\%$ fmt)
final summary table display format; default is $\% 10.0 \mathrm{~g}$
print include table summary of standard population in output
*Either popvars $\left(\right.$ casevar $_{p}$ popvar $\left._{p}\right)$ or rate $\left(\operatorname{ratevar}_{p}\left\{\# \mid\right.\right.$ crudevar $\left.\left._{p}\right\}\right)$ must be specified.
collect is allowed with dstdize and istdize; see [U] 11.1.10 Prefix commands.


## Options for dstdize

$\qquad$ Main
by (groupvars) is required for the dstdize command; it specifies the variables identifying the study populations. If base () is also specified, there must be only one variable in the by () group. If you do not have a variable for this option, you can generate one by using something like generate newvar=1 and then use newvar as the argument to this option.
using(filename) or base (\#|string) may be used to specify the standard population. You may not specify both options. using (filename) supplies the name of a .dta file containing the standard population. The standard population must contain the popvar and the stratavars. If using() is not specified, the standard population distribution will be obtained from the data. base (\#|string) lets you specify one of the values of groupvar-either a numeric value or a string-to be used as the standard population. If neither base() nor using() is specified, the entire dataset is used to determine an estimate of the standard population.
level(\#) specifies the confidence level, as a percentage, for a confidence interval of the adjusted rate. The default is level (95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals.

Options
saving (filename) saves the computed standard population distribution as a Stata dataset that can be used in further analyses.
format (\% fmt) specifies the format in which to display the final summary table. The default is $\% 10.0 \mathrm{~g}$.
print includes a table summary of the standard population before displaying the study population results.
nores suppresses storing results in $r()$. This option is seldom specified. Some results are stored in matrices. If there are more groups than can fit in a matrix, dstdize will report the "unable to allocate matrix" error message. In this case, you must specify nores. The nores option does not change how results are calculated but specifies that results need not be left behind for use by other programs.

## Options for istdize

$\qquad$ Main
popvars (casevar popvar $_{p}$ ) or rate (ratevar $\left\{\# \mid\right.$ crudevar $\left._{p}\right\}$ ) must be specified with istdize. Only one of these two options is allowed. These options are used to describe the standard population's data.
With popvars (casevar popvar $_{p}$ ), casevar $_{p}$ records the number of cases (deaths) for each stratum in the standard population, and popvar ${ }_{p}$ records the total number of individuals in each stratum (individuals at risk).
With rate (ratevar ${ }_{p}\left\{\# \mid\right.$ crudevar $\left._{p}\right\}$ ), ratevar $_{p}$ contains the stratum-specific rates. \#|crudevar ${ }_{p}$ specifies the crude case rate either by a variable name or by the crude case rate value. If a crude rate variable is used, it must be the same for all observations, although it could be missing for some.
level(\#) specifies the confidence level, as a percentage, for a confidence interval of the adjusted rate. The default is level (95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals.
by (groupvars) specifies variables identifying study populations when more than one exists in the data. If this option is not specified, the entire study population is treated as one group.
format (\%fmt) specifies the format in which to display the final summary table. The default is $\% 10.0 \mathrm{~g}$.
print outputs a table summary of the standard population before displaying the study population results.

## Remarks and examples

Remarks are presented under the following headings:

## Direct standardization

Indirect standardization
In epidemiology and other fields, you will often need to compare rates for some characteristic across different populations. These populations often differ on factors associated with the characteristic under study; thus, directly comparing overall rates may be misleading.

See van Belle et al. (2004, 642-684), Fleiss, Levin, and Paik (2003, chap. 19), or Kirkwood and Sterne (2003, chap. 25) for a discussion of direct and indirect standardization.

## Direct standardization

The direct method of adjusting for differences among populations involves computing the overall rates that would result if, instead of having different distributions, all populations had the same standard distribution. The standardized rate is defined as a weighted average of the stratum-specific rates, with the weights taken from the standard distribution. Direct standardization may be applied only when the specific rates for a given population are available.
dstdize generates adjusted summary measures of occurrence, which can be used to compare prevalence, incidence, or mortality rates between populations that may differ on certain characteristics (for example, age, gender, race). These underlying differences may affect the crude prevalence, mortality, or incidence rates.
> Example 1
We have data (Rothman 1986, 42) on mortality rates for Sweden and Panama for 1962, and we wish to compare mortality in these two countries:

| . use https://www.stata-press.com/data/r18/mortality (1962 Mortality, Sweden \& Panama) <br> . describe |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: |
| Contains data from https://www.stata-press.com/data/r18/mortality.dta |  |  |  |  |
| Observations: |  | 6 |  | 1962 Mortality, Sweden \& Panama |
| Variables: |  | 4 |  | 14 Apr 2022 16:18 |
| Variable name | Storage type | Display format | Value label | Variable label |
| nation | str6 | \%9s |  | Nation |
| age_category | byte | \%9.0g | age_lbl | Age category |
| population | float | \%10.0gc |  | Population in age category |
| deaths | float | \%9.0gc |  | Deaths in age category |

. list, sepby(nation) abbrev(12) divider

| nation | age_category | population | deaths |
| :--- | ---: | ---: | ---: |
| Sweden | $0-29$ | 3145000 | 3,523 |
| Sweden | $30-59$ | 3057000 | 10,928 |
| Sweden | $60+$ | 1294000 | 59,104 |
| Panama | $0-29$ | 741,000 | 3,904 |
| Panama | $30-59$ | 275,000 | 1,421 |
| Panama | $60+$ | 59,000 | 2,456 |

We divide the total number of cases in the population by the population to obtain the crude rate:
. collapse (sum) pop deaths, by (nation)
. list, abbrev(10) divider

|  | nation | population | deaths |
| :--- | :--- | ---: | ---: |
| 1. | Panama | 1075000 | 7,781 |
| 2. | Sweden | 7496000 | 73,555 |

. generate crude = deaths/pop
. list, abbrev(10) divider

| nation | population | deaths | crude |
| :---: | ---: | ---: | ---: |
| Panama | 1075000 | 7,781 | .0072381 |
| Sweden | 7496000 | 73,555 | .0098126 |

If we examine the total number of deaths in the two nations, the total crude mortality rate in Sweden is higher than that in Panama. From the original data, we see one possible explanation: Swedes are older than Panamanians, making direct comparison of the mortality rates difficult.

Direct standardization lets us remove the distortion caused by the different age distributions. The adjusted rate is defined as the weighted sum of the crude rates, where the weights are given by the standard distribution. Suppose that we wish to standardize these mortality rates to the following age distribution:

```
. use https://www.stata-press.com/data/r18/1962, clear
(Standard population distribution)
. list, abbrev(12) divider
```

|  | age_category | population |
| ---: | ---: | ---: |
| 1. | $0-29$ | .35 |
| 2. | $30-59$ | .35 |
| 3. | $60+$ | .3 |

. save 1962
file 1962.dta saved

If we multiply the above weights for the age strata by the crude rate for the corresponding age category, the sum gives us the standardized rate.

```
. use https://www.stata-press.com/data/r18/mortality
(1962 Mortality, Sweden & Panama)
```

```
. generate crude=deaths/pop
. drop pop
. merge m:1 age_cat using 1962
\begin{tabular}{lrl} 
Result & Number of obs & \\
\hline Not matched & 0 & \\
Matched & 6 & (_merge==3)
\end{tabular}
. list, sepby(age_category) abbrev(12)
```

|  | nation | age_category | deaths | crude | population | _merge |
| :--- | :--- | ---: | ---: | ---: | ---: | ---: |
| 1. | Sweden | $0-29$ | 3,523 | .0011202 | .35 | Matched (3) |
| 2. | Panama | $0-29$ | 3,904 | .0052686 | .35 | Matched (3) |
|  | 3. | Panama | $30-59$ | 1,421 | .0051673 | .35 |
| 4. Matched (3) |  |  |  |  |  |  |
| 4. | Sweden | $30-59$ | 10,928 | .0035747 | .35 | Matched (3) |
|  | Panama | $60+$ | 2,456 | .0416271 | .3 | Matched (3) |
| 6. | Sweden | $60+$ | 59,104 | .0456754 | .3 | Matched (3) |

. generate product = crude*pop
. by nation, sort: egen adj_rate = sum(product)
. drop _merge
. list, sepby(nation)

|  | nation | age_ca~y | deaths | crude | popula~n | product | adj_rate |
| :--- | :--- | ---: | ---: | ---: | ---: | ---: | ---: |
|  | 1. | Panama | $60+$ | 2,456 | .0416271 | .3 | .0124881 |
| 2. | Panama | $30-59$ | 1,421 | .0051673 | .0161407 |  |  |
| 3. | Panama | $0-29$ | 3,904 | .0052686 | .35 | .0018085 | .0161407 |
|  |  |  |  |  | .001844 | .0161407 |  |
|  | Sweden | $60+$ | 59,104 | .0456754 | .3 | .0137026 | .0153459 |
| 5. | Sweden | $30-59$ | 10,928 | .0035747 | .35 | .0012512 | .0153459 |
| 6. | Sweden | $0-29$ | 3,523 | .0011202 | .35 | .0003921 | .0153459 |

Comparing the standardized rates indicates that the Swedes have a slightly lower mortality rate.

To perform the above analysis with dstdize, type

```
. use https://www.stata-press.com/data/r18/mortality, clear
(1962 Mortality, Sweden & Panama)
. dstdize deaths pop age_cat, by(nation) using(1962)
Direct standardization
```

| Stratum | Pop. | Cases | nadjus Pop. dist |  | Std. <br> pop. <br> dist. | $s * P$ |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 0-29 | 741,000 | 3,904 | 0.689 | 0.0053 | 0.350 | 0.0018 |
| 30-59 | 275,000 | 1,421 | 0.256 | 0.0052 | 0.350 | 0.0018 |
| 60+ | 59,000 | 2,456 | 0.055 | 0.0416 | 0.300 | 0.0125 |
| Total: 1,075,000 |  | 7,781 |  |  |  |  |
| Note: s*P is Stratum rate multiplied by Std. pop. dist. |  |  |  |  |  |  |
|  |  | Adjusted cases $=17,351.2$ |  |  |  |  |
|  |  | Crude |  |  | te | 0.0072 |
|  |  |  |  | djusted | te $=$ | 0.0161 |
|  |  | 95\% conf. interval: [0.0156, 0.0166] |  |  |  |  |


| Stratum | Pop. | Cases | adjus Pop. dist. | ed $\qquad$ <br> Stratum rate | Std. pop. dist. | s* |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 0-29 | 3,145,000 | 3,523 | 0.420 | 0.0011 | 0.350 | 0.0004 |
| 30-59 | 3,057,000 | 10,928 | 0.408 | 0.0036 | 0.350 | 0.0013 |
| 60+ | 1,294,000 | 59,104 | 0.173 | 0.0457 | 0.300 | 0.0137 |
| Total: 7,496,000 73,555 | 7,496,000 | 73,555 |  |  |  |  |
| Note: $s * P$ is Stratum rate multiplied by Std. pop. dist.$\begin{array}{rrr} \text { Adjusted cases }= & 115,032.5 \\ \text { Crude rate }= & 0.0098 \\ \text { Adjusted rate }= & 0.0153 \\ 95 \% \text { conf. interval: [0.0152, } 0.0155] \end{array}$ |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

Summary of study populations

| nation | N | Crude rate | Adjusted rate | $[95 \%$ conf . interval] |  |
| :--- | ---: | ---: | ---: | ---: | ---: |
| Panama | $1,075,000$ | 0.007238 | 0.016141 | 0.015645 | 0.016637 |
| Sweden | $7,496,000$ | 0.009813 | 0.015346 | 0.015235 | 0.015457 |

The summary table above lets us make a quick inspection of the results within the study populations, and the detail tables give the behavior among the strata within the study populations.

## Example 2

We have individual-level data on persons in four cities over several years. Included in the data is a variable indicating whether the person has high blood pressure, together with information on the person's age, sex, and race. We wish to obtain standardized high blood pressure rates for each city for 1990 and 1992, using, as the standard, the age, sex, and race distribution of the four cities and two years combined.

Our dataset contains

| Contains data from https://www.stata-press.com/data/r18/hbp.dta Observations:$1,130$ |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: |
| Variable name | Storage type | Display format | Value label | Variable label |
| id | str10 | \%10s |  | Record identification number |
| city | byte | \%8.0g |  | City |
| year | int | \%8.0g |  | Year |
| sex | byte | \%8.0g | sexfmt | Sex |
| age_group | byte | \%8.0g | agefmt | Age group |
| race | byte | \%8.0g | racefmt | Race |
| hbp | byte | \%8.0g | yn | High blood pressure |

Sorted by:
The dstdize command is designed to work with aggregate data but will work with individuallevel data only if we create a variable recording the population represented by each observation. For individual-level data, this is one:

```
. generate pop = 1
```

On the next page, we specify print to obtain a listing of the standard population and level (90) to request $90 \%$ rather than $95 \%$ confidence intervals. Typing if year==1990| year==1992 restricts the data to the two years for both summary tables and the standard population.
. dstdize hbp pop age race sex if year==1990 | year==1992, by(city year) print > level (90)

Standard population

|  |  |  | Pop. | Dist. |
| ---: | ---: | ---: | ---: | ---: |
|  |  |  |  |  |
| $15-19$ | Black | Female | 35 | 0.077 |
| $15-19$ | Black | Male | 44 | 0.097 |
| $15-19$ | Hispanic | Female | 5 | 0.011 |
| $15-19$ | Hispanic | Male | 10 | 0.022 |
| $15-19$ | White | Female | 7 | 0.015 |
| $15-19$ | White | Male | 5 | 0.011 |
| $20-24$ | Black | Female | 43 | 0.095 |
| $20-24$ | Black | Male | 67 | 0.147 |
| $20-24$ | Hispanic | Female | 14 | 0.031 |
| $20-24$ | Hispanic | Male | 13 | 0.029 |
| $20-24$ | White | Female | 4 | 0.009 |
| $20-24$ | White | Male | 21 | 0.046 |
| $25-29$ | Black | Female | 17 | 0.037 |
| $25-29$ | Black | Male | 44 | 0.097 |
| $25-29$ | Hispanic | Female | 7 | 0.015 |
| $25-29$ | Hispanic | Male | 13 | 0.029 |
| $25-29$ | White | Female | 9 | 0.020 |
| $25-29$ | White | Male | 16 | 0.035 |
| $30-34$ | Black | Female | 16 | 0.035 |
| $30-34$ | Black | Male | 32 | 0.070 |
| $30-34$ | Hispanic | Female | 2 | 0.004 |
| $30-34$ | Hispanic | Male | 3 | 0.007 |
| $30-34$ | White | Female | 5 | 0.011 |
| $30-34$ | White | Male | 23 | 0.051 |

Total:
455
(6 observations excluded because of missing values)
Direct standardization

| St |  |  | Pop. | Unadjusted-_ |  |  | Std. pop. dist. | $s * P$ |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  | Cases | Pop. dist. | Stratum rate |  |  |
| 15-19 | Black | Female |  | 6 | 2 | 0.128 | 0.3333 | 0.077 | 0.0256 |
| 15-19 | Black | Male | 6 | 0 | 0.128 | 0.0000 | 0.097 | 0.0000 |
| 15-19 | Hispanic | Male | 1 | 0 | 0.021 | 0.0000 | 0.022 | 0.0000 |
| 20-24 | Black | Female | 3 | 0 | 0.064 | 0.0000 | 0.095 | 0.0000 |
| 20-24 | Black | Male | 11 | 0 | 0.234 | 0.0000 | 0.147 | 0.0000 |
| 25-29 | Black | Female | 4 | 0 | 0.085 | 0.0000 | 0.037 | 0.0000 |
| 25-29 | Black | Male | 6 | 1 | 0.128 | 0.1667 | 0.097 | 0.0161 |
| 25-29 | Hispanic | Female | 2 | 0 | 0.043 | 0.0000 | 0.015 | 0.0000 |
| 25-29 | White | Female | 1 | 0 | 0.021 | 0.0000 | 0.020 | 0.0000 |
| 30-34 | Black | Female | 1 | 0 | 0.021 | 0.0000 | 0.035 | 0.0000 |
| 30-34 | Black | Male | 6 | 0 | 0.128 | 0.0000 | 0.070 | 0.0000 |

Total: 47
Note: s*P is Stratum rate multiplied by Std. pop. dist.
Adjusted cases $=2.0$
Crude rate $=0.0638$
Adjusted rate $=0.0418$
$90 \%$ conf. interval: [0.0074, 0.0761]

| Stratum |  |  | Pop. | Cases | nadjus Pop. dist. | ed $\qquad$ <br> Stratum <br> rate | Std. pop. dist. | s*P |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 15-19 | Black | Female | 3 | 0 | 0.054 | 0.0000 | 0.077 | 0.0000 |
| 15-19 | Black | Male | 9 | 0 | 0.161 | 0.0000 | 0.097 | 0.0000 |
| 15-19 | Hispanic | Male | 1 | 0 | 0.018 | 0.0000 | 0.022 | 0.0000 |
| 20-24 | Black | Female | 7 | 0 | 0.125 | 0.0000 | 0.095 | 0.0000 |
| 20-24 | Black | Male | 9 | 0 | 0.161 | 0.0000 | 0.147 | 0.0000 |
| 20-24 | Hispanic | Female | 1 | 0 | 0.018 | 0.0000 | 0.031 | 0.0000 |
| 25-29 | Black | Female | 2 | 0 | 0.036 | 0.0000 | 0.037 | 0.0000 |
| 25-29 | Black | Male | 11 | 1 | 0.196 | 0.0909 | 0.097 | 0.0088 |
| 25-29 | Hispanic | Male | 1 | 0 | 0.018 | 0.0000 | 0.029 | 0.0000 |
| 30-34 | Black | Female | 7 | 0 | 0.125 | 0.0000 | 0.035 | 0.0000 |
| 30-34 | Black | Male | 4 | 0 | 0.071 | 0.0000 | 0.070 | 0.0000 |
| 30-34 | White | Female | 1 | 0 | 0.018 | 0.0000 | 0.011 | 0.0000 |

Total:
56
1
Note: s*P is Stratum rate multiplied by Std. pop. dist.
Adjusted cases $=0.5$
Crude rate $=0.0179$
Adjusted rate $=0.0088$
90\% conf. interval: [0.0000, 0.0226]

| Stratum |  |  | Pop. | Unadjusted_   <br> Pop. Stratum  <br> Cases dist. rate  |  |  | Std. pop. dist | s*P |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 15-19 | Black | Female | 5 | 0 | 0.078 | 0.0000 | 0.077 | 0.0000 |
| 15-19 | Black | Male | 7 | 1 | 0.109 | 0.1429 | 0.097 | 0.0138 |
| 15-19 | Hispanic | Male | 1 | 0 | 0.016 | 0.0000 | 0.022 | 0.0000 |
| 20-24 | Black | Female | 7 | 1 | 0.109 | 0.1429 | 0.095 | 0.0135 |
| 20-24 | Black | Male | 8 | 0 | 0.125 | 0.0000 | 0.147 | 0.0000 |
| 20-24 | Hispanic | Female | 5 | 0 | 0.078 | 0.0000 | 0.031 | 0.0000 |
| 20-24 | Hispanic | Male | 2 | 0 | 0.031 | 0.0000 | 0.029 | 0.0000 |
| 20-24 | White | Male | 2 | 0 | 0.031 | 0.0000 | 0.046 | 0.0000 |
| 25-29 | Black | Female | 3 | 0 | 0.047 | 0.0000 | 0.037 | 0.0000 |
| 25-29 | Black | Male | 9 | 0 | 0.141 | 0.0000 | 0.097 | 0.0000 |
| 25-29 | Hispanic | Female | 2 | 0 | 0.031 | 0.0000 | 0.015 | 0.0000 |
| 25-29 | White | Female | 1 | 0 | 0.016 | 0.0000 | 0.020 | 0.0000 |
| 25-29 | White | Male | 2 | 1 | 0.031 | 0.5000 | 0.035 | 0.0176 |
| 30-34 | Black | Female | 1 | 0 | 0.016 | 0.0000 | 0.035 | 0.0000 |
| 30-34 | Black | Male | 5 | 0 | 0.078 | 0.0000 | 0.070 | 0.0000 |
| 30-34 | Hispanic | Female | 2 | 0 | 0.031 | 0.0000 | 0.004 | 0.0000 |
| 30-34 | White | Female | 1 | 0 | 0.016 | 0.0000 | 0.011 | 0.0000 |
| 30-34 | White | Male | 1 | 0 | 0.016 | 0.0000 | 0.051 | 0.0000 |
| Total: 64 |  |  |  |  |  |  |  |  |
| Note: s*P is Stratum rate multiplied by Std. pop. dist.$\begin{array}{rr} \text { Adjusted cases } & =2.9 \\ \text { Crude rate } & =0.0469 \\ \text { Adjusted rate } & =0.0449 \\ 90 \% \text { conf. interval: [0.0091, } & 0.0807] \end{array}$ |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |


| Stratum |  |  | Pop. | Conadjusted—_  <br> Pop. Stratum <br> Cases dist. rate  |  |  | $\begin{aligned} & \text { Std. } \\ & \text { pop. } \\ & \text { dist. } \end{aligned}$ | $s * P$ |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 15-19 | Black | Female | 1 | 0 | 0.015 | 0.0000 | 0.077 | 0.0000 |
| 15-19 | Black | Male | 5 | 0 | 0.075 | 0.0000 | 0.097 | 0.0000 |
| 15-19 | Hispanic | Female | 3 | 0 | 0.045 | 0.0000 | 0.011 | 0.0000 |
| 15-19 | Hispanic | Male | 1 | 0 | 0.015 | 0.0000 | 0.022 | 0.0000 |
| 15-19 | White | Male | 1 | 0 | 0.015 | 0.0000 | 0.011 | 0.0000 |
| 20-24 | Black | Female | 8 | 0 | 0.119 | 0.0000 | 0.095 | 0.0000 |
| 20-24 | Black | Male | 11 | 0 | 0.164 | 0.0000 | 0.147 | 0.0000 |
| 20-24 | Hispanic | Female | 6 | 0 | 0.090 | 0.0000 | 0.031 | 0.0000 |
| 20-24 | Hispanic | Male | 4 | 2 | 0.060 | 0.5000 | 0.029 | 0.0143 |
| 20-24 | White | Female | 1 | 0 | 0.015 | 0.0000 | 0.009 | 0.0000 |
| 20-24 | White | Male | 2 | 0 | 0.030 | 0.0000 | 0.046 | 0.0000 |
| 25-29 | Black | Female | 2 | 0 | 0.030 | 0.0000 | 0.037 | 0.0000 |
| 25-29 | Black | Male | 3 | 0 | 0.045 | 0.0000 | 0.097 | 0.0000 |
| 25-29 | Hispanic | Female | 2 | 0 | 0.030 | 0.0000 | 0.015 | 0.0000 |
| 25-29 | Hispanic | Male | 4 | 0 | 0.060 | 0.0000 | 0.029 | 0.0000 |
| 25-29 | White | Female | 4 | 0 | 0.060 | 0.0000 | 0.020 | 0.0000 |
| 25-29 | White | Male | 2 | 0 | 0.030 | 0.0000 | 0.035 | 0.0000 |
| 30-34 | Black | Female | 1 | 0 | 0.015 | 0.0000 | 0.035 | 0.0000 |
| 30-34 | Black | Male | 2 | 0 | 0.030 | 0.0000 | 0.070 | 0.0000 |
| 30-34 | Hispanic | Male | 1 | 0 | 0.015 | 0.0000 | 0.007 | 0.0000 |
| 30-34 | White | Female | 2 | 0 | 0.030 | 0.0000 | 0.011 | 0.0000 |
| 30-34 | White | Male | 1 | 0 | 0.015 | 0.0000 | 0.051 | 0.0000 |

Total: $67 \quad 2$
Note: s*P is Stratum rate multiplied by Std. pop. dist.
Adjusted cases $=1.0$
Crude rate $=0.0299$
Adjusted rate $=0.0143$
90\% conf. interval: [0.0025, 0.0260]

| city year = 31990 |  |  | Pop. | Unadjusted-_ |  |  | Std. pop. dist. | s*P |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  | Cases | $\begin{array}{r} \text { Pop. } \\ \text { dist. } \end{array}$ | Stratum rate |  |  |
| 15-19 | Black | Female |  | 3 | 0 | 0.043 | 0.0000 | 0.077 | 0.0000 |
| 15-19 | Black | Male | 1 | 0 | 0.014 | 0.0000 | 0.097 | 0.0000 |
| 15-19 | Hispanic | Female | 1 | 0 | 0.014 | 0.0000 | 0.011 | 0.0000 |
| 15-19 | White | Female | 3 | 0 | 0.043 | 0.0000 | 0.015 | 0.0000 |
| 15-19 | White | Male | 1 | 0 | 0.014 | 0.0000 | 0.011 | 0.0000 |
| 20-24 | Black | Female | 1 | 0 | 0.014 | 0.0000 | 0.095 | 0.0000 |
| 20-24 | Black | Male | 9 | 0 | 0.130 | 0.0000 | 0.147 | 0.0000 |
| 20-24 | Hispanic | Male | 3 | 0 | 0.043 | 0.0000 | 0.029 | 0.0000 |
| 20-24 | White | Female | 2 | 0 | 0.029 | 0.0000 | 0.009 | 0.0000 |
| 20-24 | White | Male | 8 | 1 | 0.116 | 0.1250 | 0.046 | 0.0058 |
| 25-29 | Black | Female | 1 | 0 | 0.014 | 0.0000 | 0.037 | 0.0000 |
| 25-29 | Black | Male | 8 | 3 | 0.116 | 0.3750 | 0.097 | 0.0363 |
| 25-29 | Hispanic | Male | 4 | 0 | 0.058 | 0.0000 | 0.029 | 0.0000 |
| 25-29 | White | Female | 1 | 0 | 0.014 | 0.0000 | 0.020 | 0.0000 |
| 25-29 | White | Male | 6 | 0 | 0.087 | 0.0000 | 0.035 | 0.0000 |
| 30-34 | Black | Male | 6 | 2 | 0.087 | 0.3333 | 0.070 | 0.0234 |
| 30-34 | White | Male | 11 | 5 | 0.159 | 0.4545 | 0.051 | 0.0230 |

Adjusted cases $=6.1$
Crude rate $=0.1594$
Adjusted rate $=0.0885$
90\% conf. interval: [0.0501, 0.1268]

| -> city year = 31992 |  |  | Pop. | Unadjusted-_ |  |  | Std. pop. dist. | s*P |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  | Cases | Pop. dist. | Stratum rate |  |  |
| 15-19 | Black | Female |  | 2 | 0 | 0.054 | 0.0000 | 0.077 | 0.0000 |
| 15-19 | Hispanic | Male | 3 | 0 | 0.081 | 0.0000 | 0.022 | 0.0000 |
| 15-19 | White | Female | 2 | 0 | 0.054 | 0.0000 | 0.015 | 0.0000 |
| 15-19 | White | Male | 1 | 0 | 0.027 | 0.0000 | 0.011 | 0.0000 |
| 20-24 | Black | Male | 3 | 0 | 0.081 | 0.0000 | 0.147 | 0.0000 |
| 20-24 | Hispanic | Female | 1 | 0 | 0.027 | 0.0000 | 0.031 | 0.0000 |
| 20-24 | Hispanic | Male | 3 | 0 | 0.081 | 0.0000 | 0.029 | 0.0000 |
| 20-24 | White | Female | 1 | 0 | 0.027 | 0.0000 | 0.009 | 0.0000 |
| 20-24 | White | Male | 6 | 1 | 0.162 | 0.1667 | 0.046 | 0.0077 |
| 25-29 | Hispanic | Male | 1 | 0 | 0.027 | 0.0000 | 0.029 | 0.0000 |
| 25-29 | White | Male | 5 | 1 | 0.135 | 0.2000 | 0.035 | 0.0070 |
| 30-34 | Black | Male | 1 | 0 | 0.027 | 0.0000 | 0.070 | 0.0000 |
| 30-34 | White | Male | 8 | 5 | 0.216 | 0.6250 | 0.051 | 0.0316 |

Note: s*P is Stratum rate multiplied by Std. pop. dist.
Adjusted cases $=1.7$
Crude rate $=0.1892$
Adjusted rate $=0.0463$
90\% conf. interval: [0.0253, 0.0674]

| $\rightarrow$ city year $=51990$ |  |  | Pop. | Unadjusted-_ |  |  | Std. pop. dist. | s*P |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  | Cases | Pop. dist. | Stratum rate |  |  |
| 15-19 | Black | Female |  | 9 | 0 | 0.196 | 0.0000 | 0.077 | 0.0000 |
| 15-19 | Black | Male | 7 | 0 | 0.152 | 0.0000 | 0.097 | 0.0000 |
| 15-19 | Hispanic | Male | 1 | 0 | 0.022 | 0.0000 | 0.022 | 0.0000 |
| 15-19 | White | Male | 1 | 0 | 0.022 | 0.0000 | 0.011 | 0.0000 |
| 20-24 | Black | Female | 4 | 0 | 0.087 | 0.0000 | 0.095 | 0.0000 |
| 20-24 | Black | Male | 6 | 0 | 0.130 | 0.0000 | 0.147 | 0.0000 |
| 20-24 | Hispanic | Female | 1 | 0 | 0.022 | 0.0000 | 0.031 | 0.0000 |
| 25-29 | Black | Female | 3 | 1 | 0.065 | 0.3333 | 0.037 | 0.0125 |
| 25-29 | Black | Male | 5 | 0 | 0.109 | 0.0000 | 0.097 | 0.0000 |
| 25-29 | Hispanic | Female | 1 | 0 | 0.022 | 0.0000 | 0.015 | 0.0000 |
| 25-29 | White | Female | 2 | 1 | 0.043 | 0.5000 | 0.020 | 0.0099 |
| 30-34 | Black | Female | 2 | 0 | 0.043 | 0.0000 | 0.035 | 0.0000 |
| 30-34 | Black | Male | 3 | 0 | 0.065 | 0.0000 | 0.070 | 0.0000 |
| 30-34 | White | Male | 1 | 0 | 0.022 | 0.0000 | 0.051 | 0.0000 |
| Total: |  |  | 46 | 2 |  |  |  |  |
| Note: $s * P$ is Stratum rate multiplied by Std. pop. dist. |  |  |  |  |  |  |  |  |
|  |  |  |  | Adjusted cases $=1.0$ |  |  |  |  |
|  |  |  |  | Crude rate $=0.0435$ |  |  |  |  |
|  |  |  |  |  |  | Adjuste | rate | 0.0223 |
| 0\% conf. interval: [0.0020, 0.0426] |  |  |  |  |  |  |  |  |



## Indirect standardization

Standardization of rates can be performed via the indirect method whenever the stratum-specific rates are either unknown or unreliable. If the stratum-specific rates are known, the direct standardization method is preferred.

To apply the indirect method, you must have the following information:

- The observed number of cases in each population to be standardized, $O$. For example, if deathrates in two states are being standardized using the US deathrate for the same period, you must know the total number of deaths in each state.
- The distribution across the various strata for the population being studied, $n_{1}, \ldots, n_{k}$. If you are standardizing the deathrate in the two states, adjusting for age, you must know the number of individuals in each of the $k$ age groups.
- The stratum-specific rates for the standard population, $p_{1}, \ldots, p_{k}$. For example, you must have the US deathrate for each stratum (age group).
- The crude rate of the standard population, $C$. For example, you must have the uS mortality rate for the year.

The indirect adjusted rate is then

$$
R_{\text {indirect }}=C \frac{O}{E}
$$

where $E$ is the expected number of cases (deaths) in each population. See Methods and formulas for a more detailed description of calculations.

## > Example 3

This example is borrowed from Kahn and Sempos (1989, 95-105). We want to compare 1970 mortality rates in California and Maine, adjusting for age. Although we have age-specific population counts for the two states, we lack age-specific deathrates. Direct standardization is not feasible here. We can use the US population census data for the same year to produce indirectly standardized rates for these two states.

From the US census, the standard population for this example was entered into Stata and saved in popkahn.dta.

```
. use https://www.stata-press.com/data/r18/popkahn, clear
. list age pop deaths rate, sep(4)
```

| age | population | deaths | rate |
| ---: | ---: | ---: | ---: |
| $<15$ | $57,900,000$ | 103,062 | .00178 |
| $15-24$ | $35,441,000$ | 45,261 | .00128 |
| $25-34$ | $24,907,000$ | 39,193 | .00157 |
| $35-44$ | $23,088,000$ | 72,617 | .00315 |
| $45-54$ | $23,220,000$ | 169,517 | .0073 |
| $55-64$ | $18,590,000$ | 308,373 | .01659 |
| $65-74$ | $12,436,000$ | 445,531 | .03583 |
| $75+$ | $7,630,000$ | 736,758 | .09656 |

The standard population contains for each age stratum the total number of individuals (pop) and both the age-specific mortality rate (rate) and the number of deaths. The standard population need not contain all three. If we have only the age-specific mortality rate, we can use the rate (ratevar ${ }_{p}$ crudevar $_{p}$ ) or rate (ratevar ${ }_{p} \#$ ) option, where crudevar $_{p}$ refers to the variable containing the total population's crude deathrate or \# is the total population's crude deathrate.

Now, let's look at the states' data (study population):

```
. use https://www.stata-press.com/data/r18/kahn
. list, sep(4)
```

| state | age | populat~n | death | st | death_~e |
| :--- | ---: | ---: | ---: | ---: | ---: |
| California | $<15$ | $5,524,000$ | 166,285 | 1 | .0016 |
| California | $15-24$ | $3,558,000$ | 166,285 | 1 | .0013 |
| California | $25-34$ | $2,677,000$ | 166,285 | 1 | .0015 |
| California | $35-44$ | $2,359,000$ | 166,285 | 1 | .0028 |
| California | $45-54$ | $2,330,000$ | 166,285 | 1 | .0067 |
| California | $55-64$ | $1,704,000$ | 166,285 | 1 | .0154 |
| California | $65-74$ | $1,105,000$ | 166,285 | 1 | .0328 |
| California | $75+$ | 696,000 | 166,285 | 1 | .0917 |
| Maine | $<15$ | 286,000 | 11,051 | 2 | .0019 |
| Maine | $15-24$ | 168,000 | . | 2 | .0011 |
| Maine | $25-34$ | 110,000 | . | 2 | .0014 |
| Maine | $35-44$ | 109,000 | . | 2 | .0029 |
| Maine | $45-54$ | 110,000 | . | 2 | .0069 |
| Maine | $55-64$ | 94,000 | . | 2 | .0173 |
| Maine | $65-74$ | 69,000 | . | 2 | .039 |
| Maine | $75+$ | 46,000 | . | 2 | .1041 |

For each state, the number of individuals in each stratum (age group) is contained in the pop variable. The death variable is the total number of deaths observed in the state during the year. It must have the same value for all observations in the group, as for California, or it could be missing in all but one observation per group, as for Maine.

To match these two datasets, the strata variables must have the same name in both datasets and ideally the same levels. If a level is missing from either dataset, that level will not be included in the standardization.

With kahn.dta in memory, we now execute the command. We will use the print option to obtain the standard population's summary table, and because we have both the standard population's age-specific count and deaths, we will specify the popvars (casevar popvar $_{p}$ ) option. Or, we could specify the rate (rate 0.00945 ) option because we know that 0.00945 is the US crude deathrate for 1970.
. istdize death pop age using https://www.stata-press.com/data/r18/popkahn,
> by (state) pop(deaths pop) print
Standard population

| Stratum | Rate |
| ---: | ---: |
| $<15$ | 0.00178 |
| $15-24$ | 0.00128 |
| $25-34$ | 0.00157 |
| $35-44$ | 0.00315 |
| $45-54$ | 0.00730 |
| $55-64$ | 0.01659 |
| $65-74$ | 0.03583 |
| $75+$ | 0.09656 |

Crude rate $=0.00945$
Indirect standardization

| -> state = <br> Stratum | fornia <br> Standard population rate | Observed population | Expected cases |
| :---: | :---: | :---: | :---: |
| <15 | 0.0018 | 5,524,000 | 9,832.72 |
| 15-24 | 0.0013 | 3,558,000 | 4,543.85 |
| 25-34 | 0.0016 | 2,677,000 | 4,212.46 |
| 35-44 | 0.0031 | 2,359,000 | 7,419.59 |
| 45-54 | 0.0073 | 2,330,000 | 17,010.10 |
| 55-64 | 0.0166 | 1,704,000 | 28,266.14 |
| 65-74 | 0.0358 | 1,105,000 | 39,587.63 |
| 75+ | 0.0966 | 696,000 | 67,206.23 |
| Total: |  | 19,953,000 | 178,078.73 |
| $\begin{array}{rrr}\text { Observed cases } & =166,285 \\ \text { SMR (Obs/Exp) } & = & 0.93 \\ \text { SMR exact } 95 \% \text { conf. interval: [0.9293, } & 0.9383]\end{array}$ |  |  |  |
|  |  |  |  |
| $\begin{array}{rr}\text { Crude rate } & =0.0083 \\ \text { Adjusted rate } & =0.0088 \\ \text { 95\% conf. interval: [0.0088, } & 0.0089]\end{array}$ |  |  |  |
|  |  |  |  |

```
-> state = Maine
```

| Stratum | Standard <br> population <br> rate | Observed <br> population | Expected <br> cases |
| ---: | ---: | ---: | ---: |
| $<15$ | 0.0018 | 286,000 | 509.08 |
| $15-24$ | 0.0013 | 168,000 | 214.55 |
| $25-34$ | 0.0016 | 110,000 | 173.09 |
| $35-44$ | 0.0031 | 109,000 | 342.83 |
| $45-54$ | 0.0073 | 110,000 | 803.05 |
| $55-64$ | 0.0166 | 94,000 | $1,559.28$ |
| $65-74$ | 0.0358 | 69,000 | $2,471.99$ |
| $75+$ | 0.0966 | 46,000 | $4,441.79$ |
| Total: |  | 992,000 | $10,515.67$ |
|  |  | Observed cases $=11,051$ |  |
|  |  | SMR (Obs/Exp) $=$ |  |
|  |  |  | 1.05 |

SMR exact 95\% conf. interval: [1.0314, 1.0707]
Crude rate $=0.0111$
Adjusted rate $=0.0099$
95\% conf. interval: [0.0097, 0.0101]

Summary of study populations, reporting rates

| state | Observed <br> cases | Crude rate | Adjusted <br> rate | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| California | 166,285 | 0.008334 | 0.008824 | 0.008782 | 0.008866 |
| Maine | 11,051 | 0.011140 | 0.009931 | 0.009747 | 0.010118 |
| Summary of study populations, |  |  |  |  |  |
| reporting SMRs |  |  |  |  |  |
| State | Ebserved <br> cases | Expected <br> cases | SMR | [95\% conf. interval] |  |
| California | 166,285 | $178,078.73$ |  |  |  |
| Maine | 11,051 | $10,515.67$ | 0.934 | 0.929290 | 0.938271 |

## Stored results

dstdize stores the following in $r()$ :
Scalars
$r(k) \quad$ number of populations
Macros

$$
\begin{array}{ll}
r(\text { by }) & \text { variable names specified in by }() \\
r(c \#) & \text { values of } r(b y) \text { for \#th group }
\end{array}
$$

Matrices

```
    r(se) 1\timesk vector of standard errors of adjusted rates
    r(ub_adj) 1 1 k vector of upper bounds of confidence intervals for adjusted rates
    r(lb_adj) 1 <k vector of lower bounds of confidence intervals for adjusted rates
    r(Nobs) 1 1 k vector of number of observations
    r(crude) 1 1 k vector of crude rates (*)
    r(adj) 1\timesk vector of adjusted rates (*)
```

$\left({ }^{*}\right)$ If, in a group, the number of observations is 0 , then 9 is stored for the corresponding crude and adjusted rates.
istdize stores the following in r() :
Scalars $r(k) \quad$ number of populations
Macros
$r$ (by) variable names specified in by ()
$r(c \#) \quad$ values of $r(b y)$ for \#th group
Matrices
r (cases_obs) $1 \times k$ vector of number of observed cases
r (cases_exp) $1 \times k$ vector of number of expected cases
$r$ (ub_adj) $\quad 1 \times k$ vector of upper bounds of confidence intervals for adjusted rates
$r$ (lb_adj) $1 \times k$ vector of lower bounds of confidence intervals for adjusted rates
$r$ (crude) $\quad 1 \times k$ vector of crude rates
$\mathrm{r}(\mathrm{adj}) \quad 1 \times k$ vector of adjusted rates
$\mathrm{r}(\mathrm{smr}) \quad 1 \times k$ vector of SMRs
$r$ (ub_smr) $1 \times k$ vector of upper bounds of confidence intervals for SMRs
$r$ (lb_smr) $\quad 1 \times k$ vector of lower bounds of confidence intervals for SMRs

## Methods and formulas

The directly standardized rate, $S_{\mathrm{R}}$, is defined by

$$
S_{\mathrm{R}}=\frac{\sum_{i=1}^{k} w_{i} R_{i}}{\sum_{i=1}^{k} w_{i}}
$$

(Rothman 1986, 44), where $R_{i}$ is the stratum-specific rate in stratum $i$ and $w_{i}$ is the weight for stratum $i$ derived from the standard population.

If $n_{i}$ is the population of stratum $i$, the standard error, $\operatorname{se}\left(S_{\mathrm{R}}\right)$, in stratified sampling for proportions (ignoring the finite population correction) is

$$
\operatorname{se}\left(S_{\mathrm{R}}\right)=\frac{1}{\sum w_{i}} \sqrt{\sum_{i=1}^{k} \frac{w_{i}^{2} R_{i}\left(1-R_{i}\right)}{n_{i}}}
$$

(Cochran 1977, 108), from which the confidence intervals are calculated.
For indirect standardization, define $O$ as the observed number of cases in each population to be standardized; $n_{1}, \ldots, n_{k}$ as the distribution across the various strata for the population being studied; $R_{1}, \ldots, R_{k}$ as the stratum-specific rates for the standard population; and $C$ as the crude rate of the standard population. The expected number of cases (deaths), $E$, in each population is obtained by applying the standard population stratum-specific rates, $R_{1}, \ldots, R_{k}$, to the study populations:

$$
E=\sum_{i=1}^{k} n_{i} R_{i}
$$

The indirectly adjusted rate is then

$$
R_{\text {indirect }}=C \frac{O}{E}
$$

and $O / E$ is the study population's SMR if death is the event of interest or the SIR for studies of disease (or other) incidence.

The exact confidence interval is calculated for each estimated SMR by assuming a Poisson process as described in Breslow and Day (1987, 69-71). These intervals are obtained by first calculating the upper and lower bounds for the confidence interval of the Poisson-distributed observed events, $O$-say, $L$ and $U$, respectively-and then computing $\operatorname{SMR}_{L}=L / E$ and $\operatorname{SMR}_{U}=U / E$.

## Acknowledgments

We gratefully acknowledge the collaboration of Dr. Joel A. Harrison, consultant; Dr. José Maria Pacheco of the Departamento de Epidemiologia, Faculdade de Saúde Pública/USP, Sao Paulo, Brazil; and Dr John L. Moran of the Queen Elizabeth Hospital, Woodville, Australia.

## References

Breslow, N. E., and N. E. Day. 1987. Statistical Methods in Cancer Research: Vol. 2—The Design and Analysis of Cohort Studies. Lyon: IARC.

Cochran, W. G. 1977. Sampling Techniques. 3rd ed. New York: Wiley.
Consonni, D. 2012. A command to calculate age-standardized rates with efficient interval estimation. Stata Journal 12: 688-701.

Fleiss, J. L., B. Levin, and M. C. Paik. 2003. Statistical Methods for Rates and Proportions. 3rd ed. New York: Wiley.

Forthofer, R. N., and E. S. Lee. 1995. Introduction to Biostatistics: A Guide to Design, Analysis, and Discovery. New York: Academic Press.

Juul, S., and M. Frydenberg. 2021. An Introduction to Stata for Health Researchers. 5th ed. College Station, TX: Stata Press.

Kahn, H. A., and C. T. Sempos. 1989. Statistical Methods in Epidemiology. New York: Oxford University Press. Kirkwood, B. R., and J. A. C. Sterne. 2003. Essential Medical Statistics. 2nd ed. Malden, MA: Blackwell.

Pagano, M., and K. Gauvreau. 2022. Principles of Biostatistics. 3rd ed. Boca Raton, FL: CRC Press.
Rothman, K. J. 1986. Modern Epidemiology. Boston: Little, Brown.
van Belle, G., L. D. Fisher, P. J. Heagerty, and T. S. Lumley. 2004. Biostatistics: A Methodology for the Health Sciences. 2nd ed. New York: Wiley.

## Also see

[R] Epitab - Tables for epidemiologists
[SVY] Direct standardization - Direct standardization of means, proportions, and ratios

## Title

dtable - Create a table of descriptive statistics

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Methods and formulas | Appendix |
| Acknowledgments | Reference | Also see |  |

## Description

The dtable command allows you to easily create a table of descriptive (summary) statistics, commonly known as "Table 1 ". Optionally, you can add $p$-values from test statistics comparing groups on summary statistics. You can also create a table complete with a title, notes, and more, and then export it to a variety of file types.

## Quick start

Create a table describing continuous variables $x 1$ and $x 2$ and factor variables $f 1$ and $f 2$ dtable x1 x2 i.f1 i.f2

Same as above, but report statistics for each level of variable group, with and without the total sample
dtable x1 x2 i.f1 i.f2, by (group)
dtable x1 x2 i.f1 i.f2, by(group, nototals)
Same as above, but test each variable for equality between groups using linear regressions for continuous variables and Pearson $\chi^{2}$ tests for factor variables
dtable x1 x2 i.f1 i.f2, by (group, tests)
Same as above, but rather than putting sample frequencies in the first row, put them in the column header under each group label

```
dtable x1 x2 i.f1 i.f2, by(group, tests) sample(, place(seplabels))
```

Create a table of the median and interquartile interval for variables $\mathrm{x} 1-\mathrm{x} 4$

```
dtable, continuous(x1-x4, statistics(q2 iqi)) ///
    define(iqi = q1 q3) sformat("[%s]" iqi)
```

Same as above, but add the Kruskal-Wallis test for equality between levels of variable group

```
dtable, continuous(x1-x4, statistics(q2 iqi) test(kwallis)) ///
    define(iqi = q1 q3) sformat("[%s]" iqi) by(group, tests)
```

Create a table to describe variables using survey weights
dtable x* i.f*, svy
Same as above, but restrict the sample to the subpopulation where group equals 4
dtable x* i.f*, svy subpop(if group==4)

## Menu

Statistics $>$ Summaries, tables, and tests $>$ Table of descriptive statistics

## Syntax

dtable $[$ varlist $][$ if $][$ in $][$ weight $][$, options $]$
options
Main
by (varname [, by_opts])
column (col_opts)
Survey
svy
subpop ([varname $][$ if $]$ )
Sample
nosample
sample ([title][, samp_opts])

## Continuous

continuous $\left(\left[\right.\right.$ varlist $\left._{c}\right][$, cont_opts $]$ )
Factors
factor $\left(\left[\right.\right.$ varlist $\left._{f}\right][$, fact_opts $\left.]\right)$
Composite
define $($ name $=$ elements $[$, def_opts $])$ define a composite result
Formats
nformat (\% fint [results][, basestyle]) sformat (sfint [results])
Title
title(string)
titlestyles(text_styles)
Notes
note (string)
notestyles(text_styles)
Export
export (filename.suffix [, export_opts])
Options
[no $]$ varlabel
[no]fvlabel
[no]listwise
halign(hvalue)
name (cname)
replace
label (filename)
style(filename[, override])
warn
group results by categories of varname change column header styles and titles
use survey settings
identify a survey subpopulation
suppress sample frequency statistics
select sample frequency statistics
select continuous variable statistics
select factor-variable statistics
specify numeric format
specify string format
add table title
change table title styles
add table note
change table note styles
export table
display or suppress variable names or labels
display or suppress factor values or labels specify how to handle missing values
specify horizontal alignment of item cells
work with collection cname; default is name (DTable)
replace the collection
specify the collection labels
specify the collection style
show collect warnings

[^20]by_opts

## Description

[no]tests
$[$ no]testnotes
$[$ no] totals
$[$ no]missing
show or suppress tests across groups show or suppress notes about tests across groups show or suppress statistics for the total sample show or suppress numeric missing values in varname
col_opts
Description
summary ([title $][$, hcol_opts $]$ ) change header styles for the summary column by ([by_style $][$, by_hcol_opts $])$ change header styles for the by () variable columns; by_style may be label, name, or hide
test ([title $][$, hcol_opts $]$ change header styles for the test column total $([$ title $][$, hcol_opts $]) \quad$ change header styles for the total sample column
hcol_opts
Description
[no]hide
halign (hvalue)
text_styles
show or hide the column header title
nohide and hide do not appear in the dialog box for summary ().
by_hcol_opts Description

| $[$ no $]$ fvlabel | display by () variable values or value labels |
| :--- | :--- |
| halign(hvalue) | specify horizontal alignment of column header cells <br> text_styles |

text_styles Description

| font $([$ fontfamily $][$, font_opts $])$ | specify font style |
| :--- | :--- |
| $\operatorname{smcl}($ smcl $)$ | specify formatting for SMCL files |
| latex (latex) | specify LATEX macro |
| shading $($ sspec $)$ | set background color, foreground color, and fill pattern |

font_opts

Description

| size(\# [unit]) | specify font size |
| :--- | :--- |
| color(color) | specify font color |
| variant(variant) | specify font variant and capitalization |
| [no]bold | specify whether to format text as bold |
| [no]italic | specify whether to format text as italic |
| $[$ no]strikeout | specify whether to strike out text |
| $[$ no]underline | specify whether to underline text |
| underline(upattern) | specify underline pattern for text |

samp_opts
Description
statistics (nstats)
place(place)
text_styles
select sample frequency statistics
select location for sample frequency statistics; place may be items, inlabels, or seplabels
change row header text styles
cont_opts

## statistics(cstats)

test (ctest)
text_styles

Description
select continuous variable statistics
select continuous variable test
change row header text styles
fact_opts Description
statistics(fstats)
test (ftest)
text_styles

Description
select factor-variable statistics
select factor-variable test
change row header text styles

| def_opts | Description |
| :--- | :--- |
| delimiter (char) | use character as delimiter between elements |
| $[$ no $]$ trim | preserve or trim extra spaces from numeric formats |
| $[$ no $]$ override | preserve or override trim property of elements |
| replace | redefine name if it already exists |


| suffix | fileformat | Output format |
| :--- | :--- | :--- |
| docx | as (docx) | Microsoft Word |
| html | as (html) | HTML 5 with CSS |
| pdf | as (pdf) | PDF |
| xlsx | as (xlsx) | Microsoft Excel 2007/2010 or newer |
| xls | as (xls) | Microsoft Excel 1997/2003 |
| tex | as (latex) | LATEX |
| smcl | as (smcl) | SMCL |
| txt | as (txt) | plain text |
| markdown | as (markdown) | Markdown |
| md | as (markdown) | Markdown |

export_opts
Description
as (fileformat)
replace
docx_options
html_options
pdf_options
excel_options
tex_options
smcl_option
txt_option
md_option
specify document type
overwrite existing file
available when exporting to .docx files
available when exporting to . html files
available when exporting to .pdf files
available when exporting to . xls and . xl sx files
available when exporting to .tex files
available when exporting to . smcl files
available when exporting to .txt files
available when exporting to .markdown and .md files
docx_options
noisily
dofile(filename[, replace]) save the putdocx commands used for exporting to the named do-file
html_options

## Description

append
tableonly
cssfile (cssfile)
prefix(prefix)
append to an existing file
export only the table to the specified file
define the styles in cssfile instead of filename
use prefix to identify style classes
pdf_options
noisily
dofile(filename[, replace])

## Description

show the putpdf commands used to export to the PDF file save the putpdf commands used for exporting to the named do-file
excel_options
noisily
dofile(filename[, replace])
sheet (sheetname[, replace])
cell (cell)
modify
noopen

Description
show the putexcel commands used to export to the Excel file save the putexcel commands used for exporting to the named do-file
specify the worksheet to use; the default sheet name is Sheet1
specify the Excel upper-left cell as the starting position to export the table; the default is cell (A1)
modify Excel file
do not open Excel file in memory
noopen does not appear in the dialog box.
tex_options
Description

| append <br> tableonly | append to an existing file <br> export only the table to the specified file |
| :--- | :--- |
| smcl_option | Description |
| append | append to an existing file |
| txt_option | Description |
| append | append to an existing file |
| md_option | Description |
| append | append to an existing file |

varlist may contain i. notation, including level selection. Time-series operators and interactions are not allowed.
fweights, aweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
results is a list of names taken from nstats, cstats, ctest, fstats, ftest, and composite results.
hvalue may be left, center, or right.
fontfamily specifies a valid font family.
unit may be in (inch), pt (point), or cm (centimeter). An inch is equivalent to 72 points and 2.54 centimeters. The default is pt.
variant may be allcaps, smallcaps, or normal.
variant (allcaps) changes the text to all uppercase letters; applicable when publishing items from a collection to Microsoft Word, PDF, LATEX, and HTML files.
variant (smallcaps) changes the text to use large capitals for uppercase letters and smaller capitals for lowercase letters; applicable when publishing items from a collection to Microsoft Word, ${ }^{\text {ETSX}}$, and HTML files.
variant (normal) changes the font variant back to normal and leaves the capitalization unchanged from the original text; applicable when publishing items from a collection to Microsoft Word, PDF, LATEX, and HTML files.
upattern may be any of the patterns listed in the Appendix. For example, underline (none) removes the underline from the text, and underline (single) underlines the text. All other upatterns are available only when publishing items from a collection to Microsoft Word.
smcl specifies the name of the SMCL directive to render text for SMCL output. The supported SMCL directives are input, error, result, and text.
latex specifies the name of a $\mathrm{LAT}_{\mathrm{E}} \mathrm{X}$ macro to render text for $\mathrm{LAT}_{\mathrm{E}} \mathrm{X}$ output. Example $\mathrm{LAT}_{\mathrm{E}} \mathrm{X}$ macro names are textbf, textsf, textrm, and texttt. Custom LATEX macros are also allowed. If text is to be rendered in a cell, title, or note, then latex is translated to the following when you export to $\mathrm{LAT}_{\mathrm{E}} \mathrm{X}$ :
$\backslash$ latex $\{$ text $\}$
sspec is
[background(bgcolor) foreground(fgcolor) pattern(fpattern)]
bgcolor specifies the background color.
fgcolor specifies the foreground color.
fpattern specifies the fill pattern. A complete list of fill patterns is shown in the Appendix.
bgcolor, fgcolor, and color may be one of the colors listed in the Appendix; a valid RGB value in the form \#\#\# \#\#\# \#\#\#, for example, 171248 103; or a valid RRGGBB hex value in the form \#\#\#\#\#\#, for example, ABF867.
sfint is the specification for a string format in option sformat () and may contain a mix of text and $\%$ s. Here $\%$ s refers to the statistic value that is formatted as specified using nformat (). The text will be placed around the statistic as it is placed around $\% s$ in this option. For instance, to place parentheses around the statistic, you can specify sformat (" (\%s)").
Two text characters must be specified using a special character sequence if you want them to be displayed in your collection. To include $\%$, type $\% \%$. To include $\backslash$, type $\backslash \backslash$. For instance, to place a percent sign after a statistic, you can specify sformat("\%s\%\%").

## Options

$\underset{\text { by (varname }[\text {, by_opts }] \text { ) specifies that the statistics be displayed separately for each unique value }}{\text { Main }}$ of varname; varname may be numeric or string.
by_opts are the following:
tests and notests specify whether to show the tests comparing groups. tests displays the tests comparing groups; notests suppresses them.
When tests is combined with the missing suboption, the test will be performed with the missing values composing an additional group. By default, missing values are excluded from tests.
testnotes and notestnotes specify whether to show a note detailing each test comparing groups. testnotes displays the notes; notestnotes suppresses them.
totals and nototals specify whether to show the statistics for the total sample. totals displays the statistics for the total sample; nototals suppresses them.
missing and nomissing specify whether to show numeric missing values in varname. missing displays the numeric missing values; nomissing suppresses them.
When missing is combined with the tests suboption, missing values will also be included in the test performed. By default, missing values are excluded from tests.
column (col_opts) specifies the column header styles and titles. col_opts are one or more of the following.
summary ([title][, [no]hide halign(hvalue) text_styles $]$ ) specifies the column header and styles when global option by () is not specified. The default title is Summary.
hide and nohide specify whether to show the title. hide suppresses the title; nohide displays the title.
halign(hvalue) specifies how the column header cell is horizontally aligned. hvalue may be left, center, or right.
text_styles affect the text styles for the column header.
by ([by_style][, [no]fvlabel halign(hvalue) text_styles]) specifies the by() variable's column header and styles.
by_style specifies the column header style for the by () variable. by_style may be label, name, or hide.
label specifies that the by () variable's label be used. If the by () variable does not have a label, then the name is used.
name specifies that the by () variable's name be used.
hide specifies that the by () variable not be shown in the column header.
fvlabel and nofvlabel specify whether value labels for the levels of the by () variable should be displayed. fvlabel displays the value labels; nofvlabel displays the values.
halign(hvalue) specifies how the column header cells are horizontally aligned. hvalue may be left, center, or right.
text_styles affect the text styles for the column header.
test ([title][, [no]hide halign(hvalue) text_styles $]$ ) specifies the column header and text styles for the test column. The default title is Test.
hide and nohide specify whether to show the title for the test column. hide suppresses the title; nohide displays the title.
halign(hvalue) specifies how the test column header cell is horizontally aligned. hvalue may be left, center, or right.
text_styles affect the text styles for the test column header.
total ([title $][,[$ no $]$ hide halign(hvalue) text_styles $]$ ) specifies the column header and text styles for the total sample. The default title is Total.
hide and nohide specify whether to show the title for the total sample. hide suppresses the title; nohide displays the title.
halign(hvalue) specifies how the total sample column header cell is horizontally aligned. hvalue may be left, center, or right.
text_styles affect the text styles for the total sample column header.
text_styles are font ([fontfamily] [, font_opts]), smcl(smcl), latex (latex), and shading(sspec).
font ([fontfamily] [, size(\# [unit]) color(color) variant (variant) [no]bold
[no]italic [no]strikeout [no]underline underline(upattern)]) specifies the font style. These font style properties are applicable when exporting the table to Microsoft Word, Microsoft Excel, PDF, HTML, and LATEX files, unless otherwise specified.
fontfamily specifies a valid font family. This font style property is applicable when publishing items from a collection to Microsoft Word, Microsoft Excel, PDF, and HTML files.
$\operatorname{size}(\#[$ unit $]$ ) specifies the font size as a number optionally followed by units. This font style property is applicable when publishing items from a collection to Microsoft Word, Microsoft Excel, PDF, and HTML files.
color (color) specifies the text color.
variant (variant) specifies the font variant and capitalization.
bold and nobold specify the font weight. bold changes the font weight to bold; nobold changes the font weight back to normal.
italic and noitalic specify the font style. italic changes the font style to italic; noitalic changes the font style back to normal.
strikeout and nostrikeout specify whether to add a strikeout mark to the text. strikeout adds a strikeout mark to the text; nostrikeout changes the text back to normal.
underline (upattern), underline, and nounderline specify how to underline the text.
Only one of strikeout or underline is allowed when publishing to HTML files.
$\operatorname{smcl}(s m c l)$ specifies how to render the text for SMCL output. This style property is applicable only when publishing items from a collection to a SMCL file.
latex (latex) specifies how to render the text for $\mathrm{LAT}_{\mathrm{E}} \mathrm{X}$ output. This style property is applicable only when publishing items from a collection to a $\mathrm{LAT}_{\mathrm{E}} \mathrm{X}$ file.
shading (sspec) sets the background color, foreground color, and fill pattern. The background color is applicable when exporting the table to Microsoft Word, Microsoft Excel, PDF, HTML, and ${ }^{L A T} T_{E} \mathrm{X}$ files. The foreground color and fill pattern are applicable when exporting the table to Microsoft Word and Microsoft Excel.

Survey
svy specifies that dtable use the survey design information from svyset. This means all statistics will be computed using the specified survey weights, where applicable. This option changes the list of supported tests in continuous() and factor().
$\operatorname{subpop}([$ varname $][$ if $]$ ) specifies that estimates be computed for the single subpopulation identified by the observations for which varname $\neq 0$ and that also meet the if conditions. Typically, varname $=1$ defines the subpopulation, and varname $=0$ indicates observations not belonging to the subpopulation. For observations whose subpopulation status is uncertain, varname should be set to a missing value; such observations are dropped from the estimation sample. This option implies the svy option.
See [SVY] Subpopulation estimation for more information.
nosample and sample() control the display of sample frequency statistics. nosample prevents reporting any sample frequency statistic.
sample([title] [, statistics(nstats) place(place) text_styles]) specifies sample frequency statistics, their row header title, and how they are composed in the table.
title specifies the row header for the sample frequency statistics. The default title is N . statistics (nstats) specifies the sample frequency statistics to report.
nstats may include the following sample frequency statistics and any composite result defined from them.

| nstats | Definition |
| :--- | :--- |
| $\frac{\text { frequency }}{\text { sumw }}$ | frequency <br> sum of weights |
| percent | percentage |
| proportion | proportion |
| rawpercent | unweighted percentage <br> unweighted proportion |

Without weights, the default sample frequency statistic is frequency.
With weights, the default sample frequency statistic is sumw.
With option by (), percent is added to the default sample frequency statistic. place (place) controls where the sample frequency statistics are reported in the table.
place(items), the default, specifies that the sample frequency statistics be reported in the first row of items in the table.
place (inlabels) specifies that the sample frequency statistics be added to the column header labels.
place (seplabels) specifies that the sample frequency statistics be stacked in the column header as separate labels.
text_styles specifies the text styles for title.

Continuous
continuous ( $\left[\right.$ varlist $\left._{c}\right][$, statistics (cstats) test (ctest) text_styles $]$ ) specifies continuous variables, their row header text styles, their statistics, and their test. If $\mathrm{varlist}_{c}$ is not specified, then the specified text styles, statistics, and test become the default for all continuous variables in varlist. This option is repeatable, and when multiple text styles, statistics, or tests apply to a variable, the rightmost specification is applied.
statistics(cstats) specifies the statistics to report for varlist $c_{c}$.
cstats may contain the following continuous variable statistics and any composite result defined from them.

| cstats | Description |
| :--- | :--- |
| mean | mean |
| semean | standard error of the mean |
| sebinomial | standard error of the mean, binomial <br> sepoisson |
| standard error of the mean, Poisson |  |
| variance | variance |
| sd | standard deviation |
| $\underline{\text { skewness }}$ | skewness |
| $\underline{\text { kurtosis }}$ | kurtosis |
| cv | coefficient of variation |
| svycv | coefficient of variation (svy) |
| geomean | geometric mean |
| geosd | geometric standard deviation |
| count | number of nonmissing values |
| median | median |
| p\# | \#th percentile |
| q1 | first quartile |
| q2 | second quartile |
| q3 | third quartile |
| iqr | interquartile range |
| min | minimum value |
| max | maximum value |
| range | range |
| first | first value |
| last | last value |
| firstnm | first nonmissing value |
| lastnm | last nonmissing value |
| total | total |
| $\underline{\text { rawtotal }}$ | unweighted total |

The default continuous variable statistics are mean and sd.
test (ctest) specifies that ctest be computed for the continuous variables in varlist. This option has no effect if the by () option is not specified.
ctest is one of the following:

| ctest | Description |
| :--- | :--- |
| regress | main effects test from a linear regression |
| poisson | main effects test from a Poisson regression |
| lnormal | main effects test from a log-normal regression |
| kwallis | Kruskal-Wallis rank test |
| kwallis_ties | Kruskal-Wallis rank test, adjusted for ties |
| none | suppress the test |

kwallis and kwallis_ties are not allowed with weights or the svy option.
The default test for continuous variables is regress.
For nonsurvey data, regress is equivalent to a one-way ANOVA, and when the by () variable has two levels, regress is equivalent to a pooled $t$ test.
text_styles specifies the row header text styles for varlist ${ }_{c}$.
$\qquad$ Factors
factor ([ varlist $\left.f_{f}\right][$, statistics (fstats) test (ftest) text_styles $]$ ) specifies factor variables, their row header text styles, their statistics, and their test. If varlist ${ }_{f}$ is not specified, then the specified text styles, statistics, and test become the default for all factor variables in varlist.

This option is repeatable, and when multiple text styles, statistics, or tests apply to a variable, the rightmost specification is applied.
statistics ( $f$ stats) specifies the statistics to report for varlist $_{f}$.
$f$ stats may contain the following factor-variable statistics and any composite result defined from them.

| fstats | Definition |
| :--- | :--- |
| fvfrequency frequency of each factor-variable level <br> fvpercent percentage within each factor-variable level <br> fvproportion proportion within each factor-variable level <br> fvrawfrequency unweighted frequency of each factor-variable level <br> fvrawpercent unweighted percentage within each factor-variable level <br> fvrawproportion unweighted proportion within each factor-variable level  |  |

The default factor-variable statistics are fvfrequency and fvpercent.
test (ftest) specifies that ftest be computed for the factor variables in varlist $f_{f}$. This option has no effect if the by () option is not specified.
ftest is one of the following:

| ftest | Description |
| :--- | :--- |
| pearson | Pearson's $\chi^{2}$ test |
| fisher | Fisher's exact test |
| lrchi2 | likelihood-ratio $\chi^{2}$ test |
| gamma | Goodman and Kruskal's gamma |
| kendall | Kendall's $\tau_{b}$ |
| cramer | Cramér's $V$ |
| svylr | survey-adjusted likelihood-ratio test <br> svywald <br> svyllwald |
| survey-adjusted Wald test |  |
| none | suppress the test |

fisher, lrchi2, gamma, kendall, and cramer are not allowed with aweights, iweights, pweights, or the svy option.
svylr, svywald, and svyllwald are allowed only with the svy option.
The default test for factor variables is pearson.
text_styles specifies the row header text styles for varlist $_{f}$.

Composite
define (name=elements [, def_opts]) defines a composite result. Composite results give more control over how statistics are composed and formatted in a table cell.
name is the name you choose for the new composite result.
elements is a list of statistics and other composite results.
def_opts are delimiter (char), [no]trim, [no]override, and replace.
delimiter (char) changes the delimiter between elements. The default is delimiter(" ").
notrim and trim control the handling of extra spaces caused by numeric formats applied to the elements. notrim preserves the extra spaces; trim, the default, removes the extra spaces.
nooverride and override control handling of the trim property when an element is a composite result. nooverride, the default, does not change the trim property of elements; override applies the specified trim property to all elements.
replace permits you to redefine name if it already exists.

Formats
nformat ( $\%$ fint [results][, basestyle]) changes the numeric format, such as the number of decimal places, for specified results. If results are not specified, the numeric format is changed for all results.
results is a list of names taken from nstats, cstats, ctest, fstats, ftest, and composite results.

This option is repeatable, and when multiple formats apply to one result, the rightmost specification is applied.
This option does not affect the format of factor variables specified in varlist or the factor () option. The default format of these variables is taken from the dataset.
basestyle indicates that the format be applied to results that do not already have their own format instead of overriding the format for all results.
sformat (sfint [results]) changes the string format for specified results. You can, for instance, add symbols or text to the values reported in the table by modifying the string format.
sfmt may contain a mix of text and $\%$ s. Here $\%$ s refers to the numeric value that is formatted as specified using nformat (). The text will be placed around the numeric values in your table as it is placed around \%s in this option. For instance, to place parentheses around the percent statistics, you can specify sformat(" (\%s)" fvpercent percent).
results is a list of names taken from nstats, cstats, ctest, fstats, ftest, and composite results.
Two text characters must be specified using a special character sequence if you want them to be displayed in your table. To include $\%$, type $\% \%$. To include $\backslash$, type $\backslash \backslash$. For instance, to place a percent sign following percent statistics, you can specify sformat ( $\% \%$ s\% \% " fvpercent percent).

This option is repeatable, and when multiple formats apply to one result, the rightmost specification is applied.

## Title

title (string) adds the text string as a title to the table.
titlestyles (text_styles) changes the style for the table title. text_styles are the following:
font ([fontfamily] [, size(\# [unit]) color (color) variant (variant) [no]bold
[no]italic [no]strikeout [no]underline]) specifies the font style. These font style properties are applicable when exporting the table to Microsoft Word, Microsoft Excel, PDF, HTML, and LATEX files, unless otherwise specified.
fontfamily specifies a valid font family. This font style property is applicable when publishing items from a collection to Microsoft Word, Microsoft Excel, PDF, and HTML files.
size (\# [unit]) specifies the font size as a number optionally followed by units. This font style property is applicable when publishing items from a collection to Microsoft Word, Microsoft Excel, PDF, and HTML files.
color (color) specifies the text color.
variant (variant) specifies the font variant and capitalization.
bold and nobold specify the font weight. bold changes the font weight to bold; nobold changes the font weight back to normal.
italic and noitalic specify the font style. italic changes the font style to italic; noitalic changes the font style back to normal.
strikeout and nostrikeout specify whether to add a strikeout mark to the title. strikeout adds a strikeout mark to the title; nostrikeout changes the title back to normal.
underline and nounderline specify whether to underline the table title. underline adds a single line under the title; nounderline removes the underline.

Only one of strikeout or underline is allowed when publishing to HTML files.
$\mathrm{smcl}(\mathrm{smcl})$ specifies how to render the table title for SMCL output. This style property is applicable only when publishing items from a collection to a SMCL file.
latex (latex) specifies how to render the table title for $\mathrm{LAT}_{\mathrm{E}} \mathrm{X}$ output. This style property is applicable only when publishing items from a collection to a ${ }^{\mathrm{AT}} \mathrm{E}_{\mathrm{E}} \mathrm{X}$ file.
shading (sspec) sets the background color, foreground color, and fill pattern. The background color is applicable when exporting the table to Microsoft Word, Microsoft Excel, PDF, HTML, and $\mathrm{LAT}_{\mathrm{E}} \mathrm{X}$ files. The foreground color and fill pattern are applicable when exporting the table to Microsoft Word and Microsoft Excel.

Notes
note (string) adds the text string as a note to the table. note() may be specified multiple times to add multiple notes. Each note is placed on a new line.
notestyles (text_styles) changes the style for the table notes. text_styles are the following:
font ([fontfamily] [, size(\# [unit]) color (color) variant (variant) [no]bold
[no]italic [no]strikeout [no]underline]) specifies the font style. These font style properties are applicable when exporting the table to Microsoft Word, Microsoft Excel, PDF, HTML, and LATEX files, unless otherwise specified.
fontfamily specifies a valid font family. This font style property is applicable when publishing items from a collection to Microsoft Word, Microsoft Excel, PDF, and HTML files.
size (\# [unit]) specifies the font size as a number optionally followed by units. This font style property is applicable when publishing items from a collection to Microsoft Word, Microsoft Excel, PDF, and HTML files.
color (color) specifies the text color.
variant (variant) specifies the font variant and capitalization.
bold and nobold specify the font weight. bold changes the font weight to bold; nobold changes the font weight back to normal.
italic and noitalic specify the font style. italic changes the font style to italic; noitalic changes the font style back to normal.
strikeout and nostrikeout specify whether to add a strikeout mark to the notes. strikeout adds a strikeout mark to the note; nostrikeout changes the note back to normal.
underline and nounderline specify whether to underline the table notes. underline adds a single line under the notes; nounderline removes the underline.

Only one of strikeout or underline is allowed when publishing to HTML files.
smcl ( smcl ) specifies how to render the table notes for SMCL output. This style property is applicable only when publishing items from a collection to a SMCL file.
latex (latex) specifies how to render the table notes for $\mathrm{LAT}_{\mathrm{E} X}$ output. This style property is applicable only when publishing items from a collection to a $\mathrm{LAT}_{\mathrm{E}} \mathrm{X}$ file.
shading (sspec) sets the background color, foreground color, and fill pattern. The background color is applicable when exporting the table to Microsoft Word, Microsoft Excel, PDF, HTML, and $\mathrm{IAT}_{\mathrm{E}} \mathrm{X}$ files. The foreground color and fill pattern are applicable when exporting the table to Microsoft Word and Microsoft Excel.
export (filename.suffix [, export_opts]) exports the table to the specified file. export_opts are the following:
as (fileformat) specifies the file format to which the table is to be exported. This option is rarely specified because, by default, dtable determines the format from the suffix of the file being created.
replace permits dtable to overwrite an existing file.
noisily specifies that dtable show the commands used to export the table to Microsoft Word, Microsoft Excel, and PDF files. The putdocx, putexcel, or putpdf command used to export the table will be displayed.
dofile(filename [, replace]) specifies that dtable save to filename the commands used to export the table to Microsoft Word, Microsoft Excel, and PDF files.

If filename already exists, it can be overwritten by specifying replace. If filename is specified without an extension, .do is assumed.
append specifies that dtable append the table to an existing file.
This option is applicable when you export the table to an HTML, a LATEX, a SMCL, a txt, or a Markdown file. When you export to HTML and $\mathrm{LA}_{\mathrm{E}} \mathrm{X}$ files, the append option implies the tableonly option. Furthermore, when you export to HTML files, if the target CSS file already exists, dtable will also append to it.
tableonly specifies that only the table be exported to the specified HTML or $\mathrm{LAT}_{\mathrm{E}} \mathrm{X}$ document. By default, dtable produces complete HTML and $\mathrm{LAT}_{\mathrm{E}} \mathrm{X}$ documents.

When you export to an HTML file, if the cssfile() option is not specified, a CSS filename is constructed from filename, with the extension replaced with .css.
cssfile(cssfile) specifies that dtable define the styles in cssfile instead of filename when you export to HTML.
prefix (prefix) specifies that dtable use prefix to identify style classes when you export to HTML.
sheet (sheetname [, replace]) saves to the worksheet named sheetname. For more information about this option, see [RPT] putexcel.
cell (cell) specifies an Excel upper-left cell as the starting position to publish the table. The default is cell(A1).
modify permits putexcel set to modify an Excel file. For more information about this option, see [RPT] putexcel.
noopen prevents putexcel from opening the Excel file in memory for modification. It does not appear on the dialog box. For more information about this option, see [RPT] putexcel.
varlabel and novarlabel specify whether variable labels for the continuous and factor variables should be displayed. varlabel displays the variable label; novarlabel displays variable names.
fvlabel and nofvlabel specify whether value labels for the factor variables should be displayed. fvlabel displays value labels; nofvlabel displays the values.
listwise and nolistwise specify how to handle missing values across the continuous and factor variables. listwise handles missing values through listwise deletion, meaning that the entire observation is omitted from the sample if any continuous or factor variable is missing for that
observation; nolistwise specifies that dtable omit an observation only if all the continuous and factor variables are missing for that observation.
halign(hvalue) specifies how the item cells are horizontally aligned. hvalue may be left, center, or right.
name (cname) specifies the collection for dtable to work with. The default is name(DTable).
replace permits dtable to overwrite the existing collection. This option is implied for name (DTable).
label (filename) specifies the filename containing the collection labels to use for your table. Labels in filename will be loaded for the table, and default labels will be used for any labels not specified in filename.
style(filename[, override]) specifies the filename containing the collection styles to use for your table. This might be a style you saved with collect style save or a predefined style shipped with Stata. The dtable collection styles will be discarded, and only the collection styles in filename will be applied. Note that the layout specification saved in filename will not be applied; dtable will always use its predefined layout.
If you prefer the dtable collection styles but also want to apply any styles in filename, specify override. If there are conflicts between the default collection styles and those in filename, the ones in filename will take precedence.
The default is to use only the collection styles set in c(dtable_style); see [TABLES] set dtable_style.

The following option is available with dtable but is not shown in the dialog box:
warn specifies that dtable display warnings from collect. By default, these warnings are suppressed.

## Remarks and examples

Remarks are presented under the following headings:

Introduction<br>A first example<br>Describe variables across groups<br>Survey data<br>Save your style choices for next time<br>Composite results<br>The default style

## Introduction

dtable allows you to easily create a table of summary statistics and export it to a variety of file types, without any knowledge of the collection system. You can specify the continuous and factor variables that you want described in your table. You can also customize the table by formatting the results, changing the variable-specific statistics, adding a variable to identify sample groups, testing each variable for equality between groups, adding a title and notes, and more.

In most cases, you will use dtable to easily create a table and export it to another format. However, you can customize the table beyond the options that are available with dtable. When you issue a dtable command, the results are stored in a collection called DTable. This collection is replaced with each new dtable command, unless you specify a different name for the collection in the name () option. You can make additional changes to the collection with the collect suite of commands. To learn more about the collect commands, see [TABLES] Intro and the entries discussed therein.

## A first example

In its simplest specification, you type dtable and a list of variables you want described in your table. For our first example, we use data from the Second National Health and Nutrition Examination Survey (NHANES II) (McDowell et al. 1981). Our table describes continuous variables for systolic blood pressure, age, weight, and factor variables for race and health status.

```
. use https://www.stata-press.com/data/r18/nhanes2l
(Second National Health and Nutrition Examination Survey)
. dtable bpsystol age weight i.race i.hlthstat
```

|  | Summary |  |
| :--- | ---: | ---: |
| N | 10,351 |  |
| Systolic blood pressure | $130.882(23.333)$ |  |
| Age (years) | $47.580(17.215)$ |  |
| Weight (kg) | 71.898 | $(15.356)$ |
| Race |  |  |
| White | $9,065(87.6 \%)$ |  |
| Black | $1,086(10.5 \%)$ |  |
| Other | $200(1.9 \%)$ |  |
| Health status |  |  |
| Excellent | $2,407(23.3 \%)$ |  |
| Very good | $2,591(25.1 \%)$ |  |
| Good | $2,938(28.4 \%)$ |  |
| Fair | $1,670(16.2 \%)$ |  |
| Poor | $729(7.1 \%)$ |  |

The default continuous variable statistics are the mean and standard deviation. The default factorvariable statistics are the factor-level frequencies and percentages. You can change the statistics reported for each continuous and factor variable; see the continuous() and factor() options for the list of supported statistics.

Many options are also available to change the formatting and style of your table. Here we set the format of means and standard deviations to show two decimal places, remove the first row of the table with the overall sample size, and change the column header to be more descriptive of the table results.


Additionally, you can complete your table with a title and notes and even export it to another format. We now add a title and export our table to the file mydtable.html.

```
. dtable bpsystol age weight i.race i.hlthstat,
> nformat(%16.2fc mean sd)
> nosample
> column(summary(M(SD) / n(%)))
> title(Table 1)
> export(mydtable.html)
Table 1
```

|  | M(SD) / n(\%) |
| :--- | ---: |
| Systolic blood pressure | $130.88(23.33)$ |
| Age (years) | $47.58(17.21)$ |
| Weight (kg) | $71.90(15.36)$ |
| Race |  |
| White | $9,065(87.6 \%)$ |
| Black | $1,086(10.5 \%)$ |
| Other | $200(1.9 \%)$ |
| Health status | $2,407(23.3 \%)$ |
| Excellent | $2,591(25.1 \%)$ |
| Very good | $2,938(28.4 \%)$ |
| Good | $1,670(16.2 \%)$ |
| Fair | $729(7.1 \%)$ |
| Poor |  |

We could also export this table to a Microsoft Word, Microsoft Excel, LATEX, Markdown, SMCL, PDF, or plain-text file by specifying the appropriate file extension.

## Describe variables across groups

Describing variables across samples in your dataset is easy with the by () option. In the following, we revisit our original table, but we add the by () option to get summary statistics from the urban, rural, and total samples.
. dtable bpsystol age weight i.race i.hlthstat, by(rural) sample(N (\%))

|  | Urban |  | Rural <br> Rural | Total |
| :--- | ---: | ---: | ---: | ---: |
| N (\%) | $6,548(63.3 \%)$ | $3,803(36.7 \%)$ | $10,351(100.0 \%)$ |  |
| Systolic blood pressure | $130.460(23.526)$ | $131.607(22.980)$ | $130.882(23.333)$ |  |
| Age (years) | $46.835(17.484)$ | $48.862(16.666)$ | $47.580(17.215)$ |  |
| Weight (kg) | $71.427(15.363)$ | $72.708(15.314)$ | $71.898(15.356)$ |  |
| Race |  |  |  |  |
| White | $5,419(82.8 \%)$ | $3,646(95.9 \%)$ | $9,065(87.6 \%)$ |  |
| Black | $968(14.8 \%)$ | $118(3.1 \%)$ | $1,086(10.5 \%)$ |  |
| Other | $161(2.5 \%)$ | $39(1.0 \%)$ | $200(1.9 \%)$ |  |
| Health status |  |  |  |  |
| Excellent | $1,609(24.6 \%)$ | $798(21.0 \%)$ | $2,407(23.3 \%)$ |  |
| Very good | $1,713(26.2 \%)$ | $878(23.1 \%)$ | $2,591(25.1 \%)$ |  |
| Good | $1,878(28.7 \%)$ | $1,060(27.9 \%)$ | $2,938(28.4 \%)$ |  |
| Fair | $950(14.5 \%)$ | $720(19.0 \%)$ | $1,670(16.2 \%)$ |  |
| Poor | $389(5.9 \%)$ | $340(9.0 \%)$ | $729(7.1 \%)$ |  |

Next, we add the tests suboption in the by () option to report tests of equality between samples and the nototals option to suppress the column of statistics from the total sample. We also specify that we would like sample statistics to be in the column headers rather than the first row of the table using the sample() option. Here we also add the column(by (hide)) option to suppress the redundant label from variable rural and add notes that detail what is being reported.

```
. dtable bpsystol age weight i.race i.hlthstat,
> by(rural, nototals tests)
> column(by(hide))
> sample(, place(seplabels))
> note(Mean (Standard deviation): p-value from a pooled t-test.)
> note(Frequency (Percent%): p-value from Pearson test.)
note: using test regress across levels of rural for bpsystol, age, and
    weight.
note: using test pearson across levels of rural for race and hlthstat.
```

|  | Urban |  | Rural |
| :--- | ---: | ---: | ---: |
|  | $6,548(63.3 \%)$ | $3,803(36.7 \%)$ | Test |
| Systolic blood pressure | $130.460(23.526)$ | $131.607(22.980)$ | 0.016 |
| Age (years) | $46.835(17.484)$ | $48.862(16.666)$ | $<0.001$ |
| Weight (kg) | $71.427(15.363)$ | $72.708(15.314)$ | $<0.001$ |
| Race |  |  |  |
| White | $5,419(82.8 \%)$ | $3,646(95.9 \%)$ | $<0.001$ |
| Black | $968(14.8 \%)$ | $118(3.1 \%)$ |  |
| Other | $161(2.5 \%)$ | $39(1.0 \%)$ |  |
| Health status |  |  |  |
| Excellent | $1,609(24.6 \%)$ | $798(21.0 \%)$ | $<0.001$ |
| Very good | $1,713(26.2 \%)$ | $878(23.1 \%)$ |  |
| Good | $1,878(28.7 \%)$ | $1,060(27.9 \%)$ |  |
| Fair | $950(14.5 \%)$ | $720(19.0 \%)$ |  |
| Poor | $389(5.9 \%)$ | $340(9.0 \%)$ |  |

Mean (Standard deviation) : p-value from a pooled t-test.
Frequency (Percent\%) : p-value from Pearson test.
For continuous variables, the Test column reports the $p$-value from linear regressions, which are equivalent to pooled $t$ tests because rural has two categories. For factor variables, the Test column reports the $p$-values from Pearson $\chi^{2}$ tests.

## Survey data

dtable supports survey data by using the survey data characteristics in your dataset when you specify the svy option. Your data must be svyset prior to calling dtable with the svy option. Our data already have survey data characteristics.

```
. svyset
Sampling weights: finalwgt
            VCE: linearized
    Single unit: missing
    Strata 1: strata
Sampling unit 1: psu
                FPC 1: <zero>
```

The following table reports the survey weighted frequency, mean, standard deviation, factor-level frequency, and factor-level percent of the specified variables.

```
. dtable bpsystol age weight i.race i.hlthstat, svy
```

|  | Summary |
| :--- | ---: |
| N | $117,157,513$ |
| Systolic blood pressure | $126.946(21.401)$ |
| Age (years) | $42.253(15.502)$ |
| Weight (kg) | $71.901 \quad(15.433)$ |
| Race |  |
| $\quad$ White | $102,999,549(87.9 \%)$ |
| $\quad$ Black | $11,189,236(9.6 \%)$ |
| Other | $2,968,728(2.5 \%)$ |
| Health status |  |
| $\quad$ Excellent | $32,187,335(27.5 \%)$ |
| Very good | $32,176,310(27.5 \%)$ |
| Good | $32,715,395(28.0 \%)$ |
| Fair | $14,380,261(12.3 \%)$ |
| Poor | $5,537,956(4.7 \%)$ |

Option by () is allowed with svy. dtable also has options for tests of equality between groups that are allowed with svy. These tests account for the survey data characteristics in your data. In the following, we revisit the urban and rural summary table, but we add the svy option, a title, and some notes. We also add the column(by (hide)) option to suppress the redundant label from variable rural and add the test () suboption to rename the Test column.

```
. dtable bpsystol age weight i.race i.hlthstat, svy
> by(rural, nototals tests)
> column(by(hide) test(p-value))
> title(Survey data summary)
> note(Mean (Standard deviation): p-value from linear regression.)
> note(Frequency (Percent%): p-value from Pearson test.)
> note(Statistics computed using the survey weights.)
> note(Tests adjusted for the survey design.)
note: using test regress across levels of rural for bpsystol, age, and
    weight.
note: using test pearson across levels of rural for race and hlthstat.
Survey data summary
```

|  | Urban | Rural | p-value |
| :---: | :---: | :---: | :---: |
| N | 79,965,794 (68.3\%) | 37,191,719 (31.7\%) |  |
| Systolic blood pressure | 126.607 (21.438) | 127.675 (21.305) | 0.406 |
| Age (years) | 41.805 (15.662) | 43.215 (15.112) | 0.024 |
| Weight (kg) | 71.322 (15.371) | 73.144 (15.493) | <0.001 |
| Race |  |  |  |
| White | 67,579,394 (84.5\%) | 35,420,155 (95.2\%) | <0.001 |
| Black | 9,936,159 (12.4\%) | 1,253,077 (3.4\%) |  |
| Other | 2,450,241 (3.1\%) | 518,487 (1.4\%) |  |
| Health status |  |  |  |
| Excellent | 22,781,784 (28.5\%) | 9,405,551 ( $25.3 \%$ ) | <0.001 |
| Very good | 22,867,496 (28.6\%) | 9,308,814 (25.1\%) |  |
| Good | 22,089,942 (27.7\%) | 10,625,453 (28.6\%) |  |
| Fair | 8,892,926 (11.1\%) | 5,487,335 (14.8\%) |  |
| Poor | 3,229,798 (4.0\%) | 2,308,158 (6.2\%) |  |

Mean (Standard deviation): p-value from linear regression.
Frequency (Percent\%) : p-value from Pearson test.
Statistics computed using the survey weights.
Tests adjusted for the survey design.

## Save your style choices for next time

If you find yourself repeating certain options each time you use dtable, you can save most of those option choices for future tables. Use collect style save to store your dtable option choices to a collection style file and then specify this file in the style() option the next time you use dtable. If you store the collection style file in your PERSONAL folder, you can set that file as the default style for dtable.

In the following, we use dtable to create a table with some option choices, save our style to a file in the PERSONAL directory, and set that style file as the default for dtable.

```
. sysuse auto
(1978 automobile data)
. dtable mpg turn i.rep78,
> sample(Sample, statistic(frequency) place(seplabels))
> sformat("n=%s" frequency)
> by(foreign, tests nototals missing)
> continuous(, test(kwallis))
> factor(, test(pearson))
> title(Table 1. Summary by group with a test.)
> note(Kruskal-Wallis test for continuous variables.)
> note(Pearson's test for factor variables.)
note: using test kwallis across levels of foreign for mpg and turn.
note: using test pearson across levels of foreign for rep78.
Table 1. Summary by group with a test.
```

|  | Domestic <br> $\mathrm{n}=52$ |  | Car origin <br> Foreign <br> $\mathrm{n}=22$ | Test |
| :--- | ---: | ---: | ---: | ---: |
| Mileage (mpg) | $19.827(4.743)$ | $24.773(6.611)$ | 0.002 |  |
| Turn circle (ft.) | $41.442(3.968)$ | $35.409(1.501)$ | $<0.001$ |  |
| Repair record 1978 |  |  |  |  |
| 1 | $2(4.2 \%)$ | $0(0.0 \%)$ | $<0.001$ |  |
| 2 | $8(16.7 \%)$ | $0(0.0 \%)$ |  |  |
| 3 | $27(56.2 \%)$ | $3(14.3 \%)$ |  |  |
| 4 | $9(18.8 \%)$ | $9(42.9 \%)$ |  |  |
| 5 | $2(4.2 \%)$ | $9(42.9 \%)$ |  |  |

```
Kruskal-Wallis test for continuous variables.
Pearson's test for factor variables.
. personal dir
your personal ado-directory is C:\ado\personal\
    (output omitted)
. collect style save "C:\ado\personal\mydtable"
(style from DTable saved to file C:\ado\personal\mydtable.stjson)
. set dtable_style mydtable
. display c(dtable_style)
mydtable
```

Now these style choices are the defaults for tables created in the future. Here we create a similarly styled table but from different data.

```
. use https://www.stata-press.com/data/r18/lbw
(Hosmer & Lemeshow data)
. dtable age i.smoke i.ht, by(low)
note: using test kwallis across levels of low for age.
note: using test pearson across levels of low for smoke and ht.
Table 1. Summary by group with a test.
```

|  | Birthweight<2500g |  |  |  |
| :---: | :---: | :---: | :---: | :---: |
|  | $\begin{gathered} 0 \\ \mathrm{n}=130 \end{gathered}$ | $\begin{gathered} 1 \\ \mathrm{n}=59 \end{gathered}$ |  | Test |
|  |  |  |  |  |
| Age of mother | 23.662 (5.585) | 22.305 | (4.511) | 0.248 |
| Smoked during pregnancy |  |  |  |  |
| Nonsmoker | 86 (66.2\%) | 29 | (49.2\%) | 0.026 |
| Smoker | 44 (33.8\%) | 30 | (50.8\%) |  |
| Has history of hypertension |  |  |  |  |
| 0 | 125 (96.2\%) | 52 | (88.1\%) | 0.036 |
| 1 | 5 (3.8\%) | 7 | (11.9\%) |  |

Kruskal-Wallis test for continuous variables.
Pearson's test for factor variables.
To reset the dtable style back to its original default, type
. set dtable_style dtable
You can add the permanently option to set the default dtable style for future Stata sessions.

## Composite results

dtable's default style does not define any composite results, so you will need to define your own with the define() option. Stata has a collection style named dtable_composites that defines the following composite results for you.

| composite | elements | Description |
| :--- | :--- | :--- |
| idi | p10 p90 | Interdecile interval |
| iqi | q1 q3 | Interquartile interval |
| rangei | min max | Range interval |

All of these results are bound in square brackets without a delimiter between their elements. To get access to these composite results, use the style(dtable_composites, override) option. You can then specify idi, iqi, and rangei in the statistic() suboption of continuous(), just as you would any other statistic.

You may want different formatting for these composite results, or you may want to create composite results based on other statistics. The define() option allows you to customize your own composite results. In the following, we use dtable to create a table that defines our own interval results with some other option choices and then save the collection's style to a file.

```
. sysuse auto
(1978 automobile data)
. dtable, define(idi = p10 p90, delimiter("; "))
> define(iqi = q1 q3, delimiter("; "))
> define(rangei = min max, delimiter("; "))
> sample(Sample, statistic(frequency) place(seplabels))
> sformat("n=%s" frequency)
> sformat("{%s}" idi)
> sformat("[%s]" iqi)
> sformat("<%s>" rangei)
> continuous(mpg, statistic(p50 idi))
> continuous(turn, statistic(q2 iqi))
> continuous(trunk, statistic(mean rangei))
> title(Table 1. Location and interval.)
> note(Median {Interdecile interval})
> note(Median [Interquartile interval])
> note(Mean <Range interval>)
```

Table 1. Location and interval.

|  | Summary $n=74$ |
| :---: | :---: |
| Mileage (mpg) | 20.000 \{14.000; 29.000\} |
| Turn circle (ft.) | 40.000 [36.000; 43.000] |
| Trunk space (cu. ft.) | 13.757 <5.000; 23.000> |
| Median \{Interdecile interval\} |  |
| Median [Interquartile interval] |  |
| Mean <Range interval> |  |
| . collect style save (style from DTable sa | dtable-intervals <br> ved to file dtable-inter |

Now we can use this style and simply reference our composite results without having to redefine them.

```
. dtable, continuous(mpg, statistic(p50 idi))
> continuous(turn, statistic(q2 iqi))
> continuous(trunk, statistic(mean rangei))
> style(dtable-intervals)
```

Table 1. Location and interval.

|  | Summary <br> $\mathrm{n}=74$ |
| :--- | ---: |
| Mileage (mpg) | 20.000$\{14.000 ;$ $29.000\}$ <br> Turn circle (ft.) 40.000 <br> [36.000; $43.000]$ <br> Trunk space (cu. ft.) 13.757 <br>  $<5.000 ;$ $23.000>$ |
| Median \{Interdecile interval\} |  |
| Median [Interquartile interval] |  |
| Mean <Range interval> |  |

## The default style

Most of dtable's option defaults are controllable in its collection style. The following options are not part of dtable's collection style:

```
svy
subpop()
export()
name()
replace
label()
style()
warn
```

When the by () option is not specified, the default column() header options are

```
summary("Summary", nohide
    halign(center)
    font(, nobold
        noitalic
        nostrikeout
        nounderline)
    smcl(text))
```

When a variable is specified in the by () option, its default options are

```
notest
testnotes
totals
nomissing
```

and its default column() header options are

```
by(label, nohide
    fvlabel
    halign(center)
    font(, nobold
        noitalic
        nostrikeout
        nounderline)
    smcl(text))
total("Total", nohide
            halign(center)
            font(, nobold
                noitalic
                        nostrikeout
                    nounderline)
            smcl(text))
test("Test", nohide
        halign(center)
        font(, nobold
                        noitalic
                nostrikeout
                nounderline)
        smcl(text))
```

The default sample frequency title is " N ". When weights are not specified, the default sample frequency statistic is frequency; otherwise, sumw is reported. When the by () option is specified, percent is added to the default sample frequency statistic. The other default sample() options are

```
font(, nobold
    noitalic
    nostrikeout
    nounderline)
smcl(text)
place(items)
```

The defaults for the continuous() option are

```
statistic(mean sd)
test(regress)
font(, nobold
    noitalic
    nostrikeout
    nounderline)
smcl(text)
```

The defaults for the factor() option are

```
statistic(fvfrequency fvpercent)
test(pearson)
font(, nobold
    noitalic
    nostrikeout
    nounderline)
smcl(text)
```

There are no predefined composite results in dtable's default style.
The default formats are

```
nformat("%21.3fc", basestyle)
nformat("%21.Ofc" N count sumw frequency fvfrequency fvrawfrequency)
nformat("%9.1fc" percent rawpercent fvpercent fvrawpercent)
sformat("(%s%%)" percent rawpercent fvpercent fvrawpercent)
sformat("(%s)" sd)
nformat("%6.3f" regress poisson lnormal kwallis kwallis_ties pearson
    fisher lrchi2 svylr svywald svyllwald)
```

There is no default table title. The defaults for the titlestyles() option are

```
font(, nobold
    noitalic
    nostrikeout
    nounderline)
smcl(text)
```

There are no default table notes. The defaults for the notestyles() option are

```
font(, nobold
    noitalic
    nostrikeout
    nounderline)
smcl(text)
```

The remaining default options for dtable are

```
varlabel
fvlabel
nolistwise
halign(right)
```


## Methods and formulas

See Methods and formulas in [R] table for details on the statistics supported by dtable. When the svy or subpop() option is specified, the weights from svyset are used to compute statistics that support weights. In addition, the subpopulation variance $\widehat{\sigma}^{2}$ is computed as

$$
\widehat{\sigma}^{2}=s^{2} \frac{M}{n-1} \cdot \frac{n_{s}-1}{M_{s}}
$$

where $s^{2}$ is the sample variance as documented in [R] table, $M$ is the estimated population size, $n$ is the full sample size, $M_{s}$ is the estimated subpopulation size, and $n_{s}$ is the subpopulation sample size. The subpopulation standard deviation is computed as $\widehat{\sigma}=\sqrt{\widehat{\sigma}^{2}}$ and coefficient of variation (cv) is computed as $\widehat{\sigma} / \bar{x}$.

Let y be a continuous variable and by (group, tests) be in effect. For test (regress), dtable reports the $p$-value from test [\#1] after

$$
\text { regress y i.group }[i f][\text { in }][\text { weight }]
$$

For test (poisson), dtable reports the $p$-value from test [\#1] after

```
poisson y i.group [if] [in] [weight]
```

For test (lnormal), dtable reports the $p$-value from test [\#1] after

```
gsem y <- i.group [if] [in] [weight], family(lognormal)
```

For test(kwallis) and test(kwallis_ties), dtable reports the $p$-values corresponding with the $\chi^{2}$ statistics reported by

```
kwallis y [if] [in], by(group)
```

When the svy or subpop() option is specified, the prefix
svy, subpop([varname] [if]) :
is used with regress, poisson, and gsem. For more details, see [R] test, [R] regress, $[R]$ poisson, [SEM] gsem, [R] kwallis, and [SVY] svy.

Let $f$ be a factor variable and by (group, tests) be in effect. For nonsurvey data, the factorvariable test across groups is performed using
tabulate f group $[i f][i n][$ weight $]$, opt
where opt is an option of tabulate corresponding with the specified test (ftest) using the following table:

| ftest | opt |
| :--- | :--- |
| pearson | chi2 |
| fisher | exact |
| lrchi2 | lrchi2 |
| gamma | gamma |
| kendall | taub |
| cramer | V |

For survey data, dtable reports the $p$-value from

```
svy, subpop([varname] [if]): tabulate f group [if] [in], opt
```

where opt is an option of svy: tabulate corresponding with the specified test (ftest) using the following table:

| ftest | opt |
| :--- | :--- |
| pearson | pearson |
| svylr | lr |
| svywald | wald |
| svyllwald | llwald |

pearson, fisher, lrchi2, svylr, svywald, and svyllwald report a $p$-value. gamma, kendall, and cramer report the computed measure of association. For more details, see [R] tabulate twoway and [SVY] svy: tabulate twoway.

## Appendix

## Colors

bgcolor, fgcolor, and color

| aliceblue | darkslategray | lightsalmon | palevioletred |
| :---: | :---: | :---: | :---: |
| antiquewhite | darkturquoise | lightseagreen | papayawhip |
| aqua | darkviolet | lightskyblue | peachpuff |
| aquamarine | deeppink | lightslategray | peru |
| azure | deepskyblue | lightsteelblue | pink |
| beige | dimgray | lightyellow | plum |
| bisque | dodgerblue | lime | powderblue |
| black | firebrick | limegreen | purple |
| blanchedalmond | floralwhite | linen | red |
| blue | forestgreen | magenta | rosybrown |
| blueviolet | fuchsia | maroon | royalblue |
| brown | gainsboro | mediumaquamarine | saddlebrown |
| burlywood | ghostwhite | mediumblue | salmon |
| cadetblue | gold | mediumorchid | sandybrown |
| chartreuse | goldenrod | mediumpurple | seagreen |
| chocolate | gray | mediumseagreen | seashell |
| coral | green | mediumslateblue | sienna |
| cornflowerblue | greenyellow | mediumspringgreen | silver |
| cornsilk | honeydew | mediumturquoise | skyblue |
| crimson | hotpink | mediumvioletred | slateblue |
| cyan | indianred | midnightblue | slategray |
| darkblue | indigo | mintcream | snow |
| darkcyan | ivory | mistyrose | springgreen |
| darkgoldenrod | khaki | moccasin | steelblue |
| darkgray | lavender | navajowhite | $t a n$ |
| darkgreen | lavenderblush | navy | teal |
| darkkhaki | lawngreen | oldlace | thistle |
| darkmagenta | lemonchiffon | olive | tomato |
| darkolivegreen | lightblue | olivedrab | turquoise |
| darkorange | lightcoral | orange | violet |
| darkorchid | lightcyan | orangered | wheat |
| darkred | lightgoldenrodyellow | orchid | white |
| darksalmon | lightgray | palegoldenrod | whitesmoke |
| darkseagreen | lightgreen | palegreen | yellow |
| darkslateblue | lightpink | paleturquoise | yellowgreen |

## Underline patterns

upattern

| none | dashLong |
| :--- | :--- |
| single | dashLongHeavy |
| words | dotDash |
| double | dashDotHeavy |
| thick | dotDotDash |
| dotted | dashDotDotHeavy |
| dottedHeavy | wave |
| dash | wavyHeavy |
| dashedHeavy | wavyDouble |

## Shading patterns

fpattern

```
nil
clear
pct20
pct25
solid pct30
horzStripe pct35
vertStripe pct37
reverseDiagStripe pct40
diagStripe pct45
horzCross pct50
diagCross pct55
thinHorzStripe pct60
thinVertStripe pct62
thinReverseDiagStripe pct65
thinDiagStripe pct70
thinHorzCross pct75
thinDiagCross pct80
pct5 pct85
pct10 pct87
pct12 pct90
pct15 pct95
```


## Acknowledgments

We thank Phil Clayton of the AnZDATA Registry at the Royal Adelaide Hospital, Mark Chatfield of the University of Queensland, and Niels Henrik Bruun of Region Nordjylland for writing similar commands for creating tables of descriptive statistics. We also thank Joanna Dipnall of Monash University, Australia, for suggesting a unified framework for tables.

## Reference

McDowell, A., A. Engel, J. T. Massey, and K. Maurer. 1981. Plan and operation of the Second National Health and Nutrition Examination Survey, 1976-1980. Vital and Health Statistics 1(15): 1-144.

## Also see

$[R]$ table intro - Introduction to tables of frequencies, summaries, and command results
[R] table summary - Table of summary statistics
[TABLES] Intro - Introduction

## Title

dydx - Calculate numeric derivatives and integrals

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options for dydx | Options for integ | Remarks and examples | Stored results |
| Methods and formulas | Acknowledgment | Reference | Also see |

## Description

dydx and integ calculate derivatives and integrals of numeric "functions".

## Quick start

For variables y and x corresponding to function $y=f(x)$, compute $d y / d x$ using cubic splines and store result in dy
dydx y x, generate(dy)
Evaluate the integral of $f(x)$ using cubic splines
integ y x
Evaluate the integral using the trapezoidal rule
integ $\mathrm{y} x$, trapezoid
Same as above, and generate variable iy containing integral evaluated for each value of x integ $\mathrm{y} x$, trapezoid generate(iy)

## Menu

dydx
Data $>$ Create or change data $>$ Other variable-creation commands $>$ Calculate numerical derivatives

## integ

Data $>$ Create or change data $>$ Other variable-creation commands $>$ Calculate numeric integrals

## Syntax

Derivatives of numeric functions

```
dydx yvar xvar [if] [in], generate(newvar) [dydx_options]
```

Integrals of numeric functions
integ yvar xvar [if] [in] [, integ_options]
dydx_options Description
Main

* generate (newvar)
replace
double
store results in variable named newvar
overwrite the existing variable
store new variable as double
*generate (newvar) is required.
integ_options
Description
Main
trapezoid
generate(newvar)
replace
double
initial(\#)
use trapezoidal rule to compute integrals; default is cubic splines store results in variable named newvar
overwrite the existing variable
store new variable as double
initial value of integral; default is initial(0)
by and collect are allowed with dydx and integ; see [U] 11.1.10 Prefix commands.


## Options for dydx

$\qquad$ Main
generate (newvar) specifies that results be stored in a new variable. generate() is required. replace specifies that if an existing variable is specified for generate(), it should be overwritten. double specifies that the new variable in generate() be stored as double. If the double option is not specified, the variable is stored using the current type as set by set type, which is float by default.

## Options for integ

$\qquad$ Main
trapezoid requests that the trapezoidal rule [the sum of $\left(x_{i}-x_{i-1}\right)\left(y_{i}+y_{i-1}\right) / 2$ ] be used to compute integrals. The default is cubic splines, which give superior results for most smooth functions; for irregular functions, trapezoid may give better results. generate(newvar) specifies that results be stored in a new variable.
replace specifies that if an existing variable is specified for generate(), it should be overwritten. double specifies that the new variable in generate() be stored as double. If the double option is not specified, the variable is stored using the current type as set by set type.
initial (\#) specifies the initial condition for calculating definite integrals; see Methods and formulas below. The default is initial(0).

## Remarks and examples

dydx and integ lets you extend Stata's graphics capabilities beyond data analysis and into mathematics.

## > Example 1

We graph $y=e^{-x / 6} \sin (x)$ over the interval $[0,12.56]$ :

```
. range x 0 12.56 100
Number of observations (_N) was 0, now 100.
. generate y = exp(-x/6)*sin(x)
. label variable y "exp(-x/6)*sin(x)"
. twoway connected y x, connect(i) yline(0)
```



We estimate the derivative by using $d y d x$ and compute the relative difference between this estimate and the true derivative.
. dydx y x, gen(dy)
. generate dytrue $=\exp (-x / 6) *(\cos (x)-\sin (x) / 6)$
. generate error $=$ abs(dy - dytrue)/dytrue

The error is greatest at the endpoints, as we would expect. The error is approximately $0.5 \%$ at each endpoint, but the error quickly falls to less than $0.01 \%$.

```
. label variable error "Error in derivative estimate"
. twoway line error x, ylabel(0(.002).006)
```



We now estimate the integral by using integ:

```
. integ y x, gen(iy)
number of points = 100
integral = . }8531639
. generate iytrue = (36/37)*(1-\operatorname{exp}(-x/6)*(\operatorname{cos}(x)+\operatorname{sin}(x)/6))
. display iytrue[_N]
. }8531590
. display abs(r(integral) - iytrue[_N])/iytrue[_N]
5.811e-06
. generate diff = iy - iytrue
```

The relative difference between the estimate [stored in r(integral)] and the true value of the integral is about $6 \times 10^{-6}$. A graph of the absolute difference (diff) is shown below. Here error is cumulative. Again, most of the error is due to a relatively poorer fit near the endpoints.
. label variable diff "Error in integral estimate"
. twoway line diff x, ylabel(0(5.00e-06).00001)


## Stored results

dydx stores the following in $r()$ :
Macros
$\mathrm{r}(\mathrm{y}) \quad$ name of $y \mathrm{var}$
integ stores the following in $r()$ :
Scalars
$r$ (N_points) number of unique $x$ points
$r$ (integral) estimate of the integral

## Methods and formulas

Consider a set of data points, $\left(x_{1}, y_{1}\right), \ldots,\left(x_{n}, y_{n}\right)$, generated by a function $y=f(x)$. dydx and integ first fit these points with a cubic spline, which is then analytically differentiated (integrated) to give an approximation for the derivative (integral) of $f$.

The cubic spline (see, for example, Press et al. [2007]) consists of $n-1$ cubic polynomials $P_{i}(x)$, with the $i$ th one defined on the interval $\left[x_{i}, x_{i+1}\right]$,

$$
P_{i}(x)=y_{i} a_{i}(x)+y_{i+1} b_{i}(x)+y_{i}^{\prime \prime} c_{i}(x)+y_{i+1}^{\prime \prime} d_{i}(x)
$$

where

$$
\begin{array}{ll}
a_{i}(x)=\frac{x_{i+1}-x}{x_{i+1}-x_{i}} & b_{i}(x)=\frac{x-x_{i}}{x_{i+1}-x_{i}} \\
c_{i}(x)=\frac{1}{6}\left(x_{i+1}-x_{i}\right)^{2} a_{i}(x)\left[\left\{a_{i}(x)\right\}^{2}-1\right] & d_{i}(x)=\frac{1}{6}\left(x_{i+1}-x_{i}\right)^{2} b_{i}(x)\left[\left\{b_{i}(x)\right\}^{2}-1\right]
\end{array}
$$

and $y_{i}^{\prime \prime}$ and $y_{i+1}^{\prime \prime}$ are constants whose values will be determined as described below. The notation for these constants is justified because $P_{i}^{\prime \prime}\left(x_{i}\right)=y_{i}^{\prime \prime}$ and $P_{i}^{\prime \prime}\left(x_{i+1}\right)=y_{i+1}^{\prime \prime}$.

Because $a_{i}\left(x_{i}\right)=1, a_{i}\left(x_{i+1}\right)=0, b_{i}\left(x_{i}\right)=0$, and $b_{i}\left(x_{i+1}\right)=1$. Therefore, $P_{i}\left(x_{i}\right)=y_{i}$, and $P_{i}\left(x_{i+1}\right)=y_{i+1}$. Thus, the $P_{i}$ jointly define a function that is continuous at the interval boundaries. The first derivative should be continuous at the interval boundaries; that is,

$$
P_{i}^{\prime}\left(x_{i+1}\right)=P_{i+1}^{\prime}\left(x_{i+1}\right)
$$

The above $n-2$ equations (one equation for each point except the two endpoints) and the values of the first derivative at the endpoints, $P_{1}^{\prime}\left(x_{1}\right)$ and $P_{n-1}^{\prime}\left(x_{n}\right)$, determine the $n$ constants $y_{i}^{\prime \prime}$.

The value of the first derivative at an endpoint is set to the value of the derivative obtained by fitting a quadratic to the endpoint and the two adjacent points; namely, we use

$$
P_{1}^{\prime}\left(x_{1}\right)=\frac{y_{1}-y_{2}}{x_{1}-x_{2}}+\frac{y_{1}-y_{3}}{x_{1}-x_{3}}-\frac{y_{2}-y_{3}}{x_{2}-x_{3}}
$$

and a similar formula for the upper endpoint.
dydx approximates $f^{\prime}\left(x_{i}\right)$ by using $P_{i}^{\prime}\left(x_{i}\right)$.
integ approximates $F\left(x_{i}\right)=F\left(x_{1}\right)+\int_{x_{1}}^{x_{i}} f(x) d x$ by using

$$
I_{0}+\sum_{k=1}^{i-1} \int_{x_{k}}^{x_{k+1}} P_{k}(x) d x
$$

where $I_{0}$ (an estimate of $F\left(x_{1}\right)$ ) is the value specified by the initial(\#) option. If the trapezoid option is specified, integ approximates the integral by using the trapezoidal rule:

$$
I_{0}+\sum_{k=1}^{i-1} \frac{1}{2}\left(x_{k+1}-x_{k}\right)\left(y_{k+1}+y_{k}\right)
$$

If there are ties among the $x_{i}$, the mean of $y_{i}$ is computed at each set of ties and the cubic spline is fit to these values.

## Acknowledgment

The present versions of dydx and integ were inspired by the dydx2 command written by Patrick Royston of the MRC Clinical Trials Unit, London, and coauthor of the Stata Press book Flexible Parametric Survival Analysis Using Stata: Beyond the Cox Model.

Maria Gaetana Agnesi (1718-1799) was an Italian mathematician and philosopher.
Born in Milan into a wealthy family, she was recognized as a child prodigy. At age nine, she published a detailed argument in Latin on the importance of education for women. At age 15, her father, a mathematics professor at the University of Bologna, presented her talents in language and philosophical reasoning to Bologna's intellectual elite. Uncomfortable with public life, she educated her twenty siblings and published on mathematics. After her father's death in 1752, she studied theology and devoted the rest of her life to helping the poor, homeless, and sick.

Agnesi's best known work, Instituzioni analitiche ad uso della gioventù italiana (Analytical Institutions for the Use of Italian Youth), written in 1748, helped develop the analysis of finite quantities and infinitesimals. At the time, it was hailed as the best introduction to calculus. The work also discussed an asymptotic curve that, because of mistranslation, would come to be known as the "Witch of Agnesi". In 1750, Pope Benedict XIV appointed her to the chair of mathematics and natural philosophy at Bologna, though she never served.

In addition to being recognized as an important mathematician, Agnesi is revered in the Basilica of San Nazaro in Milan. At her death, she was mourned by radical authors as a proponent for women's rights and by the Catholic faithful as a symbol of personal piety.

## Reference

Press, W. H., S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. 2007. Numerical Recipes: The Art of Scientific Computing. 3rd ed. New York: Cambridge University Press.


#### Abstract

Also see [D] obs - Increase the number of observations in a dataset [D] range - Generate numerical range


## Title

eform_option - Displaying exponentiated coefficients

## Description Remarks and examples Reference Also see

## Description

An eform_option causes the coefficient table to be displayed in exponentiated form: for each coefficient, $e^{b}$ rather than $b$ is displayed. Standard errors and confidence intervals (CIs) are also transformed.

An eform_option is one of the following:

| eform_option | Description |
| :--- | :--- |
| eform(string) | use string for the column title |
| eform | exponentiated coefficient, string is $\exp (b)$ |
| hr | hazard ratio, string is Haz . ratio |
| shr | subhazard ratio, string is SHR |
| $\underline{\mathrm{irr}}$ | incidence-rate ratio, string is IRR <br> or |
| $\underline{\mathrm{rrr}}$ | odds ratio, string is Odds ratio <br> relative-risk ratio, string is RRR |

## Remarks and examples

## > Example 1

Here is a simple example of the or option with svy: logit. The CI for the odds ratio is computed by transforming (by exponentiating) the endpoints of the CI for the corresponding coefficient.

```
. use https://www.stata-press.com/data/r18/nhanes2d
. svy, or: logit highbp female black
(running logit on estimation sample)
    (output omitted)
```

| highbp | Odds ratioLinearized <br> std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |  |
| ---: | ---: | ---: | :---: | :---: | ---: | ---: |
| female | .6107011 | .0326159 | -9.23 | 0.000 | .5476753 | .6809798 |
| black | 1.384865 | .1336054 | 3.37 | 0.002 | 1.137507 | 1.686011 |
| _cons | .7249332 | .0551062 | -4.23 | 0.000 | .6208222 | .8465035 |

Note: _cons estimates baseline odds.
We also could have specified the following command and received the same results as above:

## Reference

Buis, M. L. 2012. Stata tip 107: The baseline is now reported. Stata Journal 12: 165-166.

## Also see

[R] ml - Maximum likelihood estimation

## Title

eivreg - Errors-in-variables regression

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

eivreg fits errors-in-variables regression models when one or more of the independent variables are measured with error. To use eivreg, you must have an estimate of each independent variable's reliability or assume it is measured without error.

## Quick start

Regression of y on x 1 , x 2 , and x 3 adjusted for x 1 measured with $90 \%$ reliability eivreg y x1 x2 x3, reliab(x1 .9)

Same as above, but also specify $80 \%$ reliability for x 2

```
    eivreg y x1 x2 x3, reliab(x1 .9 x2 .8)
```


## Menu

Statistics $>$ Linear models and related $>$ Errors-in-variables regression

## Syntax

eivreg depvar $[$ indepvars $][$ if $][$ in $][$ weight $][$, options $]$
options Description
Model
reliab(indepvar \# [indepvar \# [...] ])
specify measurement reliability for each indepvar measured with error
Reporting
level(\#)
display_options
coeflegend
set confidence level; default is level (95)
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
display legend instead of statistics
indepvars may contain factor variables; see [U] 11.4.3 Factor variables. bootstrap, by, collect, jackknife, rolling, and statsby are allowed; see [U] 11.1.10 Prefix commands.
Weights are not allowed with the bootstrap prefix; see $[R]$ bootstrap.
aweights are not allowed with the jackknife prefix; see [R] jackknife.
aweights and fweights are allowed; see [U] 11.1.6 weight.
coeflegend does not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

$\qquad$ Model
reliab (indepvar \# [indepvar \# [...]]) specifies the measurement reliability for each independent variable measured with error. Reliabilities are specified as pairs consisting of an independent variable name (a name that appears in indepvars) and the corresponding reliability $r, 0<r \leq 1$. Independent variables for which no reliability is specified are assumed to have reliability 1 . If the option is not specified, all variables are assumed to have reliability 1 , and the result is thus the same as that produced by regress (the ordinary least-squares results).

## Reporting

level(\#); see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\% fint), pformat (\% fint), sformat (\% fint), and nolstretch; see [R] Estimation options.

The following option is available with eivreg but is not shown in the dialog box:
coeflegend; see [R] Estimation options.

## Remarks and examples

For an introduction to errors-in-variables regression, see Draper and Smith (1998, 89-91) or Kmenta (1997, 352-357). Treiman (2009, 258-261) compares the results of errors-in-variables regression with conventional regression. Also see Lockwood and McCaffrey (2020) for how to use sem (see [SEM] sem) to fit errors-in-variables regression.

Errors-in-variables regression models are useful when one or more of the independent variables are measured with additive noise. Standard regression (as performed by regress) would underestimate the effect of the variable, and the other coefficients in the model can be biased to the extent that they are correlated with the poorly measured variable. You can adjust for the biases if you know the reliability:

$$
r=1-\frac{\text { noise variance }}{\text { total variance }}
$$

That is, given the model $\mathbf{y}=\mathbf{X} \boldsymbol{\beta}+\mathbf{u}$, for some variable $\mathbf{x}_{i}$ in $\mathbf{X}$, the $\mathbf{x}_{i}$ is observed with error, $\mathbf{x}_{i}=\mathbf{x}_{i}^{*}+\mathbf{e}$, and the noise variance is the variance of $\mathbf{e}$. The total variance is the variance of $\mathbf{x}_{i}$.

## > Example 1

Say that in our automobile data, the weight of cars was measured with error, and the reliability of our measured weight is 0.85 . The result of this would be to underestimate the effect of weight in a regression of, say, price on weight and foreign, and it would also bias the estimate of the coefficient on foreign (because being of foreign manufacture is correlated with the weight of cars). We would ignore all of this if we fit the model with regress:

| Source | SS | df | MS | Number of obs F (2, 71) <br> Prob > F <br> R-squared <br> Adj R-squared <br> Root MSE |  | 74 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Model | 316859273 | 2 | 158429637 |  |  | 0.0000 |
| Residual | 318206123 | 71 | 4481776.38 |  |  | 0.4989 |
| Total | 635065396 | 73 | 8699525.97 |  |  | $\begin{array}{r} 0.4848 \\ 2117 \end{array}$ |
| price | Coefficient | Std. err. | t | $P>\|t\|$ | [95\% conf. interval] |  |
| weight | 3.320737 | . 3958784 | 8.39 | 0.000 | 2.531378 | 4.110096 |
| foreign | 3637.001 | 668.583 | 5.44 | 0.000 | 2303.885 | 4970.118 |
| _cons | -4942.844 | 1345.591 | -3.67 | 0.000 | -7625.876 | -2259.812 |

With eivreg, we can account for our measurement error:

| . eivreg price weight foreign, reliab(weight .85) |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Variable | $\begin{array}{r} \text { Assumed } \\ \text { reliability } \end{array}$ |  |  |  |  |  |
|  |  |  |  | Number of | obs | 74 |
| weight | 0.85 |  |  | F ( 2 , | 71) | 18.46 |
| * | 1.00 |  |  | Prob > F |  | 0.0000 |
|  |  |  |  | R -squared |  | 0.6483 |
|  |  |  |  | Root MSE |  | 1773.54 |
| price | Coefficient | Std. err. | t | $P>\|t\|$ | [95\% conf | interval] |
| weight | 4.31985 | . 7134251 | 6.06 | 0.000 | 2.89732 | 5.742379 |
| foreign | 4637.32 | 849.0221 | 5.46 | 0.000 | 2944.418 | 6330.222 |
| _cons | -8257.017 | 2390.337 | -3.45 | 0.001 | -13023.21 | -3490.821 |

The effect of weight is increased - as we knew it would be-and here the effect of foreign manufacture is also increased. A priori, we knew only that the estimate of foreign might be biased; we did not know the direction.

## - Technical note

Swept under the rug in our example is how we would determine the reliability, $r$. We can easily see that a variable is measured with error, but we may not know the reliability because the ingredients for calculating $r$ depend on the unobserved noise.

For our example, we made up a value for $r$, and in fact we do not believe that weight is measured with error at all, so the reported eivreg results have no validity. The regress results were the statistically correct results here.

But let's say that we do suspect that weight is measured with error and that we do not know $r$. We could then experiment with various values of $r$ to describe the sensitivity of our estimates to possible error levels. We may not know $r$, but $r$ does have a simple interpretation, and we could probably produce a sensible range for $r$ by thinking about how the data were collected.

If the reliability, $r$, is less than the $R^{2}$ from a regression of the poorly measured variable on all the other variables, including the dependent variable, the information might as well not have been collected; no adjustment to the final results is possible. For our automobile data, running a regression of weight on foreign and price would result in an $R^{2}$ of 0.6743 . Thus, the reliability must be at least 0.6743 here. If we specify a reliability that is too small, eivreg will inform us and refuse to fit the model:

```
. eivreg price weight foreign, reliab(weight .6742)
reliability r() too small
r(399);
```

Returning to our problem of how to estimate $r$, too small or not, if the measurements are summaries of scaled items, the reliability may be estimated using the alpha command; see [MV] alpha. If the score is computed from factor analysis and the data are scored using predict's default options (see [MV] factor postestimation), the square of the standard deviation of the score is an estimate of the reliability.

## $>$ Example 2

Consider a model with more than one variable measured with error. For instance, say that our model is that price is a function of weight, foreign, and mpg and that both weight and mpg are measured with error.

```
. eivreg price weight foreign mpg, reliab(weight . }85\textrm{mpg . 95)
```

Errors-in-variables regression
Assumed
Variable reliability

|  |  |  |  | Number of | obs | 74 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| weight | 0.85 |  |  | F ( 3, | 70) = | 9.58 |
| mpg | 0.95 |  |  | Prob > F | = | 0.0000 |
| * | 1.0000 |  |  | R-squared | = | 0.8522 |
|  |  |  |  | Root MSE | = | 1158.04 |
| price | Coefficient | Std. err. | t | $P>\|t\|$ | [95\% conf. interval] |  |
| weight | 9.69903 | 3.768985 | 2.57 | 0.012 | 2.182027 | 17.21603 |
| foreign | 6918.624 | 2259.531 | 3.06 | 0.003 | 2412.132 | 11425.12 |
| mpg | 627.6764 | 431.0284 | 1.46 | 0.150 | -231.9826 | 1487.335 |
| _cons | -38545.27 | 20960.72 | -1.84 | 0.070 | -80350.11 | 3259.564 |

## Stored results

eivreg stores the following in e() :
Scalars
e(N)
e(df_m)
e(df_r)
e(r2)
e(F)
e(rmse)
e(rank)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(rellist)
e(wtype)
e(wexp)
e(properties)
e(predict)
e(marginsok)
e(asbalanced)
e(asobserved)
Matrices
e(b)
e(V)
Functions
e(sample) marks estimation sample

In addition to the above, the following is stored in r() :
Matrices
r(table)
matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

Let the model to be fit be

$$
\begin{aligned}
\mathbf{y} & =\mathbf{X}^{*} \boldsymbol{\beta}+\boldsymbol{\epsilon} \\
\mathbf{X} & =\mathbf{X}^{*}+\mathbf{U}
\end{aligned}
$$

where $\mathbf{X}^{*}$ are the true values and $\mathbf{X}$ are the observed values. $\boldsymbol{\epsilon}$ and $\mathbf{U}$ are assumed to be independent and have zero means and finite fourth moments. $\operatorname{Var}(\mathbf{U})$ is assumed to be diagonal.

Let $\mathbf{W}$ be the user-specified weights. If no weights are specified, $\mathbf{W}=\mathbf{I}$. If weights are specified, let $\mathbf{v}$ be the specified weights. If fweight frequency weights are specified, then $\mathbf{W}=\operatorname{diag}(\mathbf{v})$. If aweight analytic weights are specified, then $\mathbf{W}=\operatorname{diag}\left\{\mathbf{v} /\left(\mathbf{1}^{\prime} \mathbf{v}\right)\left(\mathbf{1}^{\prime} \mathbf{1}\right)\right\}$, meaning that the weights are normalized to sum to the number of observations.

The estimates bof $\boldsymbol{\beta}$ are obtained as $\mathbf{A}^{-1} \mathbf{X}^{\prime} \mathbf{W} \mathbf{y}$, where $\mathbf{A}=\mathbf{X}^{\prime} \mathbf{W} \mathbf{X}-\mathbf{S} . \mathbf{S}$ is a diagonal matrix with elements $\left(1-r_{j}\right) s_{j}^{2}$. Here $r_{j}$ is the user-specified reliability coefficient for the $j$ th explanatory variable (or 1 if not specified), and $s_{j}^{2}$ is the (appropriately weighted) sample variance of the variable.

The root mean squared error is $\left(\mathbf{y}^{\prime} \mathbf{W} \mathbf{y}-\mathbf{b A} \mathbf{b}^{\prime}\right) /(n-p)$, where $n$ is the number of observations and $p$ is the number of estimated parameters. The variance-covariance matrix of the estimators is obtained based on the formulas provided in Stefanski and Boos (2002), Buonaccorsi (2010), and Fuller (1987). For each $i=1,2, \ldots, n$, let residual $e_{i}=y_{i}-\mathbf{x}_{i} \mathbf{b}$, where $\mathbf{x}_{i}$ is the $i$ th row of $\mathbf{X}$. Consider matrix $\mathbf{H}$, where the $i$ th row of $\mathbf{H}, \mathbf{h}_{i}$, is

$$
\mathbf{h}_{i}^{\prime}=\left(\begin{array}{c}
e_{i} x_{i 1}+\left(x_{i 1}-\bar{x}_{1}\right)^{2}\left(1-r_{1}\right) b_{1} \\
e_{i} x_{i 2}+\left(x_{i 2}-\bar{x}_{2}\right)^{2}\left(1-r_{2}\right) b_{2} \\
\vdots \\
e_{i} x_{i p}+\left(x_{i p}-\bar{x}_{p}\right)^{2}\left(1-r_{p}\right) b_{p}
\end{array}\right)
$$

where $\bar{x}_{j}$ is the weighted mean of the $j$ th variable.
If analytic weights, aweights, are specified, the variance-covariance matrix is $\mathbf{A}^{-1}$ $\mathbf{H}^{\prime} \mathbf{W W H A}{ }^{-1}$; otherwise, it is $\mathbf{A}^{-1} \mathbf{H}^{\prime} \mathbf{W H A} \mathbf{H}^{-1}$.

## References

Buonaccorsi, J. P. 2010. Measurement Error: Models, Methods, and Applications. Boca Raton, FL: CRC Press.
Draper, N., and H. Smith. 1998. Applied Regression Analysis. 3rd ed. New York: Wiley.
Erickson, T., R. Parham, and T. M. Whited. 2017. Fitting the errors-in-variables model using high-order cumulants and moments. Stata Journal 17: 116-129.

Fuller, W. A. 1987. Measurement Error Models. New York: Wiley.
Kmenta, J. 1997. Elements of Econometrics. 2nd ed. Ann Arbor: University of Michigan Press.
Lee, Y. J., and D. Wilhelm. 2020. Testing for the presence of measurement error in Stata. Stata Journal 20: 382-404.

Lockwood, J. R., and D. F. McCaffrey. 2020. Recommendations about estimating errors-in-variables regression in Stata. Stata Journal 20: 116-130.

Stefanski, L. A., and D. D. Boos. 2002. The calculus of M-estimation. American Statistician 56: 29-38. https://doi.org/10.1198/000313002753631330.
Treiman, D. J. 2009. Quantitative Data Analysis: Doing Social Research to Test Ideas. San Francisco: Jossey-Bass.

## Also see

[R] eivreg postestimation - Postestimation tools for eivreg
[R] regress - Linear regression
[SEM] Example 24 - Reliability
[U] 20 Estimation and postestimation commands

## Title

eivreg postestimation - Postestimation tools for eivreg
Postestimation commands predict margins Remarks and examples

## Postestimation commands

The following postestimation commands are available after eivreg:

| Command | Description |
| :--- | :--- |
| contrast | contrasts and ANOVA-style joint tests of estimates |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estimates | cataloging estimation results <br> etable |
| lable of estimation results |  |
| lincom | point estimates, standard errors, testing, and inference for linear combinations of <br> coefficients |
| margins | marginal means, predictive margins, marginal effects, and average marginal effects <br> marginsplot <br> nlcom |
| preint estimates, standard errors, testing, and inference for nonlinear combinations |  |
| predict | of coefficients <br> pwcompare |
| linear predictions |  |
| pairwise comparisons of estimates |  |$\quad$| Wald tests of simple and composite linear hypotheses |
| :--- |
| testnl |$\quad$| Wald tests of nonlinear hypotheses |
| :--- |

## predict

## Description for predict

predict creates a new variable containing the linear prediction assuming that values of the covariates used for the prediction were measured without error.

## Menu for predict

Statistics > Postestimation

## Syntax for predict

predict [type] newvar [if] [in]
Available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

## margins

## Description for margins

margins estimates margins of response for linear predictions.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

margins [marginlist] [, options]

## Remarks and examples

## > Example 1

We return to example 1 from [ R$]$ eivreg:

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. eivreg price weight foreign, reliab(weight .85)
Errors-in-variables regression
```

| Variable | Assumed |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: |
|  | reliability |  |  |  |  |
|  |  | Number of obs |  | $=$ | 74 |
| weight | 0.8500 | F ( 2 , | 71) | = | 18.46 |
| * | 1.0000 | Prob > F |  | = | 0.0000 |
|  |  | R-squared |  | = | 0.6483 |
|  |  | Root MSE |  | = | 1773.54 |


| price | Coefficient | Std. err. | t | P>\|t| | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | ---: | ---: |
| weight | 4.31985 | .7134251 | 6.06 | 0.000 | 2.89732 | 5.742379 |
| foreign | 4637.32 | 849.0221 | 5.46 | 0.000 | 2944.418 | 6330.222 |
| _cons | -8257.017 | 2390.337 | -3.45 | 0.001 | -13023.21 | -3490.821 |

We wish to predict the price of a foreign car that weighs 2,300 pounds. We can use predict because 2,300 pounds is the true weight, not the result of an error-prone measurement.

To make this prediction, first we add the new observation to the dataset.

```
. set obs 75
Number of observations (_N) was 74, now 75.
. replace foreign = 1 in 75
(1 real change made)
. replace weight = 2300 in 75
(1 real change made)
```

Now, we use predict to predict the price of the car.

```
. predict newprice in 75
(option xb assumed; fitted values)
(74 missing values generated)
(predictions assume covariates measured without error)
. list weight foreign newprice in 75
```

75. 

| weight | foreign | newprice |
| ---: | :--- | :--- |
| 2,300 | Foreign | 6315.957 |

predict issued a note reminding us that the computed predictions assume that the covariates used for prediction are measured without error. In general, you should avoid using predict to obtain in-sample predictions unless you first replace the measurement-error covariates with values that are error free.

## Also see

[R] eivreg - Errors-in-variables regression
[U] 20 Estimation and postestimation commands

## Title

Epitab — Tables for epidemiologists
Description
Options
Acknowledgments

Quick start<br>Remarks and examples<br>References

Menu
Stored results
Also see

Syntax
Methods and formulas

## Description

ir is used with incidence-rate (incidence-density or person-time) data. It calculates point estimates and confidence intervals for the incidence-rate ratio (IRR) and incidence-rate difference (IRD), along with attributable or prevented fractions for the exposed and total population. iri is the immediate form of ir; see [U] 19 Immediate commands. Also see [R] poisson and [ST] stcox for related commands.
cs is used with cohort study data with equal follow-up time per subject and sometimes with crosssectional data. Risk is then the proportion of subjects who become cases. It calculates point estimates and confidence intervals for the risk difference, risk ratio, and (optionally) the odds ratio, along with attributable or prevented fractions for the exposed and total population. csi is the immediate form of cs; see [U] 19 Immediate commands. Also see [R] logistic for related commands.
cc is used with case-control and cross-sectional data. It calculates point estimates and confidence intervals for the odds ratio, along with attributable or prevented fractions for the exposed and total population. cci is the immediate form of cc; see [U] 19 Immediate commands. Also see [R] logistic for related commands.
tabodds is used with case-control and cross-sectional data. It tabulates the odds of failure against a categorical explanatory variable expvar. If expvar is specified, tabodds performs an approximate $\chi^{2}$ test of homogeneity of odds and a test for linear trend of the log odds against the numerical code used for the categories of expvar. Both tests are based on the score statistic and its variance; see Methods and formulas. When expvar is absent, the overall odds are reported. The variable $v a r_{\text {case }}$ is coded $0 / 1$ for individual and simple frequency records and equals the number of cases for binomial frequency records.

Optionally, tabodds tabulates adjusted or unadjusted odds ratios, using either the lowest levels of expvar or a user-defined level as the reference group. If adjust (varlist) is specified, it produces odds ratios adjusted for the variables in varlist along with a (score) test for trend.
mhodds is used with case-control and cross-sectional data. It estimates the ratio of the odds of failure for two categories of expvar, controlled for specified confounding variables, vars adjust , and tests whether this odds ratio is equal to one. When expvar has more than two categories but none are specified with the compare () option, mhodds assumes that expvar is a quantitative variable and calculates a 1-degree-of-freedom test for trend. It also calculates an approximate estimate of the log odds-ratio for a one-unit increase in expvar. This is a one-step Newton-Raphson approximation to the maximum likelihood estimate calculated as the ratio of the score statistic, $U$, to its variance, $V$ (Clayton and Hills 1993, 103).
mec is used with matched case-control data. It calculates McNemar's $\chi^{2}$; point estimates and confidence intervals for the difference, ratio, and relative difference of the proportion with the factor; and the odds ratio and its confidence interval. mcci is the immediate form of mcc; see [U] 19 Immediate commands. Also see $[R]$ clogit and $[R]$ symmetry for related commands.

## Quick start

## Cohort studies

IRR and IRD for the number of cases stored in cases for exposure indicator exposed given time exposed time
ir cases exposed time
Crude and Mantel-Haenszel combined IRRs with test of homogeneity for strata defined by svar
ir cases exposed time, by (svar)
Same as above, and standardize the IRR by weighting variable wvar1
ir cases exposed time, by (svar) standard(wvar1)
Same as above, but use person-time of the unexposed group as weights
ir cases exposed time, by (svar) estandard
IRR and IRD for 10 cases over 50 person-years in the exposed group and 15 cases over 100 person-years in the unexposed group
iri 101550100
Risk difference and ratio with binary indicators case and exposed using cumulative incidence data cs case exposed [fweight=wvar2]

Add odds ratios and calculate Fisher's exact $p$
cs case exposed [fweight=wvar2], or exact
Internally standardized risk ratio for strata defined by svar
cs case exposed [fweight=wvar2], by(svar) istandard
Risk difference and ratio for 12 cases and 55 noncases among exposed subjects and 16 cases and 125 noncases among unexposed subjects
csi 121655125

Case-control studies
Odds ratios from summary data with binary indicators case and exposed and frequency weight wvar3
cc case exposed [fweight=wvar3]
Same as above, but stratify analysis by svar and perform Breslow-Day and Tarone's homogeneity tests
cc case exposed [fweight=wvar3], by(svar) bd tarone
Odds ratios for 37 exposed cases, 148 unexposed cases, 7 exposed controls, and 137 unexposed controls
cci 371487137
Odds of binary event against catvar using summary data with frequency weight wvar4 tabodds event catvar [fweight=wvar4]

Same as above, but report odds ratios with the fourth level of catvar as the reference
tabodds event catvar [fweight=wvar4], or base(4)

Same as above, but tabulate Mantel-Haenszel adjusted odds ratios adjusting for values of categorical variable a
tabodds event catvar [fweight=wvar4], base(4) adjust(a)
Graph odds and confidence intervals against categories of catvar tabodds event catvar [fweight=wvar4], ciplot

Odds ratios for the effect of catvar on event controlling for categorical variable a using summary data with frequency weight wvar5
mhodds event catvar a [fweight=wvar5]
Same as above, but calculate odds ratios for each level of svar
mhodds event catvar a [fweight=wvar5], by (svar)
Maximum likelihood estimate of odds ratio for a equal to 4 compared with a equal to 1
mhodds event a [fweight=wvar5], compare $(4,1)$
Statistics on the difference in the proportion with the factor for exposed cases indicated in expcase and exposed controls indicated in expcontrol using summary data with frequency weight wvar6 mcc expcase expcontrol [fweight=wvar6]

Same as above, but indicate that there are 4 pairs where both cases and controls were exposed, 9 pairs where the case was exposed but the control was not, 3 pairs where the control was exposed but the case was not, and 14 pairs where neither subject was exposed

## Menu

ir

Statistics $>$ Epidemiology and related $>$ Tables for epidemiologists $>$ Incidence-rate ratio

## iri

Statistics $>$ Epidemiology and related $>$ Tables for epidemiologists $>$ Incidence-rate-ratio calculator

## CS

Statistics $>$ Epidemiology and related $>$ Tables for epidemiologists $>$ Cohort study risk-ratio etc.

## csi

Statistics $>$ Epidemiology and related $>$ Tables for epidemiologists $>$ Cohort study risk-ratio etc. calculator

## cc

Statistics $>$ Epidemiology and related $>$ Tables for epidemiologists $>$ Case-control odds ratio

## cci

Statistics $>$ Epidemiology and related $>$ Tables for epidemiologists $>$ Case-control odds-ratio calculator

## tabodds

Statistics $>$ Epidemiology and related $>$ Tables for epidemiologists $>$ Tabulate odds of failure by category

## mhodds

Statistics $>$ Epidemiology and related $>$ Tables for epidemiologists $>$ Ratio of odds of failure for two categories

## mcc

Statistics $>$ Epidemiology and related $>$ Tables for epidemiologists $>$ Matched case-control studies

## mcci

Statistics $>$ Epidemiology and related $>$ Tables for epidemiologists $>$ Matched case-control calculator

## Syntax

Cohort studies

```
ir \(v a r_{\text {case }} v a r_{\text {exposed }} v a r_{\text {time }}[\) if \(][\) in \(][\) weight \(][\), ir_options \(]\)
iri \(\#_{a} \#_{b} \#_{N_{1}} \#_{N_{2}}[\), iri_options \(]\)
cs \(\mathrm{var}_{\text {case }} \mathrm{var}_{\text {exposed }}[\) if \(][\) in \(][\) weight \(][\), cs_options \(]\)
csi \(\#_{a} \#_{b} \#_{c} \#_{d}[\), csi_options \(]\)
```

Case-control studies

```
cc varcase var exposed [if][in] [weight] [, cc_options]
cci # a # # # # # #d [, cci_options]
tabodds varcase [expvar] [if] [in] [weight] [,tabodds_options]
mhodds varcase expvar [vars adjust ] [if] [in] [weight] [, mhodds_options]
```

Matched case-control studies

```
mcc var exposed_case var exposed_control [if] [in] [weight] [, level(#)]
mcci #a # b # c # #d [, level(#)]
```

ir_options Description
Options
by (varname [, missing]) stratify on varname
estandard combine external weights with within-stratum statistics
istandard
standard (varname)
pool
nocrude
nohom
ird
midp
exact
level(\#) combine internal weights with within-stratum statistics combine user-specified weights with within-stratum statistics display pooled estimate
do not display crude estimate
do not display homogeneity test
calculate standardized IRD
display $p$-values calculated using mid- $p$ adjustment (unstratified only); the default
display exact $p$-values without mid- $p$ adjustment (unstratified only)
set confidence level; default is level (95)
iri_options
Description
midp
exact
level(\#)
display $p$-values calculated using mid- $p$ adjustment; the default display exact $p$-values without mid- $p$ adjustment set confidence level; default is level (95)

Options

```
by(varlist[, missing]) stratify on varlist
estandard
istandard
standard(varname)
pool
nocrude
nohom
rd
b_inomial(varname)
or
woolf
exact
level(#)
```

stratify on varlist
combine external weights with within-stratum statistics
combine internal weights with within-stratum statistics
combine user-specified weights with within-stratum statistics
display pooled estimate
do not display crude estimate
do not display homogeneity test
calculate standardized risk difference
number of subjects variable
report odds ratio
use Woolf approximation to calculate SE and CI of the odds ratio calculate Fisher's exact $p$
set confidence level; default is level (95)
csi_options
or
woolf
exact
level(\#)

Description
report odds ratio
use Woolf approximation to calculate SE and CI of the odds ratio calculate Fisher's exact $p$
set confidence level; default is level (95)
cc_options
Description
Options
by (varname [, missing]) stratify on varname
estandard
istandard
standard (varname)
pool
nocrude
nohom
bd
tarone
binomial (varname)
cornfield
woolf
exact
level(\#)
combine external weights with within-stratum statistics
combine internal weights with within-stratum statistics
combine user-specified weights with within-stratum statistics
display pooled estimate
do not display crude estimate
do not display homogeneity test
perform Breslow-Day homogeneity test
perform Tarone's homogeneity test
number of subjects variable
use Cornfield approximation to calculate CI of the odds ratio
use Woolf approximation to calculate SE and CI of the odds ratio
calculate Fisher's exact $p$
set confidence level; default is level (95)

Description
cornfield
woolf
exact
level(\#)
use Cornfield approximation to calculate CI of the odds ratio
use Woolf approximation to calculate SE and CI of the odds ratio
calculate Fisher's exact $p$
set confidence level; default is level (95)
tabodds_options
Main
binomial (varname)
level(\#)
or
adjust (varlist)
base(\#)
cornfield
woolf
graph
ciplot
CI plot
ciopts(rcap_options)
Plot
marker_options
marker_label_options
cline_options
Add plots
addplot (plot) add other plots to the generated graph

Y axis, X axis, Titles, Legend, Overall
twoway_options
any options other than by () documented in [G-3] twoway_options
mhodds_options
Description

## Options

by (varlist[, missing])
binomial (varname)
compare ( $v_{1}, v_{2}$ )
level(\#)
number of subjects variable
set confidence level; default is level (95)
report odds ratio
report odds ratios adjusted for the variables in varlist
reference group of control variable for odds ratio
use Cornfield approximation to calculate CI of the odds ratio
use Woolf approximation to calculate SE and CI of the odds ratio
graph odds against categories
same as graph option, except include confidence intervals
affect rendition of the confidence bands
change look of markers (color, size, etc.)
add marker labels; change look or position
affect rendition of the plotted points
collect is allowed with ir, iri, cs, csi, cc, cci, tabodds, mhodds, mcc, and mcci; see [U] 11.1.10 Prefix commands.
fweights are allowed; see [U] 11.1.6 weight.

## Options

Options are listed in the order that they appear in the syntax tables above. The commands for which the option is valid are indicated in parentheses immediately after the option name.

Options (ir, cs, cc, and mhodds)/Main (tabodds)
by (varname [, missing]) (ir, cs, cc, and mhodds) specifies that the tables be stratified on varname. Missing categories in varname are omitted from the stratified analysis, unless option missing is specified within by (). Within-stratum statistics are shown and then combined with Mantel-Haenszel weights. If estandard, istandard, or standard() is also specified (see below), the weights specified are used in place of Mantel-Haenszel weights.
estandard, istandard, and standard(varname) (ir, cs, and cc) request that within-stratum statistics be combined with external, internal, or user-specified weights to produce a standardized estimate. These options are mutually exclusive and can be used only when by () is also specified. (When by () is specified without one of these options, Mantel-Haenszel weights are used.)
estandard external weights are the person-time for the unexposed (ir), the total number of unexposed (cs), or the number of unexposed controls (cc).
istandard internal weights are the person-time for the exposed (ir), the total number of exposed (cs), or the number of exposed controls (cc). istandard can be used to produce, among other things, standardized mortality ratios (SMRs).
standard (varname) allows user-specified weights. varname must contain a constant within stratum and be nonnegative. The scale of varname is irrelevant.
pool (ir, cs, and cc) specifies that, in a stratified analysis, the directly pooled estimate also be displayed. The pooled estimate is a weighted average of the stratum-specific estimates using inverse-variance weights, which are the inverse of the variance of the stratum-specific estimate. pool is relevant only if by () is also specified.
nocrude (ir, cs, and cc) specifies that in a stratified analysis the crude estimate-an estimate obtained without regard to strata-not be displayed. nocrude is relevant only if by () is also specified.
nohom (ir, cs, and cc) specifies that a $\chi^{2}$ test of homogeneity not be included in the output of a stratified analysis. This tests whether the exposure effect is the same across strata and can be performed for any pooled estimate-directly pooled or Mantel-Haenszel. nohom is relevant only if by () is also specified.
ird (ir) may be used only with estandard, istandard, or standard(). It requests that ir calculate the standardized IRD rather than the default IRR.
midp (ir without by () and iri), the default, displays mid- $p$-adjusted $p$-values for one-sided and two-sided tests of IRD. The tests of IRD are not available with ir for stratified analysis, so midp is not allowed in combination with by (). Only one of exact or midp may be specified.
exact (ir without by() and iri) displays exact $p$-values for one-sided and two-sided tests of IRD instead of the default mid- $p$-adjusted $p$-values. This option produces $p$-values that are more conservative than the mid- $p$-adjusted $p$-values. When counts of exposed and unexposed cases are both large, exact and midp give similar results. The tests of IRD are not available with ir for stratified analysis, so exact is not allowed in combination with by (). Only one of exact or midp may be specified.
rd (cs) may be used only with estandard, istandard, or standard(). It requests that cs calculate the standardized risk difference rather than the default risk ratio.
bd (cc) specifies that Breslow and Day's $\chi^{2}$ test of homogeneity be included in the output of a stratified analysis. This tests whether the exposure effect is the same across strata. bd is relevant only if by () is also specified.
tarone (cc) specifies that Tarone's $\chi^{2}$ test of homogeneity, which is a correction to the Breslow-Day test, be included in the output of a stratified analysis. This tests whether the exposure effect is the same across strata. tarone is relevant only if by () is also specified.
binomial (varname) (cs, cc, tabodds, and mhodds) supplies the number of subjects (cases plus controls) for binomial frequency records. For individual and simple frequency records, this option is not used.
or (cs, csi, and tabodds), for cs and csi, reports the calculation of the odds ratio in addition to the risk ratio if by() is not specified. With by (), or specifies that a Mantel-Haenszel estimate of the combined odds ratio be made rather than the Mantel-Haenszel estimate of the risk ratio. In either case, this is the same calculation that would be made by cc and cci. Typically, cc, cci, or tabodds is preferred for calculating odds ratios. For tabodds, or specifies that odds ratios be produced; see base() for details about selecting a reference category. By default, tabodds will calculate odds.
adjust (varlist) (tabodds) specifies that odds ratios adjusted for the variables in varlist be calculated.
base(\#) (tabodds) specifies that the \#th category of expvar be used as the reference group for calculating odds ratios. If base () is not specified, the first category, corresponding to the minimum value of expvar, is used as the reference group.
cornfield (cc, cci, and tabodds) requests that the Cornfield (1956) approximation be used to calculate the confidence interval of the odds ratio. By default, cc and cci report an exact interval and tabodds reports a standard-error-based interval, with the standard error coming from the square root of the variance of the score statistic.
woolf (cs, csi, cc, cci, and tabodds) requests that the Woolf (1955) approximation, also known as the Taylor expansion, be used for calculating the standard error and confidence interval for the odds ratio. By default, cs and csi with the or option report the Cornfield (1956) interval; cc and cci report an exact interval; and tabodds reports a standard-error-based interval, with the standard error coming from the square root of the variance of the score statistic.
exact (cs, csi, cc, and cci) requests that Fisher's exact $p$ be calculated rather than the $\chi^{2}$ and its significance level. We recommend specifying exact whenever samples are small. When the least-frequent cell contains 1,000 cases or more, there will be no appreciable difference between the exact significance level and the significance level based on the $\chi^{2}$, but the exact significance level will take considerably longer to calculate. exact does not affect whether exact confidence intervals are calculated. Commands always calculate exact confidence intervals where they can, unless cornfield or woolf is specified.
compare ( $v_{1}, v_{2}$ ) (mhodds) indicates the categories of expvar to be compared; $v_{1}$ defines the numerator and $v_{2}$, the denominator. When compare() is not specified and there are only two categories, the second is compared with the first; when there are more than two categories, an approximate estimate of the odds ratio for a unit increase in expvar, controlled for specified confounding variables, is given.
level(\#) (ir, iri, cs, csi, cc, cci, tabodds, mhodds, mcc, and mcci) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [ $R$ ] level.

The following options are for use only with tabodds.
$\qquad$ Main
graph (tabodds) produces a graph of the odds against the numerical code used for the categories of expvar. All graph options except connect () are allowed. This option is not allowed with the or option or the adjust() option.
ciplot (tabodds) produces the same plot as the graph option, except that it also includes the confidence intervals. This option may not be used with either the or option or the adjust() option.
$\qquad$ Cl plot
ciopts(rcap_options) (tabodds) is allowed only with the ciplot option. It affects the rendition of the confidence bands; see [G-3] rcap_options.
$\qquad$ Plot
marker_options (tabodds) affect the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G-3] marker_options.
marker_label_options (tabodds) specify if and how the markers are to be labeled; see [G-3] marker_label_options.
cline_options (tabodds) affect whether lines connect the plotted points and the rendition of those lines; see [G-3] cline_options.

Add plots
addplot (plot) (tabodds) provides a way to add other plots to the generated graph; see [G-3] addplot_option.

Y axis, X axis, Titles, Legend, Overall
twoway_options (tabodds) are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and options for saving the graph to disk (see [G-3] saving_option).

## Remarks and examples

Remarks are presented under the following headings:

[^21]To calculate appropriate statistics and suppress inappropriate statistics, the ir, cs, cc, tabodds, mhodds, and mcc commands, along with their immediate counterparts, are organized in the way epidemiologists conceptualize data. ir processes incidence-rate data from prospective studies; cs, cohort study data with equal follow-up time (cumulative incidence); cc, tabodds, and mhodds, case-control or cross-sectional (prevalence) data; and mcc, matched case-control data. With the exception of mcc, these commands work with both simple and stratified tables.

Epidemiological data are often summarized in a contingency table from which various statistics are calculated. The rows of the table reflect cases and noncases or cases and person-time, and the columns reflect exposure to a risk factor. To an epidemiologist, cases and noncases refer to the outcomes of the process being studied. For instance, a case might be a person with cancer and a noncase might be a person without cancer.

A factor is something that might affect the chances of being ultimately designated a case or a noncase. Thus, a case might be a cancer patient, and the factor might be smoking behavior. A person is said to be exposed or unexposed to the factor. Exposure can be classified as a dichotomy, smokes or does not smoke, or as multiple levels, such as number of cigarettes smoked per week.

For an introduction to epidemiological methods, see Walker (1991). For an intermediate treatment, see Clayton and Hills (1993) and Schneider and Lilienfeld (2015). For other advanced discussions, see Kleinbaum, Kupper, and Morgenstern (1982) and Lash et al. (2021). For an analysis of incidence rates, see, for instance, Cummings (2019). For an anthology of writings on epidemiology since World War II, see Greenland (1987). See Jewell (2004) for a text aimed at graduate students in the medical professions that uses Stata for much of the analysis. See Dohoo, Martin, and Stryhn (2010) for a graduate-level text on the principles and methods of veterinary epidemiologic research; Stata datasets and do-files are available. Also see Dohoo, Martin, and Stryhn (2012) for a text that is a revision of their veterinary epidemiology text, but examples from human epidemiology are used.

## Incidence-rate data

In incidence-rate data from a prospective study, you observe the transformation of noncases into cases. Starting with a group of noncase subjects, you monitor them to determine whether they become cases (for example, stricken with cancer). You monitor two populations: those exposed and those unexposed to the factor (for example, multiple X-rays). A summary of the data is

|  | Exposed | Unexposed | Total |
| :--- | :---: | :---: | :---: |
| Cases | $a$ | $b$ | $a+b$ |
| Person-time | $N_{1}$ | $N_{0}$ | $N_{1}+N_{0}$ |

$>$ Example 1: iri
It will be easiest to understand these commands if we start with the immediate forms. Remember, in the immediate form, we specify the data on the command line rather than specifying names of variables containing the data; see [U] 19 Immediate commands. We have data (Boice and Monson [1977]; reported in Lash et al. [2021, 408]) on breast cancer cases and person-years of observation for women with tuberculosis repeatedly exposed to multiple X-ray fluoroscopies, and those not so exposed:

|  | X-ray fluoroscopy |  |
| :--- | ---: | ---: |
| Exposed | Unexposed |  |
| Breast cancer cases | 41 | 15 |
| Person-years | 28,010 | 19,017 |

Using iri, the immediate form of ir, we specify the values in the table following the command:

iri shows the table, reports the incidence rates for the exposed and unexposed populations, and then shows the point estimates of the difference and ratio of the two incidence rates along with their confidence intervals. The incidence rate is simply the frequency with which noncases are transformed into cases.

Next, iri reports the attributable fraction among the exposed (AFE), an estimate of the proportion of exposed cases attributable to exposure. We estimate that $46.1 \%$ of the 41 breast cancer cases among the exposed were due to exposure. (Had the IRR been less than 1, iri would have reported the prevented fraction among the exposed (PFE), an estimate of the net proportion of all potential cases in the exposed population that was prevented by exposure; see the following technical note.)

After that, the table shows the attributable fraction for the population (AFP), which is the net proportion of all cases attributable to exposure. This number, of course, depends on the proportion of cases that are exposed in the base population, which here is taken to be $41 / 56$ and may not be relevant in all situations. We estimate that $33.8 \%$ of the 56 cases were due to exposure. We estimate that 18.9 cases were caused by exposure; that is, $0.338 \times 56=0.461 \times 41=18.9$.

At the bottom of the table, iri reports one- and two-sided tests of the IRD. For the one-sided test of the number of exposed cases being 41 or greater, the $p$-value is 0.0177 . The two-sided test is twice the smallest one-sided $p$-value and is 0.0355 . These $p$-values are calculated using the mid- $p$ adjustment to exact $p$-values.

Exact $p$-values can be seen by specifying the exact option.


The exact $p$-values are slightly larger than those calculated using the mid- $p$ adjustment. This is always the case. However, when counts of exposed and unexposed cases are both large, they will be nearly identical. See Methods and formulas below.

## Technical note

When the IRR is less than 1, iri (and ir, cs, csi, cc, and cci) substitutes the prevented fraction for the attributable fraction. Let's reverse the roles of exposure in the above data, treating as exposed a person who did not receive the X-ray fluoroscopy. You can think of this as a new treatment for preventing breast cancer-the suggested treatment being not to use fluoroscopy.
. iri 15411901728010
Incidence-rate comparison


The PFE is the net proportion of all potential cases in the exposed population that were prevented by exposure. We estimate that $46.1 \%$ of potential cases among the women receiving the new "treatment" were prevented by the treatment. (Previously, we estimated that the same percentage of actual cases among women receiving the X-rays was caused by the X-rays.)

The prevented fraction for the population (PFP), which is the net proportion of all potential cases in the total population that was prevented by exposure, as with the attributable fraction, depends on the proportion of cases that are exposed in the base population - here taken as $15 / 56$-so it may not be relevant in all situations. We estimate that $18.6 \%$ of the potential cases were prevented by exposure.

See Greenland and Robins (1988) for a discussion of how to interpret attributable and prevented fractions.

## Example 2: ir

ir works like iri, except that it obtains the entries in the tables by summing data. You specify three variables: the first represents the number of cases represented by this observation, the second indicates whether the observation is for subjects exposed to the factor, and the third records the total time the subjects in this observation were observed. An observation may reflect one subject or a group of subjects.

For instance, here is a 2-observation dataset for the table in the previous example:
. use https://www.stata-press.com/data/r18/irxmpl
. list

|  | cases | exposed | time |
| :--- | ---: | ---: | ---: |
| 1. | 41 | 0 | 28010 |
| 2. | 15 | 19017 |  |

If we typed ir cases exposed time, we would obtain the same output that we obtained above. Another way the data might be recorded is

```
. use https://www.stata-press.com/data/r18/irxmpl2
```

. list

Here the first 2 observations will be automatically summed by ir because both are exposed. Finally, the data might be individual-level data:

```
. use https://www.stata-press.com/data/r18/irxmpl3
. list in 1/5
```

|  | cases | exposed | time |
| :--- | ---: | ---: | ---: |
| 1. | 1 | 1 | 10 |
| 2. | 0 | 1 | 8 |
| 3. | 0 | 0 | 9 |
| 4. | 1 | 0 | 2 |
| 5. | 0 | 1 | 1 |

The first observation represents a woman who got cancer, was exposed, and was observed for 10 years. The second is a woman who did not get cancer, was exposed, and was observed for 8 years, and so on.

## - Technical note

ir (and all the other commands) assumes that a subject was exposed if the exposed variable is nonzero and not missing, assumes the subject was not exposed if the variable is zero, and ignores the observation if the variable is missing. For ir, the case variable and the time variable are restricted to nonnegative integers and are summed within the exposed and unexposed groups to obtain the entries in the table.

## Stratified incidence-rate data

## Example 3: ir with stratified data

ir can work with stratified tables, as well as with single tables. For instance, Rothman (1986, 185) discusses data from Rothman and Monson (1973) on mortality by sex and age for patients with trigeminal neuralgia:

|  | Age through 64 |  | Age 65+ |  |
| :--- | ---: | ---: | ---: | ---: |
|  | Males | Females | Males | Females |
| Deaths | 14 | 10 | 76 | 121 |
| Person-years | 1516 | 1701 | 949 | 2245 |

Entering the data into Stata, we have the following dataset:

```
. use https://www.stata-press.com/data/r18/rm
(Rothman and Monson 1973 data)
. list
```

| age | male | deaths | pyears |
| ---: | ---: | ---: | ---: |
| $<65$ | Male | 14 | 1516 |
| $<65$ | Female | 10 | 1701 |
| $65+$ | Male | 76 | 949 |
| $65+$ | Female | 121 | 2245 |

The stratified analysis of the IRR is

```
. ir deaths male pyears, by(age)
Stratified incidence-rate analysis
```

| Age category | IRR | [95\% conf. interval] | M-H weight |  |
| ---: | ---: | ---: | ---: | ---: |
| <65 | 1.570844 | .6489373 | 3.952809 | 4.712465 (exact) |
| $65+$ | 1.485862 | 1.100305 | 1.99584 | 35.95147 (exact) |
|  |  |  |  |  |
| Crude | 1.099794 | .831437 | 1.449306 |  |
| M-H combined | 1.49571 | 1.141183 | 1.960377 |  |

```
Test of homogeneity (M-H): chi2(1) = 0.02 Pr>chi2 = 0.8992
```

The row labeled $\mathrm{M}-\mathrm{H}$ combined reflects the combined Mantel-Haenszel estimates.
As with the previous example, it is not important that each entry in the table correspond to 1 observation in the data-ir sums the time (pyears) and case (deaths) variables within the exposure (male) category.

The difference between the unadjusted crude estimate and the Mantel-Haenszel estimate suggests confounding by age: women in the study are older, and older patients are more likely to die. But we should not use the Mantel-Haenszel estimate without checking its homogeneity assumption. The $\chi^{2}$ test of homogeneity gives a $p$-value of 0.8992 , so we have no evidence that the exposure effect (the effect of being male) differs across age categories. We are justified in using the Mantel-Haenszel estimate.

## - Technical note

Stratification is one way to deal with confounding; that is, perhaps sex affects the incidence of trigeminal neuralgia and so does age, so the table was stratified by age in an attempt to uncover the sex effect. (We are concerned that age may confound the true association between sex and the incidence of trigeminal neuralgia because the age distributions are so different for males and females. If age affects incidence, the difference in the age distributions would induce different incidences for males and females and thus confound the true effect of sex.)

We do not, however, have to use tables to uncover effects; the estimation alternative when we have aggregate data is Poisson regression, and we can use the same data on which we ran ir with poisson. Poisson regression also works with individual-level data.
(Although age in the previous example appears to be a string, it is actually a numeric variable taking on values 1 and 2 . We attached a value label to produce the labels <65 and $65+$ to make ir's output look better; see [U] 12.6.3 Value labels. Stata's estimation commands will ignore this labeling.)

```
. poisson deaths male age, exposure(pyears) irr
Iteration 0: Log likelihood = -10.836732
Iteration 1: Log likelihood = -10.734087
Iteration 2: Log likelihood = -10.733944
Iteration 3: Log likelihood = -10.733944
```

Poisson regression $\quad$ Number of obs $=\quad 4$
LR chi2(2) $=164.01$
Prob > chi2 $=0.0000$
Pseudo R2 $=0.8843$

Log likelihood = -10.733944

| Number of obs | $=4$ |
| :--- | :--- |
| LR chi2(2) | $=164.01$ |
| Prob > chi2 | $=0.0000$ |
| Pseudo R2 | $=0.8843$ |


| deaths | IRR | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | :---: | ---: |
| male | 1.495096 | .2060997 | 2.92 | 0.004 | 1.141118 | 1.95888 |
| age | 8.888775 | 1.934943 | 10.04 | 0.000 | 5.801616 | 13.61867 |
| _cons | .0006805 | .0002908 | -17.07 | 0.000 | .0002945 | .0015724 |
| ln(pyears) | 1 | (exposure) |  |  |  |  |

Note: _cons estimates baseline incidence rate.
Compare these results with the Mantel-Haenszel estimates produced by ir:

| Source | IRR | 95\% conf. interval |  |
| :--- | :---: | :---: | :---: |
| Mantel-Haenszel (ir) | 1.50 | 1.14 | 1.96 |
| poisson | 1.50 | 1.14 | 1.96 |

The results from poisson agree with the Mantel-Haenszel estimates to two decimal places. But poisson also estimates an IRR for age. Here the estimate is not of much interest, because the outcome variable is total mortality and we already knew that older people have a higher mortality rate. In other contexts, however, the estimate might be of greater interest.

See [R] poisson for an explanation of the poisson command.

## - Technical note

Both the model fit above and the preceding table asserted that exposure effects are the same across age categories and, if they are not, then both of the previous results are equally inappropriate. The table presented a test of homogeneity, reassuring us that the exposure effects do indeed appear to be constant. The Poisson-regression alternative can be used to reproduce that test by including interactions between the age groups and exposure:

```
. poisson deaths male age male#c.age, exposure(pyears) irr
Iteration 0: Log likelihood = -10.898799
Iteration 1: Log likelihood = -10.726225
Iteration 2: Log likelihood = -10.725904
Iteration 3: Log likelihood = -10.725904
```

Poisson regression
Log likelihood $=-10.725904$

| Number of obs | $=4$ |
| :--- | :--- | ---: |
| LR chi2(3) | $=164.03$ |
| Prob > chi2 | $=0.0000$ |
| Pseudo R2 | $=0.8843$ |


| deaths | IRR | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| male | 1.660688 | 1.396496 | 0.60 | 0.546 | .3195218 | 8.631283 |
| age | 9.167973 | 3.01659 | 6.73 | 0.000 | 4.810583 | 17.47226 |
| male\#c.age |  |  |  |  |  |  |
| Male | .9459 | .41539 | -0.13 | 0.899 | .3999832 | 2.236911 |
| _cons | .0006412 | .0004097 <br> (exposure) | -11.51 | 0.000 | .0001833 | .0022434 |

Note: _cons estimates baseline incidence rate.
The significance level of the male\#c.age effect is 0.899 , the same as previously reported by ir.
Here forming the male-times-age interaction was easy because there were only two age groups. Had there been more groups, the test would have been slightly more difficult-see the following technical note.

## - Technical note

A word of caution is in order when applying poisson (or any estimation technique) to more than two age categories. Say that in our data, we had three age categories, which we will call categories 0,1 , and 2 , and that they are stored in the variable agecat. We might think of the categories as corresponding to age less than $35,35-64$, and 65 and above.

With such data, we might type ir deaths male pyears, by (agecat), but we would not type poisson deaths male agecat, exposure(pyears) to obtain the equivalent Poisson-regression estimated results. Such a model might be reasonable, but it is not equivalent because we would be constraining the age effect in category 2 to be (multiplicatively) twice the effect in category 1.

To poisson (and all of Stata's estimation commands other than anova), agecat is simply one variable, and only one estimated coefficient is associated with it. Thus, the model is

$$
\text { Poisson index }=P=\beta_{0}+\beta_{1} \text { male }+\beta_{2} \text { agecat }
$$

The expected number of deaths is then $e^{P}$, and the IRR associated with a variable is $e^{\beta}$; see $[\mathrm{R}]$ poisson. Thus, the value of the Poisson index when male $==0$ and agecat $==1$ is $\beta_{0}+\beta_{2}$, and the possibilities are

|  | male $==0$ | mal $e==1$ |
| :--- | :---: | :---: |
| agecat $==0$ | $\beta_{0}$ | $\beta_{0}+\beta_{1}$ |
| agecat $==1$ | $\beta_{0}+\beta_{2}$ | $\beta_{0}+\beta_{2}+\beta_{1}$ |
| agecat $==2$ | $\beta_{0}+2 \beta_{2}$ | $\beta_{0}+2 \beta_{2}+\beta_{1}$ |

The age effect for agecat $=2$ is constrained to be twice the age effect for agecat $==1$ - the only difference between lines 3 and 2 of the table is that $\beta_{2}$ is replaced with $2 \beta_{2}$. Under certain circumstances, such a constraint might be reasonable, but it does not correspond to the assumptions made in generating the Mantel-Haenszel combined results.

To obtain results equivalent to the Mantel-Haenszel result, we must estimate a separate effect for each age group, meaning that we must replace $2 \beta_{2}$, the constrained effect, with $\beta_{3}$, a new coefficient that is free to take on any value. We can achieve this by creating two new variables and using them in place of agecat. agecat 1 will take on the value 1 when agecat is 1 and 0 otherwise; agecat 2 will take on the value 1 when agecat is 2 and 0 otherwise:

```
. generate agecat1 = (agecat==1)
. generate agecat2 = (agecat==2)
. poisson deaths male agecat1 agecat2 [fweight=pop], exposure(pyears) irr
```

In Stata, we do not have to generate these variables for ourselves. We could use factor variables:

```
. poisson deaths male i.agecat [fweight=pop], exposure(pyears) irr
```

See [U] 11.4.3 Factor variables.
To reproduce the homogeneity test with multiple age categories, we could type

```
. poisson deaths agecat##male [fweight=pop], exp(pyears) irr
. testparm agecat#male
```

Poisson regression combined with factor variables generalizes to multiway tables. Suppose that there are three exposure categories. Assume exposure variable burn takes on the values 1, 2, and 3 for first-, second-, and third-degree burns. The table itself is estimated by typing

```
. poisson deaths i.burn i.agecat [fweight=pop], exp(pyears) irr
```

and the test of homogeneity is estimated by typing
. poisson deaths burn\#\#agecat [fweight=pop], exp(pyears) irr
. testparm burn\#agecat

## Standardized estimates with stratified incidence-rate data

The by () option specifies that the data are stratified and, by default, will produce a Mantel-Haenszel combined estimate of the IRR. With the estandard, istandard, or standard(varname) options, you can specify your own weights and obtain standardized estimates of the IRR or IRD.
> Example 4: ir with stratified data, using standardized estimates
Lash et al. $(2021,417)$ report results from Doll and Hill (1966) on age-specific coronary disease deaths among British male doctors from cigarette smoking:

|  | Smokers |  | Nonsmokers |  |
| :--- | ---: | ---: | ---: | ---: |
| Age | Deaths | Person-years | Deaths | Person-years |
| $35-44$ | 32 | 52,407 | 2 | 18,790 |
| $45-54$ | 104 | 43,248 | 12 | 10,673 |
| $55-64$ | 206 | 28,612 | 28 | 5,710 |
| $65-74$ | 186 | 12,663 | 28 | 2,585 |
| $75-84$ | 102 | 5,317 | 31 | 1,462 |

We have entered these data into Stata:

```
. use https://www.stata-press.com/data/r18/dollhill3
(Doll and Hill (1966))
. list
```

|  | agecat | smokes | deaths | pyears |
| ---: | ---: | ---: | ---: | ---: |
| 1. | $35-44$ | 1 | 32 | 52,407 |
| 2. | $45-54$ | 1 | 104 | 43,248 |
| 3. | $55-64$ | 1 | 206 | 28,612 |
| 4. | $65-74$ | 1 | 186 | 12,663 |
| 5. | $75-84$ | 1 | 102 | 5,317 |
| 6. | $35-44$ | 0 | 2 | 18,790 |
| 7. | $45-54$ | 0 | 12 | 10,673 |
| 8. | $55-64$ | 0 | 28 | 5,710 |
| 9. | $65-74$ | 0 | 28 | 2,585 |
| 10. | $75-84$ | 0 | 31 | 1,462 |

We can obtain the Mantel-Haenszel combined estimate along with the crude estimate for ignoring stratification of the IRR and $90 \%$ confidence intervals by typing
. ir deaths smokes pyears, by(age) level(90)
Stratified incidence-rate analysis

| Age category | IRR | [90\% conf. interval] | M-H weight |  |
| ---: | ---: | ---: | ---: | ---: |
| $35-44$ | 5.736638 | 1.704271 | 33.61646 | 1.472169 (exact) |
| $45-54$ | 2.138812 | 1.274552 | 3.813282 | 9.624747 (exact) |
| $55-64$ | 1.46824 | 1.044915 | 2.110422 | 23.34176 (exact) |
| $65-74$ | 1.35606 | .9626026 | 1.953505 | 23.25315 (exact) |
| $75-84$ | .9047304 | .6375194 | 1.305412 | 24.31435 (exact) |
| Crude | 1.719823 | 1.437544 | 2.0688 |  |
| M-H combined | 1.424682 | 1.194375 | 1.699399 |  |

Test of homogeneity $(M-H): \operatorname{chi2}(4)=10.41 \quad \operatorname{Pr}>\operatorname{chi} 2=0.0340$
Note the presence of heterogeneity revealed by the test; the effect of smoking is not the same across age categories. Moreover, the listed stratum-specific estimates show an effect that appears to be declining with age. (Even if the test of homogeneity is not significant, you should always examine estimates carefully when stratum-specific effects occur on both sides of 1 for ratios and 0 for differences.)

Lash et al. $(2021,422)$ obtain the standardized IRR and $90 \%$ confidence intervals, weighting each age category by the population of the exposed group, thus producing the standardized mortality ratio (SMR). This calculation can be reproduced by specifying by (age) to indicate that the table is stratified and istandard to specify that we want the internally standardized rate. We may also specify that we would like to see the pooled estimate (weighted average where the weights are based on the variance of the strata calculations):
. ir deaths smokes pyears, by(age) level(90) istandard pool
Stratified incidence-rate analysis

| Age category | IRR | [90\% conf. interval] | Weight |  |
| ---: | ---: | :---: | ---: | ---: |
| $35-44$ | 5.736638 | 1.704271 | 33.61646 | 52407 (exact) |
| $45-54$ | 2.138812 | 1.274552 | 3.813282 | 43248 (exact) |
| $55-64$ | 1.46824 | 1.044915 | 2.110422 | 28612 (exact) |
| $65-74$ | 1.35606 | .9626026 | 1.953505 | 12663 (exact) |
| $75-84$ | .9047304 | .6375194 | 1.305412 | 5317 (exact) |
| Crude | 1.719823 | 1.437544 | 2.0688 |  |
| Pooled (direct) | 1.355343 | 1.134356 | 1.619382 |  |
| I. standardized | 1.417609 | 1.186541 | 1.693676 |  |
| Test of homogeneity (direct): chi2 (4) $=10.20$ | Pr>chi2 $=0.0372$ |  |  |  |

We obtained the simple pooled results because we specified the pool option. Note the significance of the homogeneity test; it provides the motivation for standardizing the rate ratios.

If we wanted the externally standardized ratio (weights proportional to the population of the unexposed group), we would substitute estandard for istandard in the above command.

We are not limited to IRRs; ir can also estimate IRDs. Differences may be standardized internally or externally. We will obtain the internally weighted difference (Lash et al. 2021, 418-419):
. ir deaths smokes pyears, by(age) level(90) istandard ird
Stratified incidence-rate analysis

| Age category | IRD | [90\% conf. interval] | Weight |  |
| ---: | ---: | ---: | ---: | ---: |
| $35-44$ | .0005042 | .0002877 | .0007206 | 52407 |
| $45-54$ | .0012804 | .0006205 | .0019403 | 43248 |
| $55-64$ | .0022961 | .0005628 | .0040294 | 28612 |
| $65-74$ | .0038567 | .0000521 | .0076614 | 12663 |
| $75-84$ | -.0020201 | -.0090201 | .00498 | 5317 |
| Crude | .0018537 | .001342 | .0023654 |  |
| I. standardized | .0013047 | .000712 | .0018974 |  |

## > Example 5: ir with user-specified weights

In addition to calculating results by using internal or external weights, ir (and cs and cc) can calculate results for arbitrary weights. If we wanted to obtain the IRR weighting each age category equally, we would type

```
. generate conswgt=1
. ir deaths smokes pyears, by(age) standard(conswgt)
Stratified incidence-rate analysis
```

| Age category | IRR | [95\% conf. interval] | Weight |  |
| ---: | ---: | ---: | ---: | ---: |
| $35-44$ | 5.736638 | 1.463557 | 49.40468 | 1 (exact) |
| $45-54$ | 2.138812 | 1.173714 | 4.272545 | 1 (exact) |
| $55-64$ | 1.46824 | .9863624 | 2.264107 | 1 (exact) |
| $65-74$ | 1.35606 | .9081925 | 2.096412 | 1 (exact) |
| $75-84$ | .9047304 | .6000757 | 1.399687 | 1 (exact) |
| Crude | 1.719823 | 1.391992 | 2.14353 |  |
| Standardized | 1.155026 | .9006199 | 1.481295 |  |

## - Technical note

estandard and istandard are convenience features; they do nothing different from what you could accomplish by creating the appropriate weights and using the standard() option. For instance, we could duplicate the previously shown results of istandard (example before last) by typing

```
. sort age smokes
. by age: generate wgt=pyears[_N]
. list in 1/4
```

|  | agecat | smokes | deaths | pyears | conswgt | wgt |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: |
| 1. | $35-44$ | 0 | 2 | 18,790 | 1 | 52407 |
| 2. | $35-44$ | 1 | 32 | 52,407 | 1 | 52407 |
| 3. | $45-54$ | 0 | 12 | 10,673 | 1 | 43248 |
| 4. | $45-54$ | 1 | 104 | 43,248 | 1 | 43248 |

```
. ir deaths smokes pyears, by(age) level(90) standard(wgt) ird
    (output omitted)
```

sort age smokes made the exposed group (smokes $=1$ ) the last observation within each age category. by age: gen wgt=pyears [_N] created wgt equal to the last observation in each age category.

## Cumulative incidence data

Cumulative incidence data are "follow-up data with denominators consisting of persons rather than person-time" (Rothman 1986, 172). A group of noncases is monitored for some time, during which some become cases. Each subject is also known to be exposed or unexposed. A summary of the data is

|  | Exposed | Unexposed | Total |
| :--- | :---: | :---: | :---: |
| Cases | $a$ | $b$ | $a+b$ |
| Noncases | $c$ | $d$ | $c+d$ |
| Total | $a+c$ | $b+d$ | $a+b+c+d$ |

Data of this type are generally summarized using the risk ratio, $\{a /(a+c)\} /\{b /(b+d)\}$. A ratio of 2 means that an exposed subject is twice as likely to become a case than is an unexposed subject, a ratio of one-half means half as likely, and so on. The "null" value-the number corresponding to no effect-is a ratio of 1 . If cross-sectional data are analyzed in this format, the risk ratio becomes a prevalence ratio.

## Example 6: csi

We have data on diarrhea during a 10-day follow-up period among 30 breast-fed infants colonized with Vibrio cholerae 01 according to antilipopolysaccharide antibody titers in the mother's breast milk (Glass et al. [1983]; reported in Lash et al. [2021, 403]):

|  | Antibody level |  |
| :--- | ---: | ---: |
|  | High | Low |
| Diarrhea | 7 | 12 |
| No diarrhea | 9 | 2 |

The csi command works much like the iri command. Our sample is small, so we will specify the exact option.

|  | Exposed Unexposed | Total |  |
| :---: | :---: | :---: | :---: |
| Cases <br> Noncases | $\begin{array}{rr}7 & 12 \\ 9 & 2\end{array}$ | $\begin{aligned} & 19 \\ & 11 \end{aligned}$ |  |
| Total <br> Risk | 16 14 <br> .4375 .8571429 <br> Point estimate | $\begin{array}{r} 30 \\ .6333333 \\ {[95 \% \text { cor }} \end{array}$ | interval] |
| Risk difference <br> Risk ratio <br> Prev. frac. ex. <br> Prev. frac. pop | $\begin{array}{r} -.4196429 \\ .5104167 \\ .4895833 \\ .2611111 \end{array}$ | $\begin{array}{r} -.7240828 \\ .2814332 \\ .0742914 \end{array}$ | $\begin{array}{r} -.1152029 \\ .9257086 \\ .7185668 \end{array}$ |
| $\begin{aligned} & \text { 1-sided Fisher's exact } P=0.0212 \\ & \text { 2-sided Fisher's exact } P=0.0259 \end{aligned}$ |  |  |  |

We find that high antibody levels reduce the risk of diarrhea (the risk falls from 0.86 to 0.44 ). The difference is just significant at the $2.59 \%$ two-sided level. (Had we not specified the exact option, a $\chi^{2}$ value and its significance level would have been reported in place of Fisher's exact $p$. The calculated $\chi^{2}$ two-sided significance level would have been 0.0173 , but this calculation is inferior for small samples.)

## - Technical note

By default, cs and csi do not report the odds ratio, but they will if you specify the or option. If you want odds ratios, however, use the cc or cci commands - the commands appropriate for case-control data-because cs and csi calculate the attributable (prevented) fraction with the risk ratio, even if you specify or:
. csi 71292 , or exact


## Technical note

As with iri and ir, csi and cs report the AFE, AFP, PFE, or PFP; see the discussion under Incidencerate data above. In example 6 , we estimated that $49 \%$ of potential cases in the exposed population were prevented by exposure. We also estimated that exposure accounted for a $26 \%$ reduction in cases over the entire population, but that is based on the exposure distribution of the (small) population $(16 / 30)$ and probably is of little interest.

Fleiss, Levin, and Paik (2003, 128) report infant mortality by birthweight for 72,730 live white births in 1974 in New York City:
. csi 618422459767093

|  | Exposed | Unexposed | Total |  |
| :---: | :---: | :---: | :---: | :---: |
| Cases | 618 | 422 | 1040 |  |
| Noncases | 4597 | 67093 | 71690 |  |
| Total | 5215 | 67515 | 72730 |  |
| Risk | . 1185043 | . 0062505 | . 0142995 |  |
|  | Point estimate |  | [95\% con | interval] |
| Risk difference | . 1122539 |  | . 1034617 | . 121046 |
| Risk ratio | 18.95929 |  | 16.80661 | 21.38769 |
| Attr. frac. ex. | . 9472554 |  | . 9404996 | . 9532441 |
| Attr. frac. pop | . 5628883 |  |  |  |

In these data, exposed means a premature baby (birthweight $\leq 2,500 \mathrm{~g}$ ), and a case is a baby who is dead at the end of one year. We find that being premature accounts for $94.7 \%$ of deaths among the premature population. We also estimate, paraphrasing from Fleiss, Levin, and Paik (2003, 128), that $56.3 \%$ of all white infant deaths in New York City in 1974 could have been prevented if prematurity had been eliminated. (Moreover, Fleiss, Levin, and Paik put a standard error on the AFP. The formula is given in Methods and formulas but is appropriate only for the population on which the estimates are based because other populations may have different probabilities of exposure.)
> Example 7: cs
cs works like csi, except that it obtains its information from the data. The data equivalent to typing csi 71292 are

```
. use https://www.stata-press.com/data/r18/csxmpl, clear
. list
```

We could then type cs case exp [fweight=pop]. If we had individual-level data, so that each observation reflected a patient and we had 30 observations, we would type cs case exp.

## Stratified cumulative incidence data

> Example 8: cs with stratified data
Lash et al. $(2021,419)$ reprint the following age-specific information for deaths from all causes for tolbutamide and placebo treatment groups (University Group Diabetes Program 1970):

|  | Age through 54 |  | Age 55 |  |
| :--- | ---: | ---: | ---: | ---: |
|  | Tolbutamide above | Placebo | Tolbutamide | Placebo |
| Dead | 8 | 5 | 22 | 16 |
| Surviving | 98 | 115 | 76 | 69 |

The data corresponding to these results are

```
. use https://www.stata-press.com/data/r18/ugdp
(University Group Diabetes Program 1970)
. list
```

| age | case | exposed | pop |
| ---: | ---: | ---: | ---: |
| $<55$ | Surviving | Placebo | 115 |
| $<55$ | Surviving | Tolbutamide | 98 |
| $<55$ | Dead | Placebo | 5 |
| $<55$ | Dead | Tolbutamide | 8 |
| $55+$ | Surviving | Placebo | 69 |
| $55+$ | Surviving | Tolbutamide | 76 |
| $55+$ | Dead | Placebo | 16 |
| $55+$ | Dead | Tolbutamide | 22 |

The order of the observations is unimportant. If we were now to type cs case exposed [fweight=pop], we would obtain a summary for all the data, ignoring the stratification by age. To incorporate the stratification, we type


Mantel-Haenszel weights are appropriate when the risks may differ according to the strata but the risk ratio is believed to be the same (homogeneous across strata). Under these assumptions, Mantel-Haenszel weights are designed to use the information efficiently. They are not intended to measure a composite risk ratio when the within-stratum risk ratios differ. Then, we want a standardized ratio (see below).

The risk ratios above appear to differ markedly, but the confidence intervals are also broad because of the small sample sizes. The test of homogeneity shows that the differences can be attributed to chance; the use of the Mantel-Haenszel combined test is sensible.

## - Technical note

Stratified cumulative incidence tables are not the only way to control for confounding. Another way is logistic regression. However, logistic regression measures effects with odds ratios, not with risk ratios. So before we fit a logistic model, let's use cs to estimate the Mantel-Haenszel odds ratio:
. cs case exposed [fweight=pop], by(age) or

| Age category | Odds ratio | [95\% conf. interval] | M-H weight |  |
| ---: | ---: | ---: | ---: | ---: |
| $<55$ | 1.877551 | .6238165 | 5.637046 | 2.168142 (Cornfield) |
| $55+$ | 1.248355 | .6112772 | 2.547411 | 6.644809 (Cornfield) |
| Crude | 1.510673 | .8381198 | 2.722012 |  |
| M-H combined | 1.403149 | .7625152 | 2.582015 |  |

Test of homogeneity (M-H) chi2(1) $=0.347 \quad \operatorname{Pr}>\operatorname{chi2}=0.5556$
Test that combined odds ratio $=1$ :
Mantel-Haenszel chi2(1) = 1.19 $\operatorname{Pr}>\operatorname{chi2}=0.2750$

The Mantel-Haenszel odds ratio is 1.40 . It measures the association between death and treatment while adjusting for age. A more general way to adjust for age is logistic regression; the outcome variable is case, and it is explained by age and exposed. (As in the incidence-rate example, age may appear to be a string variable in our data-we listed the data in the previous example-but it is actually a numeric variable taking on values 0 and 1 with value labels disguising that fact; see [U] 12.6.3 Value labels.)

| Logistic regression | Number of obs | $=409$ |
| :---: | :---: | :---: |
|  | LR chi2(2) | $=22.47$ |
|  | Prob > chi2 | $=0.0000$ |
| Log likelihood = -142.6212 | Pseudo R2 | $=0.0730$ |


| case | Odds ratio | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | :---: | :---: |
| exposed | 1.404674 | .4374454 | 1.09 | 0.275 | .7629451 | 2.586175 |
| age | 4.216299 | 1.431519 | 4.24 | 0.000 | 2.167361 | 8.202223 |
| _cons | .0513818 | .0170762 | -8.93 | 0.000 | .0267868 | .0985593 |

Note: _cons estimates baseline odds.
Compare these results with the Mantel-Haenszel estimates obtained with cs:

| Source | Odds ratio | $95 \%$ conf. interval |  |
| :--- | ---: | :---: | :---: |
| Mantel-Haenszel (cs) | 1.40 | 0.76 | 2.58 |
| logistic | 1.40 | 0.76 | 2.59 |

They are virtually identical.
Logistic regression has advantages over the stratified-table approach. First, we obtained an estimate of the age effect: being 55 years or over significantly increases the odds of death. In addition to the point estimate, 4.22, we have a confidence interval for the effect: 2.17 to 8.20 .

A discrete effect at age 55 is not a plausible model of aging. It would be more reasonable to assume that a 54 -year-old patient has a higher probability of death, due merely to age, than does a 53 -year-old patient; a 53-year-old, a higher probability than a 52 -year-old patient; and so on. If we had the underlying data, where each patient's age is presumably known, we could include the actual age in the model and so better control for the age effect. This would improve our estimate of the effect of being exposed to tolbutamide.

See [R] logistic for an explanation of the logistic command. Also see the technical note in Stratified incidence-rate data concerning categorical variables, which applies to logistic regression as well as Poisson regression.

## Standardized estimates with stratified cumulative incidence data

As with ir, cs can produce standardized estimates, and the method is basically the same, although the options for which estimates are to be combined or standardized make it confusing. We showed above that cs can produce Mantel-Haenszel weighted estimates of the risk ratio (the default) or the odds ratio (obtained by specifying or). cs can also produce standardized estimates of the risk ratio (the default) or the risk difference (obtained by specifying rd).
> Example 9: cs with stratified data, using standardized estimates
To produce an estimate of the internally standardized risk ratio by using our age-specific data on deaths from all causes for tolbutamide and placebo treatment groups (example above), we type

| . cs case exposed [fweight=pop], by (age) istandard |  |  |  |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| Age category | Risk ratio | [95\% conf. interval] | Weight |  |
| $<55$ | 1.811321 | .6112044 | 5.367898 | 106 |
| $55+$ | 1.192602 | .6712664 | 2.11883 | 98 |
| Crude | 1.435574 | .8510221 | 2.421645 |  |
| I. Standardized | 1.312122 | .7889772 | 2.182147 |  |

We could obtain externally standardized estimates by substituting estandard for istandard.
To produce an estimate of the risk ratio weighting each age category equally, we could type

```
. generate wgt=1
```

. cs case exposed [fweight=pop], by(age) standard(wgt)

| Age category | Risk ratio | [95\% conf. interval] | Weight |  |
| ---: | ---: | ---: | ---: | ---: |
| $<55$ | 1.811321 | .6112044 | 5.367898 | 1 |
| $55+$ | 1.192602 | .6712664 | 2.11883 | 1 |
| Crude | 1.435574 | .8510221 | 2.421645 |  |
| Standardized | 1.304737 | .7844994 | 2.169967 |  |

If we instead wanted the risk difference, we would type
. cs case exposed [fweight=pop], by(age) standard(wgt) rd

| Age category | Risk diff. | [95\% conf. interval] | Weight |  |
| ---: | ---: | ---: | ---: | ---: |
| $<55$ | .033805 | -.0278954 | .0955055 | 1 |
| $55+$ | .0362545 | -.0809204 | .1534294 | 1 |
| Crude | .0446198 | -.0192936 | .1085332 |  |
| Standardized | .0350298 | -.0311837 | .1012432 |  |

If we wanted to weight the less-than-55 age group five times as heavily as the 55 -and-over group, we would create wgt to contain 5 for the first age group and 1 for the second (or 10 for the first group and 2 for the second-the scale of the weights does not matter).

## Case-control data

In case-control data, you select a sample on the basis of the outcome under study; that is, cases and noncases are sampled at different rates. If you were examining the link between coffee consumption and heart attacks, for instance, you could select a sample of subjects with and without the heart problem and then examine their coffee-drinking behavior. A subject who has suffered a heart attack is called a case just as with cohort study data. A subject who has never suffered a heart attack, however, is called a control rather than merely a noncase, emphasizing that the sampling was performed with respect to the outcome.

In case-control data, all hope of identifying the risk (that is, incidence) of the outcome (heart attacks) associated with the factor (coffee drinking) vanishes, at least without information on the underlying sampling fractions, but you can examine the proportion of coffee drinkers among the two populations and reason that, if there is a difference, coffee drinking may be associated with the risk of heart attacks. Remarkably, even without the underlying sampling fractions, you can also measure the ratio of the odds of heart attacks if a subject drinks coffee to the odds if a subject does not-the so-called odds ratio.

What is lost is the ability to compare absolute rates, which is not always the same as comparing relative rates; see Fleiss, Levin, and Paik (2003, 123).

## > Example 10: cci

cci calculates the odds ratio and the attributable risk associated with a $2 \times 2$ table. Rothman et al. (1979; reprinted in Rothman [1986, 161], and Lash et al. [2021, 411]) present case-control data on the history of chlordiazopoxide use in early pregnancy for mothers of children born with and without congenital heart defects:

| Chlordiazopoxide use |  |  |
| :--- | ---: | ---: |
|  | Yes | No |
| Case mothers | 4 | 386 |
| Control mothers | 4 | 1250 |

. cci 43864 1250, level(90)

|  | Exposed | Unexposed | Total | Proportion exposed |
| :---: | :---: | :---: | :---: | :---: |
| Cases | 4 | 386 | 390 | 0.0103 |
| Controls | 4 | 1250 | 1254 | 0.0032 |
| Total |  | 1636 | 1644 | 0.0049 |
|  | Point estimate |  | [90\% conf | interval] |
| Odds ratio | 3.238342 |  | . 7698467 | 13.59664 |
| Attr. frac. ex. | . 6912 |  | -. 2989599 | . 9264524 |
| Attr. frac. pop | . 0070892 |  |  |  |

We obtain a point estimate of the odds ratio as 3.24 and a $\chi^{2}$ value, which is a test that the odds ratio is 1 , significant at the $10 \%$ level.

## - Technical note

The epitab commands can calculate three different confidence intervals for the odds ratio: the exact, Woolf, and Cornfield intervals. The exact interval, illustrated in example 10, is the default. The interval is "exact" because it uses an exact sampling distribution-a distribution with no unknown parameters under the null hypothesis. An exact interval does not use a normal or $\chi^{2}$ approximation. "Exact" does not describe the coverage probability; the coverage probability of a $90 \%$ exact interval is not exactly $90 \%$. The coverage probability is actually bounded below by $90 \%$ (Agresti 2013, 606), so a $90 \%$ exact interval will always cover the odds ratio with probability at least $90 \%$ (if the model is correct).

The Woolf and Cornfield intervals, on the other hand, are approximate. They approximate the exact sampling distribution with a normal model and are not guaranteed to maintain their nominal coverage: the coverage probability of a $90 \%$ approximate interval fluctuates above and below $90 \%$. The coverage approaches $90 \%$ only in the limit as the sample size increases. Exact intervals are conservative; approximate intervals can be conservative or anticonservative (Agresti 2013, 607).

If you wish to maintain nominal coverage, then you should use the exact interval. But you will pay a price for the coverage: the exact interval will usually be wider than the approximate intervals. Example 10 is no exception:

| Method | $90 \%$ | conf. interval | Command |
| :--- | :---: | :--- | :--- |
| exact | 0.77 | 13.60 | cci |
| Woolf | 1.01 | 10.40 | cci, woolf |
| Cornfield | 1.07 | 9.83 | cci, cornfield |

The exact interval is the widest of the three-so wide that it includes the null value of one-even though the $\chi^{2}$ test $p$-value of 0.0799 was significant at the $10 \%$ level. The exact interval and $\chi^{2}$ test come from different models, so we should not expect them to always agree on sharp conclusions such as statistical significance.

The odds-ratio intervals are all frequentist methods, so we cannot compare them rigorously with one example. See Brown (1981), Gart and Thomas (1982), and Agresti (1999) for more rigorous comparisons. Agresti (1999) found that the Woolf interval performed well, even for small samples.

Jerome Cornfield (1912-1979) was born in New York City. He majored in history at New York University and took courses in statistics at the US Department of Agriculture Graduate School but otherwise had little formal training. Cornfield held positions at the Bureau of Labor Statistics, the National Cancer Institute, the National Institutes of Health, Johns Hopkins University, the University of Pittsburgh, and George Washington University. He worked on many problems in biomedical statistics, including the analysis of clinical trials, epidemiology (especially case-control studies), and Bayesian approaches.

Barnet Woolf (1902-1983) was born in London. His parents were immigrants from Lithuania. Woolf was educated at Cambridge, where he studied physiology and biochemistry, and proposed methods for linearizing plots in enzyme chemistry that were later rediscovered by others (see Haldane [1957]). His later career in London, Birmingham, Rothamsted, and Edinburgh included lasting contributions to nutrition, epidemiology, public health, genetics, and statistics. He was also active in left-wing causes and penned humorous poems, songs, and revues.

## - Technical note

By default, cc and cci report exact confidence intervals but an approximate significance test. You can replace the approximate test with Fisher's exact test by specifying the exact option. We recommend specifying exact whenever any cell count is less than 1,000 .

```
. cci 4 3864 1250, exact level(90)
```

|  | Exposed | Unexposed | TotalProportion <br> exposed |
| ---: | ---: | ---: | ---: |
| Cases <br> Controls | 4 | 386 |  |
| Total | 4 | 1250 | 390 |

In this table, the one- and two-sided significance values are equal. This is not a mistake, but it does not happen often. Exact significance values are calculated by summing the probabilities for tables that have the same marginals (row and column sums) but that are less likely (given an odds ratio of 1) than the observed table. When considering each possible table, we might ask if the table is in the same or opposite tail as the observed table. If it is in the same tail, we would count the table under consideration in the one-sided test and, either way, we would count it in the two-sided test. Here all the tables more extreme than this table are in the same tail, so the one- and two-sided tests are the same.

The $p$-value of 0.0964 is significant at the $10 \%$ level, but the exact confidence interval is not (it includes the null odds ratio of one). It was not surprising that the exact interval disagreed with the $\chi^{2}$ test; after all, they come from different models. Now, the exact interval and Fisher's exact test also disagree, even though they come from the same model!

The test and interval disagree because the exact sampling distribution is asymmetric, and the test and interval handle the asymmetry differently. The two-sided test, as we have seen, sums the probabilities of all tables at least as unlikely as the observed table, and in example 10, all the unlikely tables fall in the same tail of the distribution. The other tail does not contribute to the $p$-value. The exact interval, on the other hand, must always use both tails of the distribution, because the interval inverts two one-sided tests, not one two-sided test (Breslow and Day 1980, 128-129).

## Technical note

The reported value of the AFE or PFE is calculated using the odds ratio as a proxy for the risk ratio. This can be justified only if the outcome is rare in the population. The extrapolation to the AFP or PFP assumes that the control group is a random sample of the corresponding group in the underlying population.
> Example 11: cc equivalent to cci
Equivalent to typing cci 438641250 would be typing cc case exposed [fweight=pop] with the following data:

```
. use https://www.stata-press.com/data/r18/ccxmpl, clear
. list
```

|  | case | exposed | pop |
| :--- | ---: | ---: | ---: |
| 1. | 1 | 1 | 4 |
| 2. | 1 | 0 | 386 |
| 3. | 0 | 1 | 4 |
| 4. | 0 | 0 | 1250 |

## Stratified case-control data

## Example 12: cc with stratified data

cc can work with stratified tables. Lash et al. $(2021,429)$ reprint and discuss data from a case-control study on infants with congenital heart disease and Down syndrome and healthy controls, according to maternal spermicide use before conception and maternal age at delivery (Rothman 1982):

|  | Maternal age to 34 |  | Maternal age 35+ |  |
| :--- | ---: | ---: | ---: | ---: | ---: |
|  | Spermicide used | not used | Spermicide used | not used |
| Down syndrome | 3 | 9 | 1 | 3 |
| Controls | 104 | 1059 | 5 | 86 |

The data corresponding to these tables are

```
. use https://www.stata-press.com/data/r18/downs
(Congenital heart disease and Down syndrome)
. list
```

1. 

| case | exposed | pop | age |
| ---: | ---: | ---: | ---: |
| 1 | 1 | 3 | $<35$ |
| 1 | 0 | 9 | $<35$ |
| 0 | 1 | 104 | $<35$ |
| 0 | 0 | 1059 | $<35$ |
| 1 | 1 | 1 | $35+$ |
| 1 | 0 | 3 | $35+$ |
| 0 | 1 | 5 | $35+$ |
| 0 | 0 | 86 | $35+$ |

The stratified results for the odds ratio are


For no particular reason, we also specified the woolf option to obtain Woolf approximations to the within-stratum confidence intervals rather than the default. Had we wanted Tarone's test of homogeneity, we would have used

| Maternal age | Odds ratio | [95\% conf | interval] | M-H weight |
| :---: | :---: | :---: | :---: | :---: |
| <35 | 3.394231 | . 5812415 | 13.87412 | . 7965957 |
| 35+ | 5.733333 | . 0911619 | 85.89602 | . 1578947 |
| Crude | 3.501529 | . 8080857 | 11.78958 |  |
| M-H combined | 3.781172 | 1.18734 | 12.04142 |  |
| Test of homogeneity (M-H) <br> Test of homogeneity (Tarone) |  | $2(1)=$ | 0.14 Pr>c | i2 $=0.7105$ |
|  |  | $2(1)=$ | 0.14 Pr>c | i2 $=0.7092$ |
| Test that combined odds ratio $=1$ : |  |  |  |  |
| Mantel-Haenszel chi2(1) = 5.81 |  |  |  |  |
|  |  |  | Pr>chi2 | 0.0159 |

Whatever method you choose for calculating confidence intervals, Stata will report a test of homogeneity, which here is $\chi^{2}(1)=0.14$ and not significant. That is, the odds of Down syndrome might vary with maternal age, but we cannot reject the hypothesis that the association between Down syndrome and spermicide is the same in the two maternal age strata. This is thus a test to reject the appropriateness of the single, Mantel-Haenszel combined odds ratio-a rejection not justified by these data.

## Technical note

The cc command includes four tests of homogeneity: Mantel-Haenszel (the default); directly pooled, also known as the Woolf test (available with the pool option); Tarone (available with the tarone option); and Breslow-Day (available with the bd option). The preferred test is Tarone's (Tarone 1985, 94), which corrected an error in the Breslow-Day test; see Breslow (1996, 17-18) for details of the error and Tarone's correction.

The other two homogeneity tests, the Mantel-Haenszel and directly pooled, are less useful: they use the logs of the stratum-specific odds ratios, so they are undefined when any stratum has a zero cell. The epitab commands deal with the problem differently: cs omits the offending strata, while cc substitutes the Tarone test. The Tarone test does not use the stratum-specific odds ratios, so it can still be calculated when there are zero cells.

None of the tests is appropriate for finely stratified (many strata with only a few observations each) studies (Lash et al. 2021, 429). If you have fine stratification, one alternative is multilevel logistic regression; see [ME] melogit.

## - Technical note

As with cohort study data, an alternative to stratified tables for uncovering effects is logistic regression. From the logistic point of view, case-control data are no different from cohort study data-you must merely ignore the estimated intercept. The intercept is meaningless in case-control data because it reflects the baseline prevalence of the outcome, which you controlled by sampling.

The data we used with cc can be used directly by logistic. (The age variable, which appears to be a string, is really numeric with an associated value label; see [U] 12.6.3 Value labels. age takes on the value 0 for the age-less-than- 35 group and 1 for the $35+$ group.)

| . logistic case exposed age [fweight=pop] |  |
| :--- | :--- |
| Logistic regression | Number of obs $=1,270$ |
|  | LR chi2 2 ( $)$ |
|  | $=8.74$ |
| Log likelihood $=-81.517532$ | Prob $>$ chi2 $=0.0127$ |
|  | Pseudo R2 |


| case | Odds ratio | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | :--- | :---: | :---: | ---: | ---: |
| exposed | 3.787779 | 2.241922 | 2.25 | 0.024 | 1.187334 | 12.0836 |
| age | 4.582857 | 2.717352 | 2.57 | 0.010 | 1.433594 | 14.65029 |
| _cons | .0082631 | .0027325 | -14.50 | 0.000 | .0043218 | .0157988 |

Note: _cons estimates baseline odds.
We compare the results with those presented by cc in the previous example:

| Source | Odds ratio | $95 \%$ CI |  |
| :--- | ---: | :---: | :---: |
| Mantel-Haenszel (cc) | 3.78 | 1.19 | 12.04 |
| logistic | 3.79 | 1.19 | 12.08 |

As with the cohort study data in example 8 , results are virtually identical, and all the same comments we made previously apply once again.

To demonstrate an advantage of logistic regression, let's now ask a question that would be difficult to answer on the basis of a stratified table analysis. We now know that spermicide use appears to increase the risk of having a baby with Down syndrome, and we know that the mother's age also increases the risk. Is the effect of spermicide use statistically different for mothers in the two age groups?
. logistic case exposed age c.age\#exposed [fweight=pop]
Logistic regression
Log likelihood = -81.451332

Note: _cons estimates baseline odds.
The answer is no. The odds ratio and confidence interval reported for exposed now measure the spermicide effect for an age $==0$ (age $<35$ ) mother. The odds ratio and confidence interval reported for c.age\#exposed are the (multiplicative) difference in the spermicide odds ratio for an age==1 (age 35+) mother relative to a young mother. The point estimate is that the effect is larger for older mothers, suggesting grounds for future research, but the difference is not significant.

See $[R]$ logistic for an explanation of the logistic command. Also see the technical note under Incidence-rate data above concerning Poisson regression, which applies equally to logistic regression.

## Case-control data with multiple levels of exposure

In a case-control study, subjects with the disease of interest (cases) are compared with disease-free individuals (controls) to assess the relationship between exposure to one or more risk factors and disease incidence. Often exposure is measured qualitatively at several discrete levels or measured on a continuous scale and then grouped into three or more levels. The data can be summarized as

|  | Exposure level |  |  |  |  |
| :--- | :---: | :---: | :---: | ---: | :---: |
|  | 1 | 2 | $\ldots$ | $k$ | Total |
| Cases | $a_{1}$ | $a_{2}$ | $\ldots$ | $a_{k}$ | $M_{1}$ |
| Controls | $c_{1}$ | $c_{2}$ | $\ldots$ | $c_{k}$ | $M_{0}$ |
| Total | $N_{1}$ | $N_{2}$ | $\ldots$ | $N_{k}$ | $T$ |

An advantage afforded by having multiple levels of exposure is the ability to examine dose-response relationships. If the association between a risk factor and a disease or outcome is real, we expect the strength of that association to increase with the level and duration of exposure. A dose-response relationship provides strong support for a direct or even causal relationship between the risk factor and the outcome. On the other hand, the lack of a dose-response is usually seen as an argument against causality.

We can use the tabodds command to tabulate the odds of failure or odds ratios against a categorical exposure variable. The test for trend calculated by tabodds can serve as a test for dose-response if the exposure variable is at least ordinal. If the exposure variable has no natural ordering, the trend test is meaningless and should be ignored. See the technical note at the end of this section for more information regarding the test for trend.

Before looking at an example, consider three possible data arrangements for case-control and prevalence studies. The most common data arrangement is individual records, where each subject in the study has his or her own record. Closely related are frequency records where identical individual records are included only once, but with a variable giving the frequency with which the record occurs. The fweight weight option is used for these data to specify the frequency variable. Data can also be arranged as binomial frequency records where each record contains a variable, $D$, the number of cases; another variable, N, the number of total subjects (cases plus controls); and other variables. An advantage of binomial frequency records is that large datasets can be entered succinctly into a Stata database.

## Example 13: tabodds

Consider the following data from the Ille-et-Vilaine study of esophageal cancer, discussed in Breslow and Day (1980, chap. 4 and app. I), corresponding to subjects age 55-64 who use from 0 to 9 g of tobacco per day:

|  | Alcohol consumption (g/day) |  |  |  |  |
| :--- | ---: | ---: | ---: | ---: | ---: |
|  | $0-39$ | $40-79$ | $80-119$ | $120+$ | Total |
| Cases | 2 | 9 | 9 | 5 | 25 |
| Controls | 47 | 31 | 9 | 5 | 92 |
| Total | 49 | 40 | 18 | 10 | 117 |

The study included 24 such tables, each representing one of four levels of tobacco use and one of six age categories. We can create a binomial frequency-record dataset by typing

| . input alcohol D N agegrp tobacco |  |  |  |  |  |
| :--- | ---: | ---: | ---: | ---: | ---: |
| alcohol |  |  |  |  |  |
| a | D | N | agegrp | tobacco |  |
| 1. | 1 | 2 | 49 | 4 | 1 |
| 2. | 2 | 9 | 40 | 4 | 1 |
| 3. | 3 | 9 | 18 | 4 | 1 |
| 4. | 4 | 5 | 10 | 4 | 1 |
| 5. end |  |  |  |  |  |

where $D$ is the number of esophageal cancer cases and $N$ is the number of total subjects (cases plus controls) for each combination of six age groups (agegrp), four levels of alcohol consumption in $\mathrm{g} /$ day (alcohol), and four levels of tobacco use in $\mathrm{g} /$ day (tobacco).

Both the tabodds and mhodds commands can correctly handle all three data arrangements. Binomial frequency records require that the number of total subjects (cases plus controls) represented by each record $N$ be specified with the binomial () option.

We could also enter the data as frequency-weighted data:

| . input alcohol case freq agegrp |  |  |  |  |  |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: |
|  | alcohol | case | freq | agegrp | tobacco |
| 1. | 1 | 1 | 2 | 4 | 1 |
| 2. | 1 | 0 | 47 | 4 | 1 |
| 3. | 2 | 1 | 9 | 4 | 1 |
| 4. | 2 | 0 | 31 | 4 | 1 |
| 5. | 3 | 1 | 9 | 4 | 1 |
| 6. | 3 | 0 | 9 | 4 | 1 |
| 7. | 4 | 1 | 5 | 4 | 1 |
| 8. | 4 | 0 | 5 | 4 | 1 |
| 9. end |  |  |  |  |  |

If you are planning on using any of the other estimation commands, such as poisson or logistic, we recommend that you enter your data either as individual records or as frequency-weighted records and not as binomial frequency records, because the estimation commands currently do not recognize the binomial() option.

We have entered all the esophageal cancer data into Stata as a frequency-weighted record dataset as previously described. In our data, case indicates the esophageal cancer cases and controls, and freq is the number of subjects represented by each record (the weight).

We added value labels to the agegrp, alcohol, and tobacco variables in our dataset to ease interpretation in outputs, but these variables are numeric.

We are interested in the association between alcohol consumption and esophageal cancer. We first use tabodds to tabulate the odds of esophageal cancer against alcohol consumption:

```
. use https://www.stata-press.com/data/r18/bdesop, clear
(Ille-et-Vilaine study of esophageal cancer)
. tabodds case alcohol [fweight=freq]
```

| alcohol | Cases | Controls | Odds | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| $0-39$ | 29 | 386 | 0.07513 | 0.05151 | 0.10957 |
| $40-79$ | 75 | 280 | 0.26786 | 0.20760 | 0.34560 |
| $80-119$ | 51 | 87 | 0.58621 | 0.41489 | 0.82826 |
| $120+$ | 45 | 22 | 2.04545 | 1.22843 | 3.40587 |



The test of homogeneity clearly indicates that the odds of esophageal cancer differ by level of alcohol consumption, and the test for trend indicates a significant increase in odds with increasing alcohol use. This suggests a strong dose-response relation. The graph option can be used to study the shape of the relationship of the odds with alcohol consumption. Most of the heterogeneity in these data can be "explained" by the linear increase in risk of esophageal cancer with increased dosage (alcohol consumption).

We also could have requested that the odds ratios at each level of alcohol consumption be calculated by specifying the or option. For example, tabodds case alcohol [fweight=freq], or would produce odds ratios using the minimum value of alcohol-that is, alcohol $=1$ ( $0-39$ ) -as the reference group, and the command tabodds case alcohol [fweight=freq], or base(2) would use alcohol $=2(40-79)$ as the reference group.

Although our results appear to provide strong evidence supporting an association between alcohol consumption and esophageal cancer, we need to be concerned with the possible existence of confounders, specifically age and tobacco use, in our data. We can again use tabodds to tabulate the odds of esophageal cancer against age and against tobacco use, independently:
. tabodds case agegrp [fweight=freq]

| agegrp | Cases | Controls | Odds | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| $25-34$ | 1 | 115 | 0.00870 | 0.00121 | 0.06226 |
| $35-44$ | 9 | 190 | 0.04737 | 0.02427 | 0.09244 |
| $45-54$ | 46 | 167 | 0.27545 | 0.19875 | 0.38175 |
| $55-64$ | 76 | 166 | 0.45783 | 0.34899 | 0.60061 |
| $65-74$ | 55 | 106 | 0.51887 | 0.37463 | 0.71864 |
| $75+$ | 13 | 31 | 0.41935 | 0.21944 | 0.80138 |


. tabodds case tobacco [fweight=freq]

| tobacco | Cases | Controls | Odds | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| $0-9$ | 78 | 447 | 0.17450 | 0.13719 | 0.22194 |
| $10-19$ | 58 | 178 | 0.32584 | 0.24228 | 0.43823 |
| $20-29$ | 33 | 99 | 0.33333 | 0.22479 | 0.49428 |
| $30+$ | 31 | 51 | 0.60784 | 0.38899 | 0.94983 |

Test of homogeneity (equal odds): chi2(3) = 29.33
$\operatorname{Pr}>\operatorname{chi} 2=0.0000$
Score test for trend of odds: chi2(1) = 26.93
$\operatorname{Pr}>\operatorname{chi} 2=0.0000$
We can see that there is evidence to support our concern that both age and tobacco use are potentially important confounders. Clearly, before we can make any statements regarding the association between esophageal cancer and alcohol use, we must examine and, if necessary, adjust for the effect of any confounder. We will return to this example in the following section.

## - Technical note

The score test for trend performs a test for linear trend of the log odds against the numerical code used for the exposure variable. The test depends not only on the relationship between dose level and the outcome but also on the numeric values assigned to each level or, to be more accurate, to the distance between the numeric values assigned. For example, the trend test on a dataset with four exposure levels coded $1,2,3$, and 4 gives the same results as coding the levels $10,20,30$, and 40 because the distance between the levels in each case is constant. In the first case, the distance is 1 unit, and in the second case, it is 10 units. However, if we code the exposure levels as 1,10 , 100 , and 1,000 , we would obtain different results because the distance between exposure levels is not constant. Thus, be careful when assigning values to exposure levels. You must determine whether equally spaced numbers make sense for your data or if other more meaningful values should be used.

Remember that we are testing whether a log-linear relationship exists between the odds and the exposure variable. For your particular problem, this relationship may not be correct or even make sense, so you must be careful in interpreting the output of this trend test.

## Case-control data with confounders and possibly multiple levels of exposure

In the esophageal cancer data example introduced earlier, we determined that the apparent association between alcohol consumption and esophageal cancer could be confounded by age and tobacco use. You can adjust for the effect of possible confounding factors by stratifying on these factors. This is the method used by both tabodds and mhodds to adjust for other variables in the dataset. We will compare and contrast these two commands in the following example.
> Example 14: tabodds, adjusting for confounding factors
We begin by using tabodds to tabulate unadjusted odds ratios.
. tabodds case alcohol [fweight=freq], or

| alcohol | Odds ratio |
| ---: | :--- |
| $0-39$ | 1.000000 |
| $40-79$ | 3.565271 |

The alcohol $=1$ group ( $0-39$ ) was used by tabodds as the reference category for calculating the odds ratios. We could have selected a different group by specifying the base() option; however, because the lowest dosage level is most often the appropriate reference group, as it is in these data, the base() option is seldom used.

We use tabodds with the adjust () option to tabulate Mantel-Haenszel age-adjusted odds ratios:
. tabodds case alcohol [fweight=freq], adjust(age)
Mantel-Haenszel odds ratios adjusted for agegrp

| alcohol | Odds ratio | chi2 | P>chi2 | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| $0-39$ | 1.000000 |  |  |  |  |
| $40-79$ | 4.268155 | 37.36 | 0.0000 | 2.570025 | 7.088314 |
| $80-119$ | 8.018305 | 59.30 | 0.0000 | 4.266893 | 15.067922 |
| $120+$ | 28.570426 | 139.70 | 0.0000 | 12.146409 | 67.202514 |

We observe that the age-adjusted odds ratios are just slightly higher than the unadjusted ones, so it appears that age is not as strong a confounder as it first appeared. Even after adjusting for age, the dose-response relationship, as measured by the trend test, remains strong.

We now perform the same analysis but this time adjust for tobacco use instead of age.
. tabodds case alcohol [fweight=freq], adjust(tobacco)
Mantel-Haenszel odds ratios adjusted for tobacco

| alcohol | Odds ratio | chi2 | P>chi2 | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| $0-39$ | 1.000000 | . |  |  |  |
| $40-79$ | 3.261178 | 28.53 | 0.0000 | 2.059764 | 5.163349 |
| $80-119$ | 6.771638 | 62.54 | 0.0000 | 3.908113 | 11.733306 |
| $120+$ | 19.919526 | 123.93 | 0.0000 | 9.443830 | 42.015528 |

Again we observe a significant dose-response relationship and not much difference between the adjusted and unadjusted odds ratios. We could also adjust for the joint effect of both age and tobacco use by specifying adjust (tobacco age), but we will not bother here.

A different approach to analyzing these data is to use the mhodds command. This command estimates the ratio of the odds of failure for two categories of an exposure variable, controlling for any specified confounding variables, and it tests whether this odds ratio is equal to one. For multiple exposures, if two exposure levels are not specified with compare (), then mhodds assumes that exposure is quantitative and calculates a 1-degree-of-freedom test for trend. This test for trend is the same one that tabodds reports.

## > Example 15: mhodds, controlling for confounding factors

We first use mhodds to estimate the effect of alcohol controlled for age:

```
. mhodds case alcohol agegrp [fweight=freq]
Score test for trend of odds with alcohol
    controlling for agegrp
```

| Odds ratio | chi2(1) | P>chi2 | $[95 \%$ conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: |
| 2.845895 | 135.09 | 0.0000 | 2.385749 | 3.394792 |

Note: The Odds ratio estimate is an approximation to the odds ratio for a one-unit increase in alcohol.

Because alcohol has more than two levels, mhodds estimates and reports an approximate ageadjusted odds ratio for a one-unit increase in alcohol consumption. The $\chi^{2}$ value reported is identical to that reported by tabodds for the score test for trend on the previous page.

We now use mhodds to estimate the effect of alcohol controlled for age, and while we are at it, we do this by levels of tobacco consumption:

```
. mhodds case alcohol agegrp [fweight=freq], by(tobacco)
Score test for trend of odds with alcohol
    controlling for agegrp
    by tobacco
```

| tobacco | Odds ratio | chi2(1) | P>chi2 | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| $0-9$ | 3.579667 | 75.95 | 0.0000 | 2.68710 | 4.76871 |
| $10-19$ | 2.303580 | 25.77 | 0.0000 | 1.66913 | 3.17920 |
| $20-29$ | 2.364135 | 13.27 | 0.0003 | 1.48810 | 3.75589 |
| $30+$ | 2.217946 | 8.84 | 0.0029 | 1.31184 | 3.74992 |

Notes: Only 19 of the 24 strata formed in this analysis contribute information about the effect of the explanatory variable. The Odds ratio estimate is an approximation to the odds ratio for a one-unit increase in alcohol.
Mantel-Haenszel estimate controlling for agegrp and tobacco

| Odds ratio | chi2(1) | $\mathrm{P}>\mathrm{chi2}$ | [95\% conf. | interval] |
| :---: | :---: | :---: | :---: | :---: |
| 2.751236 | 118.37 | 0.0000 | 2.292705 | 3.301471 |
| Approximate test of homogeneity of odds ratios: |  |  | chi2(3) | 5.46 |
|  |  |  | Pr>chi2 | 0.1409 |

The first table reports estimates of the effect of alcohol for each level of tobacco use, controlling for age.

From the second table, we find that the effect of alcohol is about $\times 2.8$ when we control for both age and tobacco use. Again, because alcohol has more than two levels, mhodds estimates and reports an approximate Mantel-Haenszel age and tobacco-use adjusted odds ratio for a one-unit increase in alcohol consumption.

The $\chi^{2}$ test for trend reported with the Mantel-Haenszel estimate is again the same one that tabodds produces if adjust (agegrp tobacco) is specified.

To instead estimate the effect of tobacco use for each level of alcohol consumption, controlling for age, we type

```
. mhodds case tobacco agegrp [fweight=freq], by(alcohol)
Score test for trend of odds with tobacco
    controlling for agegrp
    by alcohol
```

| alcohol | Odds ratio | chi2(1) | P>chi2 | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| $0-39$ | 2.420650 | 15.61 | 0.0001 | 1.56121 | 3.75320 |
| $40-79$ | 1.427713 | 5.75 | 0.0165 | 1.06717 | 1.91007 |
| $80-119$ | 1.472218 | 3.38 | 0.0659 | 0.97483 | 2.22339 |
| $120+$ | 1.214815 | 0.59 | 0.4432 | 0.73876 | 1.99763 |

Notes: Only 18 of the 24 strata formed in this analysis contribute information about the effect of the explanatory variable.
The Odds ratio estimate is an approximation to the odds ratio for a
one-unit increase in tobacco.
Mantel-Haenszel estimate controlling for agegrp and alcohol

| Odds ratio | chi2(1) | $\mathrm{P}>$ chi2 | [95\% conf | interval] |
| :---: | :---: | :---: | :---: | :---: |
| 1.553437 | 20.07 | 0.0000 | 1.281160 | 1.883580 |
| Approximate test of homogeneity of odds ratios: |  |  | $\begin{aligned} & \text { chi2 (3) } \\ & \text { Pr>chi2 } \end{aligned}$ | $\begin{array}{r} 5.26 \\ 0.1540 \end{array}$ |

From the second table, we find that the effect of tobacco, controlled for both age and alcohol consumption, is about $\times 1.6$.

Comparisons between particular levels of alcohol and tobacco consumption can be made by generating a new variable with levels corresponding to all combinations of alcohol and tobacco, as in

```
. egen alctob = group(alcohol tobacco)
. mhodds case alctob [fweight=freq], compare(16,1)
Maximum likelihood estimate of the odds ratio comparing alctob==16
    vs. alctob==1
\begin{tabular}{rrrr}
\hline Odds ratio & chi2(1) & P>chi2 & [95\% conf. interval] \\
\hline 93.333333 & 103.21 & 0.0000 & 14.766136589 .938431 \\
\hline
\end{tabular}
```

which yields an odds ratio of 93 between subjects with the highest levels of alcohol and tobacco and those with the lowest levels. Similar results can be obtained simultaneously for all levels of alctob using alctob $=1$ as the comparison group by specifying tabodds D alctob, binomial(N) or.

## Standardized estimates with stratified case-control data

## Example 16: cc with stratified data, using standardized estimates

You obtain standardized estimates (here for the odds ratio) by using cc just as you obtain standardized estimates by using ir or cs. Along with the by () option, you specify one of estandard, istandard, or standard(varname).

Case-control studies can provide standardized rate-ratio estimates when density sampling is used, or when the disease is rare (Lash et al. 2021, 422). Lash et al. (2021, 429) report the SMR for the case-control study on infants with congenital heart disease and Down syndrome. We can reproduce their estimates along with the pooled estimates by typing

| Maternal age | Odds ratio | [95\% conf | interval] | Weight |
| :---: | :---: | :---: | :---: | :---: |
| <35 | 3.394231 | . 5812415 | 13.87412 | 104 |
| 35+ | 5.733333 | . 0911619 | 85.89602 | 5 |
| Crude | 3.501529 | . 8080857 | 11.78958 |  |
| Pooled (direct) | 3.824166 | 1.196437 | 12.22316 |  |
| I. Standardized | 3.779749 | 1.180566 | 12.10141 |  |
| Test of homogeneity (direct) |  | chi2(1) = | 0.14 Pr>chi2 $=0.7109$ |  |

Using the distribution of the nonexposed subjects in the source population as the standard, we can obtain an estimate of the standardized rate ratio (SRR):
. cc case exposed [fweight=pop], by(age) estandard

| Maternal age | Odds ratio | [95\% conf. interval] | Weight |  |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| $<35$ | 3.394231 | .5812415 | 13.87412 | 1059 (exact) |  |
| $35+$ | 5.733333 | .0911619 | 85.89602 | 86 (exact) |  |
|  |  |  |  |  |  |
| Crude | 3.501529 | .8080857 | 11.78958 |  | (exact) |
| E. Standardized | 3.979006 | 1.176096 | 13.46191 |  |  |

Finally, if we wanted to weight the two age groups equally, we could type

```
. generate wgt=1
```

. cc case exposed [fweight=pop], by(age) standard(wgt)

| Maternal age | Odds ratio | [95\% conf. interval] | Weight |  |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| $<35$ | 3.394231 | .5812415 | 13.87412 | 1 (exact) |  |
| $35+$ | 5.733333 | .0911619 | 85.89602 | 1 (exact) |  |
| Crude | 3.501529 | .8080857 | 11.78958 |  |  |
| Standardized | 5.275104 | .6233794 | 44.6385 |  | (exact) |

## Matched case-control data

Matched case-control studies are performed to gain sample-size efficiency and to control for important confounding factors. In a matched case-control design, each case is matched with a control on the basis of demographic characteristics, clinical characteristics, etc. Thus, their difference with respect to the outcome must be due to something other than the matching variables. If the only difference between them was exposure to the factor, we could attribute any difference in outcome to the factor.

A summary of the data is

Controls

| Cases | Exposed | Unexposed | Total |
| :--- | :---: | :---: | :---: |
| Exposed | $a$ | $b$ | $M_{1}$ |
| Unexposed | $c$ | $d$ | $M_{0}$ |
| Total | $N_{1}$ | $N_{0}$ | $T=a+b+c+d$ |

Each entry in the table represents the number of case-control pairs. For instance, in $a$ of the pairs, both members were exposed; in $b$ of the pairs, the case was exposed but the control was not; and so on. In total, $T$ pairs were observed.
> Example 17: mcci
Rothman $(1986,257)$ discusses data from Jick et al. (1973) on a matched case-control study of myocardial infarction and drinking six or more cups of coffee per day (persons drinking from one to five cups per day were excluded):

Controls

| Cases | $6+$ cups | 0 cups |
| :--- | :---: | :---: |
| $6+$ cups | 8 | 8 |
| 0 cups | 3 | 8 |

mcci analyzes matched case-control data:
. mcci 8838

| Cases | Controls |  |  |
| ---: | ---: | ---: | ---: |
|  | Exposed | Unexposed | Total |
| Exposed | 8 | 8 | 16 |
| Unexposed | 3 | 8 | 11 |
| Total | 11 | 16 | 27 |

McNemar's chi2(1) = $2.27 \quad$ Prob $>$ chi2 $=0.1317$
Exact McNemar significance probability $\quad=0.2266$
Proportion with factor
Cases . 5925926
Controls . 4074074 [95\% conf. interval]
difference $.1851852-.0822542 \quad .4526246$
ratio 1.454545 . 8911012.374257
rel. diff. . $3125-.0243688$. 6493688
odds ratio 2.666667 . 640036415.6064 (exact)

The point estimate states that the odds of drinking 6 or more cups of coffee per day is 2.67 times greater among the myocardial infarction patients. The confidence interval is wide, however, and the $p$-value of 0.1317 from McNemar's test is not statistically significant.
mcc works like the other nonimmediate commands but does not handle stratified data. If you have stratified matched case-control data, you can use conditional logistic regression to estimate odds ratios; see [R] clogit.

Matched case-control studies can also be analyzed using mhodds by controlling on the variable used to identify the matched sets. For example, if the variable set is used to identify the matched set for each subject,

```
. mhodds fail xvar set
```

will do the job. Any attempt to control for further variables will restrict the analysis to the comparison of cases and matched controls that share the same values of these variables. In general, this would lead to the omission of many records from the analysis. Similar considerations usually apply when investigating effect modification by using the by () option. An important exception to this rule is that a variable used in matching cases to controls may appear in the by () option without loss of data.

## > Example 18: mhodds with matched case-control data

Let's use mhodds to analyze matched case-control studies using the study of endometrial cancer and exposure to estrogen described in Breslow and Day (1980, chap. 5). In this study, there are four controls matched to each case. Cases and controls are matched on age, marital status, and time living in the community. The data collected include information on the daily dose of conjugated estrogen therapy. Breslow and Day created four levels of the dose variable and began by analyzing the $1: 1$ study formed by using the first control in each set. We examine the effect of exposure to estrogen:


```
. mhodds fail est set
Mantel-Haenszel estimate of the odds ratio comparing est==1 vs.
    est==0
    controlling for set
```

| Odds ratio | chi2(1) | P>chi2 | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: |
| 9.666667 | 21.12 | 0.0000 |  | 2.944702 | $\mathbf{3 1 . 7 3 3 0 7 2}$

Note: Only 32 of the 63 strata formed in this analysis contribute information about the effect of the explanatory variable.

For the 1:1 matched study, the Mantel-Haenszel methods are equivalent to conditional likelihood methods. The maximum conditional likelihood estimate of the odds ratio is given by the ratio of the off-diagonal frequencies in the two-way (case-control) table below. The data must be in the 1-observation-per-group format; that is, the matched case and control must appear in 1 observation (the same format as required by the mcc command; see also [R] clogit).

```
. keep fail est set
. reshape wide est, i(set) j(fail)
(j = 0 1)
\begin{tabular}{lrll} 
Data & Long & \(\rightarrow>\) & Wide \\
\hline Number of observations & 126 & \(->\) & 63 \\
Number of variables & 3 & \(\rightarrow\) & 3 \\
j variable (2 values) & fail & \(\rightarrow\) & (dropped) \\
xij variables: & est & \(\rightarrow\) & est0 est1 \\
\hline
\end{tabular}
```

```
. rename est1 case
. rename estO control
. label variable case case
. label variable control control
```

. tabulate case control

| case | con 0 | 1 | Total |
| :---: | :---: | :---: | :---: |
| 0 | 4 | 3 | 7 |
| 1 | 29 | 27 | 56 |
| Total | 33 | 30 | 63 |

The odds ratio is $29 / 3=9.67$, which agrees with the value obtained from mhodds. In the more general 1: $m$ matched study, however, the Mantel-Haenszel methods are no longer equivalent to maximum conditional likelihood, although they are usually close.

To illustrate the use of the by () option in matched case-control studies, we look at the effect of exposure to estrogen, stratified by age3, which codes the sets into three age groups (55-64, 65-74, and 75+) as follows:

```
. use https://www.stata-press.com/data/r18/bdendo11, clear
(Endometrial cancer and estrogen exposure)
. generate age3 = agegrp
. recode age3 1/2=1 3/4=2 5/6=3
(124 changes made to age3)
. mhodds fail est set, by(age3)
Mantel-Haenszel estimate of the odds ratio comparing est==1 vs. est==0
    controlling for set
    by age3
```

| age3 | Odds ratio | chi2(1) | P>chi2 | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| 1 | 6.000000 | 3.57 | 0.0588 | 0.72235 | 49.83724 |
| 2 | 15.000000 | 12.25 | 0.0005 | 1.98141 | 113.55557 |
| 3 | 8.000000 | 5.44 | 0.0196 | 1.00059 | 63.96252 |

Note: Only 32 of the 63 strata formed in this analysis contribute information about the effect of the explanatory variable.
Mantel-Haenszel estimate controlling for set and age3

| Odds ratio | chi2 (1) | P>chi2 | [95\% conf. interval] |  |
| ---: | :---: | :---: | ---: | ---: |
| 9.666667 | 21.12 | 0.0000 | 2.944702 | 31.733072 |
| Approximate test of homogeneity of odds ratios: chi2 (2) | $=$ | 0.41 |  |  |

There is no further loss of information when we stratify by age3 because age was one of the matching variables.

The full set of matched controls can be used in the same way. For example, the effect of exposure to estrogen is obtained (using the full dataset) with

```
. use https://www.stata-press.com/data/r18/bdendo, clear
(Endometrial cancer and estrogen exposure)
. mhodds fail est set
Mantel-Haenszel estimate of the odds ratio comparing est==1 vs.
    est==0
    controlling for set
```

| Odds ratio | chi2(1) | P>chi2 | $[95 \%$ conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: |
| 8.461538 | 31.16 | 0.0000 | 3.437773 | 20.826746 |

Note: Only 58 of the 63 strata formed in this analysis contribute information about the effect of the explanatory variable.

The effect of exposure to estrogen, stratified by age3, is obtained with

```
. generate age3 =agegrp
. recode age3 1/2=1 3/4=2 5/6=3
(310 changes made to age3)
. mhodds fail est set, by(age3)
Mantel-Haenszel estimate of the odds ratio comparing est==1 vs. est==0
    controlling for set
    by age3
```

| age3 | Odds ratio | chi2(1) | P>chi2 | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| 1 | 3.800000 | 3.38 | 0.0660 | 0.82165 | 17.57438 |
| 2 | 10.666667 | 18.69 | 0.0000 | 2.78773 | 40.81376 |
| 3 | 13.500000 | 9.77 | 0.0018 | 1.59832 | 114.02620 |

Note: Only 58 of the 63 strata formed in this analysis contribute information about the effect of the explanatory variable.

Mantel-Haenszel estimate controlling for set and age3

| Odds ratio | chi2(1) | $\mathrm{P}>\mathrm{chi2}$ | [95\% conf. interval] |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 8.461538 | 31.16 | 0.0000 | 3.437773 |  | 20.82 | 26746 |
| pproximate test of homogeneity of odds ratios: chi2 2 ) $=1.41$ |  |  |  |  |  |  |

## Video examples

Incidence-rate ratios calculator
Risk ratios calculator
Odds ratios for case-control data
Stratified analysis of case-control data
Odds ratios calculator

## Glossary

attributable fraction. An attributable fraction is the reduction in the risk of a disease or other condition of interest when a particular risk factor is removed.
case-control studies. In case-control studies, cases meeting a fixed criterion are matched to noncases ex post to study differences in possible covariates. Relative sample sizes are usually fixed at $1: 1$ or 1:2 but sometimes vary once the survey is complete. In any case, sample sizes do not reflect the distribution in the underlying population.
cohort studies. In cohort studies, a group that is well defined is monitored over time to track the transition of noncases to cases. Cohort studies differ from incidence studies in that they can be retrospective as well as prospective.
confounding. In the analysis of contingency tables, factor or interaction effects are said to be confounded when the effect of one factor is combined with that of another. For example, the effect of alcohol consumption on esophageal cancer may be confounded with the effects of age, smoking, or both. In the presence of confounding, it is often useful to stratify on the confounded factors that are not of primary interest, in the above example, age and smoking.
cross-sectional or prevalence studies. Cross-sectional studies sample distributions of healthy and diseased subjects in the population at one point in time.
crude estimate. A crude estimate has not been adjusted for the effects of other variables. Disregarding a stratification variable, for example, yields a crude estimate.
incidence and incidence rate. Incidence is the number of new failures (for example, number of new cases of a disease) that occur during a specified period in a population at risk (for example, of the disease).

Incidence rate is incidence divided by the sum of the length of time each individual was exposed to the risk.

Do not confuse incidence with prevalence. Prevalence is the fraction of a population that has the disease. Incidence refers to the rate at which people contract a disease, whereas prevalence is the total number actually sick at a given time.
incidence studies, longitudinal studies, and follow-up studies. Whichever word is used, these studies monitor a population for a time to track the transition of noncases into cases. Incidence studies are prospective. Also see cohort studies.
matched case-control study. Also known as a retrospective study, a matched case-control study is a study in which persons with positive outcomes are each matched with one or more persons with negative outcomes but with similar characteristics.
odds and odds ratio. The odds in favor of an event are $o=p /(1-p)$, where $p$ is the probability of the event. Thus if $p=0.2$, the odds are 0.25 , and if $p=0.8$, the odds are 4 .
The $\log$ of the odds is $\ln (o)=\operatorname{logit}(p)=\ln \{p /(1-p)\}$, and logistic-regression models, for instance, fit $\ln (o)$ as a linear function of the covariates.
The odds ratio is a ratio of two odds: $o_{1} / o_{0}$. The individual odds that appear in the ratio are usually for an experimental group and a control group, or two different demographic groups.
prevented fraction. A prevented fraction is the reduction in the risk of a disease or other condition of interest caused by including a protective risk factor or public-health intervention.
prospective study. Also known as a prospective longitudinal study, a prospective study is a study based on observations over the same subjects for a given period.
risk factor. This is a variable associated with an increased or decreased risk of failure.
risk ratio. In a log-linear model, this is the ratio of probability of survival associated with a one-unit increase in a risk factor relative to that calculated without such an increase, that is, $R(x+1) / R(x)$. Given the exponential form of the model, $R(x+1) / R(x)$ is constant and is given by the exponentiated coefficient.

SMR. See standardized mortality (morbidity) ratio.
standardized mortality (morbidity) ratio. Standardized mortality (morbidity) ratio (SMR) is the observed number of deaths divided by the expected number of deaths. It is calculated using indirect standardization: you take the population of the group of interest-say, by age, sex, and other factors-and calculate the expected number of deaths in each cell (expected being defined as the number of deaths that would have been observed if those in the cell had the same mortality
as some other population). You then take the ratio to compare the observed with the expected number of deaths. For instance,

|  | $(1)$ <br> Population <br> of group | Deaths per 100,000 <br> in general pop. | $(1) \times(2)$ <br> Expected \# <br> of deaths | $(4)$ <br> Observed <br> deaths |
| :--- | :---: | :---: | :---: | :---: |
| $25-34$ | 95,965 | 105.2 | 100.9 | 92 |
| $34-44$ | 78,280 | 203.6 | 159.4 | 180 |
| $44-54$ | 52,393 | 428.9 | 224.7 | 242 |
| $55-64$ | 28,914 | 964.6 | 278.9 | 312 |
| Total |  | 763.9 | 826 |  |
|  | SMR $=826 / 763.9=1.08$ |  |  |  |

stratified test. A stratified test is performed separately for each stratum. The stratum-specific results are then combined into an overall test statistic.

## Stored results

ir (without by ()) and iri store the following in $r()$ :
Scalars

```
r(ird)
r(irr)
r(lb_irr)
r(ub_irr)
r(afe)
r(lb_afe)
r(ub_afe)
r(afp)
r(p_lower_midp)
r(p_upper_midp)
r(p_twosided_midp)
r(p_lower_exact)
r(p_upper_exact)
r(p_twosided_exact)
```

$r$ (lb_ird) lower CI bound for IRD
$r$ (ub_ird) upper CI bound for IRD
IRD
IRR
lower CI bound for IRR
upper CI bound for IRR
AFE
lower CI bound for AFE
upper CI bound for AFE
AFP
lower one-sided $p$-value with mid- $p$ adjustment
upper one-sided $p$-value with mid- $p$ adjustment
two-sided $p$-value with mid- $p$ adjustment
lower one-sided exact $p$-value
upper one-sided exact $p$-value
two-sided exact $p$-value
ir, by() stores the following in $r()$ :

Scalars
r(irr)
$r\left(l b \_i r r\right)$
r(ub_irr)
r(ird)
r(lb_ird)
r(ub_ird)
r(crude)
r(lb_crude)
$r$ (ub_crude)
$r$ (pooled)
r(lb_pooled)
r(ub_pooled)
r (df)
r(chi2_mh)
r(chi2_p)

Mantel-Haenszel IRR, if option ird is not specified lower CI bound for Mantel-Haenszel IRR upper CI bound for Mantel-Haenszel IRR Mantel-Haenszel IRD, if option ird is specified lower CI bound for Mantel-Haenszel IRD upper CI bound for Mantel-Haenszel IRD crude IRR or, if option ird is specified, crude IRD lower CI bound for the crude IRR or IRD upper CI bound for the crude IRR or IRD pooled IRR or, if option ird is specified, pooled IRD lower CI bound for pooled IRR or IRD upper CI bound for pooled IRR or IRD degrees of freedom for homogeneity $\chi^{2}$ test
Mantel-Haenszel homogeneity $\chi^{2}$
pooled homogeneity $\chi^{2}$, if option pool is specified
cs and csi store the following in $r()$ :
Scalars

```
r(p)
    r(rd)
    r(lb_rd)
    r(ub_rd)
    r(rr)
    r(lb_rr)
    r(ub_rr)
    r(or)
    r(lb_or)
    r(ub_or)
    r(afe)
    r(lb_afe)
    r(ub_afe)
    r(afp)
    r(crude)
    r(lb_crude)
    r(ub_crude)
r(pooled)
r(lb_pooled)
r(ub_pooled)
r(chi2_mh)
r(chi2_p)
r(df)
r(chi2)
r(p_exact)
r(p1_exact)
```

two-sided $p$-value
risk difference
lower CI bound for risk difference
upper CI bound for risk difference
risk ratio
lower CI bound for risk ratio
upper CI bound for risk ratio
odds ratio
lower CI bound for odds ratio
upper CI bound for odds ratio
AFE
lower CI bound for AFE
upper CI bound for AFE
AFP
crude estimate (cs only)
lower CI bound for crude estimate upper CI bound for crude estimate pooled estimate (cs only)
lower CI bound for pooled estimate
upper CI bound for pooled estimate
Mantel-Haenszel heterogeneity $\chi^{2}$ (cs only)
pooled heterogeneity $\chi^{2}$
degrees of freedom (cs only)
$\chi^{2}$
2-sided Fisher's exact $p$ (exact only)
1 -sided Fisher's exact $p$ (exact only)
cc and cci store the following in r() :
Scalars

```
r(p)
    r(p1_exact)
    r(p_exact)
    r(or)
    r(lb_or)
    r(ub_or)
    r(afe)
    r(lb_afe)
    r(ub_afe)
    r(afp)
    r(crude)
    r(lb_crude)
    r(ub_crude)
    r(pooled)
    r(lb_pooled)
    r(ub_pooled)
    r(chi2_p)
    r(chi2_bd)
    r(df_bd)
    r(chi2_t)
    r(df_t)
    r(df)
    r(chi2)
```

    two-sided \(p\)-value
    one-sided \(p\)-value for Fisher's exact test
    two-sided \(p\)-value for Fisher's exact test
    odds ratio
    lower CI bound for odds ratio
    upper CI bound for odds ratio
    AFE
    lower CI bound for AFE
    upper CI bound for AFE
    AFP
    crude estimate (cc only)
    lower CI bound for crude estimate
    upper CI bound for crude estimate
    pooled estimate (cc only)
    lower CI bound for pooled estimate
    upper CI bound for pooled estimate
    pooled heterogeneity \(\chi^{2}\)
    Breslow-Day \(\chi^{2}\)
    degrees of freedom for Breslow-Day \(\chi^{2}\) test
    Tarone \(\chi^{2}\)
    degrees of freedom for Tarone \(\chi^{2}\) test
    degrees of freedom
    \(\chi^{2}\)
    tabodds stores the following in $r()$ :
Scalars

```
r(odds)
r(lb_odds)
r(ub_odds)
r(chi2_hom)
r(p_hom)
r(df_hom)
r(chi2_tr)
r(p_trend)
```

odds
lower CI bound for odds
upper CI bound for odds
$\chi^{2}$ for test of homogeneity
$p$-value for test of homogeneity
degrees of freedom for test of homogeneity
$\chi^{2}$ for score test for trend
$p$-value for score test for trend
mhodds stores the following in r() :
Scalars

```
r(p)
    r(or)
    r(lb_or)
    r(ub_or)
    r(chi2_hom)
    r(df_hom)
    r(chi2)
```

```
two-sided p-value
```

two-sided p-value
odds ratio
odds ratio
lower CI bound for odds ratio
lower CI bound for odds ratio
upper CI bound for odds ratio
upper CI bound for odds ratio
\chi }\mp@subsup{}{}{2}\mathrm{ for test of homogeneity
\chi }\mp@subsup{}{}{2}\mathrm{ for test of homogeneity
degrees of freedom for test of homogeneity
degrees of freedom for test of homogeneity
\chi

```
\chi
```

Matrices
r(strata_table) odds ratios for strata, if by () specified
mcc and mcci store the following in r() :
Scalars

| $\begin{aligned} & r\left(p \_e x a c t\right) \\ & r(o r) \end{aligned}$ | two-sided $p$-value for McNemar's test odds ratio |
| :---: | :---: |
| r(lb_or) | lower CI bound for odds ratio |
| r(ub_or) | upper CI bound for odds ratio |
| r(D_f) | difference in proportion with factor |
| r(lb_D_f) | lower CI bound for difference in proportion |
| r(ub_D_f) | upper CI bound for difference in proportion |
| r(R_f) | ratio of proportion with factor |
| r(lb_R_f) | lower CI bound for ratio of proportion |
| r (ub_R_f) | upper CI bound for ratio of proportion |
| r(RD_f) | relative difference in proportion with factor |
| r(lb_RD_f) | lower CI bound for relative difference in proportion |
| r(ub_RD_f) | upper CI bound for relative difference in proportion |
| $r$ (chi2) |  |

```
    (p_exact)
    r(or)
    r(lb_or)
```

    lower CI bound for odds ratio
    upper CI bound for odds ratio
    difference in proportion with factor
    lower CI bound for difference in proportion
    upper CI bound for difference in proportion
    ratio of proportion with factor
    lower CI bound for ratio of proportion
    upper CI bound for ratio of proportion
    relative difference in proportion with factor
    lower CI bound for relative difference in proportion
    upper CI bound for relative difference in proportion
    \(\chi^{2}\)
    
## Methods and formulas

The notation for incidence-rate data is

|  | Exposed | Unexposed | Total |
| :--- | :---: | :---: | :---: |
| Cases | $a$ | $b$ | $M_{1}$ |
| Person-time | $N_{1}$ | $N_{0}$ | $T$ |

The notation for $2 \times 2$ tables is

|  | Exposed | Unexposed | Total |
| :--- | :---: | :---: | :---: |
| Cases | $a$ | $b$ | $M_{1}$ |
| Controls | $c$ | $d$ | $M_{0}$ |
| Total | $N_{1}$ | $N_{0}$ | $T$ |

The notation for $2 \times k$ tables is

|  | Exposure level |  |  |  |  |
| :--- | :---: | :---: | :---: | :---: | :---: |
|  | 1 | 2 | $\ldots$ | k | Total |
| Cases | $a_{1}$ | $a_{2}$ | $\ldots$ | $a_{k}$ | $M_{1}$ |
| Controls | $c_{1}$ | $c_{2}$ | $\ldots$ | $c_{k}$ | $M_{0}$ |
| Total | $N_{1}$ | $N_{2}$ | $\ldots$ | $N_{k}$ | $T$ |

If the tables are stratified, all quantities are indexed by $i$, the stratum number.
We will refer to Fleiss, Levin, and Paik (2003); Kleinbaum, Kupper, and Morgenstern (1982); and Rothman (1986) so often that we will adopt the notation F-23 to mean Fleiss, Levin, and Paik (2003) page 23; KKM-52 to mean Kleinbaum, Kupper, and Morgenstern (1982) page 52; and R-164 to mean Rothman (1986) page 164.

We usually avoid making the continuity corrections to $\chi^{2}$ statistics, following the advice of KKM-292: ". . . the use of a continuity correction has been the subject of considerable debate in the statistical literature .... On the basis of our evaluation of this debate and other evidence, we do not recommend the use of the continuity correction." Breslow and Day (1980, 133), on the other hand, argue for inclusion of the correction, but not strongly. Their summary is that for small datasets, one should use exact statistics. In practice, we believe that the adjustment makes little difference for reasonably sized datasets.

Methods and formulas are presented under the following headings:

```
Unstratified incidence-rate data (ir and iri)
Unstratified cumulative incidence data (cs and csi)
Unstratified case-control data (cc and cci)
Unstratified matched case-control data (mcc and mcci)
Stratified incidence-rate data (ir with the by() option)
Stratified cumulative incidence data (cs with the by() option)
Stratified case-control data (cc with by() option, mhodds, tabodds)
```


## Unstratified incidence-rate data (ir and iri)

The IRD is defined as $I_{d}=a / N_{1}-b / N_{0}$ (R-164). The standard error of the difference is $s_{I_{d}} \approx \sqrt{a / N_{1}^{2}+b / N_{0}^{2}}$ (R-170), from which confidence intervals are calculated.

The IRR is defined as $I_{r}=\left(a / N_{1}\right) /\left(b / N_{0}\right)$ (R-164). Let $p_{l}$ and $p_{u}$ be the exact confidence interval of the binomial probability for observing $a$ successes in $M_{1}$ trials (obtained from cii proportions; see $[\mathrm{R}] \mathbf{c i})$. The exact confidence interval for the incidence ratio is then $\left(p_{l} N_{0}\right) /\left\{\left(1-p_{l}\right) N_{1}\right\}$ to $\left(p_{u} N_{0}\right) /\left\{\left(1-p_{u}\right) N_{1}\right\}$ (R-166).

The AFE is defined as AFE $=\left(I_{r}-1\right) / I_{r}$ for $I_{r} \geq 1$ (КKM-164; R-38); the confidence interval is obtained by similarly transforming the interval values of $I_{r}$. The AFP is AFP $=$ AFE $\cdot a / M_{1}$ (KKM-161); no confidence interval is reported. For $I_{r}<1$, the PFE is defined as PFE $=1-I_{r}$ (KKM-166; R-39); the confidence interval is obtained by similarly transforming the interval values of $I_{r}$. The PFP is $\mathrm{PFP}=\mathrm{PFE} \cdot N_{1} / T$ (KKM-165); no confidence interval is reported.

Exact one-sided $p$-values are calculated as the binomial probabilities (with $n=M_{1}$ and $p=N_{1} / T$ ) $\operatorname{Pr}(k \leq a)$ and $\operatorname{Pr}(k \geq a)$. Exact $p$-values tend to be overly conservative, so the mid- $p$ adjustment ( $\mathrm{R}-155$ ) reduces the exact $p$-values by subtracting half the probability of the observed result from each one-sided $p$-value. That is, one-sided $p$-values with the mid- $p$ adjustment are the binomial probabilities $\operatorname{Pr}(k \leq a)-\operatorname{Pr}(k=a) / 2$ and $\operatorname{Pr}(k \geq a)-\operatorname{Pr}(k=a) / 2$. The two-sided $p$-value is twice the smallest one-sided $p$-value for both the exact and mid- $p$-adjustment calculations. Rather than using twice the smallest one-sided $p$-value for the two-sided $p$-value, there is an another formula for the two-sided $p$-value that is sometimes used. The command bitest uses this alternative; see [R] bitest for details.

## Unstratified cumulative incidence data (cs and csi)

The risk difference is defined as $R_{d}=a / N_{1}-b / N_{0}$ (R-164). Its standard error is

$$
s_{R_{d}} \approx\left\{\frac{a c}{N_{1}^{3}}+\frac{b d}{N_{0}^{3}}\right\}^{1 / 2}
$$

(R-172), from which confidence intervals are calculated.
The risk ratio is defined as $R_{r}=\left(a / N_{1}\right) /\left(b / N_{0}\right)$ (R-165). The standard error of $\ln R_{r}$ is

$$
s_{\ln R_{r}} \approx\left(\frac{c}{a N_{1}}+\frac{d}{b N_{0}}\right)^{1 / 2}
$$

(R-173), from which confidence intervals are calculated.
For $R_{r} \geq 1$, the AFE is calculated as $\mathrm{AFE}=\left(R_{r}-1\right) / R_{r}$ (KKM-164; R-38); the confidence interval is obtained by similarly transforming the interval values for $R_{r}$. The AFP is calculated as $\mathrm{AFP}=\mathrm{AFE} \cdot a / M_{1}$ (KKM-161); no confidence interval is reported, but $\mathrm{F}-128$ provides

$$
\left\{\frac{c+(a+d) \mathrm{AFP}}{b T}\right\}^{1 / 2}
$$

as the approximate standard error of $\ln (1-\mathrm{AFP})$.
For $R_{r}<1$, the PFE is calculated as PFE $=1-R_{r}$ (KKM-166; R-39); the confidence interval is obtained by similarly transforming the interval values for $R_{r}$. The PFP is calculated as PFP $=$ PFE $\cdot N_{1} / T$; no confidence interval is reported.

The odds ratio, available with the or option, is defined as $\psi=(a d) /(b c)$ ( $\mathrm{R}-165$ ). Several confidence intervals are available. The default interval for cs and csi is the Cornfield (1956) approximate interval. If we let $z_{\alpha}$ be the index from a normal distribution for an $\alpha$ significance level, the Cornfield interval $\left(\psi_{l}, \psi_{u}\right)$ is calculated from

$$
\begin{aligned}
\psi_{l} & =a_{l}\left(M_{0}-N_{1}+a_{l}\right) /\left\{\left(N_{1}-a_{l}\right)\left(M_{1}-a_{l}\right)\right\} \\
\psi_{u} & =a_{u}\left(M_{0}-N_{1}+a_{u}\right) /\left\{\left(N_{1}-a_{u}\right)\left(M_{1}-a_{u}\right)\right\}
\end{aligned}
$$

where $a_{u}$ and $a_{l}$ are determined iteratively from

$$
a_{i+1}=a \pm z_{\alpha}\left(\frac{1}{a_{i}}+\frac{1}{N_{1}-a_{i}}+\frac{1}{M_{1}-a_{i}}+\frac{1}{M_{0}-N_{1}+a_{i}}\right)^{-1 / 2}
$$

(Newman 2001, sec. 4.4). $a_{i+1}$ converges to $a_{u}$ using the plus sign and $a_{l}$ using the minus sign. $a_{0}$ is taken as $a$. With small numbers, the iterative technique may fail. It is then restarted by decrementing ( $a_{l}$ ) or incrementing $\left(a_{u}\right) a_{0}$. If that fails, $a_{0}$ is again decremented or incremented and iterations restarted, and so on, until a terminal condition is met ( $a_{0}<0$ or $a_{0}>M_{1}$ ), at which point the value is not calculated.

The Woolf odds-ratio confidence intervals are available with cs and csi. The Woolf method (Woolf 1955; R-173; Schlesselman 1982, 176), available with the woolf option, estimates the standard error of $\ln \psi$ by

$$
s_{\ln \psi}=\left(\frac{1}{a}+\frac{1}{b}+\frac{1}{c}+\frac{1}{d}\right)^{1 / 2}
$$

from which confidence intervals are calculated. The Woolf interval cannot be calculated when there exists a zero cell. Sometimes the Woolf interval is called the "logit interval" (Breslow and Day 1980, 134).

The $\chi^{2}$ statistic, reported by default, can be calculated as

$$
\chi^{2}=\frac{(a d-b c)^{2} T}{M_{1} M_{0} N_{1} N_{0}}
$$

(Schlesselman 1982, 179).
Fisher's exact test, available with the exact option, is calculated as described in $[R]$ tabulate twoway.

## Unstratified case-control data (cc and cci)

cc and cci report by default the same odds ratio, $\psi$, that is available with the or option in cs and csi. But cc and cci calculate the confidence interval differently: they default to the exact odds-ratio interval, not the Cornfield interval, but you can request the Cornfield interval with the cornfield option. The $1-\alpha$ exact interval $(\underline{R}, \bar{R})$ is calculated from

$$
\alpha / 2=\frac{\sum_{k=a}^{\min \left(N_{1}, M_{1}\right)}\binom{N_{1}}{k}\binom{N_{0}}{M_{1}-k} \underline{R}^{k}}{\sum_{k=\max \left(0, M_{1}-N_{0}\right)}^{\min \left(N_{1}, M_{1}\right)}\binom{N_{1}}{k}\binom{N_{0}}{M_{1}-k} \underline{R}^{k}}
$$

and

$$
1-\alpha / 2=\frac{\sum_{k=a+1}^{\min \left(N_{1}, M_{1}\right)}\binom{N_{1}}{k}\binom{N_{0}}{M_{1}-k} \bar{R}^{k}}{\sum_{k=\max \left(0, M_{1}-N_{0}\right)}^{\min \left(N_{1}, M_{1}\right)}\binom{N_{1}}{k}\binom{N_{0}}{M_{1}-k} \bar{R}^{k}}
$$

( $\mathrm{R}-169$ ). The equations invert two one-sided Fisher exact tests.
cc and cci also report the same tests of significance as cs and csi: the $\chi^{2}$ statistic is the default, and Fisher's exact test is obtained with the exact option. The odds ratio, $\psi$, is used as an estimate of the risk ratio in calculating attributable or prevented fractions. For $\psi \geq 1$, the AFE is calculated as $\mathrm{AFE}=(\psi-1) / \psi($ KKM-164 $)$; the confidence interval is obtained by similarly transforming the interval values for $\psi$. The AFP is calculated as AFP $=$ AFE $\cdot a / M_{1}$ (KKM-161). No confidence interval is reported; however, F-152 provides

$$
\left(\frac{a}{M_{1} b}+\frac{c}{M_{0} d}\right)^{1 / 2}
$$

as the standard error of $\ln (1-\mathrm{AFP})$.
For $\psi<1$, the PFE is calculated as PFE $=1-\psi$ (KKM-166); the confidence interval is obtained by similarly transforming the interval values for $\psi$. The PFP is calculated as $\operatorname{PFP}=\left\{\left(a / M_{1}\right) \mathrm{PFE}\right\} /\left\{\left(a / M_{1}\right) \mathrm{PFE}+\psi\right\}$ (KKM-165); no confidence interval is reported.

## Unstratified matched case-control data (mcc and mcci)

Referring to the table at the beginning of Matched case-control data under Remarks and examples above, the columns of the table indicate controls; the rows are cases. Each entry in the table reflects a pair of a matched case and control.

McNemar's (1947) $\chi^{2}$ is defined as

$$
\chi^{2}=\frac{(b-c)^{2}}{b+c}
$$

(ККМ-389).
The proportion of controls with the factor is $p_{1}=N_{1} / T$, and the proportion of cases with the factor is $p_{2}=M_{1} / T$.

The difference in the proportions is $P_{d}=p_{2}-p_{1}$. An estimate of its standard error when the two underlying proportions are not hypothesized to be equal is

$$
s_{P_{d}} \approx \frac{\{(a+d)(b+c)+4 b c\}^{1 / 2}}{T^{3 / 2}}
$$

(F-378), from which confidence intervals are calculated. The confidence interval uses a continuity correction (F-378, eq. 13.15).

The ratio of the proportions is $P_{r}=p_{2} / p_{1}(\mathrm{R}-276, \mathrm{R}-278)$. The standard error of $\ln P_{r}$ is

$$
s_{\ln P_{r}} \approx\left(\frac{b+c}{M_{1} N_{1}}\right)^{1 / 2}
$$

(R-276), from which confidence intervals are calculated.
The relative difference in the proportions is $P_{e}=(b-c) /(b+d)$ (F-379). Its standard error is

$$
s_{P_{e}} \approx(b+d)^{-2}\{(b+c+d)(b c+b d+c d)-b c d\}^{1 / 2}
$$

(F-379), from which confidence intervals are calculated.
The odds ratio is $\psi=b / c$ ( $\mathrm{F}-376$ ), and the exact Fisher confidence interval is obtained by transforming into odds ratios the exact binomial confidence interval for the binomial parameter from observing $b$ successes in $b+c$ trials (R-264). Binomial confidence limits are obtained from cii proportions (see [R] ci) and are transformed by $p /(1-p)$.

The exact McNemar significance probability is a two-tailed exact test of $H_{0}: \psi=1$. The $p$-value, calculated from the binomial distribution, is

$$
\min \left\{1,2 \sum_{k=0}^{\min (b, c)}\binom{b+c}{k}\left(\frac{1}{2}\right)^{b+c}\right\}
$$

(Agresti 2013, 416).

Quinn McNemar (1900-1986) was born in West Virginia and attended college there and in Pennsylvania. After a brief spell of high school teaching, he began graduate study of psychology at Stanford and then joined the faculty. McNemar's text Psychological Statistics, first published in 1949, was widely influential, and he made many substantive and methodological contributions to the application of statistics in psychology.

## Stratified incidence-rate data (ir with the by() option)

Statistics presented for each stratum are calculated independently according to the formulas in Unstratified incidence-rate data (ir and iri) above. Within strata, the Mantel-Haenszel style weight is $W_{i}=b_{i} N_{1 i} / T_{i}$, and the Mantel-Haenszel combined incidence-rate ratio (Rothman and Boice 1982) is

$$
I_{\mathrm{mh}}=\frac{\sum_{i} a_{i} N_{0 i} / T_{i}}{\sum_{i} W_{i}}
$$

( $\mathrm{R}-196$ ). The standard error for the $\log$ of the incidence-rate ratio was derived by Greenland and Robins $(1985,63)$ and appears in R-213:

$$
s_{\ln I_{\mathrm{mh}}} \approx\left\{\frac{\sum_{i} M_{1 i} N_{1 i} N_{0 i} / T_{i}^{2}}{\left(\sum_{i} a_{i} N_{0 i} / T_{i}\right)\left(\sum_{i} b_{i} N_{1 i} / T_{i}\right)}\right\}^{1 / 2}
$$

The confidence interval is calculated first on the $\log$ scale and then is transformed.
For standardized rates, let $w_{i}$ be the user-specified weight within stratum $i$. The standardized rate difference (the ird option) and rate ratio are defined as

$$
\begin{aligned}
\mathrm{SRD} & =\frac{\sum_{i} w_{i}\left(R_{1 i}-R_{0 i}\right)}{\sum_{i} w_{i}} \\
\mathrm{SRR} & =\frac{\sum_{i} w_{i} R_{1 i}}{\sum_{i} w_{i} R_{0 i}}
\end{aligned}
$$

( $\mathrm{R}-229$ ). The standard error of SRD is

$$
s_{\mathrm{SRD}} \approx\left\{\frac{1}{\left(\sum_{i} w_{i}\right)^{2}} \sum_{i} w_{i}^{2}\left(\frac{a_{i}}{N_{1 i}^{2}}+\frac{b_{i}}{N_{0 i}^{2}}\right)\right\}^{1 / 2}
$$

(R-231), from which confidence intervals are calculated. The standard error of $\ln (S R R)$ is

$$
s_{\ln (\mathrm{SRR})} \approx\left\{\frac{\sum_{i} w_{i}^{2} a_{i} / N_{1 i}^{2}}{\left(\sum_{i} w_{i} R_{1 i}\right)^{2}}+\frac{\sum_{i} w_{i}^{2} b_{i} / N_{0 i}^{2}}{\left(\sum_{i} w_{i} R_{0 i}\right)^{2}}\right\}^{1 / 2}
$$

( $\mathrm{R}-231$ ), from which confidence intervals are calculated.
Internally and externally standardized measures are calculated using $w_{i}=N_{1 i}$ and $w_{i}=N_{0 i}$, respectively, and are obtained with the istandard and estandard options, respectively.

Directly pooled estimates are available with the pool option. The directly pooled estimate is a weighted average of stratum-specific estimates; each weight, $w_{i}$, is inversely proportional to the variance of the estimate for stratum $i$. The variances for rate differences come from the formulas in Unstratified incidence-rate data (ir and iri), while the variances of log rate-ratios are estimated by $\left(1 / a_{i}+1 / b_{i}\right)$ (R-184). Ratios are averaged in the log scale before being exponentiated. The standard error of the directly pooled estimate is calculated as $1 / \sqrt{\sum w_{i}}$, from which confidence intervals are calculated (R-183-185); the calculation for ratios again uses the log scale.

For rate differences, the $\chi^{2}$ test of homogeneity is calculated as $\sum\left(R_{d i}-\widehat{R}_{d}\right)^{2} / \operatorname{var}\left(R_{d i}\right)$, where $R_{d i}$ are the stratum-specific rate differences and $\widehat{R}_{d}$ is the directly pooled estimate. The number of degrees of freedom is one less than the number of strata ( $\mathrm{R}-222$ ).

For rate ratios, the same calculation is made, except that it is made on a logarithmic scale using $\ln \left(R_{r i}\right)$ (R-222), and $\ln \left(\widehat{R}_{d}\right)$ may be the $\log$ of either the directly pooled estimate or the Mantel-Haenszel estimate.

## Stratified cumulative incidence data (cs with the by() option)

Statistics presented for each stratum are calculated independently according to the formulas in Unstratified cumulative incidence data (cs and csi) above. The Mantel-Haenszel $\chi^{2}$ test (Mantel and Haenszel 1959) is

$$
\chi_{\mathrm{mh}}^{2}=\frac{\left\{\sum_{i}\left(a_{i}-N_{1 i} M_{1 i} / T_{i}\right)\right\}^{2}}{\sum_{i}\left(N_{1 i} N_{0 i} M_{1 i} M_{0 i}\right) /\left\{T_{i}^{2}\left(T_{i}-1\right)\right\}}
$$

(R-206).
For the odds ratio (available with the or option), the Mantel-Haenszel weight is $W_{i}=b_{i} c_{i} / T_{i}$, and the combined odds ratio (Mantel and Haenszel 1959) is

$$
\psi_{\mathrm{mh}}=\frac{\sum_{i} a_{i} d_{i} / T_{i}}{\sum_{i} W_{i}}
$$

( $\mathrm{R}-195$ ). The standard error (Robins, Breslow, and Greenland 1986) is

$$
s_{\ln \psi_{\mathrm{mh}}} \approx\left\{\frac{\sum_{i} P_{i} R_{i}}{2\left(\sum_{i} R_{i}\right)^{2}}+\frac{\sum_{i} P_{i} S_{i}+Q_{i} R_{i}}{2 \sum_{i} R_{i} \sum_{i} S_{i}}+\frac{\sum_{i} Q_{i} S_{i}}{2\left(\sum_{i} S_{i}\right)^{2}}\right\}^{1 / 2}
$$

where

$$
\begin{aligned}
P_{i} & =\left(a_{i}+d_{i}\right) / T_{i} \\
Q_{i} & =\left(b_{i}+c_{i}\right) / T_{i} \\
R_{i} & =a_{i} d_{i} / T_{i} \\
S_{i} & =b_{i} c_{i} / T_{i}
\end{aligned}
$$

(R-220).
For the risk ratio (the default), the Mantel-Haenszel-style weight is $W_{i}=b_{i} N_{1 i} / T_{i}$, and the combined risk ratio (Rothman and Boice 1982) is

$$
R_{\mathrm{mh}}=\frac{\sum_{i} a_{i} N_{0 i} / T_{i}}{\sum_{i} W_{i}}
$$

( R -196). The standard error (Greenland and Robins 1985) is

$$
s_{\ln R_{\mathrm{mh}}} \approx\left\{\frac{\sum_{i}\left(M_{1 i} N_{1 i} N_{0 i}-a_{i} b_{i} T_{i}\right) / T_{i}^{2}}{\left(\sum_{i} a_{i} N_{0 i} / T_{i}\right)\left(\sum_{i} b_{i} N_{1 i} / T_{i}\right)}\right\}^{1 / 2}
$$

( $\mathrm{R}-216$ ), from which confidence intervals are calculated.
For standardized rates, let $w_{i}$ be the user-specified weight within stratum $i$. The standardized rate difference (SRD, the rd option) and rate ratios (SRR, the default) are defined as in Stratified incidence-rate data (ir with the by() option), where the individual risks are defined $R_{1 i}=a_{i} / N_{1 i}$ and $R_{0 i}=b_{i} / N_{0 i}$. The standard error of SRD is

$$
s_{\mathrm{SRD}} \approx\left[\frac{1}{\left(\sum_{i} w_{i}\right)^{2}} \sum_{i} w_{i}^{2}\left\{\frac{a_{i}\left(N_{1 i}-a_{i}\right)}{N_{1 i}^{3}}+\frac{b_{i}\left(N_{0 i}-b_{i}\right)}{N_{0 i}^{3}}\right\}\right]^{1 / 2}
$$

( $\mathrm{R}-231$ ), from which confidence intervals are calculated. The standard error of $\ln (\mathrm{SRR})$ is

$$
s_{\ln (\mathrm{SRR})} \approx\left\{\frac{\sum_{i} w_{i}^{2} a_{i}\left(N_{1 i}-a_{i}\right) / N_{1 i}^{3}}{\left(\sum_{i} w_{i} R_{1 i}\right)^{2}}+\frac{\sum_{i} w_{i}^{2} b_{i}\left(N_{0 i}-b_{i}\right) / N_{0 i}^{3}}{\left(\sum_{i} w_{i} R_{0 i}\right)^{2}}\right\}^{1 / 2}
$$

( $\mathrm{R}-231$ ), from which confidence intervals are calculated.
Internally and externally standardized measures are calculated using $w_{i}=N_{1 i}$ and $w_{i}=N_{0 i}$, respectively, and are obtained with the istandard and estandard options, respectively.

Directly pooled estimates of the odds ratio are available when you specify both the pool and or options. The directly pooled estimate is a weighted average of stratum-specific log odds-ratios; each weight, $w_{i}$, is inversely proportional to the variance of the log odds-ratio for stratum $i$. The variances of the log odds-ratios are estimated by Woolf's method, described under Unstratified cumulative incidence data (cs and csi). The standard error of the directly pooled log odds-ratio is calculated as $1 / \sqrt{\sum w_{i}}$, from which confidence intervals are calculated and then exponentiated (Kahn and Sempos 1989, 113-115).

Direct pooling is also available for risk ratios and risk differences; the variance formulas may be found in Unstratified cumulative incidence data (cs and csi). The directly pooled risk ratio is provided when the pool option is specified. The directly pooled risk difference is provided only when you specify the pool and rd options, and one of the estandard, istandard, and standard() options.

For risk differences, the $\chi^{2}$ test of homogeneity is calculated as $\sum\left(R_{d i}-\widehat{R}_{d}\right)^{2} / \operatorname{var}\left(R_{d i}\right)$, where $R_{d i}$ are the stratum-specific risk differences and $\widehat{R}_{d}$ is the directly pooled estimate. The number of degrees of freedom is one less than the number of strata ( $\mathrm{R}-222$ ).

For risk and odds ratios, the same calculation is made, except that it is made in the log scale using $\ln \left(R_{r i}\right)$ or $\ln \left(\psi_{i}\right)$ (R-222), and $\ln \left(\widehat{R}_{d}\right)$ may be the $\log$ of either the directly pooled estimate or the Mantel-Haenszel estimate.

## Stratified case-control data (cc with by() option, mhodds, tabodds)

Statistics presented for each stratum are calculated independently according to the formulas in Unstratified cumulative incidence data (cs and csi) above. The combined odds ratio, $\psi_{\mathrm{mh}}$, and the test that $\psi_{\mathrm{mh}}=1\left(\chi_{\mathrm{mh}}^{2}\right)$ are calculated as described in Stratified cumulative incidence data (cs with the by() option) above.

For standardized weights, let $w_{i}$ be the user-specified weight within stratum $i$. The standardized odds ratio (the standard() option) is calculated as

$$
\mathrm{SOR}=\frac{\sum_{i} w_{i} a_{i} / c_{i}}{\sum_{i} w_{i} b_{i} / d_{i}}
$$

(Greenland 1986, 473). The standard error of $\ln (S O R)$ is

$$
s_{\ln (\mathrm{SOR})}=\left\{\frac{\sum_{i}\left(w_{i} a_{i} / c_{i}\right)^{2}\left(\frac{1}{a_{i}}+\frac{1}{b_{i}}+\frac{1}{c_{i}}+\frac{1}{d_{i}}\right)}{\left(\sum_{i} w_{i} a_{i} / c_{i}\right)^{2}}\right\}^{1 / 2}
$$

from which confidence intervals are calculated. The internally and externally standardized odds ratios are calculated using $w_{i}=c_{i}$ and $w_{i}=d_{i}$, respectively.

The directly pooled estimate of the odds ratio (the pool option) is calculated as described in Stratified cumulative incidence data (cs with the by() option) above.

The directly pooled and Mantel-Haenszel $\chi^{2}$ tests of homogeneity are calculated as $\sum\left\{\ln \left(R_{r i}\right)-\right.$ $\left.\ln \left(\widehat{R}_{r}\right)\right\}^{2} / \operatorname{var}\left\{\ln \left(R_{r i}\right)\right\}$, where $R_{r i}$ are the stratum-specific odds ratios and $\widehat{R}_{r}$ is the pooled estimate (Mantel-Haenszel or directly pooled). The number of degrees of freedom is one less than the number of strata (R-222).

The Breslow-Day $\chi^{2}$ test of homogeneity is available with the bd option. Let $\widehat{\psi}$ be the MantelHaenszel estimate of the common odds ratio, and let $A_{i}(\widehat{\psi})$ be the fitted count for cell $a$; $A_{i}(\widehat{\psi})$ is found by solving the quadratic equation

$$
A\left(M_{0}-N_{1}+A\right)=(\widehat{\psi})\left(M_{1}-A\right)\left(N_{1}-A\right)
$$

and choosing the root that makes all cells in stratum $i$ positive. Let $\operatorname{Var}\left(a_{i} ; \widehat{\psi}\right)$ be the estimated variance of $a_{i}$ conditioned on the margins and on an odds ratio of $\widehat{\psi}$ :

$$
\operatorname{Var}\left(a_{i} ; \widehat{\psi}\right)=\left\{\frac{1}{A_{i}(\widehat{\psi})}+\frac{1}{M_{1 i}-A_{i}(\widehat{\psi})}+\frac{1}{N_{1 i}-A_{i}(\widehat{\psi})}+\frac{1}{M_{0 i}-N_{1 i}+A_{i}(\widehat{\psi})}\right\}^{-1}
$$

The Breslow-Day $\chi^{2}$ statistic is then

$$
\sum_{i} \frac{\left\{a_{i}-A_{i}(\widehat{\psi})\right\}^{2}}{\operatorname{Var}\left(a_{i} ; \widehat{\psi}\right)}
$$

The Tarone $\chi^{2}$ test of homogeneity (the tarone option) is calculated as

$$
\sum_{i} \frac{\left\{a_{i}-A_{i}(\widehat{\psi})\right\}^{2}}{\operatorname{Var}\left(a_{i} ; \widehat{\psi}\right)}-\frac{\left\{\sum_{i} a_{i}-\sum_{i} A_{i}(\widehat{\psi})\right\}^{2}}{\sum_{i} \operatorname{Var}\left(a_{i} ; \widehat{\psi}\right)}
$$

Tarone (1985) provides this correction to the Breslow-Day statistic to ensure that its distribution is asymptotically $\chi^{2}$. Without the correction, the Breslow-Day statistic does not necessarily follow a $\chi^{2}$ distribution because it is based on the Mantel-Haenszel estimate, $\widehat{\psi}$, which is an inefficient estimator of the common odds ratio.

When the exposure variable has multiple levels, mhodds calculates an approximate estimate of the $\log$ odds-ratio for a one-unit increase in exposure as the ratio of the score statistic, $U$, to its variance, $V$ (Clayton and Hills 1993, 103), which are defined below. This is a one-step NewtonRaphson approximation to the maximum likelihood estimate. Within-stratum estimates are combined with Mantel-Haenszel weights.

By default, both tabodds and mhodds produce test statistics and confidence intervals based on score statistics (Clayton and Hills 1993). tabodds reports confidence intervals for the odds of the $i$ th exposure level, unless the adjust () or or option is specified. The confidence interval for odds ${ }_{i}$, $i=1, \ldots, k$, is given by

$$
\operatorname{odds}_{i} \cdot \exp \left( \pm z \sqrt{1 / a_{i}+1 / c_{i}}\right)
$$

The score $\chi^{2}$ test of homogeneity of odds is calculated as

$$
\chi_{k-1}^{2}=\frac{T(T-1)}{M_{1} M_{0}} \sum_{i=1}^{k} \frac{\left(a_{i}-E_{i}\right)^{2}}{N_{i}}
$$

where $E_{i}=\left(M_{1} N_{i}\right) / T$.

Let $l_{i}$ denote the value of the exposure at the $i$ th level. The score $\chi^{2}$ test for trend of odds is calculated as

$$
\chi_{1}^{2}=\frac{U^{2}}{V}
$$

where

$$
U=\frac{M_{1} M_{0}}{T}\left(\sum_{i=1}^{k} \frac{a_{i} l_{i}}{M_{1}}-\sum_{i=1}^{k} \frac{c_{i} l_{i}}{M_{0}}\right)
$$

and

$$
V=\frac{M_{1} M_{0}}{T}\left\{\frac{\sum_{i=1}^{k} N_{i} l_{i}^{2}-\left(\sum_{i=1}^{k} N_{i} l_{i}\right)^{2} / T}{T-1}\right\}
$$

## Acknowledgments

We thank Hal Morgenstern, professor emeritus of the Department of Epidemiology at the University of Michigan, Ardythe Morrow of Cincinnati Children's Hospital, and the late Stewart West of Baylor College of Medicine for their assistance in designing these commands.

We thank Jonathan Freeman (1939-2000) of the Department of Epidemiology at Harvard School of Public Health for encouraging us to extend these commands to include tests for homogeneity, for helpful comments on the default behavior of the commands, and for his comments on an early draft of this section.

We thank David Clayton (retired) of the Cambridge Institute for Medical Research and Michael Hills (1934-2021) of the London School of Hygiene and Tropical Medicine, who wrote the original versions of mhodds and tabodds.

Finally, we thank William Dupont and Dale Plummer, both at the Department of Biostatistics, Vanderbilt University, for their contribution to the implementation of exact confidence intervals for the odds ratios for cc and cci.

John Snow (1813-1858) was born in York, England. From age 14, he worked as an apprentice and assistant to surgeons in northeast England and Yorkshire. In 1836, Snow moved to London; he was admitted to the Royal College of Surgeons in 1838 and the Royal College of Physicians in 1850. He made outstanding contributions to the adoption of anesthesia and is considered one of the originators of modern epidemiology. Snow died following a stroke in 1858.

Snow calculated dosages for ether and chloroform. He personally administered chloroform to Queen Victoria for the births of her last two children, which helped obstetric anesthesia gain wider acceptance.

Snow was skeptical of the miasma theory that cholera was caused by foul air. His essay On the Mode of Communication of Cholera was first published in 1849 and then greatly enlarged in 1855 with the results of his very detailed investigation of the role of water supply in the epidemic of 1854 in the Soho district of London. Snow identified the source of the outbreak as the public water pump on Broad Street (now Broadwick Street), leading the local council to remove the pump handle. It was later discovered that the well had been dug very close to an old cesspit. He also mapped the clustering of cholera cases around the pump and related mortality to water sources, clearly showing higher deathrates in areas supplied by the Southwark and Vauxhall Waterworks Company, which was taking water from sewage-polluted sections of the River Thames. Snow is widely regarded as a pioneer in public health, epidemiology, and medical geography.

Janet Elizabeth Lane-Claypon (1877-1967) was a pioneer in the use of cohort and case-control studies. She was born in Lincolnshire county, England, and began her studies at the London School of Medicine for Women in 1898. From 1907 to 1912, she was at the Lister Institute of Preventive Medicine, where she was a colleague of Major Greenwood. By the end of her studies, she had obtained a doctorate in both physiology and medicine.

In 1912, Lane-Claypon published one of the first retrospective cohort studies, examining the weight gain of babies fed cow's milk versus babies fed breast milk. Using statistical techniques, she determined that babies fed breast milk gained weight faster; she later employed that knowledge to become a public health advocate for breast feeding.

She also conducted one of the first case-control studies, examining risk factors associated with breast cancer. Her study included 500 women without breast cancer and 500 women with breast cancer. To obtain what was at the time a remarkably large sample, she coordinated data collection from nine different hospitals. Carefully controlling for variables including occupation and infant mortality, she determined that factors like age at first pregnancy, age at menopause, and number of children all influence the incidence of breast cancer; these factors are still considered to be among the prime determinants.

In conjunction with the Ministry of Health, in 1926 Lane-Claypon published one of the first studies to contain long-term follow-up results. In that study, she followed patients who had undergone surgery for breast cancer for up to 10 years after the operation. As is still the case today, her study showed that the sooner the cancer was treated, the better the woman's chance for long-term survival. Notably, her study was also among the first to consider survivorship bias.

## References

Agresti, A. 1999. On logit confidence intervals for the odds ratio with small samples. Biometrics 55: 597-602. https://doi.org/10.1111/j.0006-341X.1999.00597.x.
——. 2013. Categorical Data Analysis. 3rd ed. Hoboken, NJ: Wiley.
Boice, J. D., Jr., and R. R. Monson. 1977. Breast cancer in women after repeated fluoroscopic examinations of the chest. Journal of the National Cancer Institute 59: 823-832. https://doi.org/10.1093/jnci/59.3.823.

Breslow, N. E. 1996. Statistics in epidemiology: The case-control study. Journal of the American Statistical Association 91: 14-28. https://doi.org/10.2307/2291379.
Breslow, N. E., and N. E. Day. 1980. Statistical Methods in Cancer Research: Vol. 1-The Analysis of Case-Control Studies. Lyon: IARC.

Brown, C. C. 1981. The validity of approximation methods for interval estimation of the odds ratio. American Journal of Epidemiology 113: 474-480. https://doi.org/10.1093/oxfordjournals.aje.a113115.
Clayton, D. G., and M. Hills. 1993. Statistical Models in Epidemiology. Oxford: Oxford University Press.
Cornfield, J. 1956. A statistical problem arising from retrospective studies. In Vol. 4 of Proceedings of the Third Berkeley Symposium, ed. J. Neyman, 135-148. Berkeley, CA: University of California Press.
Cummings, P. 2009. Methods for estimating adjusted risk ratios. Stata Journal 9: 175-196.
—_. 2019. Analysis of Incidence Rates. Boca Raton, FL: CRC Press.
Dohoo, I., W. Martin, and H. Stryhn. 2010. Veterinary Epidemiologic Research. 2nd ed. Charlottetown, Prince Edward Island: VER Inc.
__. 2012. Methods in Epidemiologic Research. Charlottetown, Prince Edward Island: VER Inc.
Doll, R., and A. B. Hill. 1966. Mortality of British doctors in relation to smoking: Observations on coronary thrombosis. Journal of the National Cancer Institute, Monographs 19: 205-268.

Dupont, W. D. 2009. Statistical Modeling for Biomedical Researchers: A Simple Introduction to the Analysis of Complex Data. 2nd ed. Cambridge: Cambridge University Press.
Fagerland, M. W. 2012. Exact and mid-p confidence intervals for the odds ratio. Stata Journal 12: 505-514.
Fleiss, J. L., B. Levin, and M. C. Paik. 2003. Statistical Methods for Rates and Proportions. 3rd ed. New York: Wiley.
Gart, J. J., and D. G. Thomas. 1982. The performance of three approximate confidence limit methods for the odds ratio. American Journal of Epidemiology 115: 453-470. https://doi.org/10.1093/oxfordjournals.aje.a113323.

Gini, R., and J. Pasquini. 2006. Automatic generation of documents. Stata Journal 6: 22-39.
Glass, R. I., A. M. Svennerholm, B. J. Stoll, M. R. Khan, K. M. Hossain, M. I. Huq, and J. Holmgren. 1983. Protection against cholera in breast-fed children by antibodies in breast milk. New England Journal of Medicine 308: 1389-1392. https://doi.org/10.1056/NEJM198306093082304.

Greenhouse, S. W., and J. B. Greenhouse. 1998. Cornfield, Jerome. In Vol. 1 of Encyclopedia of Biostatistics, ed. P. Armitage and T. Colton, 955-959. Chichester, UK: Wiley.

Greenland, S. 1986. Estimating variances of standardized estimators in case-control studies and sparse data. Journal of Chronic Diseases 39: 473-477. https://doi.org/10.1016/0021-9681(86)90114-1.

Greenland, S., ed. 1987. Evolution of Epidemiologic Ideas: Annotated Readings on Concepts and Methods. Newton Lower Falls, MA: Epidemiology Resources.
Greenland, S., and J. M. Robins. 1985. Estimation of a common effect parameter from sparse follow-up data. Biometrics 41: 55-68. https://doi.org/10.2307/2530643.
-_. 1988. Conceptual problems in the definition and interpretation of attributable fractions. American Journal of Epidemiology 128: 1185-1197. https://doi.org/10.1093/oxfordjournals.aje.a115073.
Haldane, J. B. S. 1957. Graphical methods in enzyme chemistry. Nature 179: 832. https://doi.org/10.1038/179832b0.
Hastorf, A. H., E. R. Hilgard, and R. R. Sears. 1988. Quinn McNemar (1900-1986). American Psychologist 43: 196-197. https://doi.org/10.1037/h0091955.

Hempel, S. 2006. The Medical Detective: John Snow, Cholera and the Mystery of the Broad Street Pump. London: Granta Books.
Hill, W. G. 1984. Barnet Woolf. Year Book, Royal Society of Edinburgh 1984 214-219.

Jewell, N. P. 2004. Statistics for Epidemiology. Boca Raton, FL: Chapman and Hall/CRC.
Jick, H., O. S. Miettinen, R. K. Neff, S. Shapiro, O. P. Heinonen, and D. Slone. 1973. Coffee and myocardial infarction. New England Journal of Medicine 289: 63-67. https://doi.org/10.1056/NEJM197307122890203.

Johnson, S. 2006. The Ghost Map: The Story of London's Most Terrifying Epidemic—and How It Changed Science, Cities, and the Modern World. London: Penguin Books.

Kahn, H. A., and C. T. Sempos. 1989. Statistical Methods in Epidemiology. New York: Oxford University Press.
Kleinbaum, D. G., L. L. Kupper, and H. Morgenstern. 1982. Epidemiologic Research: Principles and Quantitative Methods (Industrial Health and Safety). New York: Wiley.

Lash, T. L., T. J. VanderWeele, S. Haneuse, and K. J. Rothman. 2021. Modern Epidemiology. 4th ed. Philadelphia: Wolters Kluwer.

Linden, A. 2019. Assessing medication adherence using Stata. Stata Journal 19: 820-831.
MacMahon, B., S. Yen, D. Trichopoulos, K. Warren, and G. Nardi. 1981. Coffee and cancer of the pancreas. New England Journal of Medicine 304: 630-633. https://doi.org/10.1056/NEJM198103123041102.

Mantel, N., and W. Haenszel. 1959. Statistical aspects of the analysis of data from retrospective studies of disease. Journal of the National Cancer Institute 22: 719-748. Reprinted in Evolution of Epidemiologic Ideas: Annotated Readings on Concepts and Methods, ed. S. Greenland, pp. 112-141. Newton Lower Falls, MA: Epidemiology Resources.

Markel, H. 2013. Happy birthday, Dr Snow. Journal of the American Medical Association 309: 995-996. https://doi.org/10.1001/jama.2013.1304.

McNemar, Q. 1947. Note on the sampling error of the difference between correlated proportions or percentages. Psychometrika 12: 153-157. https://doi.org/10.1007/BF02295996.

Newman, S. C. 2001. Biostatistical Methods in Epidemiology. New York: Wiley.
Orsini, N., R. Bellocco, M. Bottai, A. Wolk, and S. Greenland. 2008. A tool for deterministic and probabilistic sensitivity analysis of epidemiologic studies. Stata Journal 8: 29-48.

Robins, J. M., N. E. Breslow, and S. Greenland. 1986. Estimators of the Mantel-Haenszel variance consistent in both sparse data and large-strata limiting models. Biometrics 42: 311-323. https://doi.org/10.2307/2531052.

Rothman, K. J. 1982. Spermicide use and Down's syndrome. American Journal of Public Health 72: 399-401. https://doi.org/10.2105/ajph.72.4.399.
—_. 1986. Modern Epidemiology. Boston: Little, Brown.
——. 2012. Epidemiology: An Introduction. 2nd ed. New York: Oxford University Press.
Rothman, K. J., and J. D. Boice, Jr. 1982. Epidemiologic Analysis with a Programmable Calculator. Brookline, MA: Epidemiology Resources.

Rothman, K. J., D. C. Fyler, A. Goldblatt, and M. B. Kreidberg. 1979. Exogenous hormones and other drug exposures of children with congenital heart disease. American Journal of Epidemiology 109: 433-439. https://doi.org/10.1093/oxfordjournals.aje.a112701.

Rothman, K. J., and R. R. Monson. 1973. Survival in trigeminal neuralgia. Journal of Chronic Diseases 26: 303-309. https://doi.org/10.1016/0021-9681(73)90033-7.

Royston, P., and A. G. Babiker. 2002. A menu-driven facility for complex sample size calculation in randomized controlled trials with a survival or a binary outcome. Stata Journal 2: 151-163.

Schlesselman, J. J. 1982. Case-Control Studies: Design, Conduct, Analysis. New York: Oxford University Press.
Schneider, D., and D. E. Lilienfeld. 2015. Lilienfeld's Foundations of Epidemiology. 4th ed. New York: Oxford University Press.

Selvin, S. 2011. Statistical Tools for Epidemiologic Research. New York: Oxford University Press.
Snow, J. 1855. On the Mode of Communication of Cholera. 2nd ed. London: Churchill.
Suárez, E. L., C. M. Pérez, R. Rivera, and M. N. Martínez. 2017. Applications of Regression Models in Epidemiology. Hoboken, NJ: Wiley.

Tarone, R. E. 1985. On heterogeneity tests based on efficient scores. Biometrika 72: 91-95.
https://doi.org/10.2307/2336337.

University Group Diabetes Program. 1970. A study of the effects of hypoglycemic agents on vascular complications in patients with adult-onset diabetes, II: Mortality results. Diabetes 19, supplement 2: 789-830.

Vach, W. 2013. Regression Models as a Tool in Medical Research. Boca Raton, FL: CRC Press.
Vinten-Johansen, P., H. Brody, N. Paneth, S. Rachman, and M. Rip. 2003. Cholera, Chloroform, and the Science of Medicine: A Life of John Snow. New York: Oxford University Press.

Walker, A. M. 1991. Observation and Inference: An Introduction to the Methods of Epidemiology. Newton Lower Falls, MA: Epidemiology Resources.
Wang, Z. 2007. Two postestimation commands for assessing confounding effects in epidemiological studies. Stata Journal 7: 183-196.

Woodward, M. 2014. Epidemiology: Study Design and Data Analysis. 3rd ed. Boca Raton, FL: Chapman and Hall/CRC.
Woolf, B. 1955. On estimating the relation between blood group disease. Annals of Human Genetics 19: 251-253. Reprinted in Evolution of Epidemiologic Ideas: Annotated Readings on Concepts and Methods, ed. S. Greenland, pp. 108-110. Newton Lower Falls, MA: Epidemiology Resources.

## Also see

[ST] stcox - Cox proportional hazards model
[R] bitest - Binomial probability test
$[\mathrm{R}]$ ci - Confidence intervals for means, proportions, and variances
$[\mathrm{R}]$ clogit - Conditional (fixed-effects) logistic regression
[R] dstdize - Direct and indirect standardization
[R] logistic - Logistic regression, reporting odds ratios
[R] poisson - Poisson regression
$[R]$ reri - Relative excess risk due to interaction
$[\mathrm{R}]$ symmetry - Symmetry and marginal homogeneity tests
[R] tabulate twoway - Two-way table of frequencies
[META] meta - Introduction to meta
[U] 19 Immediate commands

## Title

Error messages - Error messages and return codes

## Description Also see

## Description

Whenever Stata detects that something is wrong-that what you typed is uninterpretable, that you are trying to do something you should not be trying to do, or that you requested the impossible - Stata responds by typing a message describing the problem, together with a return code. For instance,

```
. lsit
command lsit is unrecognized
r(199);
. list myvar
variable myvar not found
r(111);
. test a=b
last estimates not found
r(301);
```

In each case, the message is probably sufficient to guide you to a solution. When we typed lsit, Stata responded with "unrecognized command". We meant to type list. When we typed list myvar, Stata responded with "variable myvar not found". There is no variable named myvar in our data. When we typed test $\mathrm{a}=\mathrm{b}$, Stata responded with "last estimates not found". test tests hypotheses about previously fit models, and we have not yet fit a model.

The numbers in parentheses in the $r(199), r(111)$, and $r(301)$ messages are called the return codes. To find out more about these messages, type search rc \#, where \# is the number returned in the parentheses.

## > Example 1

. search rc 301
[P] error messages . . . . . . . . . . . . . . . . . . . . Return code 301
last estimates not found;
You typed an estimation command, such as regress, without arguments or attempted to perform a test or typed predict, but there were no previous estimation results.

Programmers should see [P] error for details on programming error messages.

## Also see

[R] search - Search Stata documentation and other resources

## Title

esize - Effect size based on mean comparison

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

esize calculates effect sizes for comparing the difference between the means of a continuous variable for two groups. In the first form, esize calculates effect sizes for the difference between the mean of varname for two groups defined by groupvar. In the second form, esize calculates effect sizes for the difference between varname $_{1}$ and varname ${ }_{2}$, assuming unpaired data.
esizei is the immediate form of esize; see [U] 19 Immediate commands. In the first form, esizei calculates the effect size for comparing the difference between the means of two groups. In the second form, esizei calculates the effect size for an $F$ test after an ANOVA.

## Quick start

Cohen's $d$ and Hedges's $g$ comparing the difference in means of v for two independent groups in catvar

```
esize twosample v, by(catvar)
```

Same as above, but with group data stored in v1 and v2
esize unpaired $\mathrm{v} 1==\mathrm{v} 2$
Same as above, but use $90 \%$ confidence level
esize unpaired v1==v2, level(90)
Cohen's $d$ and Hedges's $g$ for means of v for groups in catvar1 calculated over each level of catvar2
by catvar2: esize twosample v, by(catvar1)

## Menu

esize
Statistics $>$ Summaries, tables, and tests $>$ Classical tests of hypotheses $>$ Effect size based on mean comparison

## esizei

Statistics $>$ Summaries, tables, and tests $>$ Classical tests of hypotheses $>$ Effect-size calculator

## Syntax

Effect sizes for two independent samples using groups

```
esize twosample varname [if] [in], by(groupvar) [options]
```

Effect sizes for two independent samples using variables

```
esize unpaired varname}\mp@subsup{1}{1}{\prime== varname 2 [if] [in], [options]
```

Immediate form of effect sizes for two independent samples

$$
\text { esizei } \#_{\text {obs1 }} \#_{\text {mean } 1} \#_{\text {sd1 }} \#_{\text {obs2 }} \#_{\text {mean2 }} \#_{\text {sd2 }}[\text {, options }]
$$

Immediate form of effect sizes for $F$ tests after an ANOVA

```
esizei # df1 # #ff2 #F [, level(#)]
```

options
Description
Main
cohensd report Cohen's $d$ (1988)
hedgesg report Hedges's $g$ (1981)
glassdelta report Glass's $\Delta$ (Smith and Glass 1977) using each group's standard deviation
pbcorr
report the point-biserial correlation coefficient (Pearson 1909)
all report all estimates of effect size
unequal use unequal variances
welch use Welch's (1947) approximation
level(\#) set confidence level; default is level(95)
by is allowed with esize, and collect is allowed with esize and esizei; see [U] 11.1.10 Prefix commands.

## Options

$\qquad$ Main
by (groupvar) specifies the groupvar that defines the two groups that esize will use to estimate the effect sizes. Do not confuse the by () option with the by prefix; you can specify both.
cohensd specifies that Cohen's $d$ (1988) be reported.
hedgesg specifies that Hedges's $g$ (1981) be reported.
glassdelta specifies that Glass's $\Delta$ (Smith and Glass 1977) be reported.
pbcorr specifies that the point-biserial correlation coefficient (Pearson 1909) be reported.
all specifies that all estimates of effect size be reported. The default is Cohen's $d$ and Hedges's $g$. unequal specifies that the data not be assumed to have equal variances.
welch specifies that the approximate degrees of freedom for the test be obtained from Welch's formula (1947) rather than from Satterthwaite's approximation formula (1946), which is the default when unequal is specified. Specifying welch implies unequal.
level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals.

## Remarks and examples

Remarks are presented under the following headings:

Introduction<br>Estimating effect sizes<br>Immediate form<br>Video example

## Introduction

Whereas $p$-values are used to assess the statistical significance of a result, measures of effect size are used to assess the practical significance of a result. Effect sizes can be broadly categorized as "measures of group differences" (the $d$ family) and "measures of association" (the $r$ family); see Ellis (2010, table 1.1). The $d$ family includes estimators such as Cohen's $d$, Hedges's $g$, and Glass's $\Delta$. The $r$ family includes estimators such as the point-biserial correlation coefficient, $\eta^{2}, \varepsilon^{2}$, and $\omega^{2}$ (also see estat esize in [R] regress postestimation). For an introduction to the concepts and calculation of effect sizes, see Kline (2013) and Thompson (2006). For a more detailed discussion, see Kirk (1996), Ellis (2010), Cumming (2012), Grissom and Kim (2012), and Kelley and Preacher (2012).

Note that there is much variation in the definitions of measures of effect size (Kline 2013). As Ellis (2010, 27) cautions, "However, beware the inconsistent terminology. What is labeled here as $g$ was labeled by Hedges and Olkin as $d$ and vice versa. For these authors writing in the early 1980s, $g$ was the mainstream effect-size index developed by Cohen and refined by Glass (hence $g$ for Glass). However, since then $g$ has become synonymous with Hedges's equation (not Glass's) and the reason it is called Hedges's $g$ and not Hedges's $h$ is because it was originally named after Glass-even though it was developed by Larry Hedges. Confused?"

To avoid confusion, esize and esizei closely follow the notation of Hedges (1981), Smithson (2001), Kline (2013), and Ellis (2010).

## Estimating effect sizes

> Example 1: Effect size for two independent samples using by()
Suppose we are interested in question 1 from the fictitious depression.dta: "My statistical software makes me feel sad". We might have conducted a $t$ test to test the null hypothesis that there is no difference in response by sex. We could then compute various measures of effect size to describe the magnitude of the effect of sex.

| (Fictitious depression inventory dat esize twosample qu1, by (sex) all Effect size based on mean comparison |  |  |  |
| :---: | :---: | :---: | :---: |
| Obs per group: |  |  |  |
|  |  | Female $=$ | 712 |
|  |  | Male | 288 |
| Effect size | Estimate | [95\% conf | interval] |
| Cohen's d | -. 0512417 | -. 1881184 | . 0856607 |
| Hedges's $g$ | -. 0512032 | -. 187977 | . 0855963 |
| Glass's Delta 1 | -. 0517793 | -. 1886587 | . 0851364 |
| Glass's Delta 2 | -. 0499786 | -. 1868673 | . 086997 |
| Point-biserial r | -. 0232208 | -. 0849629 | . 0387995 |

Cohen's $d$, Hedges's $g$, and both estimates of Glass's $\Delta$ indicate that the score for females is 0.05 standard deviations lower than the score for males. The point-biserial correlation coefficient indicates that there is a small, negative correlation between the scores for females and males.

## - Technical note

Glass's $\Delta$ has traditionally been estimated for experimental studies using the control group standard deviation rather than the pooled standard deviation. Kline (2013) notes that the choice of group becomes arbitrary for data arising from observational studies and recommends the reporting of Glass's $\Delta$ using each group standard deviation.

Example 2: Effect size for two independent samples by a third variable
If we are interested in the same effect sizes from example 1 stratified by race, we could use the by prefix with the sort option to accomplish this task.

```
. by race, sort: esize twosample qu1, by(sex)
```

```
-> race = Hispanic
Effect size based on mean comparison
```

        Obs per group:
        Female \(=\quad 88\)
            Male \(=\quad 45\)
    | Effect size | Estimate | $[95 \%$ conf. interval] |  |
| ---: | ---: | :---: | ---: |
| Cohen's d | -.1042883 | -.463503 | .2553235 |
| Hedges's g | -.1036899 | -.4608434 | .2538584 |

```
-> race = Black
Effect size based on mean comparison
```

    Obs per group:
        Female \(=\quad 259\)
        Male \(=\quad 95\)
    | Effect size | Estimate | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: |
| Cohen's d | -.1720681 | -.4073814 | .063489 |
| Hedges's g | -.1717011 | -.4065127 | .0633536 |

-> race = White
Effect size based on mean comparison
Obs per group:
Female $=365$
Male = 148

| Effect size | Estimate | $[95 \%$ conf. interval] |  |
| ---: | :---: | :---: | ---: |
| Cohen's d | .0479511 | -.1430932 | .2389486 |
| Hedges's g | .0478807 | -.1428831 | .2385977 |

## > Example 3: Bootstrap confidence intervals for effect sizes

Simulation studies have shown that bootstrap confidence intervals may be preferable to confidence intervals based on the noncentral $t$ distribution when the variable of interest does not have a normal distribution (Kelley 2005; Algina, Keselman, and Penfield 2006). Bootstrap confidence intervals can be easily estimated for effect sizes using the bootstrap prefix.

| (Fictitious depression inventory data based on the Beck Depression Inventory) <br> . set seed 12345 <br> . bootstrap $r(d) r(g)$, reps(1000) nodots nowarn: esize twosample qu1, by (sex) <br> Bootstrap results <br> Number of obs $=1,000$ <br> Replications $=1,000$ <br> Command: esize twosample qu1, by(sex) _bs_1: r(d) _bs_2: r(g) |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | Observed coefficient | Bootstrap std. err. | z | $P>\|z\|$ | $\begin{array}{r} \text { Norm } \\ {[95 \% \text { con }} \end{array}$ | based <br> interval] |
| _bs_1 | -. 0512417 | . 0742692 | -0.69 | 0.490 | -. 1968066 | . 0943233 |
| _bs_2 | -. 0512032 | . 0742134 | -0.69 | 0.490 | -. 1966587 | . 0942523 |

## > Example 4: Effect sizes for two independent samples using variables

Sometimes, the data of interest are stored in two separate variables. We can calculate effect sizes for the two groups by using the unpaired version of esize.

| - use https://www.stata-press.com/data/r18/fuel |  |  |  |
| :--- | :---: | :---: | :---: |
| . esize unpaired mpg1==mpg2 |  |  |  |
| Effect size based on mean comparison |  |  |  |
| Number of obs $=$ |  |  |  |
| 24    <br> Effect size Estimate [95\% conf. interval]  <br> Cohen's d -.5829654 -1.394934 .2416105 <br> Hedges's g -.5628243 -1.34674 .2332631 |  |  |  |

## Immediate form

## Example 5: Immediate form for effect sizes for two means

Often we do not have access to raw data, but we are given summary statistics in a report or manuscript. To calculate the effect sizes from summary statistics, we can use the immediate command esizei. For example, Kline (2013) in table 4.2 shows summary statistics for a hypothetical sample where mean $_{1}=13, \operatorname{sd}_{1}=2.74$, mean $_{2}=11$, and $\mathrm{sd}_{2}=2.24$; there are 30 people in each group. We can estimate the effect sizes from these summary data using esizei:

|  |  | er group: <br> Group 1 <br> Group 2 | $\begin{aligned} & 30 \\ & 30 \end{aligned}$ |
| :---: | :---: | :---: | :---: |
| Effect size | Estimate | [95\% con | interval] |
| Cohen's d | . 7991948 | . 2695509 | 1.322465 |
| Hedges's g | . 7888081 | . 2660477 | 1.305277 |

## Example 6: Immediate form for effect sizes for F tests after an ANOVA

esizei can also be used to compute $\eta^{2}, \varepsilon^{2}$, and $\omega^{2}$ for $F$ tests after an ANOVA. The following example from Smithson $(2001,623)$ illustrates the use of esizei for $\mathrm{df}_{\text {num }}=4, \mathrm{df}_{\mathrm{den}}=50$, and $F=4.2317$ :

```
. esizei 4 50 4.2317, level(90)
```

Effect sizes for linear models

| Effect size | Estimate | $[90 \%$ conf. interval] |  |
| ---: | :---: | :---: | :---: |
| Eta-squared | .2529151 | .0521585 | .3603621 |
| Epsilon-squared | .1931483 |  |  |
| Omega-squared | .1903049 |  |  |

## Video example

Tour of effect sizes

## Stored results

esize and esizei for comparing two means store the following in $r()$ :
Scalars

```
r(d)
    r(lb_d)
    r(ub_d)
    r(g)
    r(lb_g)
    r(ub_g)
    r(delta1)
    r(lb_delta1)
    r(ub_delta1)
    r(delta2)
    r(lb_delta2)
    r(ub_delta2)
    r(r_pb)
    r(lb_r_pb)
    r(ub_r_pb)
    r(N_1) sample size }\mp@subsup{n}{1}{
    r(N_2) sample size }\mp@subsup{n}{2}{
    r(df_t) degrees of freedom
    r(level)
    Cohen's d
    lower confidence bound for Cohen's d
        upper confidence bound for Cohen's d
        Hedges's g
        lower confidence bound for Hedges's g
        upper confidence bound for Hedges's }
        Glass's }\Delta\mathrm{ for group 1
        lower confidence bound for Glass's }\Delta\mathrm{ for group 1
        upper confidence bound for Glass's }\Delta\mathrm{ for group 1
        Glass's }\Delta\mathrm{ for group 2
        lower confidence bound for Glass's }\Delta\mathrm{ for group 2
        upper confidence bound for Glass's }\Delta\mathrm{ for group 2
        point-biserial correlation coefficient
        lower confidence bound for the point-biserial correlation coefficient
        upper confidence bound for the point-biserial correlation coefficient
    confidence level
```

esizei for $F$ tests after ANOVA stores the following in r() :
Scalars

```
r(eta2) 渞
    r(lb_eta2) lower confidence bound for }\mp@subsup{\eta}{}{2
    r(ub_eta2) upper confidence bound for }\mp@subsup{\eta}{}{2
    r(epsilon2) & <
    r(omega2) }\mp@subsup{\omega}{}{2
    r(level) confidence level
```


## Methods and formulas

For the $d$ family, the effect-size parameter of interest is the scaled difference between the means given by

$$
\delta=\frac{\left(\mu_{1}-\mu_{2}\right)}{\sigma}
$$

One of the most popular estimators of effect size is Cohen's $d$, given by

$$
\text { Cohen's } d=\frac{\left(\bar{x}_{1}-\bar{x}_{2}\right)}{s^{*}}
$$

where

$$
s^{*}=\sqrt{\frac{\left(n_{1}-1\right) s_{1}^{2}+\left(n_{2}-1\right) s_{2}^{2}}{n_{1}+n_{2}-2}}
$$

Hedges (1981) showed that Cohen's $d$ is biased and proposed the unbiased estimator

$$
\text { Hedges's } g=\text { Cohen's } d \times c(m)
$$

where $m=n_{1}+n_{2}-2$ and

$$
c(m)=\frac{\Gamma\left(\frac{m}{2}\right)}{\sqrt{\frac{m}{2}} \Gamma\left(\frac{m-1}{2}\right)}
$$

Glass (Smith and Glass 1977) proposed an estimator for $\delta$ in the context of designed experiments,

$$
\text { Glass's } \Delta=\frac{\left(\bar{x}_{\text {treated }}-\bar{x}_{\text {control }}\right)}{s_{\text {control }}}
$$

where $s_{\text {control }}$ is the standard deviation for the control group.
As noted above, esize and esizei report two estimates of Glass's $\Delta$ : one using the standard deviation for group 1 and the other using the standard deviation for group 2 :

$$
\text { Glass's } \Delta_{1}=\frac{\left(\bar{x}_{1}-\bar{x}_{2}\right)}{s_{1}}
$$

and

$$
\text { Glass's } \Delta_{2}=\frac{\left(\bar{x}_{1}-\bar{x}_{2}\right)}{s_{2}}
$$

For the $r$ family, the effect-size parameter of interest is the ratio of the variance attributable to an effect and the total variance:

$$
\eta^{2}=\frac{\sigma_{\text {effect }}^{2}}{\sigma_{\text {total }}^{2}}
$$

A popular estimator of $\eta$ when there are two groups is the point-biserial correlation coefficient,

$$
r_{\mathrm{PB}}=\frac{t}{\sqrt{t^{2}+\mathrm{df}}}
$$

where $t$ is the $t$ statistic for the difference between the means of the two groups, and $d f$ is the corresponding degrees of freedom. Satterthwaite's or Welch's adjustment (see [R] ttest for details) to the degrees of freedom can be used to calculate $r_{\mathrm{PB}}$ by specifying the unequal or welch option, respectively.

When more than two means are being compared, as in the case of an ANOVA with $p$ groups, a popular estimator of effect size is the correlation ratio denoted $\eta^{2}$ (Fisher 1925; Kerlinger and Lee 2000). $\eta^{2}$ can be computed directly as the ratio of the $\mathrm{SS}_{\text {effect }}$ and the $\mathrm{SS}_{\text {total }}$ or as a function of the $F$ statistic with numerator degrees of freedom equal to $\mathrm{df}_{\text {num }}$ and denominator degrees of freedom equal to $\mathrm{df}_{\text {den }}$.

$$
\widehat{\eta}^{2}=\frac{F}{F+\mathrm{df}_{\mathrm{den}} / \mathrm{df}_{\mathrm{num}}}
$$

Like its equivalent estimator $R^{2}, \eta^{2}$ has an upward bias. Less biased estimators of effect size are $\varepsilon^{2}$ and $\omega^{2}$ (Grissom and Kim 2012).

$$
\widehat{\varepsilon}^{2}=\frac{F-1}{F+\mathrm{df}_{\mathrm{den}} / \mathrm{df}_{\text {num }}}=\widehat{\eta}^{2}-\frac{\mathrm{df}_{\text {num }}}{\mathrm{df}_{\mathrm{den}}}\left(1-\widehat{\eta}^{2}\right)
$$

$$
\widehat{\omega}^{2}=\frac{F-1}{F+\left(\mathrm{df}_{\mathrm{den}}+1\right) / \mathrm{df}_{\mathrm{num}}}
$$

To calculate $\widehat{\eta}^{2}, \widehat{\varepsilon}^{2}$, and $\widehat{\omega}^{2}$ directly after anova or regress, see estat esize in $[\mathrm{R}]$ regress postestimation.

Cohen's $d$, Hedges's $g$, and Glass's $\Delta$ have been shown to have a noncentral $t$ distribution (Hedges 1981) with noncentrality parameter equal to

$$
\lambda=\delta \sqrt{\frac{n_{1} n_{2}}{n_{1}+n_{2}}}
$$

Confidence intervals are calculated by finding the noncentrality parameters $\lambda_{\text {lower }}$ and $\lambda_{\text {upper }}$ that correspond to

$$
\operatorname{Pr}\left(\mathrm{df}, \delta, \lambda_{\text {lower }}\right)=1-\frac{\alpha}{2}
$$

and

$$
\operatorname{Pr}\left(\mathrm{df}, \delta, \lambda_{\text {upper }}\right)=\frac{\alpha}{2}
$$

using the function npnt ( $d f, t, p$ ). The noncentrality parameters are then transformed back to the effect-size scale:

$$
\delta_{\text {lower }}=\lambda_{\text {lower }} \sqrt{\frac{n_{1}+n_{2}}{n_{1} n_{2}}}
$$

and

$$
\delta_{\text {upper }}=\lambda_{\text {upper }} \sqrt{\frac{n_{1}+n_{2}}{n_{1} n_{2}}}
$$

(see Venables [1975]; Steiger and Fouladi [1997]; Cumming and Finch [2001]; Smithson [2001]).
Confidence intervals for the point-biserial correlation coefficient are calculated similarly and transformed back to the effect-size scale as

$$
r_{\text {lower }}=\frac{\lambda_{\text {lower }}}{\sqrt{\lambda_{\text {lower }}^{2}+\mathrm{df}}}
$$

and

$$
r_{\text {upper }}=\frac{\lambda_{\text {upper }}}{\sqrt{\lambda_{\text {upper }}^{2}+\mathrm{df}}}
$$

Following Smithson's (2001) notation, the $F$ statistic is written as

$$
F_{\mathrm{df}_{\mathrm{num}}, \mathrm{df}_{\mathrm{den}}}=f^{2}\left(\mathrm{df}_{\mathrm{num}} / \mathrm{df}_{\mathrm{den}}\right)
$$

This equation has a noncentral $F$ distribution with noncentrality parameter:

$$
\lambda=f^{2}\left(\mathrm{df}_{\mathrm{num}}+\mathrm{df}_{\mathrm{den}}+1\right)
$$

where $f^{2}=\eta^{2} /\left(1-\eta^{2}\right)$.

Confidence intervals for $\widehat{\eta}^{2}$ are calculated by finding the noncentrality parameters $\lambda_{\text {lower }}$ and $\lambda_{\text {upper }}$ for a noncentral $F$ distribution that correspond to

$$
\operatorname{Pr}\left(\mathrm{df}_{\text {num }}, \mathrm{df}_{\text {den }}, F, \lambda_{\text {lower }}\right)=1-\frac{\alpha}{2}
$$

and

$$
\operatorname{Pr}\left(\mathrm{df}_{\text {num }}, \mathrm{df}_{\text {den }}, F, \lambda_{\text {upper }}\right)=\frac{\alpha}{2}
$$

using the function $\operatorname{npnF}\left(d f_{1}, d f_{2}, f, p\right)$. The noncentrality parameters are transformed back to the $\widehat{\eta}^{2}$ scale as

$$
\widehat{\eta}_{\text {lower }}^{2}=\frac{\lambda_{\text {lower }}}{\lambda_{\text {lower }}+\mathrm{df}_{\text {num }}+\mathrm{df}_{\text {den }}+1}
$$

and

$$
\widehat{\eta}_{\text {upper }}^{2}=\frac{\lambda_{\text {upper }}}{\lambda_{\text {upper }}+\mathrm{df}_{\text {num }}+\mathrm{df}_{\text {den }}+1}
$$

While confidence intervals for $\widehat{\varepsilon}^{2}$ can be constructed using the same transformation that links it with $\widehat{\eta}^{2}$, there are several arguments for not using them in practice. See Smithson $(2003,54)$ for further details.

Fred Nichols Kerlinger (1910-1991) was born in New York City. He studied music at New York University and graduated magna cum laude with a degree in education and philosophy. After graduation, he joined the US Army and served as a counterintelligence officer in Japan in 1946. Kerlinger earned an MA and a PhD in educational psychology from the University of Michigan and held faculty appointments at several universities, including New York University. He was president of the American Educational Research Association and is best known for his popular and influential book Foundations of Behavioral Research (1964), which introduced Fisher's (1925) $\eta^{2}$ statistic to behavioral researchers.
William Lee Hays (1926-1995) was born in Clarksville, Texas. He studied mathematics and psychology at Paris Junior College in Paris, Texas, and at East Texas State College. He earned BS and MS degrees from North Texas State University. Upon completion of his PhD in psychology at the University of Michigan, he joined the faculty, where he eventually became associate vice president for academic affairs. In 1977, Hays accepted an appointment as vice president for academic affairs at the University of Texas at Austin, where he remained until his death in 1995. Hays is best known for his book Statistics for Psychologists (1963), which introduced the $\omega^{2}$ statistic (and is actually denoted here by $\varepsilon^{2}$ ).

## References

Algina, J., H. J. Keselman, and R. D. Penfield. 2006. Confidence interval coverage for Cohen's effect size statistic. Educational and Psychological Measurement 66: 945-960. https://doi.org/10.1177/0013164406288161.
Baldwin, S. 2019. Psychological Statistics and Psychometrics Using Stata. College Station, TX: Stata Press.
Cohen, J. 1988. Statistical Power Analysis for the Behavioral Sciences. 2nd ed. Hillsdale, NJ: Erlbaum.
Cumming, G. 2012. Understanding the New Statistics: Effect Sizes, Confidence Intervals, and Meta-Analysis. New York: Routledge.

Cumming, G., and S. Finch. 2001. A primer on the understanding, use, and calculation of confidence intervals that are based on central and noncentral distributions. Educational and Psychological Measurement 61: 532-574. https://doi.org/10.1177/0013164401614002.

Ellis, P. D. 2010. The Essential Guide to Effect Sizes: Statistical Power, Meta-Analysis, and the Interpretation of Research Results. Cambridge: Cambridge University Press.

Fisher, R. A. 1925. Statistical Methods for Research Workers. Edinburgh: Oliver and Boyd.
Grissom, R. J., and J. J. Kim. 2012. Effect Sizes for Research: Univariate and Multivariate Applications. 2nd ed. New York: Routledge.

Hays, W. L. 1963. Statistics for Psychologists. New York: Holt, Rinehart and Winston.
Hedges, L. V. 1981. Distribution theory for Glass's estimator of effect size and related estimators. Journal of Educational Statistics 6: 107-128. https://doi.org/10.2307/1164588.
Huber, C. 2013. Measures of effect size in Stata 13. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2013/09/05/measures-of-effect-size-in-stata-13/.

Kelley, K. 2005. The effects of nonnormal distributions on confidence intervals around the standardized mean difference: Bootstrap and parametric confidence intervals. Educational and Psychological Measurement 65: 51-69. https://doi.org/10.1177/0013164404264850.

Kelley, K., and K. J. Preacher. 2012. On effect size. Psychological Methods 17: 137-152.
https://doi.org/10.1037/a0028086.
Kerlinger, F. N. 1964. Foundations of Behavioral Research. New York: Holt, Rinehart and Winston.
Kerlinger, F. N., and H. B. Lee. 2000. Foundations of Behavioral Research. 4th ed. Belmont, CA: Wadsworth.
Kirk, R. E. 1996. Practical significance: A concept whose time has come. Educational and Psychological Measurement 56: 746-759. https://doi.org/10.1177/0013164496056005002.
Kline, R. B. 2013. Beyond Significance Testing: Statistics Reform in the Behavioral Sciences. 2nd ed. Washington, DC: American Psychological Association.

Miller, D. J., J. T. Nguyen, and M. Bottai. 2020. emagnification: A tool for estimating effect-size magnification and performing design calculations in epidemiological studies. Stata Journal 20: 548-564.
Pearson, K. 1909. On a new method of determining correlation between a measured character A, and a character B, of which only the percentage of cases wherein B exceeds (or falls short of) a given intensity is recorded for each grade of A. Biometrika 7: 96-105. https://doi.org/10.2307/2345365.

Satterthwaite, F. E. 1946. An approximate distribution of estimates of variance components. Biometrics Bulletin 2: 110-114. https://doi.org/10.2307/3002019.
Shaw, B. P. 2022. Effect sizes for contrasts of estimated marginal effects. Stata Journal 22: 134-157.
Smith, M. L., and G. V. Glass. 1977. Meta-analysis of psychotherapy outcome studies. American Psychologist 32: 752-760. https://doi.org/10.1037/0003-066X.32.9.752.
Smithson, M. 2001. Correct confidence intervals for various regression effect sizes and parameters: The importance of noncentral distributions in computing intervals. Educational and Psychological Measurement 61: 605-632. https://doi.org/10.1177/00131640121971392.
_—. 2003. Confidence Intervals. Thousand Oaks, CA: Sage.
Steiger, J. H., and R. T. Fouladi. 1997. Noncentrality interval estimation and the evaluation of statistical models. In What If There Were No Significance Tests?, ed. L. L. Harlow, S. A. Mulaik, and J. H. Steiger, 221-257. Mahwah, NJ: Erlbaum.

Thompson, B. 2006. Foundations of Behavioral Statistics: An Insight-Based Approach. New York: Guilford Press.
Venables, W. 1975. Calculation of confidence intervals for noncentrality parameters. Journal of the Royal Statistical Society, Series B 37: 406-412. https://doi.org/10.1111/j.2517-6161.1975.tb01554.x.

Welch, B. L. 1947. The generalization of 'student's' problem when several different population variances are involved. Biometrika 34: 28-35. https://doi.org/10.2307/2332510.

## Also see

[R] bitest - Binomial probability test
$[\mathrm{R}]$ ci - Confidence intervals for means, proportions, and variances
[R] mean - Estimate means
[R] oneway - One-way analysis of variance
[R] prtest - Tests of proportions
[R] sdtest - Variance-comparison tests
$[\mathrm{R}]$ ttest $-t$ tests (mean-comparison tests)

## Title

estat - Postestimation statistics

Description Syntax

## Description

estat displays scalar- and matrix-valued statistics after estimation; it complements predict, which calculates variables after estimation. Exactly what statistics estat can calculate depends on the previous estimation command.

Three sets of statistics are so commonly used that they are available after all estimation commands that store the model log likelihood. estat ic displays Akaike's and Schwarz's Bayesian information criteria. estat summarize summarizes the variables used by the command and automatically restricts the sample to e (sample); it also summarizes the weight variable and cluster structure, if specified. estat vce displays the covariance or correlation matrix of the parameter estimates of the previous model.

## Syntax

Command Reference

Display information criteria

```
estat ic [,n(#)]
[R] estat ic
```

Summarize estimation sample
estat summarize [eqlist] [, estat_summ_options] [R] estat summarize

Display covariance matrix estimates
estat vce [, estat_vce_options]
[R] estat vce

Command-specific
estat subcommand $_{1}$ [, options ${ }_{1}$ ]

## Title

estat classification - Classification statistics and table

| Description | Quick start | Menu for estat | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

estat classification reports various summary statistics, including the classification table.
estat classification requires that the current estimation results be from logistic, logit, probit, or ivprobit; see [R] logistic, [R] logit, [R] probit, or [R] ivprobit.

## Quick start

Display classification table and related statistics for current estimation results estat classification

Change probability threshold for assignment to positive outcome to $75 \%$
estat classification, cutoff(.75)
Classification for observations with catvar $=2$
estat classification if catvar==2

## Menu for estat

Statistics $>$ Postestimation

## Syntax

estat classification [if] [in] [weight] [, options]
options
Description
Main
all display summary statistics for all observations in the data
cutoff(\#)
positive outcome threshold; default is cutoff (0.5)
estat classification is not appropriate with svy estimation results.
collect is allowed; see [U] 11.1.10 Prefix commands.
fweights are allowed; see [U] 11.1.6 weight.

## Options

$\qquad$ Main
all requests that the statistic be computed for all observations in the data, ignoring any if or in restrictions specified by the estimation command.
cutoff(\#) specifies the value for determining whether an observation has a predicted positive outcome. An observation is classified as positive if its predicted probability is $\geq \#$. The default is 0.5 .

## Remarks and examples

estat classification presents the classification statistics and classification table after logistic, logit, probit, or ivprobit.

Statistics are produced either for the estimation sample (the default) or for any set of observations. When weights, if, or in is used with the estimation command, it is not necessary to repeat the qualifier when you want statistics computed for the estimation sample. Specify if, in, or the all option only when you want statistics computed for a set of observations other than the estimation sample. Specify weights only when you want to use a different set of weights.

## - Example 1

We illustrate estat classification after logistic; see [R] logistic.


The overall rate of correct classification is estimated to be 73.54 , with $90.77 \%$ of the normal weight group correctly classified (specificity) and only $35.59 \%$ of the low weight group correctly classified (sensitivity). Classification is sensitive to the relative sizes of each component group, and always favors classification into the larger group. This phenomenon is evident here.

By default, estat classification uses a cutoff of 0.5 , although you can vary this with the cutoff() option. You can use the lsens command to review the potential cutoffs; see [R] Isens.

## Stored results

estat classification stores the following in $r()$ :
Scalars
$r$ (P_corr) percent correctly classified
$r\left(P \_p 1\right)$ sensitivity
$r\left(P \_n 0\right) \quad$ specificity
$r\left(P_{-} 0\right)$ false-positive rate given true negative
$r\left(P \_n 1\right) \quad$ false-negative rate given true positive
$r\left(P \_1 p\right) \quad$ positive predictive value
$r\left(P \_0 n\right) \quad$ negative predictive value
$r$ (P_Op) false-positive rate given classified positive
$r\left(P \_1 n\right) \quad$ false-negative rate given classified negative
Matrices
$r$ (ctable) classification table

## Methods and formulas

Let $j$ index observations. Define $c$ as the cutoff() specified by the user or, if not specified, as 0.5 . Let $p_{j}$ be the predicted probability of a positive outcome and $y_{j}$ be the actual outcome, which we will treat as 0 or 1 , although Stata treats it as 0 and non- 0 , excluding missing observations.

A prediction is classified as positive if $p_{j} \geq c$ and otherwise is classified as negative. The classification is correct if it is positive and $y_{j}=1$ or if it is negative and $y_{j}=0$.

Sensitivity is the fraction of $y_{j}=1$ observations that are correctly classified. Specificity is the percentage of $y_{j}=0$ observations that are correctly classified.

## References

Hosmer, D. W., Jr., S. A. Lemeshow, and R. X. Sturdivant. 2013. Applied Logistic Regression. 3rd ed. Hoboken, NJ: Wiley.
Kohler, U., and F. Kreuter. 2012. Data Analysis Using Stata. 3rd ed. College Station, TX: Stata Press.


#### Abstract

Also see [R] logistic - Logistic regression, reporting odds ratios [R] logit - Logistic regression, reporting coefficients [R] probit - Probit regression [R] ivprobit - Probit model with continuous endogenous covariates $[\mathrm{R}]$ Iroc - Compute area under ROC curve and graph the curve [R] Isens - Graph sensitivity and specificity versus probability cutoff [R] estat gof - Pearson or Hosmer-Lemeshow goodness-of-fit test $[R]$ roc - Receiver operating characteristic (ROC) analysis [U] 20 Estimation and postestimation commands


## Title

estat gof - Pearson or Hosmer-Lemeshow goodness-of-fit test

| Description | Quick start | Menu for estat | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

estat gof reports the Pearson goodness-of-fit test or the Hosmer-Lemeshow goodness-of-fit test.
estat gof requires that the current estimation results be from logistic, logit, or probit; see $[R]$ logistic, $[R]$ logit, or $[R]$ probit. For estat gof after poisson, see $[R]$ poisson postestimation. For estat gof after sem, see [SEM] estat gof.

## Quick start

Pearson goodness-of-fit test for current estimation results
estat gof
Same as above, but apply to all observations in dataset instead of just those in e(sample)
estat gof, all
Hosmer-Lemeshow goodness-of-fit test
estat gof, group(10)
Same as above, and display table of groups used for the test
estat gof, group(10) table

## Menu for estat

Statistics $>$ Postestimation

## Syntax

```
estat gof [if] [in] [weight] [, options]
```

options
Main
group(\#)
all
outsample
table

Description
perform Hosmer-Lemeshow goodness-of-fit test using \# quantiles execute test for all observations in the data
adjust degrees of freedom for samples outside estimation sample display table of groups used for test
estat gof is not appropriate with svy estimation results.
collect is allowed; see [U] 11.1.10 Prefix commands.
fweights are allowed; see [U] 11.1.6 weight.

## Options

## Main

group(\#) specifies the number of quantiles to be used to group the data for the Hosmer-Lemeshow goodness-of-fit test. group (10) is typically specified. If this option is not given, the Pearson goodness-of-fit test is computed using the covariate patterns in the data as groups.
all requests that the statistic be computed for all observations in the data, ignoring any if or in restrictions specified by the estimation command.
outsample adjusts the degrees of freedom for the Pearson and Hosmer-Lemeshow goodness-of-fit tests for samples outside the estimation sample. See Samples other than the estimation sample later in this entry.
table displays a table of the groups used for the Hosmer-Lemeshow or Pearson goodness-of-fit test with predicted probabilities, observed and expected counts for both outcomes, and totals for each group.

## Remarks and examples

Remarks are presented under the following headings:
Introduction
Samples other than the estimation sample

## Introduction

estat gof computes goodness-of-fit tests: either the Pearson $\chi^{2}$ test or the Hosmer-Lemeshow test.

By default, estat gof computes statistics for the estimation sample by using the last model fit by logistic, logit, or probit. However, samples other than the estimation sample can be specified; see Samples other than the estimation sample later in this entry.

## > Example 1

estat gof, typed without options, presents the Pearson $\chi^{2}$ goodness-of-fit test for the fitted model. The Pearson $\chi^{2}$ goodness-of-fit test is a test of the observed against expected number of responses using cells defined by the covariate patterns; see predict with the number option in $[\mathrm{R}]$ logistic postestimation for the definition of covariate patterns.

```
. use https://www.stata-press.com/data/r18/lbw
(Hosmer & Lemeshow data)
. logistic low age lwt i.race smoke ptl ht ui
    (output omitted)
. estat gof
Goodness-of-fit test after logistic model
Variable: low
    Number of observations = 189
Number of covariate patterns = 182
    Pearson chi2(173) = 179.24
                        Prob > chi2 = 0.3567
```

Our model fits reasonably well. However, the number of covariate patterns is close to the number of observations, making the applicability of the Pearson $\chi^{2}$ test questionable but not necessarily inappropriate. Hosmer, Lemeshow, and Sturdivant (2013, 157-160) suggest regrouping the data by ordering on the predicted probabilities and then forming, say, 10 nearly equal-sized groups. estat gof with the group() option does this:

```
. estat gof, group(10)
note: obs collapsed on 10 quantiles of estimated probabilities.
Goodness-of-fit test after logistic model
Variable: low
    Number of observations = 189
    Number of groups = 10
Hosmer-Lemeshow chi2(8) = 9.65
        Prob > chi2 = 0.2904
```

Again we cannot reject our model. If we specify the table option, estat gof displays the groups along with the expected and observed number of positive responses (low-birthweight babies):
. estat gof, group(10) table
note: obs collapsed on 10 quantiles of estimated probabilities.
Goodness-of-fit test after logistic model
Variable: low
Table collapsed on quantiles of estimated probabilities

| Group | Prob | Obs_1 | Exp_1 | Obs_0 | Exp_0 | Total |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| 1 | 0.0827 | 0 | 1.2 | 19 | 17.8 | 19 |
| 2 | 0.1276 | 2 | 2.0 | 17 | 17.0 | 19 |
| 3 | 0.2015 | 6 | 3.2 | 13 | 15.8 | 19 |
| 4 | 0.2432 | 1 | 4.3 | 18 | 14.7 | 19 |
| 5 | 0.2792 | 7 | 4.9 | 12 | 14.1 | 19 |
| 6 | 0.3138 | 7 | 5.6 | 12 | 13.4 | 19 |
| 7 | 0.3872 | 6 | 6.5 | 13 | 12.5 | 19 |
| 8 | 0.4828 | 7 | 8.2 | 12 | 10.8 | 19 |
| 9 | 0.5941 | 10 | 10.3 | 9 | 8.7 | 19 |
| 10 | 0.8391 | 13 | 12.8 | 5 | 5.2 | 18 |

Number of observations $=$
Number of groups $=$
Hosmer-Lemeshow chi2(8) $=$
Prob > chi2 $=0.2904$

In this table, the column Prob shows the upper boundaries of predicted probabilities for these 10 groups, which are the 10 th, 20 th, $\ldots$, and 100 th percentiles in this case.

## - Technical note

estat gof with the group() option puts all observations with the same predicted probabilities into the same group. If, as in the previous example, we request 10 groups, the groups that estat gof makes are $\left[p_{0}, p_{10}\right],\left(p_{10}, p_{20}\right],\left(p_{20}, p_{30}\right], \ldots,\left(p_{90}, p_{100}\right]$, where $p_{k}$ is the $k$ th percentile of the predicted probabilities, with $p_{0}$ the minimum and $p_{100}$ the maximum.

If there are many ties at the quantile boundaries, as will often happen if all independent variables are categorical and there are only a few of them, the sizes of the groups will be uneven. If the totals in some of the groups are small, the $\chi^{2}$ statistic for the Hosmer-Lemeshow test may be unreliable. In this case, fewer groups should be specified, or the Pearson goodness-of-fit test may be a better choice.

## > Example 2

The table option can be used without the group() option. We would not want to specify this for our current model because there were 182 covariate patterns in the data, caused by including the two continuous variables, age and lwt, in the model. As an aside, we fit a simpler model and specify table with estat gof:
. logistic low i.race smoke ui
Logistic regression
Number of obs = 189
LR chi2 (4) $=18.80$
Prob > chi2 $=0.0009$
Pseudo R2 $=0.0801$

| low | Odds ratio | Std. err. | $z$ | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :--- | :---: | :---: | :---: | :---: |
| race |  |  |  |  |  |  |
| Black | 3.052746 | 1.498087 | 2.27 | 0.023 | 1.166747 | 7.987382 |
| Other | 2.922593 | 1.189229 | 2.64 | 0.008 | 1.316457 | 6.488285 |
|  |  |  |  |  |  |  |
| smoke | 2.945742 | 1.101838 | 2.89 | 0.004 | 1.415167 | 6.131715 |
| ui | 2.419131 | 1.047359 | 2.04 | 0.041 | 1.035459 | 5.651788 |
| _cons | .1402209 | .0512295 | -5.38 | 0.000 | .0685216 | .2869447 |

Note: _cons estimates baseline odds.
. estat gof, table
Goodness-of-fit test after logistic model
Variable: low

| Group | Prob | Obs_1 | Exp_1 | Obs_0 | Exp_0 | Total |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| 1 | 0.1230 | 3 | 4.9 | 37 | 35.1 | 40 |
| 2 | 0.2533 | 1 | 1.0 | 3 | 3.0 | 4 |
| 3 | 0.2907 | 16 | 13.7 | 31 | 33.3 | 47 |
| 4 | 0.2923 | 15 | 12.6 | 28 | 30.4 | 43 |
| 5 | 0.2997 | 3 | 3.9 | 10 | 9.1 | 13 |
| 6 | 0.4978 | 4 | 4.0 | 4 | 4.0 | 8 |
| 7 | 0.4998 | 4 | 4.5 | 5 | 4.5 | 9 |
| 8 | 0.5087 | 2 | 1.5 | 1 | 1.5 | 3 |
| 9 | 0.5469 | 2 | 4.4 | 6 | 3.6 | 8 |
| 10 | 0.5577 | 6 | 5.6 | 4 | 4.4 | 10 |
| 11 | 0.7449 | 3 | 3.0 | 1 | 1.0 | 4 |


| Group | Prob | race | smoke | ui |
| ---: | ---: | ---: | ---: | ---: |
| 1 | 0.1230 | White | Nonsmoker | 0 |
| 2 | 0.2533 | White | Nonsmoker | 1 |
| 3 | 0.2907 | Other | Nonsmoker | 0 |
| 4 | 0.2923 | White | Smoker | 0 |
| 5 | 0.2997 | Black | Nonsmoker | 0 |
| 6 | 0.4978 | Other | Nonsmoker | 1 |
| 7 | 0.4998 | White | Smoker | 1 |
| 8 | 0.5087 | Black | Nonsmoker | 1 |
| 9 | 0.5469 | Other | Smoker | 0 |
| 10 | 0.5577 | Black | Smoker | 0 |
| 11 | 0.7449 | Other | Smoker | 1 |

Number of observations $=189$
Number of covariate patterns $=11$
Pearson chi2 (6) $=5.71$
Prob > chi2 $=0.4569$

- Technical note
logistic, logit, or probit and estat gof keep track of the estimation sample. If you type, for instance, logistic ... if $x==1$, then when you type estat gof, the statistics will be calculated on the $x==1$ subsample of the data automatically.

You should specify if or in with estat gof only when you wish to calculate statistics for a set of observations other than the estimation sample. See Samples other than the estimation sample later in this entry.

If the logistic model was fit with fweights, estat gof properly accounts for the weights in its calculations. (estat gof allows only fweights.) You do not have to specify the weights when you run estat gof. Weights should be specified with estat gof only when you wish to use a different set of weights.

## Samples other than the estimation sample

estat gof can be used with samples other than the estimation sample. By default, estat gof remembers the estimation sample used with the last logistic, logit, or probit command. To override this, simply use an if or in restriction to select another set of observations, or specify the all option to force the command to use all the observations in the dataset.

If you use estat gof with a sample that is completely different from the estimation sample (that is, no overlap), you should also specify the outsample option so that the $\chi^{2}$ statistic properly adjusts the degrees of freedom upward. For an overlapping sample, the conservative thing to do is to leave the degrees of freedom the same as they are for the estimation sample.

## > Example 3

We want to develop a model for predicting low-birthweight babies. One approach would be to divide our data into two groups, a developmental sample and a validation sample. See Lemeshow and Gall (1994) and Tilford, Roberson, and Fiser (1995) for more information on developing prediction models and severity-scoring systems.

We will do this with the low-birthweight data that we considered previously. First, we randomly divide the data into two samples.

```
. use https://www.stata-press.com/data/r18/lbw, clear
(Hosmer & Lemeshow data)
. set seed 101
. generate r = runiform()
. sort r
. generate group = 1 if _n <= _N/2
(95 missing values generated)
. replace group = 2 if group==.
(95 real changes made)
```

Then, we fit a model using the first sample (group $=1$ ), which is our developmental sample.

```
. logistic low age lwt i.race smoke ptl ht ui if group==1
```

Logistic regression $\quad$ Number of obs $=\quad 94$
LR chi2(8) $=28.03$
Prob > chi2 $=0.0005$
Log likelihood $=-42.351112$
Pseudo R2 $=0.2487$

| low | Odds ratio | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| age | .922865 | .0555349 | -1.33 | 0.182 | .8201924 | 1.03839 |
| lwt | .9825782 | .0114438 | -1.51 | 0.131 | .9604029 | 1.005265 |
| race |  |  |  |  |  |  |
| Black | 5.975476 | 4.936135 | 2.16 | 0.030 | 1.183652 | 30.16621 |
| Other | 3.364479 | 2.760784 | 1.48 | 0.139 | .6736724 | 16.803 |
|  |  |  |  |  |  |  |
| smoke | 3.442716 | 2.53779 | 1.68 | 0.094 | .8117831 | 14.60032 |
| ptl | 3.467274 | 2.337648 | 1.84 | 0.065 | .9249222 | 12.99784 |
| ht | 5.928512 | 6.047106 | 1.74 | 0.081 | .8030021 | 43.76982 |
| ui | 4.045883 | 2.947396 | 1.92 | 0.055 | .9703295 | 16.8697 |
| _cons | 3.120871 | 5.977489 | 0.59 | 0.552 | .0731049 | 133.231 |

Note: _cons estimates baseline odds.
To test calibration in the developmental sample, we calculate the Hosmer-Lemeshow goodness-of-fit test by using estat gof.

```
. estat gof, group(10)
note: obs collapsed on 10 quantiles of estimated probabilities.
Goodness-of-fit test after logistic model
Variable: low
    Number of observations = 94
    Number of groups = 10
Hosmer-Lemeshow chi2(8) = 5.64
        Prob > chi2 = 0.6871
```

We did not specify an if statement with estat gof because we wanted to use the estimation sample. Because the test is not significant, we are satisfied with the fit of our model.

Running lroc (see [R] lroc) gives a measure of the discrimination:

```
. lroc, nograph
Logistic model for low
Number of observations = 94
Area under ROC curve = 0.8145
```

Now, we test the calibration of our model by performing a goodness-of-fit test on the validation sample. We specify the outsample option so that the number of degrees of freedom is 10 rather than 8 .
. estat gof if group $==2$, group (10) table outsample
note: obs collapsed on 10 quantiles of estimated probabilitie
Goodness-of-fit test after logistic model
Variable: low
Table collapsed on quantiles of estimated probabilities

| Group | Prob | Obs_1 | Exp_1 | Obs_0 | Exp_0 | Total |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| 1 | 0.0276 | 0 | 0.2 | 10 | 9.8 | 10 |
| 2 | 0.0496 | 2 | 0.4 | 7 | 8.6 | 9 |
| 3 | 0.0875 | 1 | 0.7 | 9 | 9.3 | 10 |
| 4 | 0.1536 | 4 | 1.1 | 5 | 7.9 | 9 |
| 5 | 0.2283 | 4 | 2.0 | 6 | 8.0 | 10 |
| 6 | 0.2842 | 4 | 2.2 | 5 | 6.8 | 9 |
| 7 | 0.4190 | 3 | 3.6 | 7 | 6.4 | 10 |
| 8 | 0.5248 | 5 | 4.3 | 4 | 4.7 | 9 |
| 9 | 0.6413 | 5 | 5.8 | 5 | 4.2 | 10 |
| 10 | 0.9787 | 4 | 7.3 | 5 | 1.7 | 9 |

Number of observations $=95$
Number of groups $=10$
Hosmer-Lemeshow chi2(10) $=29.30$
Prob $>$ chi2 $=0.0011$
We must acknowledge that our model does not fit well on the validation sample. The model's discrimination in the validation sample is appreciably lower, as well.
. lroc if group==2, nograph
Logistic model for low
Number of observations $=95$
Area under ROC curve $=0.6835$

## Stored results

estat gof stores the following in r() :
Scalars

```
r(N) number of observations
    r(m) number of covariate patterns or groups
    r(df) degrees of freedom
    r(chi2) < < 
    r(p) p-value for }\mp@subsup{\chi}{}{2}\mathrm{ test
```


## Methods and formulas

Let $M$ be the total number of covariate patterns among the $N$ observations. View the data as collapsed on covariate patterns $j=1,2, \ldots, M$, and define $m_{j}$ as the total number of observations having covariate pattern $j$ and $y_{j}$ as the total number of positive responses among observations with covariate pattern $j$. Define $p_{j}$ as the predicted probability of a positive outcome in covariate pattern $j$.

The Pearson $\chi^{2}$ goodness-of-fit statistic is

$$
\chi^{2}=\sum_{j=1}^{M} \frac{\left(y_{j}-m_{j} p_{j}\right)^{2}}{m_{j} p_{j}\left(1-p_{j}\right)}
$$

This $\chi^{2}$ statistic has approximately $M-k$ degrees of freedom for the estimation sample, where $k$ is the number of independent variables, including the constant. For a sample outside the estimation sample, the statistic has $M$ degrees of freedom.

The Hosmer-Lemeshow goodness-of-fit $\chi^{2}$ (Hosmer and Lemeshow 1980; Lemeshow and Hosmer 1982; Hosmer, Lemeshow, and Klar 1988) is calculated similarly, except that rather than using the $M$ covariate patterns as the group definition, the quantiles of the predicted probabilities are used to form groups. Let $G=$ \# be the number of quantiles requested with group (\#). The smallest index $1 \leq q(i) \leq M$, such that

$$
W_{q(i)}=\sum_{j=1}^{q(i)} m_{j} \geq \frac{N}{G}
$$

gives $p_{q(i)}$ as the upper boundary of the $i$ th quantile for $i=1,2, \ldots, G$. Let $q(0)=1$ denote the first index.

The groups are then

$$
\left[p_{q(0)}, p_{q(1)}\right],\left(p_{q(1)}, p_{q(2)}\right], \ldots,\left(p_{q(G-1)}, p_{q(G)}\right]
$$

If the table option is given, the upper boundaries $p_{q(1)}, \ldots, p_{q(G)}$ of the groups appear next to the group number on the output.

The resulting $\chi^{2}$ statistic has approximately $G-2$ degrees of freedom for the estimation sample. For a sample outside the estimation sample, the statistic has $G$ degrees of freedom.

## References

Archer, K. J., and S. A. Lemeshow. 2006. Goodness-of-fit test for a logistic regression model fitted using survey sample data. Stata Journal 6: 97-105.
Fagerland, M. W., and D. W. Hosmer, Jr. 2012. A generalized Hosmer-Lemeshow goodness-of-fit test for multinomial logistic regression models. Stata Journal 12: 447-453.

Hosmer, D. W., Jr., and S. A. Lemeshow. 1980. Goodness of fit tests for the multiple logistic regression model. Communications in Statistics—Theory and Methods 9: 1043-1069. https://doi.org/10.1080/03610928008827941.
Hosmer, D. W., Jr., S. A. Lemeshow, and J. Klar. 1988. Goodness-of-fit testing for the logistic regression model when the estimated probabilities are small. Biometrical Journal 30: 911-924. https://doi.org/10.1002/bimj. 4710300805.
Hosmer, D. W., Jr., S. A. Lemeshow, and R. X. Sturdivant. 2013. Applied Logistic Regression. 3rd ed. Hoboken, NJ: Wiley.
Lemeshow, S. A., and J.-R. L. Gall. 1994. Modeling the severity of illness of ICU patients: A systems update. Journal of the American Medical Association 272: 1049-1055. https://doi.org/10.1001/jama.1994.03520130087038.

Lemeshow, S. A., and D. W. Hosmer, Jr. 1982. A review of goodness of fit statistics for the use in the development of logistic regression models. American Journal of Epidemiology 115: 92-106. https://doi.org/10.1093/oxfordjournals.aje.a113284.
Nattino, G., S. A. Lemeshow, G. Phillips, S. Finazzi, and G. Bertolini. 2017. Assessing the calibration of dichotomous outcome models with the calibration belt. Stata Journal 17: 1003-1014.

Tilford, J. M., P. K. Roberson, and D. H. Fiser. 1995. sbe12: Using lifit and lroc to evaluate mortality prediction models. Stata Technical Bulletin 28: 14-18. Reprinted in Stata Technical Bulletin Reprints, vol. 5, pp. 77-81. College Station, TX: Stata Press.

## Also see

[R] logistic - Logistic regression, reporting odds ratios
[R] logit - Logistic regression, reporting coefficients
[R] probit - Probit regression
[R] estat classification - Classification statistics and table
$[R]$ Iroc - Compute area under ROC curve and graph the curve
[R] Isens - Graph sensitivity and specificity versus probability cutoff
[U] 20 Estimation and postestimation commands

## Title

estat ic - Display information criteria

| Description | Quick start | Menu for estat | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

estat ic computes Akaike's (AIC), consistent Akaike's (CAIC), corrected Akaike's (AICc), and Schwarz's Bayesian (BIC) information criteria.

## Quick start

Display AIC and BIC
estat ic
Display CAIC and BIC
estat ic, aicconsistent
Display AICc and BIC
estat ic, aiccorrected
Display AIC, BIC, AICc, and CAIC
estat ic, all
Specify $N$ to be used in calculating BIC as 500
estat ic, n(500)
Specify $N$ and degrees of freedom to be used in calculating all information criteria as 500 and 10 , respectively
estat ic, $n(500) d f(10)$ all

## Menu for estat

Statistics $>$ Postestimation

## Syntax

estat ic [, options]
options Description

| aiccorrected | report AICc instead of AIC <br> report CAIC instead of AIC |
| :--- | :--- |
| aicconsistent | report all four information criteria: AIC, BIC, AICc, and CAIC <br> all <br> $\mathrm{n}(\#)$ |
|  | specify $N$ to be used in calculating BIC, AICc, and CAIC; <br> see $[R]$ IC note |
| specify degrees of freedom $k$ to be used in calculating AIC, BIC, |  |
| AICc, and CAIC |  |

collect is allowed; see [U] 11.1.10 Prefix commands.

## Options

aiccorrected specifies that AICc be computed instead of AIC. This information criterion is a second-order approximation and is recommended for small sample sizes.

Only one of aiccorrected, aicconsistent, or all is allowed.
aicconsistent specifies that CAIC be computed instead of AIC. This information criterion is a consistent version of AIC; that is, the probability of selecting the "true model" approaches 1 as sample size increases.

Only one of aicconsistent, aiccorrected, or all is allowed.
all produces a table showing all four information criteria: AIC, BIC, AICc, and CAIC.
Only one of all, aiccorrected, or aicconsistent is allowed.
$\mathrm{n}(\#)$ specifies $N$ to be used in calculating BIC, AICc, and CAIC; see [R] IC note.
$\mathrm{df}(\#)$ specifies degrees of freedom $k$ to be used in calculating AIC, BIC, AICc, and CAIC. By default, $k$ is the number of estimated parameters.

## Remarks and examples

estat ic calculates four information criteria used to compare models fit to the same dataset. Unlike likelihood-ratio, Wald, and similar testing procedures, the models need not be nested to compare the information criteria. The information criteria are constructed as a function of the log likelihood $\ln L$, the number of estimated parameters (degrees of freedom) $k$, and, in some cases, the number of observations $N$. Because they are based on the log-likelihood function, information criteria are available only after commands that report the log likelihood.

The use of information criteria is subjective, and no formal inference can be drawn from the reported values. In a typical approach, a set of potential models is selected, and a superior model is selected from the values of information criteria. A superior model is the model with the lowest value of information criterion. For example, given two models, the model with the lowest AIC fits the data better than the model with the larger AIC. For details, see Methods and formulas.

## $>$ Example 1

In [R] mlogit, we fit a model explaining the type of insurance a person has on the basis of age, gender, race, and site of study. Here we refit the model with and without the site dummies and compare the models.

```
. use https://www.stata-press.com/data/r18/sysdsn1
(Health insurance data)
. mlogit insure age male nonwhite
    (output omitted)
. estat ic
Akaike's information criterion and Bayesian information criterion
```

| Model | N | ll (null) | ll(model) | df | AIC | BIC |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| . | 615 | -555.8545 | -545.5833 | 8 | 1107.167 | 1142.54 |

Note: BIC uses $N$ = number of observations. See [R] IC note.
. mlogit insure age male nonwhite i.site
(output omitted)
. estat ic
Akaike's information criterion and Bayesian information criterion

| Model | N | ll (null) | ll(model) | df | AIC | BIC |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| . | 615 | -555.8545 | -534.3616 | 12 | 1092.723 | 1145.783 |

Note: BIC uses $N=$ number of observations. See [R] IC note.
The AIC indicates that the model including the site dummies fits the data better, whereas BIC indicates the opposite. As is often the case, different model-selection criteria have led to conflicting conclusions.

## > Example 2

In example 1, we compared AIC and BIC. Here we focus on comparing AIC and AICc for small sample size. For simplicity, we are using the same health insurance dataset but running mlogit with the age $<30$ condition to reduce the sample size.

```
. mlogit insure age male nonwhite if age < 30
    (output omitted)
. estat ic, all
Information criteria
```

| Model | N | ll(null) | ll(model) | df |
| ---: | ---: | ---: | ---: | ---: |
| $\cdot$ | 87 | -76.93025 | -70.36684 | 8 |

Note: BIC, AICc, and CAIC use $N=$ number of observations.
See [R] IC note.

| Model | AIC | BIC | AICc | CAIC |
| ---: | ---: | ---: | ---: | ---: |
| . | 156.7337 | 176.4609 | 158.5798 | 184.4609 |

Legend: AIC is Akaike's information criterion.
BIC is Bayesian information criterion.
AICc is corrected Akaike's information criterion.
CAIC is consistent Akaike's information criterion.
. mlogit insure age male nonwhite i.site if age < 30
(output omitted)
. estat ic, all
Information criteria

| Model | N | ll(null) | ll(model) | df |
| ---: | ---: | ---: | ---: | ---: |
| . | 87 | -76.93025 | -66.03298 | 12 |

Note: BIC, AICc, and CAIC use $N=$ number of observations.
See [R] IC note.

| Model | AIC | BIC | AICc | CAIC |
| ---: | ---: | ---: | ---: | ---: |
| . | 156.066 | 185.6569 | 160.2822 | 197.6569 |

Legend: AIC is Akaike's information criterion.
BIC is Bayesian information criterion.
AICc is corrected Akaike's information criterion.
CAIC is consistent Akaike's information criterion.
Burnham and Anderson (2002) recommend using AICc when the ratio $N / k<40$. The AIC suggests that the model with the site dummies is preferred, whereas AICc reports the opposite result.

## Example 3

As we discuss in the technical note below, for the linear mixed models fit using restricted maximum likelihood (REML), one needs to be careful when comparing models using the standard information criteria, especially when the fixed-effects specifications differ across models. In this example, we show how to use $\mathrm{n}(\#)$ and $\mathrm{df}(\#)$ to modify the the standard $N$ and $k$ used in the information criteria
when we compare such models. As in [ME] mixed, we consider the dataset from Munnell (1990) and estimate a Cobb-Douglas production function, which examines the productivity of public capital in each state's private output (Baltagi, Song, and Jung 2001).

Suppose we want to compare two models:

```
. use https://www.stata-press.com/data/r18/productivity
(Public capital productivity)
. mixed gsp private emp hwy water other unemp || region: || state:, reml
    (output omitted)
. estimates store model1
. mixed gsp private emp hwy unemp || region: hwy || state: unemp, reml
    (output omitted)
. estimates store model2
```

The two models differ in both their fixed-effects and random-effects specifications. By default, the number of degrees of freedom in estat ic is calculated as $k=k_{f}+k_{r}$, where $k_{f}$ and $k_{r}$ are the number of estimated fixed-effects and random-effects parameters, respectively. For REML, Gurka (2006) evaluates the performance of various information criteria. He discusses using $k=k_{r}$ and different possible values for $N$. Here, we follow the Vonesh and Chinchilli (1997) approach and choose $N-k_{f}$. Finally, we run estat ic to compare the models:

```
. estimates restore model1
(results model1 are active now)
. estat ic, n(809) df(3)
Akaike's information criterion and Bayesian information criterion
```

| Model | N | ll (null) | ll(model) | df | AIC | BIC |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| model1 | 809 |  | 1404.71 | 3 | -2803.42 | -2789.333 |

. estimates restore model2
(results model2 are active now)
. estat ic, $\mathrm{n}(811) \mathrm{df}(5)$

Akaike's information criterion and Bayesian information criterion

| Model | N | ll (null) | ll(model) | df | AIC | BIC |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| model2 | 811 |  | 1413.557 | 5 | -2817.114 | -2793.623 |

Both AIC and BIC indicate that the second model is preferable.

## - Technical note

glm and binreg, ml report a slightly different version of AIC and BIC; see [R] glm for the formulas used. That version is commonly used within the generalized linear models literature; see, for example, Hardin and Hilbe (2018). The literature on information criteria is vast; see, among others, Akaike (1973), Sawa (1978), and Raftery (1995). Judge et al. (1985) discuss the use of information criteria in econometrics. Royston and Sauerbrei (2008, chap. 2) examine the use of information criteria as an alternative to stepwise procedures for selecting model variables.

For linear mixed models, when restricted maximum likelihood is used, the information criteria with default degrees of freedom and the number of observations cannot be used to compare models with varying sets of fixed effects, because the likelihood of restricted maximum likelihood is dependent on the fixed-effects design matrix (Harville 1974; Gurka 2006). By default, the degrees of freedom in estat ic is the sum of the dimension of fixed-effect parameters and the number of covariance parameters. Therefore, only models with the same sets of fixed effects can be compared. However, for each model, the $\mathrm{df}(\#)$ option can be specified manually to allow comparison with different sets of fixed effects. There are also different views on which number should be used as $N$ to calculate BIC, AICc, and CAIC. For example, see Vonesh and Chinchilli (1997) and Kass and Raftery (1995). Use the n (\#) option to pass a desired number of observations to the estat ic command. For details, see [R] IC note.

## Stored results

estat ic stores the following in $r()$ :
Matrices
$r(S) \quad$ row vector with columns (N, 11 (null), 11 (model), df, and information criteria)

## Methods and formulas

There are two main large-sample notions of information criteria: efficiency and consistency (Burnham and Anderson 2002). Efficient criteria target the best finite dimension model under the assumption that the unknown "true model" has infinite dimension. In contrast, assuming that the true data-generating model is finite and fixed, the consistent criterion selects the correct model with probability approaching 1 as $N \rightarrow \infty$. The AIC and AICc belong to the efficient class, while the BIC and CAIC to the consistent class.

Akaike's (1974) information criterion is defined as

$$
\mathrm{AIC}=-2 \ln L+2 k
$$

where $\ln L$ is the maximized $\log$-likelihood of the model and $k$ is the number of parameters estimated. Some authors define AIC as the expression above divided by the sample size.

AIC performs poorly when there are too many parameters in relation to the sample size. Hurvich and Tsai (1989) derived a second-order variant of AIC called AICc,

$$
\mathrm{AICc}=\mathrm{AIC}+\frac{2 k(k+1)}{N-k-1}
$$

where $N$ is the sample size. See $[\mathrm{R}]$ IC note for additional information on calculating and interpreting $N$. Compared with AIC, AICc has an additional bias-correction term, and for large $N$ and small $k$, this term is negligible. Burnham and Anderson (2002) recommend using AICc when the ratio $N / k<40$.

Schwarz's (1978) Bayesian information criterion is another measure of fit defined as

$$
\mathrm{BIC}=-2 \ln L+k \ln N
$$

Bozdogan (1987) proposed a consistent version of AIC called CAIC,

$$
\mathrm{CAIC}=-2 \ln L+k(\ln N+1)
$$

Burnham and Anderson (2002, chap. 6) argue that employing and comparing consistent and efficient information criteria in the same situation contrasts with the fact that they were designed to answer different questions. Thus, one needs to be careful when interpreting the results.

Hirotugu Akaike (1927-2009) was born in Fujinomiya City, Shizuoka Prefecture, Japan. He was the son of a silkworm farmer. He gained BA and DSc degrees from the University of Tokyo. Akaike's career from 1952 at the Institute of Statistical Mathematics in Japan culminated in service as Director General; after 1994, he was Professor Emeritus. His best-known work in a prolific career is on what is now known as the Akaike information criterion (AIC), which was formulated to help selection of the most appropriate model from a number of candidates.

Gideon E. Schwarz (1933-2007) was a professor of statistics at the Hebrew University, Jerusalem. He was born in Salzburg, Austria, and obtained an MSc in 1956 from the Hebrew University and a PhD in 1961 from Columbia University. His interests included stochastic processes, sequential analysis, probability, and geometry. He is best known for the Bayesian information criterion (BIC).

## References

Akaike, H. 1973. Information theory and an extension of the maximum likelihood principle. In Second International Symposium on Information Theory, ed. B. N. Petrov and F. Csaki, 267-281. Budapest: Akailseoniai-Kiudo.
——. 1974. A new look at the statistical model identification. IEEE Transactions on Automatic Control 19: 716-723. https://doi.org/10.1109/TAC.1974.1100705.
Baltagi, B. H., S. H. Song, and B. C. Jung. 2001. The unbalanced nested error component regression model. Journal of Econometrics 101: 357-381. https://doi.org/10.1016/S0304-4076(00)00089-0.

Bozdogan, H. 1987. Model selection and Akaike's information criterion (AIC): The general theory and its analytical extensions. Psychometrika 52: 345-370. https://doi.org/10.1007/BF02294361.
Burnham, K. P., and D. R. Anderson. 2002. Model Selection and Multimodel Inference: A Practical InformationTheoretic Approach. 2nd ed. New York: Springer.

Findley, D. F., and E. Parzen. 1995. A conversation with Hirotugu Akaike. Statistical Science 10: 104-117. https://doi.org/10.1214/ss/1177010133.
Gurka, M. J. 2006. Selecting the best linear mixed model under REML. American Statistician 60: 19-26. https://doi.org/10.1198/000313006X90396.

Hardin, J. W., and J. M. Hilbe. 2018. Generalized Linear Models and Extensions. 4th ed. College Station, TX: Stata Press.
Harville, D. A. 1974. Bayesian inference for variance components using only error contrasts. Biometrika 61: 383-385. https://doi.org/10.2307/2334370.

Hurvich, C. M., and C.-L. Tsai. 1989. Regression and time series model selection in small samples. Biometrika 76: 297-307. https://doi.org/10.1093/biomet/76.2.297.
Judge, G. G., W. E. Griffiths, R. C. Hill, H. Lütkepohl, and T.-C. Lee. 1985. The Theory and Practice of Econometrics. 2nd ed. New York: Wiley.

Kass, R. E., and A. E. Raftery. 1995. Bayes factors. Journal of the American Statistical Association 90: 773-795. https://doi.org/10.1080/01621459.1995.10476572.
Munnell, A. H. 1990. Why has productivity growth declined? Productivity and public investment. New England Economic Review Jan./Feb.: 3-22.

Raftery, A. E. 1995. Bayesian model selection in social research. In Vol. 25 of Sociological Methodology, ed. P. V. Marsden, 111-163. Oxford: Blackwell.
Royston, P., and W. Sauerbrei. 2008. Multivariable Model-Building: A Pragmatic Approach to Regression Analysis Based on Fractional Polynomials for Modelling Continuous Variables. Chichester, UK: Wiley.

Sawa, T. 1978. Information criteria for discriminating among alternative regression models. Econometrica 46: 12731291. https://doi.org/10.2307/1913828.

Schwarz, G. 1978. Estimating the dimension of a model. Annals of Statistics 6: 461-464. https://doi.org/10.1214/aos/1176344136.
Tong, H. 2010. Professor Hirotugu Akaike, 1927-2009. Journal of the Royal Statistical Society, Series A 173: 451-454. https://doi.org/10.1111/j.1467-985X.2009.00633.x.

Vonesh, E. F., and V. M. Chinchilli. 1997. Linear and Nonlinear Models for the Analysis of Repeated Measurements. New York: Dekker.

## Also see

[R] estat - Postestimation statistics
[R] estat summarize - Summarize estimation sample
$[R]$ estat vce - Display covariance matrix estimates
[R] estimates stats - Model-selection statistics

## Title

estat summarize - Summarize estimation sample

| Description | Quick start | Menu for estat | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Also see |

## Description

estat summarize summarizes the variables used by the command and automatically restricts the sample to the estimation sample; it also summarizes the weight variable and cluster structure, if specified.

## Quick start

Summary statistics for all variables in the model using estimation sample
estat summarize
Add variable labels to output
estat summarize, labels
Obtain summary of estimation sample for each equation
estat summarize, equation
Ignore weights when calculating summary statistics after weighted estimation
estat summarize, noweights

## Menu for estat

Statistics > Postestimation

## Syntax

estat summarize [eqlist] [, estat_summ_options]
estat_summ_options Description

| equation | display summary by equation <br> display summary by group; only after sem and gsem |
| :--- | :--- |
| group | display variable labels |
| nobels | suppress the header <br> noheader |
| ignore weights |  |
| coneights | control row spacing, line width, display of omitted variables <br> and base and empty cells, and factor-variable labeling |

eqlist is rarely used and specifies the variables, with optional equation name, to be summarized. eqlist may be varlist or (eqname ${ }_{1}$ : varlist) (eqname ${ }_{2}$ : varlist) .... varlist may contain time-series operators; see [U] 11.4.4 Time-series varlists.
collect is allowed; see [U] 11.1.10 Prefix commands.

## Options

equation requests that the dependent variables and the independent variables in the equations be displayed in the equation-style format of estimation commands, repeating the summary information about variables entered in more than one equation.
group displays summary information separately for each group. group is only allowed after sem or gsem with a group() variable specified.
labels displays variable labels.
noheader suppresses the header.
noweights ignores the weights, if any, from the previous estimation command. The default when weights are present is to perform a weighted summarize on all variables except the weight variable itself. An unweighted summarize is performed on the weight variable.
display_options: noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), and fvwrapon(style); see [R] Estimation options.

## Remarks and examples

Often when fitting a model, you will also be interested in obtaining summary statistics, such as the sample means and standard deviations of the variables in the model. estat summarize makes this process simple. The output displayed is similar to that obtained by typing

```
. summarize varlist if e(sample)
```

without the need to type the varlist containing the dependent and independent variables.

## $>$ Example 1

Continuing with the example in $[\mathrm{R}]$ estat ic, here we summarize the variables by using estat summarize.

| mlogit insure (output omitted) <br> estat summari <br> Estimation sa | ale nonwh <br> omitted <br> logit | i.site |  |  |
| :---: | :---: | :---: | :---: | :---: |
| Variable | Mean | Std. dev. | Min | Max |
| insure | 1.596748 | . 6225846 | 1 | 3 |
| age | 44.46832 | 14.18523 | 18.11087 | 86.07254 |
| male | . 2504065 | . 4335998 | 0 | 1 |
| nonwhite | . 196748 | . 3978638 | 0 | 1 |
| site |  |  |  |  |
| 2 | . 3707317 | . 4833939 | 0 | 1 |
| 3 | . 3138211 | . 4644224 | 0 | 1 |

The output in the previous example contains all the variables in one table, though mlogit presents its results in a multiple-equation format. For models in which the same variables appear in all equations, that is fine; but for other multiple-equation models, we may prefer to have the variables separated by the equation in which they appear. The equation option makes this possible.

## > Example 2

Systems of simultaneous equations typically have different variables in each equation, and the equation option of estat summarize is helpful in such situations. In example 2 of $[R]$ reg3, we have a model of supply and demand. We first refit the model and then call estat summarize.


The first block of the table contains statistics on the dependent (or, more accurately, left-hand-side) variables, and because we specified quantity as the left-hand-side variable in both equations, it is listed twice. The second block refers to the variables in the first equation we specified, which we labeled "Demand" in our call to reg3; and the final block refers to the supply equation.

## Stored results

estat summarize stores the following in $r()$ :
Scalars
r(N_groups) number of groups (group only)
Matrices
r (stats) $\quad k \times 4$ matrix of means, standard deviations, minimums, and maximums
$\mathrm{r}($ stats [ \# $]$ ) $\left.\quad \begin{array}{c}k \times 4 \text { matrix } \\ \text { only }\end{array}\right)$ of means, standard deviations, minimums, and maximums for group \# (group

## Also see

[R] estat - Postestimation statistics
$[\mathrm{R}]$ estat ic - Display information criteria
[R] estat vce - Display covariance matrix estimates

## Title

estat vce - Display covariance matrix estimates

| Description | Quick start | Menu for estat | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Also see |

## Description

estat vce displays the covariance or correlation matrix of the parameter estimates of the previous model.

## Quick start

Display variance-covariance matrix of the estimates (VCE) from the previous model

> estat vce

Matrix of correlations rather than covariances
estat vce, correlation
Same as above, but report correlations using three decimal places
estat vce, correlation format(\%6.3f)
After fitting a multiple-equation model, display VCE for each equation in separate blocks
estat vce, block
Show VCE for equation y1 only
estat vce, equation(y1)

## Menu for estat

Statistics $>$ Postestimation

## Syntax

```
    estat vce [, estat_vce_options]
```

estat_vce_options Description

| covariance | display as covariance matrix; the default <br> display as correlation matrix |
| :--- | :--- |
| correlation | display only specified equations |
| equation $($ spec $)$ | display submatrices by equation <br> block |
| display submatrices by equation; diagonal blocks only |  |
| diarmat $(\%$ fint $)$ | display format for covariances and correlations |
| nolines | suppress lines between equations <br> control display of omitted variables and base and empty cells |

collect is allowed; see [U] 11.1.10 Prefix commands.

## Options

covariance displays the matrix as a variance-covariance matrix; this is the default.
correlation displays the matrix as a correlation matrix rather than a variance-covariance matrix. rho is a synonym.
equation (spec) selects part of the VCE to be displayed. If spec is eqlist, the VCE for the listed equations is displayed. If spec is eqlistl $\backslash$ eqlist 2 , the part of the VCE associated with the equations in eqlistl (rowwise) and eqlist 2 (columnwise) is displayed. If $s p e c$ is $*$, all equations are displayed. equation() implies block if diag is not specified.
block displays the submatrices pertaining to distinct equations separately.
diag displays the diagonal submatrices pertaining to distinct equations separately.
format (\% fint) specifies the number format for displaying the elements of the matrix. The default is format ( $\% 10.0 \mathrm{~g}$ ) for covariances and format (\%8.4f) for correlations. See [U] 12.5 Formats: Controlling how data are displayed for more information.
nolines suppresses lines between equations.
display_options: noomitted, noemptycells, baselevels, allbaselevels; see [R] Estimation options.

## Remarks and examples

estat vce allows you to display the VCE of the parameters of the previously fit model, as either a covariance matrix or a correlation matrix.
> Example 1
Returning to the example in $[\mathrm{R}]$ estat ic, here we display the covariance matrix of the parameters of the mlogit model by using estat vce.

| mlogit insure age male nonwhite (output omitted) |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: |
| . estat vce, block |  |  |  |  |
| Covariance matrix of coefficients of mlogit model |  |  |  |  |
| Covariances of equation Indemnity |  |  |  |  |
|  | $\begin{aligned} & \circ \\ & \text { age } \end{aligned}$ | $\begin{gathered} \circ \\ \text { male } \end{gathered}$ | nonwhite | $\begin{array}{r} 0 . \\ \text { _cons } \end{array}$ |
| o.age | 0 |  |  |  |
| o.male | 0 | 0 |  |  |
| o.nonwhite | 0 | 0 | 0 |  |
| o._cons | 0 | 0 | 0 | 0 |
| Covariances of equation Prepaid (row) by equation Indemnity (column) |  |  |  |  |
|  | $\bigcirc$. | $\bigcirc$ | o. | $\bigcirc$. |
|  | age | male | nonwhite | _cons |
| age | 0 |  |  |  |
| male | 0 | 0 |  |  |
| nonwhite | 0 | 0 | 0 |  |
| _cons | 0 | 0 | 0 | 0 |
| Covariances of equation Prepaid |  |  |  |  |
|  | age | male | nonwhite | _cons |
| age | . 00003711 |  |  |  |
| male | -. 00015303 | . 0402091 |  |  |
| nonwhite | -. 00008948 | . 00470608 | . 04795135 |  |
| _cons | -. 00159095 | -. 00398961 | -. 00628886 | . 08000462 |
| Covariances of equation Uninsure (row) by equation Indemnity (column) |  |  |  |  |
|  |  | o. |  |  |
| age | 0 |  |  |  |
| male | 0 | 0 |  |  |
| nonwhite | 0 | 0 | 0 |  |
| _cons | 0 | 0 | 0 | 0 |
| Covariances of equation Uninsure (row) by equation Prepaid (column) |  |  |  |  |
|  | age | male | nonwhite | _cons |
| age | . 00001753 | -. 00007926 | -. 00004564 | -. 00076886 |
| male | -. 00007544 | . 02188398 | . 0023186 | -. 00145923 |
| nonwhite | -. 00004577 | . 00250588 | . 02813553 | -. 00263872 |
| _cons | -. 00077045 | -. 00130535 | -. 00257593 | . 03888032 |

Covariances of equation Uninsure

|  | age | male | nonwhite | _cons |
| ---: | ---: | ---: | ---: | ---: |
| age | .00013022 |  |  |  |
| male | -.00050406 | .13248095 |  |  |
| nonwhite | -.00026145 | .01505449 | .16861327 |  |
| _cons | -.00562159 | -.01686629 | -.02474852 | .28607591 |

The block option is particularly useful for multiple-equation estimators. The first block of output here corresponds to the VCE of the estimated parameters for the first equation-the square roots of the diagonal elements of this matrix are equal to the standard errors of the first equation's parameters. Similarly, the final block corresponds to the VCE of the parameters for the second equation. The middle block shows the covariances between the estimated parameters of the first and second equations.

## Stored results

estat vce stores the following in r() :
Matrices
$r(V) \quad V C E$ or correlation matrix

## Also see

[R] estat - Postestimation statistics
$[\mathrm{R}]$ estat ic - Display information criteria
[R] estat summarize - Summarize estimation sample

## Title

estimates - Save and manipulate estimation results

Description Syntax Remarks and examples Also see

## Description

estimates allows you to store and manipulate estimation results:

- You can save estimation results in a file for use in later sessions.
- You can store estimation results in memory so that you can
a. switch among separate estimation results and
b. form tables combining separate estimation results.


## Syntax

| Command | Reference |
| :--- | :--- |
| Save and use results from disk <br> estimates save filename | $[\mathrm{R}]$ estimates save |
| $\underline{\text { estimates use filename }}$ | $[\mathrm{R}]$ estimates save |
| estimates describe using filename | $[\mathrm{R}]$ estimates describe |
| estimates esample:... | $[\mathrm{R}]$ estimates save |

Store and restore estimates in memory

| estimates store name | $[\mathrm{R}]$ estimates store |
| :--- | :--- |
| estimates restore name | $[\mathrm{R}]$ estimates store |
| estimates query | $[\mathrm{R}]$ estimates store |
| estimates dir | $[R]$ estimates store |
| estimates drop namelist | $[R]$ estimates store |
| estimates clear | $[\mathrm{R}]$ estimates store |

Set titles and notes

| estimates title: text | $[\mathrm{R}]$ estimates title |
| :--- | :--- |
| estimates title | $[\mathrm{R}]$ estimates title |
| estimates notes: text | $[\mathrm{R}]$ estimates notes |
| estimates notes | $[\mathrm{R}]$ estimates notes |
| estimates notes list.. | $[R]$ estimates notes |
| estimates notes drop.. | $[R]$ estimates notes |

Report

| estimates describe $[$ name $]$ | $[R]$ estimates describe |
| :--- | :--- |
| estimates replay $[$ namelist $]$ | $[R]$ estimates replay |

estimates replay [namelist] [R] estimates replay

## Tables and statistics

| estimates table [namelist] | $[R]$ estimates table |
| :--- | :--- |
| etable | $[R]$ etable |
| estimates selected [namelist] | $[R]$ estimates selected |
| estimates stats [namelist] | $[R]$ estimates stats |
| $\underline{\text { estimates for namelist: ... }}$ | $[R]$ estimates for |

## Remarks and examples

estimates is for use after you have fit a model, be it with regress, logistic, etc. You can use estimates after any estimation command, whether it be an official estimation command of Stata or a community-contributed one.
estimates has three separate but related capabilities:

1. You can save estimation results in a file on disk so that you can use them later, even in a different Stata session.
2. You can store up to 300 estimation results in memory so that they are at your fingertips.
3. You can make tables comparing any results you have stored in memory.

Remarks are presented under the following headings:

> Saving and using estimation results
> Storing and restoring estimation results
> Comparing estimation results
> Jargon

## Saving and using estimation results

After you have fit a model, say, with regress, type

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. regress mpg weight displ foreign
    (output omitted)
```

You can save the results in a file:

```
. estimates save basemodel
file basemodel.ster saved
```

Later, say, in a different session, you can reload those results:

```
. estimates use basemodel
```

The situation is now nearly identical to what it was immediately after you fit the model. You can replay estimation results:

```
. regress
(output omitted)
```

You can perform tests:

```
. test foreign==0
(output omitted)
```

And you can use any postestimation command or postestimation capability of Stata. The only difference is that Stata no longer knows what the estimation sample, e (sample) in Stata jargon, was. When you reload the estimation results, you might not even have the original data in memory. That is okay. Stata will know to refuse to calculate anything that can be calculated only on the original estimation sample.

If it is important that you use a postestimation command that can be used only on the original estimation sample, there is a way you can do that. You use the original data and then use estimates esample: to tell Stata what the original sample was.

See $[R]$ estimates save for details.

## Storing and restoring estimation results

Storing and restoring estimation results in memory is much like saving them to disk. You type

```
. estimates store base
```

to save the current estimation results under the name base, and you type
. estimates restore base
to get them back later. You can find out what you have stored by typing

- estimates dir

Saving estimation results to disk is more permanent than storing them in memory, so why would you want merely to store them? The answer is that, once they are stored, you can use other estimates commands to produce tables and reports from them.

See [R] estimates store for details about the estimates store and restore commands.

## Comparing estimation results

Let's say that you have done the following:

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. regress mpg weight displ
    (output omitted)
- estimates store base
. regress mpg weight displ foreign
    (output omitted)
. estimates store alt
```

You can now get a table comparing the coefficients:
. estimates table base alt

| Variable | base | alt |
| ---: | :---: | ---: |
| weight | -.00656711 | -.00677449 |
| displacement | .00528078 | .00192865 |
| foreign |  | -1.6006312 |
| _cons | 40.084522 | 41.847949 |

estimates table can do much more; see [R] estimates table. etable also produces tables from stored estimates, customizes the tables, and exports to a variety of formats, including Word, PDF, ${ }^{L A} T_{E} X$, Excel, and HTML; see [R] etable. Also see [R] estimates stats. estimates stats works like estimates table but produces model comparisons in terms of BIC and AIC.

## Jargon

You know that if you fit a model, say, by typing
. regress mpg weight displacement
then you can later replay the results by typing
. regress
and you can do tests and calculate other postestimation statistics by typing

```
. test displacement==0
. estat vif
. predict mpghat
```

As a result, we often refer to the estimation results or the current estimation results or the most recent estimation results or the last estimation results or the estimation results in memory.

With estimates store and estimates restore, you can have many estimation results in memory. One set of those-the set most recently estimated or the set most recently restored-is the current or active estimation results, which you can replay, which you can test, or from which you can calculate postestimation statistics.

Current and active are the two words we will use interchangeably from now on.

## Also see

[P] _estimates - Manage estimation results

## Title

estimates describe - Describe estimation results

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Option | Remarks and examples | Stored results | Also see |

## Description

estimates describe describes the current (active) estimates. Reported are the command line that produced the estimates, any title that was set by estimates title (see [R] estimates title), and any notes that were added by estimates notes (see [R] estimates notes).
estimates describe name does the same but reports results for estimates stored by estimates store (see [R] estimates store).
estimates describe using filename does the same but reports results for estimates saved by estimates save (see [R] estimates save). If filename contains multiple sets of estimates (saved in it by estimates save, append), the number of sets of estimates is also reported. If filename is specified without an extension, .ster is assumed.

## Quick start

Describe current estimation results

```
estimates describe
```

Describe estimation results stored in m 1
estimates describe m1
Describe estimates from first model saved in mymodels.ster
estimates describe using mymodels
Same as above
estimates describe using mymodels, number(1)
Describe estimates from third model saved in mymodels.ster
estimates describe using mymodels, number(3)

## Menu

Statistics $>$ Postestimation

## Syntax

estimates describe
estimates describe name
estimates describe using filename [, number(\#)]
collect is allowed; see [U] 11.1.10 Prefix commands.

## Option

number (\#) specifies that the \#th set of estimation results from filename be described. This assumes that multiple sets of estimation results have been saved in filename by estimates save, append. The default is number (1).

## Remarks and examples

estimates describe can be used to describe the estimation results currently in memory,

```
. estimates describe
Estimation results produced by
. regress mpg weight displ if foreign
```

or to describe results saved by estimates save in a .ster file:

```
. estimates describe using final
Estimation results "Final results" saved on 12apr2022 14:20, produced by
    . logistic myopic age sex drug1 drug2 if complete==1
Notes:
    1. Used file patient.dta
    2. "datasignature myopic age sex drug1 drug2 if complete==1"
        reports 148:5(58763):2252897466:3722318443
    3. must be reviewed by rgg
```


## > Example 1

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. regress mpg weight displ if foreign
(output omitted)
. estimates notes: file 'c(filename)'
. datasignature
    74:12(71728):3831085005:1395876116
. estimates notes: datasignature report 'r(datasignature)'
. estimates save foreign
file foreign.ster saved
. regress mpg weight displ if !foreign
    (output omitted)
```

```
estimates describe using foreign
Estimation results saved on 23mar2023 17:29, produced by
    . regress mpg weight displ if foreign
Notes:
    1. file https://www.stata-press.com/data/r18/auto.dta
    2. datasignature report 74:12(71728):3831085005:1395876116
```


## Stored results

estimates describe and estimates describe name store the following in $r()$ :
Macros
r(title) title
$r$ (cmdline) original command line
estimates describe using filename stores the above and the following in r() :
Scalars
$r$ (datetime) $\%$ tc value of date/time file saved
$r$ (nestresults) number of sets of estimation results in file

## Also see

[R] estimates - Save and manipulate estimation results

## Title

estimates for - Repeat postestimation command across models
Description Quick start Syntax Options Remarks and examples
Also see

## Description

estimates for performs postestimation_command on each estimation result specified.

## Quick start

Test for no effect of continuous covariate x 1 in stored estimates m 1 and m 2 estimates for m 1 m 2 : test $\mathrm{x} 1==0$

Same as above, but test interaction of binary covariate a and x 1
estimates for m1 m2: test 0.a\#c.x1==1.a\#c.x1
Linear combination of coefficients of x 1 and x 2 in all stored estimates
estimates for _all: lincom x1 + x2
Tables of margins for each level of a and confidence intervals using estimates m1 and m2
estimates for m1 m2: pwcompare i.a, cimargins

## Syntax

estimates for namelist [, options]: postestimation_command
where namelist is a name, a list of names,_all, or *. A name may be ., meaning the current (active) estimates. _all and $*$ mean the same thing.
options
Description
noheader
do not display title
nostop
do not stop if command fails

## Options

noheader suppresses the display of the header as postestimation_command is executed each time. nostop specifies that execution of postestimation_command is to be performed on the remaining models even if it fails on some.

## Remarks and examples

In the example that follows, we fit a model two different ways, store the results, and then use estimates for to perform the same test on both of them:

## $>$ Example 1

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. generate gpm = 1/mpg
. regress gpm i.foreign i.foreign#c.weight displ
    (output omitted)
. estimates store reg
. qreg gpm i.foreign i.foreign#c.weight displ
    (output omitted)
. estimates store qreg
. estimates for reg qreg: test 0.foreign#c.weight==1.foreign#c.weight
```


## Model reg

```
( 1) Ob.foreign#c.weight - 1.foreign#c.weight = 0
    F( 1, 69) = 4.87
        Prob > F = 0.0307
```

Model qreg
( 1) Ob.foreign\#c.weight - 1.foreign\#c.weight $=0$
$\mathrm{F}(1,69)=0.03$
Prob $>\mathrm{F}=0.8554$

## Also see

[R] estimates - Save and manipulate estimation results

## Title

estimates notes - Add notes to estimation results

Description Quick start Syntax Remarks and examples Also see

## Description

estimates notes: text adds a note to the current (active) estimation results. estimates notes and estimates notes list list the current notes.
estimates notes drop in noterange eliminates the specified notes.

## Quick start

Add "My note" to current estimation results estimates notes: My note

List all notes for current estimation results
estimates notes list
Same as above
estimates notes
Drop notes 1 to 3 from current estimation results
estimates notes drop in $1 / 3$
Drop last note applied to current estimation results
estimates notes drop in 1

## Syntax

```
estimates notes: text
estimates notes
estimates notes list [in noterange]
    estimates notes drop in noterange
```

where noterange is \# or \#/\# and where \# may be a number, the letter f (meaning first), or the letter $l$ (meaning last).

## Remarks and examples

After adding or removing notes, if estimates have been stored, do not forget to store them again. If estimates have been saved, do not forget to save them again.

Notes are most useful when you intend to save estimation results in a file; see $[R]$ estimates save. For instance, after fitting a model, you might type

```
. estimates note: I think these are final
```

. estimates save lock2
and later when going through your files, you could type

```
. estimates use lock2
. estimates notes
    1. I think these are final
```

Up to 9,999 notes can be attached to estimation results. If estimation results are important, we recommend that you add a note identifying the .dta dataset you used. The best way to do that is to type

```
. estimates notes: file 'c(filename)'
```

because 'c(filename)' will expand to include not just the name of the file but also its full path; see $[\mathrm{P}]$ creturn.

If estimation results took a long time to estimate-say, they were produced by cmmprobit or gllamm (see [CM] cmmprobit and http://www.gllamm.org) -it is also a good idea to add a data signature. A data signature takes less time to compute than reestimation when you need proof that you really have the right dataset. The easy way to do that is to type

```
. datasignature
    74:12(71728) : 3831085005:1395876116
. estimates notes: datasignature reports 'r(datasignature)'
```

Now when you ask to see the notes, you will see

```
estimates notes
```

1. I think these are final
2. file $C: \backslash p r o j e c t \backslash o n e \backslash p a t 4 . d t a$
3. datasignature reports 74:12(71728):3831085005:1395876116

## See [D] datasignature.

Notes need not be positive. You might set a note to be, "I need to check that age is defined correctly."
> Example 1

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. regress mpg weight displ if foreign
(output omitted)
. estimates notes: file 'c(filename)'
. datasignature
    74:12(71728):3831085005:1395876116
. estimates notes: datasignature report 'r(datasignature)'
. estimates save foreign
file foreign.ster saved
```

```
estimates notes list in 1/2
    1. file https://www.stata-press.com/data/r18/auto.dta
    2. datasignature report 74:12(71728):3831085005:1395876116
    estimates notes drop in 2
    (1 note dropped)
    estimates notes
    1. file https://www.stata-press.com/data/r18/auto.dta
```


## Also see

[R] estimates - Save and manipulate estimation results

## Title

estimates replay - Redisplay estimation results
Description Quick start Menu Syntax Remarks and examples
Also see

## Description

estimates replay redisplays the current (active) estimation results, just as typing the name of the estimation command would do.
estimates replay namelist redisplays each specified estimation result. The active estimation results are left unchanged.

## Quick start

Redisplay current estimation results

```
estimates replay
```

Redisplay estimation results stored as m1
estimates replay m1
Redisplay all stored estimation results

```
estimates replay *
```

Same as above

```
estimates replay _all
```


## Menu

Statistics $>$ Postestimation

## Syntax

```
estimates replay
estimates replay namelist
```

where namelist is a name, a list of names, _all, or *. A name may be ., meaning the current (active) estimates. _all and $*$ mean the same thing.

## Remarks and examples

In the example that follows, we fit a model two different ways, store the results, use estimates for to perform the same test on both of them, and then replay the results:

## Example 1

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. generate gpm = 1/mpg
. regress gpm i.foreign i.foreign#c.weight displ
    (output omitted)
. estimates store reg
. qreg gpm i.foreign i.foreign#c.weight displ
    (output omitted)
. estimates store qreg
. estimates for reg qreg: test 0.foreign#c.weight==1.foreign#c.weight
```


## Model reg

```
( 1) Ob.foreign#c.weight - 1.foreign#c.weight = 0
    F( 1, 69) = 4.87
            Prob > F = 0.0307
```

Model qreg
( 1) Ob.foreign\#c.weight - 1.foreign\#c.weight $=0$
$\mathrm{F}(1,69)=0.03$
Prob > F = 0.8554
. estimates replay

Model qreg

. estimates replay reg

Model reg


## Also see

[R] estimates - Save and manipulate estimation results

## Title

estimates save - Save and use estimation results

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Also see |

## Description

estimates save filename saves the current (active) estimation results in filename.
estimates use filename loads the results saved in filename into the current (active) estimation results.

In both cases, if filename is specified without an extension, .ster is assumed.
estimates esample: (note the colon) resets e(sample). After estimates use filename, $e$ (sample) is set to contain 0 , meaning that none of the observations currently in memory was used in obtaining the estimates.
estimates esample (without a colon) displays how e(sample) is currently set.

## Quick start

Save current estimation results to mymodels.ster estimates save mymodels

Add current estimation results to existing file mymodels.ster estimates save mymodels, append

Make active third estimation results saved in mymodels.ster estimates use mymodels, number (3)

Reset e(sample) to original estimation sample assuming command regress y x1 x2 estimates esample: y x1 x2

## Menu

Statistics $>$ Postestimation

## Syntax

```
estimates save filename [, append replace]
estimates use filename [, number(\#)]
estimates esample: [varlist] [if] [in] [weight]
    [, replace stringvars(varlist) zeroweight]
estimates esample
```

collect is allowed with estimates esample (without a colon); see [U] 11.1.10 Prefix commands.

## Options

append, used with estimates save, specifies that results be appended to an existing file. If the file does not already exist, a new file is created.
replace, used with estimates save, specifies that filename can be replaced if it already exists.
number(\#), used with estimates use, specifies that the \#th set of estimation results from filename be loaded. This assumes that multiple sets of estimation results have been saved in filename by estimates save, append. The default is number (1).
replace, used with estimates esample:, specifies that e(sample) can be replaced even if it is already set.
stringvars(varlist), used with estimates esample:, specifies string variables. Observations containing variables that contain "" will be omitted from e(sample).
zeroweight, used with estimates esample:, specifies that observations with zero weights are to be included in e(sample).

## Remarks and examples

See [R] estimates for an overview of the estimates commands.
For a description of estimates save and estimates use, see Saving and using estimation results in $[\mathrm{R}]$ estimates.

The rest of this entry concerns e(sample).
Remarks are presented under the following headings:

```
Setting e(sample)
Resetting e(sample)
Determining who set e(sample)
```


## Setting e(sample)

After estimates use filename, the situation is nearly identical to what it was immediately after you fit the model. The one difference is that e (sample) is set to 0 .
$e$ (sample) is Stata's function to mark which observations among those currently in memory were used in producing the estimates. For instance, you might type

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. regress mpg weight displ if foreign
(output omitted)
. summarize mpg if e(sample)
(output omitted)
```

and summarize would report the summary statistics for the observations regress in fact used, which would exclude not only observations for which foreign $=0$ but also any observations for which mpg, weight, or displ was missing.

If you saved the above estimation results and then reloaded them, however, summarize mpg if e(sample) would produce

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | :---: | :---: | :---: | :---: | :---: |
| $m p g$ | 0 |  |  |  |  |

Stata thinks that none of these observations was used in producing the estimates currently loaded.
What else could Stata think? When you estimates use filename, you do not have to have the original data in memory. Even if you do have data in memory that look like the original data, they might not be. Setting e (sample) to 0 is the safe thing to do. There are some postestimation statistics, for instance, that are appropriate only when calculated on the estimation sample. Setting e (sample) to 0 ensures that if you ask for one of them, you will get back a null result.

We recommend that you leave e(sample) set to 0 . But what if you really need to calculate that postestimation statistic? Well, you can get it, but you are going to be responsible for setting e(sample) correctly. Here we just happen to know that all the observations with foreign $=1$ were used, so we can type

```
. estimates esample: if foreign
```

If all the observations had been used, we could simply type

```
. estimates esample:
```

The safe thing to do, however, is to look at the estimation command-estimates describe will show it to you-and then type

```
. estimates esample: mpg weight displ if foreign
```

We include all observations with foreign $=1$, excluding any with missing values in the mpg , weight, or displ variable, that are to be treated as the estimation sample.

## Resetting e(sample)

estimates esample: will allow you to not only set but also reset e(sample). If e(sample) has already been set (say that you just fit the model) and you try to set it, you will see

```
. estimates esample: mpg weight displ if foreign
no; e(sample) already set
r(322);
```

Here you can specify the replace option:

```
. estimates esample: mpg weight displ if foreign, replace
```

We do not recommend resetting e (sample), but the situation can arise where you need to. Imagine that you estimates use filename, you set e(sample), and then you realize that you set it wrong. Here you would want to reset it.

## Determining who set e(sample)

estimates esample without a colon will report whether and how e(sample) was set. You might see

```
. estimates esample
```

$e$ (sample) set by estimation command
or

```
. estimates esample
e(sample) set by user
```

or
estimates esample
e(sample) not set (0 assumed)

## Stored results

estimates esample without the colon saves macro $r$ (who), which will contain cmd, user, or zero'd.

## Also see

[R] estimates - Save and manipulate estimation results

## Title

estimates selected - Show selected coefficients

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Also see |

## Description

estimates selected reports on coefficients from one or more estimation results. It creates a table that indicates which coefficients were estimated in each model and, if requested, reports the value of those coefficients. The results may be sorted based on the values of the estimated coefficients or based on variable names.

## Quick start

Compare coefficients for stored estimates m 1 and m 2

```
estimates selected m1 m2
```

Same as above, but display a $u$ for covariates that are not specified in the model

```
estimates selected m1 m2, display(u)
```

Compare stored estimates 11 and 12, and order the rows by absolute values of the coefficients estimates selected 11 12, sort(coef)

## Menu

Statistics $>$ Postestimation

## Syntax

estimates selected [namelist] [, options]
namelist is the name given to previously stored estimation results, a list of names, _all, or *. A name may be ., meaning the current (active) estimates. _all and $*$ mean the same thing.
options
Description
Main
display(info) display info; default is display(x)
sort(on)
Reporting
noabbrev
display_options
sort rows in order of on
do not abbreviate variable names
control row spacing, line width, and display of omitted variables and base and empty cells

[^22]
## Options

$\qquad$
display (info) specifies what to display in the table. The default is display(x).
Blank cells in the table indicate that the corresponding covariate does not have a fitted value. For some covariates without fitted values, a code that indicates the reason for omission is reported in the table. Base levels of factors and interactions are coded with the letter b. Empty levels of factors and interactions are coded with the letter e. Covariates omitted because of collinearity are coded with the letter o.
display ( $x$ ) displays an $x$ in the cell of the table where a covariate has a fitted value. This is the default.
display (u) is the same as display ( x ), except that when a covariate was not specified in the model, $u$ (for unavailable) is displayed instead of a blank cell.
display (coef [, eform format $(\% f m t)]$ ) specifies that coefficient values be displayed in the table.
eform displays coefficients in exponentiated form. For each coefficient, $e^{b}$ rather than $b$ is displayed. This option can be used to display odds ratios, incidence-rate ratios, relative-risk ratios, hazard ratios, and subhazard ratios after the appropriate estimation command.
format ( $\%$ frit ) specifies the display format for the coefficients in the table. The default is format (\%9.0g).
sort (on) specifies how to sort the rows of the table. By default, coefficients are displayed in the order in which they appear in the estimation results.
sort (none) specifies that the rows are not sorted. This is the default. The order of the coefficients is taken from their order in $e(b)$.
sort (names) orders rows alphabetically by the variable names of the covariates. In the case of factor variables, main effects and nonfactor variables are displayed first in alphabetical order; then all two-way interactions are displayed in alphabetical order, then all three-way interactions, and so on.
sort (coef) orders rows in descending order by the absolute values of the coefficients. When results from two or more estimation results are displayed, results are sorted first by the ordering for the first estimation result with rows representing coefficients not in the first estimation result last. Within the rows representing coefficients not in the first estimation result, the rows are sorted by the ordering for the second estimation result with rows representing coefficients not in the first or second estimation results last. And so on.

Reporting
noabbrev prevents variable names from being abbreviated in the row titles of the table. Long variable names are split onto multiple lines if they do not fit.
display_options: vsquish, fvwrap(\#), fvwrapon(style), and nolstretch; see [R] Estimation options.

## Remarks and examples

estimates selected produces a table based on estimated coefficients from one or more models. Results can be sorted by the values of the estimated coefficients or by variable names. Multiple
models are displayed side-by-side, making it easy to compare which covariates were included in each model or to compare the estimated values of those coefficients.
> Example 1: Compare coefficients across models
To compare coefficients from two or more models, we first need to store the results of each model using estimates store; see [R] estimates store. Then we use estimates selected to obtain a table comparing which coefficients were estimated in each model. By default, the table reports an x for each coefficient that was estimated.

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. regress mpg gear turn
    (output omitted)
. estimates store small
. regress mpg gear turn length
(output omitted)
. estimates store large
. estimates selected small large
```

|  | small | large |
| ---: | :---: | :---: |
| gear_ratio | x | x |
| turn | x | x |
| length |  | x |
| _cons | x | x |

Legend:
b - base level
e - empty cell
o - omitted
x - estimated

There is only one difference in the two regress commands; the large model includes length as a covariate, but the small model does not. Therefore, the above table displays x's for all but the length coefficient in the small model. When working with larger models or with more models, this type of table makes it easy to spot the differences in the model specification.

By default, the rows of the table in the order that covariates appear in the models. To sort the results on the covariate names, we add the sort (names) option.

```
. estimates selected small large, sort(names)
```

|  | small | large |
| ---: | :---: | :---: |
| gear_ratio | x | x |
| length |  | x |
| turn | x | x |
| _cons | x | x |

```
Legend:
    b - base level
    e - empty cell
    o - omitted
    x - estimated
```

We could have instead sorted on the values of the estimated coefficients by including the sort (coef) option.

To display the coefficients values instead of the x's, we add the display(coef) option.
. estimates selected small large, sort(names) display(coef)

|  | small | large |
| ---: | ---: | ---: |
| gear_ratio | 3.032884 | 1.35666 |
| length |  | -.1665899 |
| turn | -.7330502 | -.1219185 |
| _cons | 41.21801 | 53.3487 |

Legend:
b - base level
e - empty cell
o - omitted

## Stored results

estimates selected stores the following in $r()$ :
Macros
$r$ (names) names of results used
Matrices
r(coef) matrix $M: n \times m$
$M[i, j]=i$ th coefficient estimate for model $j ; i=1, \ldots, n ; j=1, \ldots, m$

## Also see

[R] estimates - Save and manipulate estimation results
[LASSO] lassocoef - Display coefficients after lasso estimation results

## Title

estimates stats - Model-selection statistics

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| Also see |  |  |  |

## Description

estimates stats reports model-selection statistics, including the Akaike (AIC), consistent Akaike's (CAIC), corrected Akaike's (AICc), and Schwarz's Bayesian (BIC) information criteria. These measures are appropriate for maximum likelihood models.

If estimates stats is used for a non-likelihood-based model, such as qreg, missing values are reported.

## Quick start

Display AIC and BIC for last estimation command estimates stats

Display AIC and BIC for stored estimates m1 and m2
estimates stats m1 m2
Same as above, but display CAIC instead of AIC
estimates stats m1 m2, aicconsistent
Same as above, but display AICc instead of AIC
estimates stats m 1 m 2 , aiccorrected
Same as above, but display all AIC, BIC, AICc, and CAIC
estimates stats m 1 m 2 , all
Specify $N=1,000$ for calculation of BIC
estimates stats, $\mathrm{n}(1000)$
Same as above, but use 10 degrees of freedom
estimates stats, $\mathrm{n}(1000) \mathrm{df}(10)$

## Menu

Statistics $>$ Postestimation

## Syntax

estimates stats [namelist] [, options]
where namelist is a name, a list of names,__all, or *. A name may be ., meaning the current (active) estimates. _all and $*$ mean the same thing.
options

| aiccorrected | report AICc instead of AIC <br> report CAIC instead of AIC |
| :--- | :--- |
| aicconsistent | report all four information criteria: AIC, BIC, AICc, and CAIC <br> all <br> $\mathrm{n}(\#)$ |
| $\mathrm{df}(\#)$ | specify $N$ to be used in calculating BIC, AICc, and CAIC; <br> secify degrees of freedom $k$ to be used in calculating AIC, BIC, <br> AICc, and CAIC |
| icdetail | produce a table showing the type of $N$ used in BIC, AICc, and <br>  <br> CAIC calculation |

collect is allowed; see [U] 11.1.10 Prefix commands.

## Options

aiccorrected specifies that AICc be computed instead of AIC. This information criterion is a second-order approximation and is recommended for small sample sizes.

Only one of aiccorrected, aicconsistent, or all is allowed.
aicconsistent specifies that CAIC be computed instead of AIC. This information criterion is a consistent version of AIC; that is, the probability of selecting the "true model" approaches 1 as sample size increases.
Only one of aicconsistent, aiccorrected, or all is allowed.
all produces a table showing all four information criteria: AIC, BIC, AICc, and CAIC.
Only one of all, aiccorrected, or aicconsistent is allowed.
$\mathrm{n}(\#)$ specifies $N$ to be used in calculating BIC, AICc, and CAIC; see [R] IC note.
$\mathrm{df}(\#)$ specifies degrees of freedom $k$ to be used in calculating AIC, BIC, AICc, and CAIC. By default, $k$ is the number of estimated parameters.
icdetail produces a table showing the type of $N$ used in BIC, AICc, and CAIC calculations. Most estimation commands use the number of observations in the estimation sample for the information criteria. For some models, however, other types of $N$, such as the number of cases in choice models, should be used. When the default table of estimates stats contains more than one type of $N$, specifying icdetail allows you to see the different types of $N$ used for BIC, AICc, and CAIC.

## Remarks and examples

If you type estimates stats without arguments, a table for the most recent estimation results will be shown:

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. logistic foreign mpg weight displ
    (output omitted)
. estimates stats
Akaike's information criterion and Bayesian information criterion
```

| Model | N | ll (null) | ll(model) | df | AIC | BIC |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| . | 74 | -45.03321 | -20.59083 | 4 | 49.18167 | 58.39793 |

Note: BIC uses $N=$ number of observations. See [R] IC note.
Regarding the note at the bottom of the table, $N$ is an ingredient in the calculation of BIC, AICc, and CAIC; see $[R]$ IC note. The note changes if you specify the $n()$ option, which tells estimates stats what $N$ to use. By default, $N$ is the number of observations used in fitting the model.

Regarding the table itself, 11 (null) is the log likelihood for the constant-only model, 11 (model) is the $\log$ likelihood for the model, df is the number of degrees of freedom, and AIC and BIC are the Akaike and Bayesian information criteria, respectively.

Models with smaller values of an information criterion are considered preferable.
estimates stats can compare estimation results:

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. logistic foreign mpg weight displ
(output omitted)
. estimates store full
. logistic foreign mpg weight
(output omitted)
. estimates store sub
. estimates stats full sub
Akaike's information criterion and Bayesian information criterion
```

| Model | N | ll(null) | ll(model) | df | AIC | BIC |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| full | 74 | -45.03321 | -20.59083 | 4 | 49.18167 | 58.39793 |
| sub | 74 | -45.03321 | -27.17516 | 3 | 60.35031 | 67.26251 |

Note: BIC uses $N=$ number of observations. See [R] IC note.

You can use option all to compare the models based on all four information criteria
. estimates stats full sub, all
Information criteria

| Model | N | ll(null) | ll(model) | df |
| ---: | ---: | ---: | ---: | ---: |
| full | 74 | -45.03321 | -20.59083 | 4 |
| sub | 74 | -45.03321 | -27.17516 | 3 |

Note: BIC, AICc, and CAIC use $\mathrm{N}=$ number of observations. See [R] IC note.

| Model | AIC | BIC | AICc | CAIC |
| ---: | ---: | ---: | ---: | ---: |
| full | 49.18167 | 58.39793 | 49.76138 | 62.39793 |
| sub | 60.35031 | 67.26251 | 60.69317 | 70.26251 |

Legend: AIC is Akaike's information criterion.
BIC is Bayesian information criterion.
AICc is corrected Akaike's information criterion.
CAIC is consistent Akaike's information criterion.
All four information criteria suggest that the full model is preferable.

## Stored results

estimates stats stores the following in r() :
Matrices
$r(S) \quad$ matrix with columns (N, ll(null), 11 (model), df, and information criteria) and rows corresponding to models in the table

## Methods and formulas

See [R] IC note.

## Also see

[R] estimates - Save and manipulate estimation results

## Title

estimates store - Store and restore estimation results

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Option | Remarks and examples | Stored results | References |
| Also see |  |  |  |

## Description

estimates store name stores the current (active) estimation results under the name name.
estimates restore name loads the results stored under name into the current (active) estimation results.
estimates query tells you whether the current (active) estimates have been stored and, if so, the name.
estimates dir displays a list of the stored estimates.
estimates drop namelist drops the specified stored estimation results.
estimates clear drops all stored estimation results.
estimates clear, estimates drop _all, and estimates drop $*$ do the same thing. estimates drop and estimates clear do not eliminate the current (active) estimation results.

## Quick start

Store estimation results as m 1 for use later in the same session estimates store m1

Restore estimation results from m 2
estimates restore m2
Find out whether the current estimation results have been stored
estimates query
Display table of information about all stored results
estimates dir
Drop stored estimation results m3
estimates drop m3
Drop all stored results
estimates clear

## Menu

Statistics $>$ Postestimation

## Syntax

```
estimates store name [, nocopy]
    estimates restore name
    estimates query
    estimates dir [namelist]
    estimates drop namelist
    estimates clear
```

where namelist is a name, a list of names, _all, or *. _all and * mean the same thing.
collect is allowed with estimates dir; see [U] 11.1.10 Prefix commands.

## Option

nocopy, used with estimates store, specifies that the current (active) estimation results are to be moved into name rather than copied. Typing

- estimates store hold, nocopy
is the same as typing
. estimates store hold
. ereturn clear
except that the former is faster. The nocopy option is sometimes used by programmers.


## Remarks and examples

estimates store stores estimation results in memory so that you can access them later.
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. regress mpg weight displ (output omitted)
. estimates store myreg
. . . . you do other things, including fitting other models . . .

- estimates restore myreg
. regress
(same output shown again)
After estimates restore myreg, things are once again just as they were, estimationwise, just after you typed regress mpg weight displ.
estimates store stores results in memory. When you exit Stata, those stored results vanish. If you wish to make a permanent copy of your estimation results, see [R] estimates save.

The purpose of making copies in memory is 1) so that you can quickly switch between them and 2) so that you can make tables comparing estimation results. Concerning the latter, see [R] estimates table, $[R]$ etable, and $[R]$ estimates stats.

## Stored results

estimates dir stores the following in r() :
Macros
$r$ (names) names of stored results

## References

Jann, B. 2005. Making regression tables from stored estimates. Stata Journal 5: 288-308.
——. 2007. Making regression tables simplified. Stata Journal 7: 227-244.

## Also see

[R] estimates - Save and manipulate estimation results
[LASSO] estimates store - Saving and restoring estimates in memory and on disk

## Title

estimates table - Compare estimation results

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | References |
| Also see |  |  |  |

## Description

estimates table organizes estimation results from one or more models in a single formatted table.

## Quick start

Display a table of coefficients for stored estimates m1 and m2
estimates table m1 m2
Same as above, but include standard errors
estimates table m1 m2, se
Same as above, but display coefficients and standard errors to 2 decimal places
estimates table $\mathrm{m} 1 \mathrm{~m} 2, \mathrm{~b}(\% 7.2 \mathrm{f}) \mathrm{se}(\% 7.2 \mathrm{f})$
Same as above, but include $p$-values displayed to 3 decimal places
estimates table m1 m2, b(\%7.2f) se(\%7.2f) p(\%4.3f)
Table of coefficients for m 1 and m 2 with sample size and adjusted $R^{2}$
estimates table m1 m2, stats(N r2_a)
Same as above, but replace variable names with labels
estimates table m1 m2, stats (N r2_a) varlabel
Table of coefficients with stars to denote significance
estimates table m1 m2, star
Display coefficients in exponentiated form
estimates table m3 m4, eform
Display only a subset of variables and reorder variables in table
estimates table m1 m2, keep(v2 v1 v3 _cons)

## Menu

Statistics $>$ Postestimation

## Syntax

estimates table [namelist] [, options]
namelist is the name given to previously stored estimation results, a list of names, _all, or *. A name may be ., meaning the current (active) estimates. _all and $*$ mean the same thing.
options
Main
stats (scalarlist) report scalarlist in table
star[(\#1 \#2 \#3)]
Options
keep(coeflist)
drop(coeflist)
equations(matchlist)
Numerical formats
$\mathrm{b}[(\% f m t)]$
se $[(\% f m t)]$
$\mathrm{t}[(\% f m t)]$
$\mathrm{p}[(\% f m t)]$
stfmt ( $\%$ fint)
General format
varwidth(\#)
modelwidth(\#)
eform
varlabel
newpanel
style(oneline)
style(columns)
style(noline)
coded display compact table
Reporting
display_options
title(string)
Description
use stars to denote significance
report coefficients in order specified
omit specified coefficients from table match equations of models as specified
report $t$ or $z$ and use optional format
report $p$-values and use optional format
how to format scalar statistics
suppress all vertical lines base and empty cells
title for table
how to format coefficients, which are always reported
report standard errors and use optional format
use \# characters to display variable names and statistics
use \# characters to display model names
display coefficients in exponentiated form
display variable labels rather than variable names
display statistics in separate table from coefficients
put vertical line after variable names; the default
put vertical line separating every column
control row spacing, line width, and display of omitted variables and
collect is allowed; see [U] 11.1.10 Prefix commands.
title() does not appear in the dialog box.

## Options

## Main

stats (scalarlist) specifies a list of any of or all the names of scalars stored in e() to be displayed in the table. scalarlist may also contain the following:

| aic | Akaike's information criterion |
| :--- | :--- |
| bic | Schwarz's Bayesian information criterion |
| rank | rank of $e(V)$ (\# of free parameters in model) |

The specified statistics do not have to be available for all estimation results being displayed.
For example, stats (N ll chi2 aic) specifies that e(N), e(ll), e(chi2), and AIC be included. In Stata, e(N) records the number of observations; e(ll), the log likelihood; and e(chi2), the $\chi^{2}$ test that all coefficients in the first equation of the model are equal to zero.
star and star (\#1 \#2 \#3) specify that stars (asterisks) are to be used to mark significance. The second syntax specifies the significance for one, two, and three stars. If you specify simply star, that is equivalent to specifying star(.05 .01 .001), which means one star $\left(^{*}\right.$ ) if $p<0.05$, two stars $(* *)$ if $p<0.01$, and three stars $(* * *)$ if $p<0.001$.
The star and star() options may not be combined with the se, $t$, or p option.

## Options

keep (coeflist) and drop (coeflist) are alternatives; they specify coefficients to be included or omitted from the table. The default is to display all coefficients.
If keep () is specified, it specifies not only the coefficients to be included but also the order in which they appear.

A coeflist is a list of coefficient names, each name of which may be simple (for example, price), an equation name followed by a colon (for example, mean:), or a full name (for example, mean:price). Names are separated from each other by blanks.
When full names are not specified, all coefficients that match the partial specification are included. For instance, drop (_cons) would omit _cons for all equations.
equations (matchlist) specifies how the equations of the models in namelist are to be matched. The default is to match equations by name. Matching by name usually works well when all results were fit by the same estimation command. When you are comparing results from different estimation commands, however, specifying equations() may be necessary.
The most common usage is equations(1), which indicates that all first equations are to be matched into one equation named \#1.
matchlist has the syntax

$$
\operatorname{term}[, \operatorname{term} \ldots]
$$

term is

$$
\begin{array}{ll}
{[\text { eqname }=] \#: \# \ldots: \#} & (\text { syntax } 1) \\
{[\text { eqname }=] \#} & (\text { syntax } 2)
\end{array}
$$

In syntax 1, each \# is a number or a period (.). If a number, it specifies the position of the equation in the corresponding model; 1:3:1 would indicate that equation 1 in the first model matches equation 3 in the second, which matches equation 1 in the third. A period indicates that there
is no corresponding equation in the model; 1:.:1 indicates that equation 1 in the first matches equation 1 in the third.
In syntax 2 , you specify just one number, say, 1 or 2 , and that is shorthand for $1: 1 \ldots: 1$ or $2: 2 \ldots: 2$, meaning that equation 1 matches across all models specified or that equation 2 matches across all models specified.

Now that you can specify a term, you can put that together into a matchlist by separating one term from the other by commas. In what follows, we will assume that three names were specified,

```
. estimates table alpha beta gamma, ...
```

equations(1) is equivalent to equations(1:1:1); we would be saying that the first equations match across the board.
equations ( $1: .: 1$ ) would specify that equation 1 matches in models alpha and gamma but that there is nothing corresponding in model beta.
equations $(1,2)$ is equivalent to equations $(1: 1: 1,2: 2: 2)$. We would be saying that the first equations match across the board and so do the second equations.
equations ( $1,2: .: 2$ ) would specify that the first equations match across the board, that the second equations match for models alpha and gamma, and that there is nothing equivalent to equation 2 in model beta.

If equations() is specified, equations not matched by position are matched by name.
$\mathrm{b}(\% \mathrm{fmt})$ specifies how the coefficients are to be displayed. You might specify $\mathrm{b}(\% 9.2 \mathrm{f})$ to make decimal points line up. There is also a b option, which specifies that coefficients are to be displayed, but that is just included for consistency with the $s e$, $t$, and $p$ options. Coefficients are always displayed.
se, t , and p specify that standard errors, $t$ or $z$ statistics, and $p$-values are to be displayed. The default is not to display them. se (\%fmt), $\mathrm{t}(\% f m t)$, and $\mathrm{p}(\% f m t)$ specify that each is to be displayed and specifies the display format to be used to format them.
stfmt (\% fint) specifies the format for displaying the scalar statistics included by the stats() option.

General format
varwidth (\#) specifies the number of character positions used to display the names of the variables and statistics. The default is 12 .
modelwidth(\#) specifies the number of character positions used to display the names of the models. The default is 12 .
eform displays coefficients in exponentiated form. For each coefficient, $\exp (\boldsymbol{\beta})$ rather than $\boldsymbol{\beta}$ is displayed, and standard errors are transformed appropriately. Display of the intercept, if any, is suppressed.
varlabel specifies that variable labels be displayed instead of variable names.
newpanel specifies that the statistics be displayed in a table separated by a blank line from the table with coefficients rather than in the style of another equation in the table of coefficients.
style(stylespec) specifies the style of the coefficient table.
style(oneline) specifies that a vertical line be displayed after the variables but not between the models. This is the default.
style(columns) specifies that vertical lines be displayed after each column. style(noline) specifies that no vertical lines be displayed.
coded specifies that a compact table be displayed. This format is especially useful for comparing variables that are included in a large collection of models.
$\qquad$ Reporting
display_options: noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), and fvwrapon(style); see [R] Estimation options.

The following option is available with estimates table but is not shown in the dialog box: title (string) specifies the title to appear above the table.

## Remarks and examples

estimates table lets you format estimation results and organize results from multiple models in a single table. You achieve this by combining estimates table with estimates store; see $[R]$ estimates store. When combined with putdocx or putpdf, estimates table lets you create customized tables of results in . docx or .pdf format.

Alternatively, you can use etable to create a table with the results stored with estimates store. The advantage of using etable is that you can add notes to your table and export it to a variety of file types, such as HTML and LATEX.

## > Example 1: Creating the default table

If you type estimates table without arguments, a table of the most recent estimation results will be shown:

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. regress mpg weight
    (output omitted)
. regress mpg weight displ
(output omitted)
. estimates table
\begin{tabular}{r|r}
\hline Variable & Active \\
\hline weight & -.00656711 \\
displacement & .00528078 \\
_cons & 40.084522 \\
\hline
\end{tabular}
```

In the above output table, only the results from the second regress command are displayed because they are the current (active) estimates.

By default, estimates table displays only coefficient estimates. You can request additional model statistics by specifying the stats() option. Estimation commands store e-class results. To see a list of available results from the last estimation command that can be specified in the stats() option, you can use the ereturn list command.

## > Example 2: Creating a table with multiple models

If we want to display more than one set of estimation results in a single table, we can use estimates store to store each set of estimation results in memory so that they can be accessed later by estimates table. This is one of the primary uses of estimates table-comparing estimation results:

```
. regress mpg weight displ
(output omitted)
. estimates store base
. regress mpg weight displ i.foreign
(output omitted)
. estimates store alt
. qreg mpg weight displ i.foreign
(output omitted)
. estimates store qreg
. estimates table base alt qreg, stats(r2)
```

| Variable | base | alt | qreg |
| ---: | :---: | ---: | ---: |
| weight <br> foreign <br> Foreign | -.00656711 | -.00677449 | -.00595056 |
| .00528078 | .00192865 | .00018552 |  |
| _cons | 40.084522 | 41.847949 | 39.213348 |
| $r 2$ | .6529307 | .66287957 |  |

estimates table automatically lines up the point estimates of coefficients on covariates that are common across each model. The stats (r2) option specifies that the coefficient of determination ( $R^{2}$ ) be placed under the models for which it is computed.

The foreign variable was included in the models alt and qreg as a factor variable. Because the values of foreign are labeled, the value labels are displayed by default. See [D] label for information about managing value labels.

## > Example 3: Creating and exporting a formatted table

We can add estimates of the standard errors to the table and format them along with the original coefficient estimates by using the b() and se() options.

As shown in example 2, the labeled values of any factor variables are displayed by default. To see the labels corresponding to the variables that are included in the model rather than the names of the variables, we can specify the varlabel option.

```
. estimates table alt, b(%5.4f) se(%5.4f) varlabel
```

| Variable | alt |
| ---: | ---: |
| Weight (lbs.) | -0.0068 |
|  | 0.0012 |
| Displacement (cu. in.) | 0.0019 |
|  | 0.0101 |
| Car origin |  |
| Foreign | -1.6006 |
|  | 1.1136 |
| Constant | 41.8479 |
|  | 2.3507 |

Legend: b/se
If we wanted this table to appear in a Word document, we could use the putdocx command with the etable output type to write it to a new document. To create the new document myresults.docx, we would type the following putdocx commands:

```
. putdocx begin
. putdocx table results = etable
. putdocx save myresults.docx
successfully created "myresults.docx"
```

This creates a table in Word that looks like


## Stored results

estimates table stores the following in $r()$ :
Macros
$r$ (names) names of results used
Matrices

$$
\begin{array}{ll}
\text { r(coef) } & \text { matrix } M: n \times 2 * m \\
& M[i, 2 j-1]=i \text { th parameter estimate for model } j ; \\
& M[i, 2 j]=\text { variance of } M[i, 2 j-1] ; i=1, \ldots, n ; j=1, \ldots, m \\
\text { r(stats) } \quad \text { matrix } S: k \times m \text { (if option stats() specified) } \\
& S[i, j]=i \text { th statistic for model } j ; i=1, \ldots, k ; j=1, \ldots, m
\end{array}
$$

## References

Gallup, J. L. 2012. A new system for formatting estimation tables. Stata Journal 12: 3-28.
Weiss, M. 2010. Stata tip 90: Displaying partial results. Stata Journal 10: 500-502.

## Also see

[R] estimates - Save and manipulate estimation results
[R] etable - Create a table of estimation results
[R] table regression - Table of regression results

## Title

estimates title - Set title for estimation results

Description Quick start Menu Syntax Remarks and examples Also see

## Description

estimates title: (note the colon) sets or clears the title for the current estimation results. The title is used by estimates table, estimates stats, and estimates dir.
estimates title without the colon displays the current title.

## Quick start

Set title for the current estimation results

```
estimates title: Base model
```

Display the title of the current estimation results

```
estimates title
```


## Menu

Statistics $>$ Postestimation

## Syntax

```
estimates title: [text]
estimates title
```


## Remarks and examples

After setting the title, if estimates have been stored, do not forget to store them again:

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. regress mpg gear turn
    (output omitted)
. estimates store reg
```

Now let's add a title:

```
. estimates title: "My regression"
```

. estimates store reg

## Also see <br> [R] estimates - Save and manipulate estimation results

## Title

Estimation options - Estimation options

Description Syntax Options Also see

## Description

This entry describes the options common to many estimation commands. Not all the options documented here work with all estimation commands. See the documentation for the particular estimation command; if an option is listed there, it is applicable.

## Syntax

```
estimation_cmd ... [, options]
```

options Description

Model

```
noconstant
offset(varname o)
exposure(\mp@subsup{varname e}{e}{})
constraints(constraints)
```

Reporting
level(\#)
lrmodel
nocnsreport
noci
nopvalues
noomitted
vsquish
noemptycells
baselevels
allbaselevels
nofvlabel
fvwrap(\#)
fvwrapon(style)
cformat (\% fmt)
pformat (\%fmt)
sformat (\%fmt)
Integration
intmethod(intmethod)
intpoints(\#)
nolstretch
collinear
coeflegend
suppress constant term
include varname $_{o}$ in model with coefficient constrained to 1 include $\ln \left(\right.$ varname $\left._{e}\right)$ in model with coefficient constrained to 1 apply specified linear constraints
set confidence level; default is level (95)
perform likelihood-ratio model test instead of the default Wald test do not display constraints
suppress confidence intervals
suppress $p$-values and their test statistics
do not display omitted collinear variables
suppress blank space separating factor variables or time-series variables
do not display empty interaction cells of factor variables
report base levels for factor variables and interactions
display all base levels for factor variables and interactions
display factor-variable level values rather than value labels
allow \# lines when wrapping long value labels
apply style for wrapping long value labels;
style may be word or width
format for coefficients, standard errors, and confidence limits
format for $p$-values
format for test statistics
integration method for random-effects models
use \# integration (quadrature) points
do not automatically widen coefficient table for long variable names keep collinear variables
display legend instead of statistics

## Options

## Model

noconstant suppresses the constant term (intercept) in the model.
offset $\left(\right.$ varname $\left._{o}\right)$ specifies that varname ${ }_{o}$ be included in the model with the coefficient constrained to be 1 .
exposure ( varname $_{e}$ ) specifies a variable that reflects the amount of exposure over which the depvar events were observed for each observation; $\ln \left(\right.$ varname $\left._{e}\right)$ with the coefficient constrained to be 1 is entered into the log-link function.
constraints (numlist $\mid$ matname) specifies the linear constraints to be applied during estimation. The default is to perform unconstrained estimation. See $[R]$ reg3 for the use of constraints in multiple-equation contexts.
constraints (numlist) specifies the constraints by number after they have been defined by using the constraint command; see [R] constraint. Some commands (for example, slogit) allow only constraints (numlist).
constraints (matname) specifies a matrix containing the constraints; see $[\mathrm{P}]$ makecns.

## Reporting

level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals.
lrmodel specifies to conduct a likelihood-ratio test of the full maximum-likelihood model versus the restricted model that includes only a constant term in the regression equation instead of conducting the default Wald test that all coefficients are zero. This option can substantially increase estimation time.
lrmodel may not be specified with the vce(robust), vce(cluster clustvar), vce(jackknife), vce(bootstrap), or noconstant option.
lrmodel may not be combined with constraints; see constraints (constraints). In some cases, a likelihood-ratio test is valid for models with constraints. To compute a likelihood-ratio test when constraints have been applied during estimation, use lrtest; see [R] lrtest.
lrmodel may not be specified with the jackknife, bootstrap, svy, or mi estimate prefix. In addition, lrmodel may not be specified on replay.
nocnsreport specifies that no constraints be reported. The default is to display user-specified constraints above the coefficient table.
noci suppresses confidence intervals from being reported in the coefficient table.
nopvalues suppresses $p$-values and their test statistics from being reported in the coefficient table.
noomitted specifies that variables that were omitted because of collinearity not be displayed. The default is to include in the table any variables omitted because of collinearity and to label them as "(omitted)".
vsquish specifies that the blank space separating factor-variable terms or time-series-operated variables from other variables in the model be suppressed.
noemptycells specifies that empty cells for interactions of factor variables not be displayed. The default is to include in the table interaction cells that do not occur in the estimation sample and to label them as "(empty)".
baselevels and allbaselevels control whether the base levels of factor variables and interactions are displayed. The default is to exclude from the table all base categories.
baselevels specifies that base levels be reported for factor variables and for interactions whose bases cannot be inferred from their component factor variables.
allbaselevels specifies that all base levels of factor variables and interactions be reported.
nofvlabel displays factor-variable level values rather than attached value labels. This option overrides the fvlabel setting; see $[R]$ set showbaselevels.
fvwrap(\#) specifies how many lines to allow when long value labels must be wrapped. Labels requiring more than \# lines are truncated. This option overrides the fvwrap setting; see [R] set showbaselevels.
fvwrapon(style) specifies whether value labels that wrap will break at word boundaries or break based on available space.
fvwrapon(word), the default, specifies that value labels break at word boundaries.
fvwrapon(width) specifies that value labels break based on available space.
This option overrides the fvwrapon setting; see [R] set showbaselevels.
cformat (\% fmt) specifies how to format coefficients, standard errors, and confidence limits in the coefficient table. The maximum format width is 9 . See [R] set cformat.
pformat (\%fint) specifies how to format $p$-values in the coefficient table. The maximum format width is 5 . See [R] set cformat.
sformat (\% $\% m t$ ) specifies how to format test statistics in the coefficient table. The maximum format width is 8 . See $[R]$ set cformat.
$\qquad$ Integration
intmethod(intmethod) specifies the integration method to be used for the random-effects model. It accepts one of four arguments: mvaghermite, the default for all but a crossed random-effects model, performs mean and variance adaptive Gauss-Hermite quadrature; mcaghermite performs mode and curvature adaptive Gauss-Hermite quadrature; ghermite performs nonadaptive GaussHermite quadrature; and laplace, the default for crossed random-effects models, performs the Laplacian approximation.
intpoints(\#) specifies the number of integration points to use for integration by quadrature. The default is intpoints (12); the maximum is intpoints(195). Increasing this value improves the accuracy but also increases computation time. Computation time is roughly proportional to its value.

The following options are not shown in the dialog box:
nolstretch specifies that the width of the coefficient table not be automatically widened to accommodate longer variable names. The default, lstretch, is to automatically widen the coefficient table up to the width of the Results window. Specifying lstretch or nolstretch overrides the setting given by set lstretch. If set lstretch has not been set, the default is lstretch.
collinear specifies that the estimation command not omit collinear variables. This option is seldom used because collinear variables make a model unidentified. However, you can add constraints to a model that will identify it even with collinear variables. For example, if variables x1 and x2 are collinear, but you constrain the coefficient on x 2 to be a multiple of the coefficient on x 1 , then your model is identified even with collinear variables. In such cases, you specify collinear so that both x 1 and x 2 are retained in the model.
coeflegend specifies that the legend of the coefficients and how to specify them in an expression be displayed rather than displaying the statistics for the coefficients.

## Also see

[U] 20 Estimation and postestimation commands

## Title

etable - Create a table of estimation results

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Appendix | Acknowledgments |
| Reference | Also see |  |  |

## Description

The etable command allows you to easily create a table of estimation results and export it to a variety of file types. You can create a table complete with a title, notes, stars for indicating significant results, and more.

## Quick start

Create a table from the active estimation results, reporting the coefficients, standard errors, and number of observations
etable
Same as above, and display stars for significant results and a note indicating what the stars represent etable, showstars showstarsnote

Create a table with title "My title" and note "My note"
etable, title("My title") note("My note")
Create a table with stored estimates model1 and model2, along with the number of observations and adjusted $R^{2}$ value for each model
etable, estimates(model1 model2) mstat(N) mstat(r2_a)
Same as above, and export the table to myfile.tex
etable, estimates(model1 model2) mstat(N) mstat(r2_a) export(myfile.tex)

## Menu

Statistics $>$ Summaries, tables, and tests $>$ Table of estimation results

## Syntax

etable [, options]
options
Description

## Main

estimates(namelist)
margins
replay
column (column_header)
name (cname)
append
replace

## Coefficients

keep (coeflist)
cstat (cstat[, cstat_opts])
Model
mstat (mstat $[$, mstat_opts $]$ )
Equations
equations (eqlist)
eqrecode (oldeq $=$ neweq $)$
[no]showeq
Stars
stars([starspec $][$, stars_opts $]$ )
[no]showstars
[no]showstarsnote
Title
title (string)
titlestyles (text_styles)
Notes
note (string)
notestyles(text_styles)
Export
export(filename.suffix [, export_opts])
Options
$[$ no $]$ varlabel
$[$ no $]$ fvlabel
$[$ no $]$ center
label(filename[, replace])
style(filename[, override])
warn
work with previously stored estimation results
consume results from margins
report table without consuming results
select column header
work with collection cname; default is name (ETable)
append to the collection
replace the collection
report coefficients in order specified report coefficient statistic
report model statistic
report equations in order specified
recode equation
display or suppress equations
customize rules for star labels
display or suppress star labels
display or suppress note explaining star labels
add table title
change table title styles
add table note
change table note styles
export table
display or suppress variable names or labels display or suppress factor values or labels center or right-align item cells
specify the collection labels
specify the collection style
show collect warnings
warn does not appear in the dialog box.
column_header
Description

## depvar

dvlabel
command
title
estimates
index
show dependent variable name; the default
show variable label for dependent variable
show command name show command title show estimates name show result set index
cstat_opts
Description
label (string)
font ([fontfamily] [, font_opts $]$ )
smcl (smcl)
latex (latex)
shading (sspec)
nformat (\% fint)
sformat (sfint)
cidelimiter (char)
cridelimiter (char)
specify coefficient statistic label
specify font style
specify formatting for SMCL files
specify LATEX macro
set background color, foreground color, and fill pattern specify numeric format
specify string format
use character as delimiter for confidence interval limits use character as delimiter for credible interval limits
mstat_opts
label (string)
font ([fontfamily][, font_opts])
$\mathrm{smcl}(\mathrm{smcl})$
latex (latex)
shading (sspec)
nformat (\%fmt)
sformat (sfmt)

Description
specify model statistic label
specify font style
specify formatting for SMCL files
specify LATEX macro
set background color, foreground color, and fill pattern
specify numeric format
specify string format

| font_opts | Description |
| :--- | :--- |
| size(\# [unit $]$ ) | specify font size |
| color (color) | specify font color |
| variant(variant) | specify font variant and capitalization |
| $[$ no $]$ bold | specify whether to format statistic as bold |
| $[$ no $]$ italic | specify whether to format statistic as italic |
| $[$ no $]$ strikeout | specify whether to strike out statistic |
| underline(upattern) | specify whether to underline statistic |

stars_opts
Description
attach (cstat)
increasing
decreasing
pvname (string)
delimiter (char)
nformat (\% fint)
prefix (string)
suffix (string)
clear
attach star labels to coefficient statistic cstat compose stars note with increasing $p$-values; the default compose stars note with decreasing $p$-values $p$-value name for stars note use character as delimiter for labels in stars note numeric format for stars note prefix for stars note suffix for stars note remove previous star properties
text_styles
font ([fontfamily] [, text_opts]) font style
smcl (smcl) specify formatting for SMCL files
latex (latex)
shading (sspec)

Description
specify formatting for SMCL files
specify LATEX macro $^{2}$
set background color, foreground color, and fill pattern
text_opts Description
size (\# [unit]) specify font size
color (color)
variant (variant)
specify font color
specify font variant and capitalization
[no]bold specify whether to format statistic as bold
[no]italic specify whether to format statistic as italic
[no]strikeout specify whether to strike out statistic
[no]underline
specify whether to underline statistic

| suffix | fileformat | Output format |
| :--- | :--- | :--- |
| docx | as (docx) | Microsoft Word |
| html | as (html) | HTML 5 with CSS |
| pdf | as (pdf) | PDF |
| xlsx | as (xlsx) | Microsoft Excel 2007/2010 or newer |
| xls | as (xls) | Microsoft Excel 1997/2003 |
| tex | as (latex) | LATEX |
| smcl | as (smcl) | SMCL |
| txt | as (txt) | plain text |
| markdown | as (markdown) | Markdown |
| md | as (markdown) | Markdown |

export_opts
Description
as (fileformat)
replace
docx_options
html_options
pdf_options
excel_options
tex_options
smcl_option
txt_option
md_option
specify document type
overwrite existing file
available when exporting to .docx files
available when exporting to .html files
available when exporting to .pdf files
available when exporting to . xls and . xl sx files
available when exporting to .tex files
available when exporting to .smcl files
available when exporting to .txt files
available when exporting to .markdown and .md files
docx_options Description
noisily
show the putdocx commands used to export to the Microsoft Word file
dofile(filename[, replace]) save the putdocx commands used for exporting to the named do-file
html_options
Description

| append | append to an existing file |
| :--- | :--- |
| tableonly | export only the table to the specified file |
| cssfile(cssfile) | define the styles in cssfile instead of filename |
| prefix(prefix) | use prefix to identify style classes |

pdf_options Description
noisily
dofile(filename[, replace])
show the putpdf commands used to export to the PDF file save the putpdf commands used for exporting to the named do-file
excel_options

## Description

## noisily

dofile(filename[, replace])
sheet (sheetname[, replace])
cell (cell)
modify
noopen do not open Excel file in memory

[^23]| tex_options | Description |
| :--- | :--- |
| append | append to an existing file <br> export only the table to the specified file |
| tableonly |  |

smcl_option Description
append $\quad$ append to an existing file
txt_option
Description
append append to an existing file
md_option
Description
append
append to an existing file
fontfamily specifies a valid font family.
unit may be in (inch), pt (point), or cm (centimeter). An inch is equivalent to 72 points and 2.54 centimeters. The default is pt.
variant may be allcaps, smallcaps, or normal.
variant (allcaps) changes the text to all uppercase letters; applicable when publishing items from a collection to Microsoft Word, PDF, LATEX, and HTML files.
variant (smallcaps) changes the text to use large capitals for uppercase letters and smaller capitals for lowercase letters; applicable when publishing items from a collection to Microsoft Word, LATEX, and HTML files.
variant (normal) changes the font variant back to normal and leaves the capitalization unchanged from the original text; applicable when publishing items from a collection to Microsoft Word, PDF, $\mathrm{LA}_{\mathrm{E}} \mathrm{X}$, and HTML files.
upattern may be any of the patterns listed in the Appendix. For example, underline (none) removes the underline from the statistic, and underline (single) underlines the statistic. All other upatterns are available only when publishing items from a collection to Microsoft Word.
smcl specifies the name of the SMCL directive to render text for SMCL output. The supported SMCL directives are input, error, result, and text.
latex specifies the name of a $\mathrm{LAT}_{\mathrm{E}} \mathrm{X}$ macro to render text for $\mathrm{LAT}_{\mathrm{E}} \mathrm{X}$ output. Example $\mathrm{LAT}_{\mathrm{E}} \mathrm{X}$ macro names are textbf, textsf, textrm, and texttt. Custom LATEX macros are also allowed. If text is to be rendered in a cell, title, or note, then latex is translated to the following when you export to $\mathrm{LAT}_{\mathrm{E}} \mathrm{X}$ :

## $\backslash$ latex $\{$ text $\}$

sspec is
[background(bgcolor) foreground(fgcolor) pattern(fpattern)]
bgcolor specifies the background color.
fgcolor specifies the foreground color.
fpattern specifies the fill pattern. A complete list of fill patterns is shown in the Appendix.
bgcolor, fgcolor, and color may be one of the colors listed in the Appendix; a valid RGB value in the form \#\#\# \#\#\# \#\#\#, for example, 171248 103; or a valid RRGGBB hex value in the form \#\#\#\#\#\#, for example, ABF867.
sfmt is the specification for a string format in option sformat () and may contain a mix of text and $\% s$. Here $\%$ s refers to the statistic value that is formatted as specified using nformat(). The text will be placed around the statistic as it is placed around $\% \mathrm{~s}$ in this option. For instance, to place parentheses around the statistic, you can specify sformat (" (\%s)").
Two text characters must be specified using a special character sequence if you want them to be displayed in your collection. To include $\%$, type $\% \%$. To include $\backslash$, type $\backslash \backslash$. For instance, to place a percent sign after a statistic, you can specify sformat("\%s\%\%").

## Options

estimates (namelist) specifies the estimation results to be included in the table. These are the names specified with estimates store. By default, etable creates a table with the active estimation results.
margins creates a table with the results of the immediately preceding margins command.
replay specifies that etable redisplay the table without consuming results.
column (column_header) specifies the content to be used in the column headers. column_header may be depvar, dvlabel, command, title, estimates, or index.
depvar specifies that etable use the dependent variable name for the column headers. This name is obtained from the eclass macro e(depvar). Note that this macro may contain multiple names after fitting a multivariate model.
dvlabel specifies that etable use the variable label for the dependent variable name for the column headers. etable uses the variable label associated with the variable name in the eclass macro e(depvar). If the variable does not have a variable label, the variable name will be used. column(dvlabel) will not be helpful when the estimation command stores multiple names in e(depvar).
command specifies that etable use the command name for the column headers. This name is obtained from the eclass macro e(cmd).
title specifies that etable use the command title for the column headers. This title is obtained from the eclass macro e(title).
estimates specifies that etable use the name given to previously stored estimation results for the column headers; these are the names specified with estimates store.
index specifies that etable use the result set index for the column headers.
name (cname) specifies the collection from which estimation results will be obtained, instead of the current collection. The default is name (ETable).
append specifies that etable append the results into the collection named in name().
replace permits etable to overwrite the existing collection. This option is implied for name (ETable)
when append and replay are not specified.

Coefficients
keep (coeflist) specifies the coefficients to be included in the table and the order in which to display them. Names are separated from each other by blanks. The default is to display all coefficients.
cstat (cstat[, cstat_opts]) specifies the coefficient statistic to be included in the table. Optionally, you may specify the label and style for this statistic. cstat () may be repeated to request multiple coefficient statistics.

The default is to display the coefficients ( $\_\mathbf{r} \_$b) and their standard errors ( $\_$r_se), both formatted to three decimals. Standard errors are enclosed in parentheses.

Available coefficient statistics are

| cstat | Description |
| :---: | :---: |
| -r_b | coefficients reported by estimation |
| _r_se | standard errors of _r_b |
| -r_z | test statistics for _r_b |
| -r_z_abs | absolute values of $\_$r $\quad$ z |
| -r_p | $p$-values for $\_$r_b |
| _r_lb | lower bounds of confidence intervals (CIs) for $\quad$ r_b |
| _r_ub | upper bounds of CIs for $\_$r_b |
| _r_ci | CIs for _r_b |
| _r_crlb | lower bounds of credible intervals for _r_b |
| -r_crub | upper bounds of credible intervals for $-r \_b$ |
| _r_cri | credible intervals of Bayesian estimates |
| _hide | hide coefficient statistics |

cstat_opts are label(string), font([fontfamily] [, font_opts]), smcl(smcl), latex (latex), shading (sspec), nformat (\%fmt), sformat (sfmt), cidelimiter(char), and cridelimiter (char).
label (string) is used to modify the label for the specified coefficient statistic.
font ([fontfamily] [, size(\# [unit]) color(color) variant (variant) [no]bold [no]italic [no]strikeout [no]underline underline (upattern)]) specifies the font style for the coefficient statistic. These font style properties are applicable when exporting the table to Microsoft Word, Microsoft Excel, PDF, HTML, and LETEX files, unless otherwise specified.
fontfamily specifies a valid font family. This font style property is applicable when publishing items from a collection to Microsoft Word, Microsoft Excel, PDF, and HTML files.
size (\# [unit]) specifies the font size as a number optionally followed by units. This font style property is applicable when publishing items from a collection to Microsoft Word, Microsoft Excel, PDF, and HTML files.
color (color) specifies the text color.
variant (variant) specifies the font variant and capitalization.
bold and nobold specify the font weight. bold changes the font weight to bold; nobold changes the font weight back to normal.
italic and noitalic specify the font style. italic changes the font style to italic; noitalic changes the font style back to normal.
strikeout and nostrikeout specify whether to add a strikeout mark to the coefficient statistic. strikeout adds a strikeout mark to the statistic; nostrikeout changes the statistic back to normal.
underline(upattern), underline, and nounderline specify how to underline the coefficient statistic.

Only one of strikeout or underline is allowed when publishing to HTML files.
smcl (smcl) specifies how to render the statistic value for SMCL output. This style property is applicable only when publishing items from a collection to a SMCL file.
latex (latex) specifies how to render the statistic value for $\mathrm{LAT}_{\mathrm{E}} \mathrm{X}$ output. This style property is applicable only when publishing items from a collection to a $\mathrm{LAT}_{\mathrm{E}} \mathrm{X}$ file.
shading (sspec) sets the background color, foreground color, and fill pattern. The background color is applicable when exporting the table to Microsoft Word, Microsoft Excel, PDF, HTML, and $\mathrm{LAT}_{\mathrm{E}} \mathrm{X}$ files. The foreground color and fill pattern are applicable when exporting the table to Microsoft Word and Microsoft Excel.
nformat (\%fmt) applies the Stata numeric format \% fmt to the coefficient statistic.
sformat (sfint) applies a string format to the coefficient statistic.
cidelimiter (char) changes the delimiter between confidence interval limits for coefficient statistic _r_ci. The default is cidelimiter(" ").
cridelimiter (char) changes the delimiter between credible interval limits for coefficient statistic _r_cri. The default is cridelimiter(" ").
mstat (mstat [, mstat_opts]) specifies the model statistics to be included in the table.
mstat may be a result identifier or a named expression. mstat () may be repeated to request multiple model statistics. The default is to display the number of observations, with zero decimal digits. result identifier is one of the following:

| Identifier | Result |
| :--- | :--- |
| N | number of observations |
| aic | Akaike's information criteria |
| bic | Schwarz's Bayesian information criteria |
| F | $F$ statistic |
| chi2 | $\chi^{2}$ |
| ll | log likelihood of fitted model |
| r2 | $R^{2}$ |
| r2_a | adjusted $R^{2}$ |
| rank | rank of fitted model |
| escalar | any e() scalar, by default |
| rscalar | any r() scalar with option margins |
| _hide | hide model statistics |

named expression is specified as name $=$ exp, where name may be any valid Stata name and exp is a scalar expression, typically an expression that involves one or more scalars in e(). For example, aic $=(-2 * e(l l)+2 * e(r a n k))$.
With option margins, or if you ran a postestimation command like estat, exp may also involve one or more scalars in $r()$. For example, suppose you fit a logistic model and also called estat gof, then you can show its goodness-of-fit statistic with chi2_gof=(r(chi2)).
mstat_opts are label (string), font ([fontfamily] [, font_opts]), smcl (smcl), latex(latex), shading (sspec), nformat (\%fint), and sformat (sfmt).
label (string) is used to modify the label for the specified model statistic.
font ([fontfamily] [, size(\# [unit]) color (color) variant(variant) [no]bold [no]italic [no]strikeout [no]underline underline(upattern)]) specifies the font style for the model statistic. These font style properties are applicable when exporting the table to Microsoft Word, Microsoft Excel, PDF, HTML, and LATEX files, unless otherwise specified.
fontfamily specifies a valid font family. This font style property is applicable when publishing items from a collection to Microsoft Word, Microsoft Excel, PDF, and HTML files.
size (\# [unit]) specifies the font size as a number optionally followed by units. This font style property is applicable when publishing items from a collection to Microsoft Word, Microsoft Excel, PDF, and HTML files.
color (color) specifies the text color.
variant (variant) specifies the font variant and capitalization.
bold and nobold specify the font weight. bold changes the font weight to bold; nobold changes the font weight back to normal.
italic and noitalic specify the font style. italic changes the font style to italic; noitalic changes the font style back to normal.
strikeout and nostrikeout specify whether to add a strikeout mark to the model statistic. strikeout adds a strikeout mark to the statistic; nostrikeout changes the text back to normal.
underline(upattern), underline, and nounderline specify how to underline the model statistic.

Only one of strikeout or underline is allowed when publishing to HTML files.
$\operatorname{smcl}(s m c l)$ specifies how to render the statistic value for SMCL output. This style property is applicable only when publishing items from a collection to a SMCL file.
latex (latex) specifies how to render the statistic value for $\mathrm{LAT}_{\mathrm{E}} \mathrm{X}$ output. This style property is applicable only when publishing items from a collection to a LATEX file.
shading (sspec) sets the background color, foreground color, and fill pattern.
nformat (\% fmt) applies the Stata numeric format \%fmt to the model statistic.
sformat (sfint) applies a string format to the model statistic.

Equations
equations (eqlist) specifies the equations to be included in the table and the order in which they are reported.
eqrecode (oldeq $=$ neweq $)$ changes the equation name from oldeq to neweq. eqrecode() may be repeated to recode multiple equations.
showeq and noshoweq specify whether equations should be displayed. showeq displays the equations; noshoweq suppresses the equations. manages the display of stars for indicating the significance of results.

The default is stars (. 05 "*" . 01 "**", attach (_r_b)), which will display one star (*) for $p$-values less than 0.05 and two stars $(* *)$ for $p$-values less than 0.01 ; the stars will be attached to the coefficients ( $\_$r_b).
stars_opts are attach(cstat), increasing, decreasing, pvname(string), delimiter (char), nformat(\%fmt), prefix(string), suffix(string), and clear.
attach (cstat) specifies that the star labels be attached to coefficient statistic cstat. The default is attach (_r_b).
increasing and decreasing control the order of $p$-values in the stars note.
increasing specifies that the stars note be composed with increasing $p$-values. This is the default.
decreasing specifies that the stars note be composed with decreasing $p$-values.
pvname (string) specifies the name for the $p$-value in the stars note that is displayed with showstarsnote. The default is pvname ( p ).
delimiter (char) changes the delimiter between labels in the stars note. The default is cridelimiter(",").
nformat (\%fmt) specifies the numeric format for the numbers displayed in the stars note. The default is nformat ( $\% 9.0 \mathrm{~g}$ ).
prefix (string) adds a prefix to the stars note.
suffix (string) adds a suffix to the stars note.
clear removes existing star properties.
showstars and noshowstars specify whether star labels should be displayed. showstars displays star labels; noshowstars suppresses the star labels.
showstarsnote and noshowstarsnote specify whether to display the note explaining what the star labels represent. showstarsnote displays the note; noshowstarsnote suppresses the note.
showstarsnote is ignored if noshowstars is in effect.

Title
title (string) adds the text string as a title to the table.
titlestyles(text_styles) changes the style for the table title. text_styles are the following:
font ([fontfamily] [, size(\# [unit]) color (color) variant (variant) [no]bold
[no]italic [no]strikeout [no]underline]) specifies the font style. These font style properties are applicable when exporting the table to Microsoft Word, Microsoft Excel, PDF, HTML, and LATEX files, unless otherwise specified.
fontfamily specifies a valid font family. This font style property is applicable when publishing items from a collection to Microsoft Word, Microsoft Excel, PDF, and HTML files.
size (\# [unit]) specifies the font size as a number optionally followed by units. This font style property is applicable when publishing items from a collection to Microsoft Word, Microsoft Excel, PDF, and HTML files.
color (color) specifies the text color.
variant (variant) specifies the font variant and capitalization.
bold and nobold specify the font weight. bold changes the font weight to bold; nobold changes the font weight back to normal.
italic and noitalic specify the font style. italic changes the font style to italic; noitalic changes the font style back to normal.
strikeout and nostrikeout specify whether to add a strikeout mark to the title. strikeout adds a strikeout mark to the title; nostrikeout changes the title back to normal.
underline and nounderline specify whether to underline the table title. underline adds a single line under the title; nounderline removes the underline.

Only one of strikeout or underline is allowed when publishing to HTML files.
smcl ( smcl ) specifies how to render the table title for SMCL output. This style property is applicable only when publishing items from a collection to a SMCL file.
latex (latex) specifies how to render the table title for ${ }^{[A T} T_{E} X$ output. This style property is applicable only when publishing items from a collection to a LATEX file.
shading (sspec) sets the background color, foreground color, and fill pattern. The background color is applicable when exporting the table to Microsoft Word, Microsoft Excel, PDF, HTML, and LATEX files. The foreground color and fill pattern are applicable when exporting the table to Microsoft Word and Microsoft Excel.
note (string) adds the text string as a note to the table. note() may be specified multiple times to add multiple notes. Each note is placed on a new line.
notestyles (text_styles) changes the style for the table notes. text_styles are the following:
font ([fontfamily] [, size(\# [unit]) color (color) variant (variant) [no]bold
[no]italic [no]strikeout [no]underline]) specifies the font style. These font style properties are applicable when exporting the table to Microsoft Word, Microsoft Excel, PDF, HTML, and LATEX files, unless otherwise specified.
fontfamily specifies a valid font family. This font style property is applicable when publishing items from a collection to Microsoft Word, Microsoft Excel, PDF, and HTML files.
size (\# [unit]) specifies the font size as a number optionally followed by units. This font style property is applicable when publishing items from a collection to Microsoft Word, Microsoft Excel, PDF, and HTML files.
color (color) specifies the text color.
variant (variant) specifies the font variant and capitalization.
bold and nobold specify the font weight. bold changes the font weight to bold; nobold changes the font weight back to normal.
italic and noitalic specify the font style. italic changes the font style to italic; noitalic changes the font style back to normal.
strikeout and nostrikeout specify whether to add a strikeout mark to the notes. strikeout adds a strikeout mark to the note; nostrikeout changes the note back to normal.
underline and nounderline specify whether to underline the table notes. underline adds a single line under the notes; nounderline removes the underline.

Only one of strikeout or underline is allowed when publishing to HTML files.
smcl (smcl) specifies how to render the table notes for SMCL output. This style property is applicable only when publishing items from a collection to a SMCL file.
latex (latex) specifies how to render the table notes for $\mathrm{LT}_{\mathrm{E}} \mathrm{X}$ output. This style property is applicable only when publishing items from a collection to a $\mathrm{EAT}_{\mathrm{E}}$ file.
shading (sspec) sets the background color, foreground color, and fill pattern. The background color is applicable when exporting the table to Microsoft Word, Microsoft Excel, PDF, HTML, and $\mathrm{LAT}_{\mathrm{E}} \mathrm{X}$ files. The foreground color and fill pattern are applicable when exporting the table to Microsoft Word and Microsoft Excel.

Export
export (filename.suffix [, export_opts]) exports the table to the specified file. export_opts are the following:
as (fileformat) specifies the file format to which the table is to be exported. This option is rarely specified because, by default, etable determines the format from the suffix of the file being created.
replace permits etable to overwrite an existing file.
noisily specifies that etable show the commands used to export the table to Microsoft Word, Microsoft Excel, and PDF files. The putdocx, putexcel, or putpdf command used to export the table will be displayed.
dofile(filename[, replace]) specifies that etable save to filename the commands used to export the table to Microsoft Word, Microsoft Excel, and PDF files.
If filename already exists, it can be overwritten by specifying replace. If filename is specified without an extension, .do is assumed.
append specifies that etable append the table to an existing file.
This option is applicable when you export the table to an HTML, a LATEX, a SMCL, a txt, or a Markdown file. When you export to HTML and LATEX files, the append option implies the tableonly option. Furthermore, when you export to HTML files, if the target CSS file already exists, etable will also append to it.
tableonly specifies that only the table be exported to the specified HTML or ${ }^{L A T} \mathrm{E}_{\mathrm{E}} \mathrm{X}$ document. By default, etable produces complete HTML and LATEX documents.

When you export to an HTML file, if the cssfile() option is not specified, a CSS filename is constructed from filename, with the extension replaced with .css.
cssfile(cssfile) specifies that etable define the styles in cssfile instead of filename when you export to HTML.
prefix (prefix) specifies that etable use prefix to identify style classes when you export to HTML.
sheet (sheetname [, replace]) saves to the worksheet named sheetname. For more information about this option, see [RPT] putexcel.
cell (cell) specifies an Excel upper-left cell as the starting position to publish the table. The default is cell(A1).
modify permits putexcel set to modify an Excel file. For more information about this option, see [RPT] putexcel.
noopen prevents putexcel from opening the Excel file in memory for modification. It does not appear on the dialog box. For more information about this option, see [RPT] putexcel. This
option is necessary only when you need to force etable to produce do-files as it did when etable was first introduced in Stata 17.
$\qquad$
varlabel and novarlabel specify whether variable labels should be displayed. varlabel displays variable labels; novarlabel displays variable names.
fvlabel and nofvlabel specify whether value labels should be displayed. fvlabel displays value labels; nofvlabel displays the values of the factor variable.
center and nocenter specify how item cells are horizontally aligned. center specifies that item cells are centered; nocenter specifies that item cells are right-aligned.
label (filename [, replace]) specifies the filename containing the collection labels to use for your table. Labels in filename will be loaded for the table, and default labels will be used for any labels not specified in filename.

If you prefer to replace the labels used by etable with those specified in filename, specify replace. The etable labels will be discarded, and only the labels in filename will be applied.
style(filename[, override]) specifies the filename containing the collection styles to use for your table. This might be a style you saved with collect style save or a predefined style shipped with Stata. The etable collection styles will be discarded, and only the collection styles in filename will be applied. Note that the layout specification saved in filename will not be applied; etable will always use its predefined layout.

If you prefer the etable collection styles but also want to apply any styles in filename, specify override. If there are conflicts between the default collection styles and those in filename, the ones in filename will take precedence.

The default is to use only the collection styles set in c(etable_style); see [TABLES] set etable_style.

The following option is available with etable but is not shown in the dialog box:
warn specifies that etable display warnings from collect. By default, these warnings are suppressed.

## Remarks and examples

Remarks are presented under the following headings:
Introduction
A first example
Table comparing regression results
Multiple-equation models

## Introduction

etable allows you to easily create a table of estimation results and export it to a variety of file types, without any knowledge of the collection system. You can make a standard estimation table with the active estimation results, results from a margins command, or with stored estimates. You can also customize the table by formatting the results, adding model statistics and coefficient-specific statistics, labeling statistically significant results, adding a title and notes, and more.

In most cases, you will use etable to easily create a table and export it to another format. However, you can customize the table beyond the options that are available with etable. When you issue an etable command, the results are stored in a collection called ETable. This collection is replaced with each new etable command, unless you specify the append or replay option. You can make additional changes to the collection with the collect suite of commands. To learn more about the collect commands, see [TABLES] Intro and the entries discussed therein.

## A first example

In its simplest specification, you type etable after fitting a model, and you get a table with coefficients, standard errors, and the number of observations. For example, below, we use data from the Second National Health and Nutrition Examination Survey (NHANES II) (McDowell et al. 1981). We fit a simple model for systolic blood pressure and then create our table of estimation results:

```
. use https://www.stata-press.com/data/r18/nhanes2l
(Second National Health and Nutrition Examination Survey)
. quietly: regress bpsystol age weight i.region
. etable
```

|  | bpsystol |
| :--- | ---: |
| Age (years) | 0.638 |
|  | $(0.011)$ |
| Weight (kg) | 0.407 |
|  | $(0.012)$ |
| Region | -0.240 |
| MW | $(0.564)$ |
| S | -0.619 |
|  | $(0.560)$ |
| W | -0.862 |
|  | $(0.570)$ |
| Intercept | 71.708 |
|  | $(1.108)$ |
| Number of observations | 10351 |

You can also include statistics that pertain to the coefficients, such as test statistics and confidence intervals, and model statistics, such as the $R^{2}$ value and the $F$ statistic. You can look at the cstat () and mstat () options for additional statistics.

Additionally, you can complete your table with a title, notes, and labels for significant results. For example, below, we add a title to our table, and we display stars for statistically significant results:

```
. etable, title(Model for systolic blood pressure) showstars showstarsnote
Model for systolic blood pressure
\begin{tabular}{lc}
\hline & bpsystol \\
\hline Age (years) & \(0.638)^{* *}\) \\
Weight (kg) & \((0.011)\) \\
& \(0.407)^{* *}\) \\
Region & \((0.012)\) \\
MW & -0.240 \\
S & \((0.564)\) \\
& -0.619 \\
W & \((0.560)\) \\
& -0.862 \\
Intercept & \((0.570)\) \\
& \(71.708 * *\) \\
Number of observations & \((1.108)\) \\
\hline
\end{tabular}
** p<.01, * p<.05
```

The showstars option displays stars next to the coefficients that are significant either at the $1 \%$ or $5 \%$ levels, and showstarsnote adds the note we see at the bottom, explaining what the stars represent. You can look at the stars() option to create your own rules for displaying stars or to specify your own labels for significance.

Suppose we have finalized our table and we are ready to export it to another format. Below, we export our table to the file mytable.html:

```
. etable, title(Model for systolic blood pressure)
> showstars showstarsnote export(mytable.html)
Model for systolic blood pressure
\begin{tabular}{lc} 
& bpsystol \\
\hline Age (years) & \(0.638 * *\) \\
Weight (kg) & \((0.011)\) \\
\(0.407 * *\) \\
Region & \((0.012)\) \\
MW & -0.240
\end{tabular}
    (0.564)
    S -0.619
                                    (0.560)
    W -0.862
        (0.570)
Intercept 71.708 *
    (1.108)
Number of observations 10351
** p<.01, * p<.05
(collection ETable exported to file mytable.html)
```

We could also export this table to a Microsoft Word, Microsoft Excel, LATEX, Markdown, SMCL, PDF, or plain text file by specifying the appropriate file extension.

## Table comparing regression results

If your goal is to create a table comparing regression results, you can store the results from each model with estimates store and then specify which of those models you want to include in your table with the estimates() option.

For example, below, we fit two different models for systolic blood pressure and store them under the names model1 and model2.
. quietly: regress bpsystol i.sex weight
. estimates store model1
. quietly: regress bpsystol i.sex i.agegrp weight
. estimates store model2
To include results from both of these models in our table, we specify estimates (model1 model2); the models are presented in the order we list them. Additionally, we report the number of observations and the $R^{2}$ adjusted for degrees of freedom.

```
. etable, estimates(model1 model2) mstat(N) mstat(r2_a)
```

|  | bpsystol bpsystol |  |
| :---: | :---: | :---: |
| Sex |  |  |
| Female | 1.420 | 1.041 |
|  | (0.475) | (0.415) |
| Weight (kg) | 0.452 | 0.436 |
|  | (0.015) | (0.014) |
| Age group |  |  |
| 30-39 |  | 1.195 |
|  |  | (0.633) |
| 40-49 |  | 7.252 |
|  |  | (0.684) |
| 50-59 |  | 15.942 |
|  |  | (0.681) |
| 60-69 |  | 22.839 |
|  |  | (0.546) |
| 70+ |  | 30.466 |
|  |  | (0.741) |
| Intercept | 97.634 | 86.710 |
|  | (1.246) | (1.116) |
| Number of observations | 10351 | 10351 |
| Adjusted R-squared | 0.08 | 0.30 |

We would like to make a few changes to finalize this table. First, because both models have the same dependent variable, we want to display the index of result sets instead of the variable name. Second, we add stars for significance and a note explaining what the stars represent. Third, instead of reporting standard errors, we want to report confidence intervals (_r_ci). We format the intervals with one decimal place and use a comma as the delimiter. etable will automatically report coefficients, unless you specify cstat (), in which case it will report only the coefficient statistics you specify; therefore, we add cstat (_r_b). Finally, we add a title to our table:

```
. etable, estimates(model1 model2) mstat(N) mstat(r2_a) column(index)
> showstars showstarsnote cstat(_r_b)
> title("Models for systolic blood pressure")
> cstat(_r_ci, nformat(%6.1f) cidelimiter(","))
Models for systolic blood pressure
```

|  |  | 1 |  |  | 2 |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Sex |  |  |  |  |  |  |
| Female |  | 1.420 | ** |  | 1.041 | * |
|  | [0.5, | $2.4]$ |  | [0.2, | 1.9] |  |
| Weight (kg) |  | 0.452 | ** |  | 0.436 | ** |
|  | [0.4, | 0.5] |  | [0.4, | 0.5] |  |
| Age group |  |  |  |  |  |  |
| 30-39 |  |  |  |  | 1.195 |  |
|  |  |  |  | [-0.0, | 2.4] |  |
| 40-49 |  |  |  |  | 7.252 | ** |
|  |  |  |  | [5.9, | 8.6] |  |
| 50-59 |  |  |  |  | 15.942 | ** |
|  |  |  |  | [14.6, | 17.3] |  |
| 60-69 |  |  |  |  | 22.839 | ** |
|  |  |  |  | [21.8, | $23.9]$ |  |
| 70+ |  |  |  |  | 30.466 | ** |
|  |  |  |  | [29.0, | 31.9] |  |
| Intercept |  | 97.634 | ** |  | 86.710 | ** |
|  | [95.2, | 100.1] |  | [84.5, | 88.9] |  |
| Number of observations |  | 10351 |  |  | 10351 |  |
| Adjusted R-squared |  | 0.08 |  |  | 0.30 |  |

Now our table is complete.
An alternative way to create this table is to build up the estimation table with the append option. For example, we can fit the first model and create the table. Then, after fitting the second model, we append the results as follows:

```
. quietly: regress bpsystol i.sex weight
. etable
. quietly: regress bpsystol i.sex i.agegrp weight
. etable, append mstat(N) mstat(r2_a) column(index)
> showstars showstarsnote cstat(_r_b)
> title("Models for systolic blood pressure")
> cstat(_r_ci, nformat(%6.1f) cidelimiter(","))
    (output omitted)
```


## Multiple-equation models

When you work with multiple-equation models, there is an additional option that will prove useful when creating the table of estimation results. For example, below, we fit a multivariate regression with mvreg:
. mvreg bpsystol bpdiast = age weight

| Equation | Obs | Parms | RMSE | "R-sq" | F | P>F |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: |
| bpsystol | 10,351 | 3 | 19.48051 | 0.3031 | 2250 | 0.0000 |
| bpdiast | 10,351 | 3 | 11.51474 | 0.2067 | 1348.469 | 0.0000 |


|  | Coefficient | Std. err. | t | $\mathrm{P}>\mid \mathrm{t\mid}$ | [95\% conf. interval] |  |
| ---: | ---: | :--- | :---: | :---: | :---: | :---: |
| bpsystol |  |  |  |  |  |  |
| age | .6379892 | .0111315 | 57.31 | 0.000 | .6161692 | .6598091 |
| weight | .4069041 | .0124786 | 32.61 | 0.000 | .3824435 | .4313646 |
| _cons | 71.27096 | 1.041742 | 68.42 | 0.000 | 69.22894 | 73.31297 |
| bpdiast |  |  |  |  |  |  |
| age | .187733 | .0065797 | 28.53 | 0.000 | .1748355 | .2006306 |
| weight | .3116502 | .007376 | 42.25 | 0.000 | .2971918 | .3261086 |
| _cons | 50.37585 | .615764 | 81.81 | 0.000 | 49.16884 | 51.58287 |

Next, we create our table of estimation results:
. etable

|  | bpsystol bpdiast |
| :---: | :---: |
| Age (years) | 0.638 |
|  | (0.011) |
| Weight (kg) | 0.407 |
|  | (0.012) |
| Intercept | 71.271 |
|  | (1.042) |
| Age (years) | 0.188 |
|  | (0.007) |
| Weight (kg) | 0.312 |
|  | (0.007) |
| Intercept | 50.376 |
|  | (0.616) |
| Number of observations | 10351 |

The results for both models are placed in a single column, so below we add the showeq option to display the equation names (bpsystol and bpdiast). This will help us identify which results correspond to each model. Note that there are two dependent variables in this model, and both variable names are displayed in the column header. These names are collected from the returned result e (depvar). Instead of displaying both names, we will display the index of result sets we have collected for our table by typing command (index).

| . etable, showeq column(index) |  |
| :--- | ---: |
|  | 1 |
| Systolic blood pressure |  |
| Age (years) | 0.638 |
|  | $(0.011)$ |
| Weight (kg) | 0.407 |
|  | $(0.012)$ |
| Intercept | 71.271 |
|  | $(1.042)$ |
| Diastolic blood pressure | 0.188 |
| Age (years) | $(0.007)$ |
|  | 0.312 |
| Weight (kg) | $(0.007)$ |
|  | 50.376 |
| Intercept | $(0.616)$ |
|  | 10351 |

## Appendix

## Colors

bgcolor, fgcolor, and color

| aliceblue | darkslategray | lightsalmon | palevioletred |
| :---: | :---: | :---: | :---: |
| antiquewhite | darkturquoise | lightseagreen | papayawhip |
| aqua | darkviolet | lightskyblue | peachpuff |
| aquamarine | deeppink | lightslategray | peru |
| azure | deepskyblue | lightsteelblue | pink |
| beige | dimgray | lightyellow | plum |
| bisque | dodgerblue | lime | powderblue |
| black | firebrick | limegreen | purple |
| blanchedalmond | floralwhite | linen | red |
| blue | forestgreen | magenta | rosybrown |
| blueviolet | fuchsia | maroon | royalblue |
| brown | gainsboro | mediumaquamarine | saddlebrown |
| burlywood | ghostwhite | mediumblue | salmon |
| cadetblue | gold | mediumorchid | sandybrown |
| chartreuse | goldenrod | mediumpurple | seagreen |
| chocolate | gray | mediumseagreen | seashell |
| coral | green | mediumslateblue | sienna |
| cornflowerblue | greenyellow | mediumspringgreen | silver |
| cornsilk | honeydew | mediumturquoise | skyblue |
| crimson | hotpink | mediumvioletred | slateblue |
| cyan | indianred | midnightblue | slategray |
| darkblue | indigo | mintcream | snow |
| darkcyan | ivory | mistyrose | springgreen |
| darkgoldenrod | khaki | moccasin | steelblue |
| darkgray | lavender | navajowhite | $\tan$ |
| darkgreen | lavenderblush | navy | teal |
| darkkhaki | lawngreen | oldlace | thistle |
| darkmagenta | lemonchiffon | olive | tomato |
| darkolivegreen | lightblue | olivedrab | turquoise |
| darkorange | lightcoral | orange | violet |
| darkorchid | lightcyan | orangered | wheat |
| darkred | lightgoldenrodyellow | orchid | white |
| darksalmon | lightgray | palegoldenrod | whitesmoke |
| darkseagreen | lightgreen | palegreen | yellow |
| darkslateblue | lightpink | paleturquoise | yellowgreen |

## Underline patterns

upattern

| none | dashLong |
| :--- | :--- |
| single | dashLongHeavy |
| words | dotDash |
| double | dashDotHeavy |
| thick | dotDotDash |
| dotted | dashDotDotHeavy |
| dottedHeavy | wave |
| dash | wavyHeavy |
| dashedHeavy | wavyDouble |

## Shading patterns

fpattern

```
nil
clear
pct20
pct25
solid pct30
horzStripe pct35
vertStripe pct37
reverseDiagStripe pct40
diagStripe pct45
horzCross pct50
diagCross pct55
thinHorzStripe pct60
thinVertStripe pct62
thinReverseDiagStripe pct65
thinDiagStripe pct70
thinHorzCross pct75
thinDiagCross pct80
pct5 pct85
pct10 pct87
pct12 pct90
pct15 pct95
```


## Acknowledgments

We thank Roger Newson of the King's College London, UK, Ben Jann of the Institute of Sociology at the University of Bern, Switzerland, and John Luke Gallup of Portland State University for writing similar commands for exporting tables of estimation results.

## Reference

McDowell, A., A. Engel, J. T. Massey, and K. Maurer. 1981. Plan and operation of the Second National Health and Nutrition Examination Survey, 1976-1980. Vital and Health Statistics 1(15): 1-144.

## Also see

[R] table intro - Introduction to tables of frequencies, summaries, and command results [TABLES] Intro - Introduction

## Title

exit — Exit Stata

## Description Syntax Option Remarks and examples Also see

## Description

Typing exit causes Stata to stop processing and return control to the operating system. If the dataset in memory has changed since the last save command, you must specify the clear option before Stata will let you exit.

If you wish to use exit in do-files or programs to set return codes or terminate programs, see [ P ] exit.

Stata for Windows users may also exit Stata by clicking on the Close button or by pressing Alt + F4.
Stata for Mac users may also exit Stata by pressing Command+Q.
Stata for Unix(GUI) users may also exit Stata by clicking on the Close button.

## Syntax

```
exit [, clear]
```


## Option

clear permits you to exit, even if the current dataset has not been saved.

## Remarks and examples

Type exit to leave Stata and return to the operating system. If the dataset in memory has changed since the last time it was saved, however, Stata will refuse. At that point, you can either save the dataset and then type exit, or type exit, clear:

```
. exit
no; dataset in memory has changed since last saved
r(4);
. exit, clear
```


## Also see

[P] exit - Exit from a program or do-file

## Title

exlogistic - Exact logistic regression

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

exlogistic fits an exact logistic regression model, which produces more accurate inference in small samples than the standard maximum-likelihood-based logistic regression estimator. It can also better deal with completely determined outcomes. exlogistic with the group() option conditions on the number of positive outcomes within stratum and is an alternative to the conditional (fixed-effects) logistic regression estimator.

Unlike Stata's other estimation commands, exlogistic must perform hypothesis tests during estimation rather than after estimation with standard postestimation commands.

## Quick start

Exact logistic regression of y on $\mathrm{x} 1, \mathrm{x} 2$, and x 3
exlogistic y x1 x2 x3
Same as above, but condition on values of $x 3$ to save time and memory

```
exlogistic y x1 x2, condvars(x3)
```

Same as above, and allow more memory for computing the conditional distribution of sufficient statistics
exlogistic y x1 x2, condvars(x3) memory (100m)
Using data stored in binomial form with ys successes out of n trials
exlogistic ys x1 x2 x3, binomial(n)
Report coefficients rather than odds ratios
exlogistic y x1 x2 x3, coef
Report conditional scores tests
exlogistic y x1 x2 x3, test(score)
Fit a model with strata identified by svar
exlogistic y x1 x2 x3, group(svar)

## Menu

Statistics $>$ Exact statistics $>$ Exact logistic regression

## Syntax

exlogistic depvar indepvars [if] [in] [weight] [, options]
depvar can be specified as a zero or nonzero variable or the number of positive outcomes within each trial. For a zero or nonzero variable, zero indicates failure and nonzero indicates success. To specify depvar as the number of positive outcomes, you must also specify binomial (varname|\#).

## options

Model
condvars (varlist ${ }_{c}$ )
group (varname)
binomial (varname|\#)
estconstant
noconstant
Options
memory (\# $[\mathrm{b}|\mathrm{k}| \mathrm{m} \mid \mathrm{g}]$ ) set limit on memory usage; default is memory ( 10 m )
saving (filename [, replace]) save the joint conditional distribution to filename
Reporting
level(\#)
coef
test (testopt)
mue (varlist ${ }_{m}$ )
midp
[no]log
display_options
coeflegend

Description
condition on variables in varlist $_{c}$
groups or strata are stratified by unique values of varname data are in binomial form and the number of trials is contained in varname or in \#
estimate constant term; do not condition on the number of successes
suppress constant term
set confidence level; default is level (95)
report estimated coefficients
report $p$-value for observed sufficient statistic, conditional scores test, or conditional probabilities test
compute the median unbiased estimates for varlist $_{m}$ use the mid- $p$-value rule
display or suppress the enumeration log; default is to display control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
display legend instead of statistics
indepvars, varlist $_{c}$, and varlist ${ }_{m}$ may contain factor variables; see [U] 11.4.3 Factor variables.
by, collect, and statsby are allowed; see [U] 11.1.10 Prefix commands.
fweights are allowed; see [U] 11.1.6 weight.
coeflegend does not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

> Model
> condvars ( varlist $_{c}$ ) specifies variables whose parameter estimates are not of interest to you. You can save substantial computer time and memory by moving such variables from indepvars to condvars (). Understand that you will get the same results for $x 1$ and $x 3$ whether you type
or
. exlogistic y x1 x3, condvars(x2 x4)
group (varname) specifies the variable defining the strata, if any. A constant term is assumed for each stratum identified in varname, and the sufficient statistics for indepvars are conditioned on the observed number of successes within each group. This makes the fitted model equivalent to that fit by clogit, Stata's conditional logistic regression command (see [R] clogit). group() may not be specified with estconstant or noconstant.
binomial (varname |\#) indicates that the data are in binomial form and depvar contains the number of successes. varname contains the number of trials for each observation. If all observations have the same number of trials, you can instead specify the number as an integer. The number of trials must be a positive integer at least as great as the number of successes. If binomial () is not specified, the data are assumed to be Bernoulli, meaning that depvar equaling zero or nonzero records one failure or success.
estconstant estimates the constant term. By default, the models are assumed to have an intercept (constant), but the value of the intercept is not calculated. That is, the conditional distribution of the sufficient statistics for the indepvars is computed given the number of successes in depvar, thus conditioning out the constant term of the model. Use estconstant if you want the estimate of the intercept reported. estconstant may not be specified with group().
noconstant; see [R] Estimation options. noconstant may not be specified with group().

Options
memory (\# $[\mathrm{b}|\mathrm{k}| \mathrm{m} \mid \mathrm{g}]$ ) sets a limit on the amount of memory exlogistic can use when computing the conditional distribution of the parameter sufficient statistics. The default is memory $(10 \mathrm{~m})$, where m stands for megabyte, or $1,048,576$ bytes. The following are also available: b stands for byte; k stands for kilobyte, which is equal to 1,024 bytes; and g stands for gigabyte, which is equal to 1,024 megabytes. The minimum setting allowed is 1 m , and the maximum is 2048 m or 2 g , but do not attempt to use more memory than is available on your computer. Also see the first technical note under example 4 on counting the conditional distribution.
saving (filename [, replace]) saves the joint conditional distribution to filename. This distribution is conditioned on those variables specified in condvars(). Use replace to replace an existing file with filename. A Stata data file is created containing all the feasible values of the parameter sufficient statistics. The variable names are the same as those in indepvars, in addition to a variable named _f_ containing the feasible value frequencies (sometimes referred to as the condition numbers).

## Reporting

level(\#); see [R] Estimation options. The level() option will not work on replay because confidence intervals are based on estimator-specific enumerations. To change the confidence level, you must refit the model.
coef reports the estimated coefficients rather than odds ratios (exponentiated coefficients). coef may be specified when the model is fit or upon replay. coef affects only how results are displayed and not how they are estimated.
test (sufficient |score $\mid$ probability) reports the $p$-value associated with the observed sufficient statistics, the conditional scores tests, or the conditional probabilities tests, respectively. The default is test (sufficient). If factor variables are included in the specification, the conditional scores test and the conditional probabilities test are applied to each term providing conditional inference for several parameters simultaneously. All the statistics are computed at estimation time regardless
of which is specified. Each statistic may thus also be displayed when replaying results after estimation without having to refit the model; see $[\mathrm{R}]$ exlogistic postestimation.
mue ( varlist $_{m}$ ) specifies that median unbiased estimates (MUEs) be reported for the specified variables. By default, the conditional maximum likelihood estimates (CMLEs) are reported, except for those parameters for which the CMLEs are infinite. Specify mue (_all) if you want MUEs for all the indepvars.
midp instructs exlogistic to use the mid- $p$-value rule when computing the MUEs, $p$-values, and confidence intervals. This adjustment is for the discreteness of the distribution and halves the value of the discrete probability of the observed statistic before adding it to the $p$-value. The mid- $p$-value rule cannot be applied to MUEs whose corresponding parameter CMLE is infinite.
log and nolog specify whether to display the enumeration log, which shows the progress of computing the conditional distribution of the sufficient statistics. The enumeration $\log$ is displayed by default unless you used set iterlog off to suppress it; see set iterlog in [R] set iter.
display_options: noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), and sformat (\% firt); see [R] Estimation options.

Note that the maximum widths for cformat(), pformat(), and sformat() differ from those widths listed in [R] Estimation options. The maximum width for each format is 9 for exlogistic.

The following option is available with exlogistic but is not shown in the dialog box:
coeflegend; see [R] Estimation options.

## Remarks and examples

Exact logistic regression is the estimation of the logistic model parameters by using the conditional distribution of the parameter sufficient statistics. The estimates are referred to as the conditional maximum likelihood estimates (CMLEs). This technique was first introduced by Cox and Snell (1989) as an alternative to using maximum likelihood estimation, which can perform poorly for small sample sizes. For stratified data, exact logistic regression is a small-sample alternative to conditional logistic regression. See $[R]$ logit, $[R]$ logistic, and $[R]$ clogit to obtain maximum likelihood estimates (MLEs) for the logistic model and the conditional logistic model. For a comprehensive overview of exact logistic regression, see Mehta and Patel (1995).

Let $Y_{i}$ denote a Bernoulli random variable where we observe the outcome $Y_{i}=y_{i}, i=1, \ldots, n$. Associated with each independent observation is a $1 \times p$ vector of covariates, $\mathbf{x}_{i}$. We will denote $\pi_{i}=\operatorname{Pr}\left(Y_{i} \mid \mathbf{x}_{i}\right)$ and let the logit function model the relationship between $Y_{i}$ and $\mathbf{x}_{i}$,

$$
\log \left(\frac{\pi_{i}}{1-\pi_{i}}\right)=\theta+\mathbf{x}_{i} \boldsymbol{\beta}
$$

where the constant term $\theta$ and the $p \times 1$ vector of regression parameters $\boldsymbol{\beta}$ are unknown. The probability of observing $Y_{i}=y_{i}, i=1, \ldots, n$, is

$$
\operatorname{Pr}(\mathbf{Y}=\mathbf{y})=\prod_{i=1}^{n} \pi_{i}^{y_{i}}\left(1-\pi_{i}\right)^{1-y_{i}}
$$

where $\mathbf{Y}=\left(Y_{1}, \ldots, Y_{n}\right)$ and $\mathbf{y}=\left(y_{1}, \ldots, y_{n}\right)$. The MLEs for $\theta$ and $\boldsymbol{\beta}$ maximize the log of this function.

The sufficient statistics for $\theta$ and $\beta_{j}, j=1, \ldots, p$, are $M=\sum_{i=1}^{n} Y_{i}$ and $T_{j}=\sum_{i=1}^{n} Y_{i} x_{i j}$, respectively, and we observe $M=m$ and $T_{j}=t_{j}$. By default, exlogistic tallies the conditional distribution of $\mathbf{T}=\left(T_{1}, \ldots, T_{p}\right)$ given $M=m$. This distribution will have a size of $\binom{n}{m}$. (It would have a size of $2^{n}$ without conditioning on $M=m$.) Denote one of these vectors $\mathbf{T}^{(k)}=$ $\left(t_{1}^{(k)}, \ldots, t_{p}^{(k)}\right), k=1, \ldots, N$, with combinatorial coefficient (frequency) $c_{k}, \sum_{k=1}^{N} c_{k}=\binom{n}{m}$. For each independent variable $x_{j}, j=1, \ldots, p$, we reduce the conditional distribution further by conditioning on all other observed sufficient statistics $T_{l}=t_{l}, l \neq j$. The conditional probability of observing $T_{j}=t_{j}$ has the form

$$
\operatorname{Pr}\left(T_{j}=t_{j} \mid T_{l}=t_{l}, l \neq j, M=m\right)=\frac{c e^{t_{j} \beta_{j}}}{\sum_{k} c_{k} e^{t_{j}^{(k)} \beta_{j}}}
$$

where the sum is over the subset of $\mathbf{T}$ vectors such that $\left(T_{1}^{(k)}=t_{1}, \ldots, T_{j}^{(k)}=t_{j}^{(k)}, \ldots, T_{p}^{(k)}=t_{p}\right)$ and $c$ is the combinatorial coefficient associated with the observed $\mathbf{t}$. The CMLE for $\beta_{j}$ maximizes the $\log$ of this function.

Specifying nuisance variables in condvars() will reduce the size of the conditional distribution by conditioning on their observed sufficient statistics as well as conditioning on $M=m$. This reduces the amount of memory consumed at the cost of not obtaining regression estimates for those variables specified in condvars().

Inferences from MLEs rely on asymptotics, and if your sample size is small, these inferences may not be valid. On the other hand, inferences from the CMLEs are exact in that they use the conditional distribution of the sufficient statistics outlined above.

For small datasets, the dependent variable can be completely determined by the data. Here the MLEs and the CMLEs are unbounded. When this occurs, exlogistic will compute the MUE, the regression estimate that places the observed sufficient statistic at the median of the conditional distribution.

## > Example 1

One example presented by Mehta and Patel (1995) is data from a prospective study of perinatal infection and human immunodeficiency virus type 1 (HIV-1). We use a variation of this dataset. There was an investigation (Hutto et al. 1991) into whether the blood serum levels of glycoproteins CD4 and CD8 measured in infants at 6 months of age might predict their development of HIV infection. The blood serum levels are coded as ordinal values 0,1 , and 2 .

```
. use https://www.stata-press.com/data/r18/hiv1
(Prospective study of perinatal infection of HIV-1)
. list in 1/5
```

| hiv | cd4 | cd8 |
| ---: | ---: | ---: |
| Positive | 0 | 0 |
| Negative | 0 | 0 |
| Positive | 0 | 2 |
| Positive | 1 | 0 |
| Negative | 1 | 0 |

We first obtain the MLEs from logistic so that we can compare the estimates and associated statistics with the CMLEs from exlogistic.

| Logistic regression |  |  |  |  | Number of obs <br> LR chi2(2) <br> Prob > chi2 <br> Pseudo R2 | $\begin{array}{lr} = & 47 \\ = & 15.75 \\ = & 0.0004 \\ = & 0.2751 \end{array}$ |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| hiv | Coefficient | Std. err. | z | $P>\|z\|$ | [95\% conf. | interval] |
| cd4 | -2.541669 | . 8392231 | -3.03 | 0.002 | -4.186517 | -. 8968223 |
| cd8 | 1.658586 | . 821113 | 2.02 | 0.043 | . 0492344 | 3.267938 |
| _cons | . 5132389 | . 6809007 | 0.75 | 0.451 | -. 8213019 | 1.84778 |

. exlogistic hiv cd4 cd8, coef
Enumerating sample-space combinations:
Observation 1: Enumerations = 2
Observation 2: Enumerations $=3$
(output omitted)
Observation 46: Enumerations $=601$
Observation 47: Enumerations $=326$
Exact logistic regression Number of obs $=\quad 47$
Model score $=13.34655$
$\operatorname{Pr}>=$ score $=0.0006$

| hiv | Coefficient | Suff. | $2 * \operatorname{Pr}($ Suff. $)$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| cd4 | -2.387632 | 10 | 0.0004 | -4.699633 | -.8221807 |
| cd8 | 1.592366 | 12 | 0.0528 | -.0137905 | 3.907876 |

exlogistic produced a $\log$ showing how many records are generated as it processes each observation. The primary purpose of the $\log$ is to provide feedback because generating the distribution can be time consuming, but we also see from the last entry that the joint distribution for the sufficient statistics for cd 4 and cd8 conditioned on the total number of successes has 326 unique values (but a size of $\left.\binom{47}{14}=341,643,774,795\right)$.

The statistics for logistic are based on asymptotics: for a large sample size, each $Z$ statistic will be approximately normally distributed (with a mean of zero and a standard deviation of one) if the associated regression parameter is zero. The question is whether a sample size of 47 is large enough.

On the other hand, the $p$-values computed by exlogistic are from the conditional distributions of the sufficient statistics for each parameter given the sufficient statistics for all other parameters. In this sense, these $p$-values are exact. By default, exlogistic reports the sufficient statistics for the regression parameters and the probability of observing a more extreme value. These are singleparameter tests for $H_{0}: \beta_{\mathrm{cd} 4}=0$ and $H_{0}: \beta_{\mathrm{cd} 8}=0$ versus the two-sided alternatives. The conditional scores test, located in the coefficient table header, is testing that both $H_{0}: \beta_{\mathrm{cd} 4}=0$ and $H_{0}: \beta_{\mathrm{cd} 8}=0$. We find these $p$-values to be in fair agreement with the Wald and likelihood-ratio tests from logistic.

The confidence intervals for exlogistic are computed from the exact conditional distributions. The exact confidence intervals are asymmetrical about the estimate and are wider than the normal-based confidence intervals from logistic.

Both estimation techniques indicate that the incidence of HIV infection decreases with increasing CD4 blood serum levels and increases with increasing CD8 blood serum levels. The constant term is
missing from the exact logistic coefficient table because we conditioned out its observed sufficient statistic when tallying the joint distribution of the sufficient statistics for the cd4 and cd8 parameters.

The test () option provides two other test statistics used in exact logistic regression: the conditional scores test, test (score), and the conditional probabilities test, test (probability). For comparison, we display the individual parameter conditional scores tests.

| Exact logistic regression |  |  |  | Number of obs <br> Model score <br> Pr >= score | $\begin{array}{lr} = & 47 \\ = & 13.34655 \\ = & 0.0006 \end{array}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| hiv | Coefficient | Score | Pr $>=$ Score | [95\% conf. | interval] |
| cd4 | -2.387632 | 12.88022 | 0.0003 | -4.699633 | -. 8221807 |
| cd8 | 1.592366 | 4.604816 | 0.0410 | -. 0137905 | 3.907876 |

For the probabilities test, the probability statistic is computed from (1) in Methods and formulas with $\boldsymbol{\beta}=0$. For this example, the $p$-value for the conditional probabilities tests matches the conditional scores tests, so they are not displayed here.

## - Technical note

Typically, the value of $\theta$, the constant term, is of little interest, as well as perhaps some of the parameters in $\boldsymbol{\beta}$, but we need to include all parameters in the model to correctly specify it. By conditioning out the nuisance parameters, we can reduce the size of the joint conditional distribution that is used to estimate the regression parameters of interest. The condvars () option allows you to specify a varlist of nuisance variables. By default, exlogistic conditions on the sufficient statistic of $\theta$, which is the number of successes. You can save computation time and computer memory by using the condvars() option because infeasible values of the sufficient statistics associated with the variables in condvars () can be omitted from consideration before all $n$ observations are processed.

Specifying some of your independent variables in condvars() will not change the estimated regression coefficients of the remaining independent variables. For instance, in example 1, if we instead type
. exlogistic hiv cd4, condvars(cd8) coef
the regression coefficient for $\mathrm{cd4}$ (as well as all associated inference) will be identical.

- Technical note

If you fit a clogit (see [R] clogit) model to the HIV data from example 1, you will find that the estimates differ from those with exlogistic. (To fit the clogit model, you will have to create a group variable that includes all observations.) The regression estimates will be different because clogit conditions on the constant term only, whereas the estimates from exlogistic condition on the sufficient statistic of the other regression parameter as well as the constant term.

## Example 2

The HIV data presented in table IV of Mehta and Patel (1995) are in a binomial form, where the variable hiv contains the HIV cases that tested positive and the variable $n$ contains the number of individuals with the same CD4 and CD8 levels, the binomial number-of-trials parameter. Here depvar is hiv, and we use the binomial(n) option to identify the number-of-trials variable.

```
. use https://www.stata-press.com/data/r18/hiv_n
(Prospective study of perinatal infection of HIV-1; binomial form)
. list
```

| cd4 | cd8 | hiv | n |
| ---: | ---: | ---: | ---: |
| 0 | 2 | 1 | 1 |
| 1 | 2 | 2 | 2 |
| 0 | 0 | 4 | 7 |
| 1 | 1 | 4 | 12 |
| 2 | 2 | 1 | 3 |
| 1 | 0 | 2 | 7 |
| 2 | 0 | 0 | 2 |
| 2 | 1 | 0 | 13 |

Further, the cd4 and cd8 variables are coded with ordinal values $(0,1,2)$. Another approach is to specify these variables as factor variables in the model so that indicators for all but the base level of cd4 and of cd8 will be included in the model. Similar to Mehta and Patel (1995), we will estimate the odds ratio of level 0 versus 2 and level 1 versus 2 by typing ib2.cd4 and ib2.cd8, indicating that 2 is the base level.


```
(*) median unbiased estimates (MUE)
. matrix list e(sufficient)
e(sufficient)[1,6]
            0. 1. 2b. 0. 1. 2b.
        cd4
. display e(n_possible)
1091475
```

Because we included cd4 and cd8 as factor variables, we obtained a conditional probabilities test for cd4, simultaneously testing both $0 . \operatorname{cd4}$ and $1 . c d 4$, and for cd8, simultaneously testing both $0 . c d 8$ and $1 . c d 8$. The $p$-values for the two terms are 0.0055 and 0.0323 , respectively.

This example also illustrates instances where the dependent variable is completely determined by the independent variables and CMLEs are infinite. If we try to obtain MLEs, logistic will omit each variable and then terminate with a no-data error, error number 2000.

```
. use https://www.stata-press.com/data/r18/hiv_n, clear
(Prospective study of perinatal infection of HIV-1; binomial form)
. expand n
(39 observations created)
```

. capture logistic hiv ib2.cd4 ib2.cd8

The previous exlogistic command generated the joint conditional distribution of $T_{0 . \operatorname{cd} 4}, T_{1 . \operatorname{cd} 4}$, $T_{0 . c d 8}$, and $T_{1 . \mathrm{cd} 8}$ given $M=14$ (the number of individuals that tested positive), and for reference, we listed the observed sufficient statistics that are stored in the matrix e(sufficient). Below, we take that distribution and further condition on $T_{1 . \mathrm{cd} 4}=8, T_{0 . \mathrm{cd} 8}=6$, and $T_{1 . \mathrm{cd} 8}=4$, giving the conditional distribution of $T_{0 . c d 4}$. Here we see that the observed sufficient statistic $T_{0 . c d 4}=5$ is last in the sorted listing or, equivalently, $T_{0 . c d 4}$ is at the domain boundary of the conditional probability distribution. When this occurs, the conditional probability distribution is monotonically increasing in $\beta_{0 . c d 4}$ and a maximum does not exist.

```
. use dist, clear
. keep if __1_cd4==8 & _0_cd8==6 & _1_cd8==4
(4,139 observations deleted)
. list, sep(0)
```

| _f_ | _0_cd4 | _1_cd4 | _0_cd8 | _1_cd8 | _2b_cd4 | _2b_cd8 |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| 1668667 | 0 | 8 | 6 | 4 | 0 | 0 |
| 18945542 | 1 | 8 | 6 | 4 | 0 | 0 |
| 55801053 | 2 | 8 | 6 | 4 | 0 | 0 |
| 55867350 | 3 | 8 | 6 | 4 | 0 | 0 |
| 17423175 | 4 | 8 | 6 | 4 | 0 | 0 |
| 1091475 | 5 | 8 | 6 | 4 | 0 | 0 |

When the CMLEs are infinite, the MUEs are computed (Hirji, Tsiatis, and Mehta 1989). For the 0.cd4 estimate, we compute the value $\bar{\beta}_{0 . c d 4}$ such that

$$
\operatorname{Pr}\left(T_{0 . \mathrm{cd} 4} \geq 5 \mid \beta_{0 . \mathrm{cd} 4}=\bar{\beta}_{0 . \mathrm{cd} 4}, T_{1 . \mathrm{cd} 4}=8, T_{0 . \mathrm{cd} 8}=6, T_{1 . \mathrm{cd} 8}=4, M=14\right)=1 / 2
$$

using (1) in Methods and formulas.
The output is in agreement with example 1: there is an increase in risk of HIV infection for a CD4 blood serum level of 0 relative to a level of 2 and for a level of 1 relative to a level of 2 ; there is a decrease in risk of HIV infection for a CD8 blood serum level of 0 relative to a level of 2 and for a level of 1 relative to a level of 2.

We also displayed e(n_possible). This is the combinatorial coefficient associated with the observed sufficient statistics. The same value is found in the _f_ variable of the conditional distribution dataset listed above. The size of the distribution is $\binom{47}{14}=341,643,774,795$. This can be verified by summing the $\mathrm{f}_{\text {_ }}$ variable of the generated conditional distribution dataset.

```
. use dist, clear
. summarize _f_, meanonly
. di %15.1f r(sum)
341643774795.0
```


## > Example 3

One can think of exact logistic regression as a covariate-adjusted exact binomial. To demonstrate this point, we will use exlogistic to compute a binomial confidence interval for $m$ successes of $n$ trials by fitting the constant-only model, and we will compare it with the confidence interval computed by ci proportions (see [R] ci). We will use the saving() option to retain the dataset containing the feasible values for the constant term sufficient statistic, namely, the number of successes, $m$, given $n$ trials and their associated combinatorial coefficients $\binom{n}{m}, m=0,1, \ldots, n$.
. input y
y

1. 1
2. 0
3. 1
4. 0
5. 1
6. 1
7. end
. ci proportions y

| Variable | Obs | Proportion |  | Std. err. | [95\% conf. interval] exact |  |
| ---: | ---: | ---: | ---: | ---: | ---: | :---: |
| $y$ | 6 | .6666667 | .1924501 | .2227781 | .9567281 |  |

. exlogistic y, estconstant nolog coef saving(binom)
note: saving distribution to file binom.dta.
Exact logistic regression

$$
\text { Number of obs }=6
$$

| $y$ | Coefficient | Suff. | $2 * \operatorname{Pr}($ Suff. $)$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| _cons | .6931472 | 4 | 0.6875 | -1.24955 | 3.096017 |

We use the postestimation command estat predict to transform the estimated constant term and its confidence bounds by using the inverse logit function, invlogit() (see [FN] Mathematical functions). The standard error for the estimated probability is computed using the delta method.
. estat predict

| $y$ | Predicted | Std. err. | [95\% conf. interval] |  |
| ---: | ---: | :---: | ---: | ---: |
| Probability | 0.6667 | 0.1925 | 0.2228 | 0.9567 |

. use binom, replace
. list, sep(0)

| -f | _cons_ |
| ---: | ---: |
| 1 | 0 |
| 6 | 1 |
| 15 | 2 |
| 20 | 3 |
| 15 | 4 |
| 6 | 5 |
| 1 | 6 |

Examining the listing of the generated data, the values contained in the variable _cons_ are the feasible values of $M$, and the values contained in the variable $\_\mathrm{f}$ - are the binomial coefficients $\binom{6}{m}$ with total $\sum_{m=0}^{6}\binom{6}{m}=2^{6}=64$. In the coefficient table, the sufficient statistic for the constant term, labeled Suff., is $m=4$. This value is located at record 5 of the dataset. Therefore, the two-tailed probability of the sufficient statistic is computed as $0.6875=2(15+6+1) / 64$.

The constant term is the value of $\theta$ that maximizes the probability of observing $M=4$; see (1) of Methods and formulas:

$$
\operatorname{Pr}(M=4 \mid \theta)=\frac{15 e^{4 \alpha}}{1+6 e^{\alpha}+15 e^{2 \alpha}+20 e^{3 \alpha}+15 e^{4 \alpha}+6 e^{5 \alpha}+e^{6 \alpha}}
$$

The maximum is at the value $\theta=\log 2$, which is demonstrated in the figure below.


The lower and upper confidence bounds are the values of $\theta$ such that $\operatorname{Pr}(M \geq 4 \mid \theta)=0.025$ and $\operatorname{Pr}(M \leq 4 \mid \theta)=0.025$, respectively. These probabilities are plotted in the figure below for $\theta \in[-2,4]$.

> Example 4
This example demonstrates the group() option, which allows the analysis of stratified data. Here the logistic model is

$$
\log \left(\frac{\pi_{i k}}{1-\pi_{i k}}\right)=\theta_{k}+\mathbf{x}_{k i} \boldsymbol{\beta}
$$

where $k$ indexes the $s$ strata, $k=1, \ldots, s$, and $\theta_{k}$ is the strata-specific constant term whose sufficient statistic is $M_{k}=\sum_{i=1}^{n_{k}} Y_{k i}$.

Mehta and Patel (1995) use a case-control study to demonstrate this model, which is useful in comparing the estimates from exlogistic and clogit. This study was intended to determine the role of birth complications in people with schizophrenia (Garsd 1988). Siblings from seven families took part in the study, and each individual was classified as normal or schizophrenic. A birth complication index is recorded for each individual that ranges from 0 , an uncomplicated birth, to 15 , a very complicated birth. Some of the frequencies contained in variable $f$ are greater than 1 , and these count different births at different times where the individual has the same birth complications index, found in variable BCindex.
. use https://www.stata-press.com/data/r18/schizophrenia, clear (Case-control study on birth complications for people with schizophrenia) . list, sepby(family)

|  | family | BCindex | schizo | $f$ |
| :--- | ---: | ---: | ---: | ---: |
| 1. | 1 | 6 | 0 | 1 |
| 2. | 1 | 7 | 0 | 1 |
| 3. | 1 | 3 | 0 | 2 |
| 4. | 1 | 2 | 0 | 3 |
| 5. | 1 | 5 | 0 | 1 |
| 6. | 1 | 0 | 0 | 1 |
| 7. | 1 | 15 | 1 | 1 |
| 8. | 2 | 2 | 1 | 1 |
| 9. | 2 | 0 | 0 | 1 |
| 10. | 3 | 2 | 0 | 1 |
| 11. | 3 | 9 | 1 | 1 |
| 12. | 3 | 1 | 0 | 1 |
| 13. | 4 | 2 | 1 | 1 |
| 14. | 4 | 0 | 0 | 4 |
| 15. | 5 | 3 | 1 | 1 |
| 16. | 5 | 6 | 0 | 1 |
| 17. | 5 | 0 | 1 | 1 |
| 18. | 6 | 3 | 0 | 1 |
| 19. | 6 | 0 | 1 | 1 |
| 20. | 6 | 0 | 0 | 2 |
| 21. | 7 | 2 | 0 | 1 |
| 22. | 7 | 6 | 1 | 1 |
|  |  |  |  |  |


| Enumerating sample-space combinations: |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Observation 1: Enumerations = 2 |  |  |  |  |  |  |
| Observation 2: Enumerations |  |  | 3 |  |  |  |
| Observation 3: Enumerations |  |  | 4 |  |  |  |
| Observation 4: Enumerations |  |  | 5 |  |  |  |
| Observation 5: Enumerations |  |  | 6 |  |  |  |
| Observation 6: Enumerations (output omitted) |  |  | 7 |  |  |  |
| Observation 21: Enumerations |  |  | 72 |  |  |  |
| Observation 22: Enumerations = |  |  | 40 |  |  |  |
| Exact logistic regression Group variable: family |  |  | Number of obs |  |  | 29 |
|  |  |  | Number of groups Obs per group: |  |  | 7 |
|  |  |  | Obs per group. ${ }_{\text {min }}$ |  |  | 2 |
|  |  |  | avg |  |  | 4.1 |
|  |  |  | max |  |  | 10 |
|  |  |  | Model score |  |  | 6.328033 |
|  |  |  | $\operatorname{Pr}>=$ score |  |  | 0.0167 |
| schizo | Coefficient | Score | Pr $>=$ Score | [95\% conf | . | nterval] |
| BCindex | . 3251178 | 6.328033 | 0.0167 | . 0223423 |  | . 7408832 |

The asymptotic alternative for this model can be estimated using clogit (equivalently, xtlogit, fe) and is listed below for comparison. We must expand the data because clogit will not accept frequency weights if they are not constant within the groups.

```
. expand f
(7 observations created)
. clogit schizo BCindex, group(family) nolog
note: multiple positive outcomes within groups encountered.
Conditional (fixed-effects) logistic regression Number of obs = 29
    LR chi2(1) = 5.20
    Prob > chi2 = 0.0226
    Pseudo R2 = 0.2927
```

| schizo | Coefficient | Std. err. | $z$ | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | :---: | :---: |
| BCindex | .3251178 | .1678981 | 1.94 | 0.053 | -.0039565 | .654192 |

Both techniques compute the same regression estimate for the BCindex, which might not be too surprising because both estimation techniques condition on the total number of successes in each group. The difference lies in the $p$-values and confidence intervals. The $p$-value testing $H_{0}: \beta_{\mathrm{BCindex}}=0$ is approximately 0.0167 for the exact conditional scores test and 0.053 for the asymptotic Wald test. Moreover, the exact confidence interval is asymmetric about the estimate and does not contain zero.

## - Technical note

The memory () option limits the amount of memory that exlogistic will consume when computing the conditional distribution of the parameter sufficient statistics. memory () is independent of the data maximum memory setting (see set max_memory in [D] memory), and it is possible for exlogistic to exceed the memory limit specified in set max_memory without terminating. By default, a log is provided that displays the number of enumerations (the size of the conditional distribution)
after processing each observation. Typically, you will see the number of enumerations increase, and then at some point they will decrease as the multivariate shift algorithm (Hirji, Mehta, and Patel 1987) determines that some of the enumerations cannot achieve the observed sufficient statistics of the conditioning variables. When the algorithm is complete, however, it is necessary to store the conditional distribution of the parameter sufficient statistics as a dataset. It is possible, therefore, to get a memory error when the algorithm has completed if there is not enough memory to store the conditional distribution.

## Technical note

Computing the conditional distributions and reported statistics requires data sorting and numerical comparisons. If there is at least one single-precision variable specified in the model, exlogistic will make comparisons with a relative precision of $2^{-5}$. Otherwise, a relative precision of $2^{-11}$ is used. Be careful if you use recast to promote a single-precision variable to double precision (see [D] recast). You might try listing the data in full precision (maybe $\% 20.15 \mathrm{~g}$; see [D] format) to make sure that this is really what you want. See [D] Data types for information on precision of numeric storage types.

## Stored results

exlogistic stores the following in e() :
Scalars
e(N)
e(k_groups)
e(n_possible)
e(n_trials)
e(sum_y)
e(k_indvars)
e(k_condvars)
e(condcons)
e(midp)
e(eps)
Macros
e(cmd)
e(cmdline)
e(title)
e(depvar)
e(indvars)
e(condvars)
e(groupvar)
e(binomial)
e(fvvarlist)
e(level)
e(wtype)
e(wexp)
e(datasignature)
e(datasignaturevars)
e(properties)
e(estat_cmd)
e(marginsnotok) predictions disallowed by margins
number of observations
number of groups e(sufficient)
binomial number-of-trials parameter
sum of depvar
number of independent variables
number of conditioning variables
conditioned on the constant(s) indicator
mid- $p$-value rule indicator
relative difference tolerance
exlogistic
command as typed
title in estimation output
name of dependent variable
independent variables
conditional variables
group variable
binomial number-of-trials variable
independent factor variables
confidence level
weight type
weight expression
the checksum
variables used in calculation of checksum
b
program used to implement estat
number of distinct possible outcomes where sum(sufficient) equals observed

Matrices
e(b)
e(mue_indicators)
e(se)
e(ci)
e(sum_y_groups)
e(N_g)
e(sufficient)
e(p_sufficient)
e(scoretest)
e(p_scoretest)
e(probtest)
e(p_probtest)
e(scoretest_m)
e(p_scoretest_m)
e(probtest_m)
e(p_probtest_m)
Functions
e(sample)
coefficient vector
indicator for elements of $e(b)$ estimated using MUE instead of CMLE $e$ (b) standard errors (CMLEs only)
matrix of e(level) confidence intervals for e(b)
sum of e(depvar) for each group
number of observations in each group
sufficient statistics for e(b)
$p$-value for e(sufficient)
conditional scores tests for indepvars
$p$-values for e(scoretest)
conditional probabilities tests for indepvars
$p$-value for e(probtest)
conditional scores tests for model terms
$p$-value for e(scoretest_m)
conditional probabilities tests for model terms
$p$-value for e (probtest_m)
marks estimation sample

## Methods and formulas

Methods and formulas are presented under the following headings:
Sufficient statistics
Conditional distribution and CMLE
MUEs and exact confidence intervals
Conditional hypothesis tests
Sufficient-statistic p-value

## Sufficient statistics

Let $\left\{Y_{1}, Y_{2}, \ldots, Y_{n}\right\}$ be a set of $n$ independent Bernoulli random variables, each of which can realize two outcomes, $\{0,1\}$. For each $i=1, \ldots, n$, we observe $Y_{i}=y_{i}$, and associated with each observation is the covariate row vector of length $p, \mathbf{x}_{i}=\left(x_{i 1}, \ldots, x_{i p}\right)$. Denote $\boldsymbol{\beta}=\left(\beta_{1}, \ldots, \beta_{p}\right)^{T}$ to be the column vector of regression parameters and $\theta$ to be the constant. The sufficient statistic for $\beta_{j}$ is $T_{j}=\sum_{i=1}^{n} Y_{i} x_{i j}, j=1, \ldots, p$, and for $\theta$ is $M=\sum_{i=1}^{n} Y_{i}$. We observe $T_{j}=t_{j}, t_{j}=\sum_{i=1}^{n} y_{i} x_{i j}$, and $M=m, m=\sum_{i=1}^{n} y_{i}$. The probability of observing $\left(Y_{1}=y_{1}, Y_{2}=y_{2}, \ldots, Y_{n}=y_{n}\right)$ is

$$
\operatorname{Pr}\left(Y_{1}=y_{1}, \ldots, Y_{n}=y_{n} \mid \boldsymbol{\beta}, \mathbf{X}\right)=\frac{\exp (m \theta+\mathbf{t} \boldsymbol{\beta})}{\prod_{i=1}^{n}\left\{1+\exp \left(\theta+\mathbf{x}_{i} \boldsymbol{\beta}\right)\right\}}
$$

where $\mathbf{t}=\left(t_{1}, \ldots, t_{p}\right)$ and $\mathbf{X}=\left(\mathbf{x}_{1}^{T}, \ldots, \mathbf{x}_{n}^{T}\right)^{T}$.
The joint distribution of the sufficient statistics $\mathbf{T}$ is obtained by summing over all possible binary sequences $Y_{1}, \ldots, Y_{n}$ such that $\mathbf{T}=\mathbf{t}$ and $M=m$. This probability function is

$$
\operatorname{Pr}\left(T_{1}=t_{1}, \ldots, T_{p}=t_{p}, M=m \mid \boldsymbol{\beta}, \mathbf{X}\right)=\frac{c(\mathbf{t}, m) \exp (m \theta+\mathbf{t} \boldsymbol{\beta})}{\prod_{i=1}^{n}\left\{1+\exp \left(\theta+\mathbf{x}_{i} \boldsymbol{\beta}\right)\right\}}
$$

where $c(\mathbf{t}, m)$ is the combinatorial coefficient of $(\mathbf{t}, m)$ or the number of distinct binary sequences $Y_{1}, \ldots, Y_{n}$ such that $\mathbf{T}=\mathbf{t}$ and $M=m$ (Cox and Snell 1989).

## Conditional distribution and CMLE

Without loss of generality, we will restrict our discussion to computing the CMLE of $\beta_{1}$. If we condition on observing $M=m$ and $T_{2}=t_{2}, \ldots, T_{p}=t_{p}$, the probability function of $\left(T_{1} \mid \beta_{1}, T_{2}=\right.$ $\left.t_{2}, \ldots, T_{p}=t_{p}, M=m\right)$ is

$$
\begin{equation*}
\operatorname{Pr}\left(T_{1}=t_{1} \mid \beta_{1}, T_{2}=t_{2}, \ldots, T_{p}=t_{p}, M=m\right)=\frac{c(\mathbf{t}, m) e^{t_{1} \beta_{1}}}{\sum_{u} c\left(u, t_{2}, \ldots, t_{p}, m\right) e^{u \beta_{1}}} \tag{1}
\end{equation*}
$$

where the sum in the denominator is over all possible values of $T_{1}$ such that $M=m$ and $T_{2}=t_{2}, \ldots, T_{p}=t_{p}$ and $c\left(u, t_{2}, \ldots, t_{p}, m\right)$ is the combinatorial coefficient of $\left(u, t_{2}, \ldots, t_{p}, m\right)$ (Cox and Snell 1989). The CMLE for $\beta_{1}$ is the value $\widehat{\beta}_{1}$ that maximizes the $\log$ of (1). This optimization task is carried out by ml , using the conditional frequency distribution of $\left(T_{1} \mid T_{2}=t_{2}, \ldots, T_{p}=\right.$ $\left.t_{p}, M=m\right)$ as a dataset. Generating the joint conditional distribution is efficiently computed using the multivariate shift algorithm described by Hirji, Mehta, and Patel (1987).

Difficulties in computing $\widehat{\beta}_{1}$ arise if the observed $\left(T_{1}=t_{1}, \ldots, T_{p}=t_{p}, M=m\right)$ lies on the boundaries of the distribution of $\left(T_{1} \mid T_{2}=t_{2}, \ldots, T_{p}=t_{p}, M=m\right)$, where the conditional probability function is monotonically increasing (or decreasing) in $\beta_{1}$. Here the CMLE is plus infinity if it is on the upper boundary, $\operatorname{Pr}\left(T_{1} \leq t_{1} \mid T_{2}=t_{2}, \ldots, T_{p}=t_{p}, M=m\right)=1$, and is minus infinity if it is on the lower boundary of the distribution, $\operatorname{Pr}\left(T_{1} \geq t_{1} \mid T_{2}=t_{2}, \ldots, T_{p}=t_{p}, M=m\right)=1$. This concept is demonstrated in example 2. When infinite CMLEs occur, the MUE is computed.

## MUEs and exact confidence intervals

The MUE is computed using the technique outlined by Hirji, Tsiatis, and Mehta (1989). First, we find the values of $\beta_{1}^{(u)}$ and $\beta_{1}^{(l)}$ such that

$$
\begin{align*}
& \operatorname{Pr}\left(T_{1} \leq t_{1} \mid \beta_{1}=\beta_{1}^{(u)}, T_{2}=t_{2}, \ldots, T_{p}=t_{p}, M=m\right)=  \tag{2}\\
& \operatorname{Pr}\left(T_{1} \geq t_{1} \mid \beta_{1}=\beta_{1}^{(l)}, T_{2}=t_{2}, \ldots, T_{p}=t_{p}, M=m\right)=1 / 2
\end{align*}
$$

The MUE is then $\bar{\beta}_{1}=\left(\beta_{1}^{(l)}+\beta_{1}^{(u)}\right) / 2$. However, if $T_{1}$ is equal to the minimum of the domain of the conditional distribution, $\beta^{(l)}$ does not exist and $\bar{\beta}_{1}=\beta^{(u)}$. If $T_{1}$ is equal to the maximum of the domain of the conditional distribution, $\beta^{(u)}$ does not exist and $\bar{\beta}_{1}=\beta^{(l)}$.

Confidence bounds for $\beta$ are computed similarly, except that we substitute $\alpha / 2$ for $1 / 2$ in (2), where $1-\alpha$ is the confidence level. Here $\beta_{1}^{(l)}$ would then be the lower confidence bound and $\beta_{1}^{(u)}$ would be the upper confidence bound (see example 3).

## Conditional hypothesis tests

To test $H_{0}: \beta_{1}=0$ versus $H_{1}: \beta_{1} \neq 0$, we obtain the exact $p$-value from $\sum_{u \in E} f_{1}(u)-f_{1}\left(t_{1}\right) / 2$ if the mid- $p$-value rule is used and $\sum_{u \in E} f_{1}(u)$ otherwise. Here $E$ is a critical region, and we define $f_{1}(u)=\operatorname{Pr}\left(T_{1}=u \mid \beta_{1}=0, T_{2}=t_{2}, \ldots, T_{p}=t_{p}, M=m\right)$ for ease of notation. There are two popular ways to define the critical region: the conditional probabilities test and the conditional scores test (Mehta and Patel 1995). The critical region when using the conditional probabilities test is all values of the sufficient statistic for $\beta_{1}$ that have a probability less than or equal to that of the observed $t_{1}, E_{p}=\left\{u: f_{1}(u) \leq f_{1}\left(t_{1}\right)\right\}$. The critical region of the conditional scores test is defined as all values of the sufficient statistic for $\beta_{1}$ such that its score is greater than or equal to that of $t_{1}$,

$$
E_{s}=\left\{u:\left(u-\mu_{1}\right)^{2} / \sigma_{1}^{2} \geq\left(t_{1}-\mu_{1}\right)^{2} / \sigma_{1}^{2}\right\}
$$

Here $\mu_{1}$ and $\sigma_{1}^{2}$ are the mean and variance of $\left(T_{1} \mid \beta_{1}=0, T_{2}=t_{2}, \ldots, T_{p}=t_{p}, M=m\right)$.

The score statistic is defined as

$$
\left\{\frac{\partial \ell(\beta)}{\partial \beta}\right\}^{2}\left[-E\left\{\frac{\partial^{2} \ell(\beta)}{\partial \beta^{2}}\right\}\right]^{-1}
$$

evaluated at $H_{0}: \beta=0$, where $\ell$ is the log of (1). The score test simplifies to $(t-E[T \mid \beta])^{2} / \operatorname{var}(T \mid \beta)$ (Hirji 2006), where the mean and variance are computed from the conditional distribution of the sufficient statistic with $\beta=0$ and $t$ is the observed sufficient statistic.

## Sufficient-statistic p-value

The $p$-value for testing $H_{0}: \beta_{1}=0$ versus the two-sided alternative when $\left(T_{1}=t_{1} \mid T_{2}=\right.$ $\left.t_{2}, \ldots, T_{p}=t_{p}\right)$ is computed as $2 \times \min \left(p_{l}, p_{u}\right)$, where

$$
\begin{aligned}
p_{l} & =\frac{\sum_{u \leq t_{1}} c\left(u, t_{2}, \ldots, t_{p}, m\right)}{\sum_{u} c\left(u, t_{2}, \ldots, t_{p}, m\right)} \\
p_{u} & =\frac{\sum_{u \geq t_{1}} c\left(u, t_{2}, \ldots, t_{p}, m\right)}{\sum_{u} c\left(u, t_{2}, \ldots, t_{p}, m\right)}
\end{aligned}
$$

It is the probability of observing a more extreme $T_{1}$.

## References

Cox, D. R., and E. J. Snell. 1989. Analysis of Binary Data. 2nd ed. London: Chapman and Hall.
Garsd, A. 1988. Schizophrenia and birth complications. Unpublished manuscript.
Hirji, K. F. 2006. Exact Analysis of Discrete Data. Boca Raton, FL: Chapman and Hall/CRC.
Hirji, K. F., C. R. Mehta, and N. R. Patel. 1987. Computing distributions for exact logistic regression. Journal of the American Statistical Association 82: 1110-1117. https://doi.org/10.2307/2289388.
Hirji, K. F., A. A. Tsiatis, and C. R. Mehta. 1989. Median unbiased estimation for binary data. American Statistician 43: 7-11. https://doi.org/10.2307/2685158.

Hutto, C., W. P. Parks, S. Lai, M. T. Mastrucci, C. Mitchell, J. Muñoz, E. Trapido, I. M. Master, and G. B. Scott. 1991. A hospital-based prospective study of perinatal infection with human immunodeficiency virus type 1. Journal of Pediatrics 118: 347-353. https://doi.org/10.1016/S0022-3476(05)82145-6.
Mehta, C. R., and N. R. Patel. 1995. Exact logistic regression: Theory and examples. Statistics in Medicine 14: 2143-2160. https://doi.org/10.1002/sim. 4780141908.

## Also see

$[\mathrm{R}]$ exlogistic postestimation - Postestimation tools for exlogistic
$[\mathrm{R}]$ binreg - Generalized linear models: Extensions to the binomial family
$[\mathrm{R}]$ clogit - Conditional (fixed-effects) logistic regression
[R] expoisson - Exact Poisson regression
$[\mathrm{R}]$ logistic - Logistic regression, reporting odds ratios
[R] logit - Logistic regression, reporting coefficients
[U] 20 Estimation and postestimation commands

## Title

exlogistic postestimation - Postestimation tools for exlogistic

| Postestimation commands | estat | Remarks and examples |
| :--- | :--- | :--- |
| Reference | Also see |  |

## Postestimation commands

The following postestimation commands are of special interest after exlogistic:

| Command | Description |
| :--- | :--- |
| estat predict | single-observation prediction |
| estat se | report ORs or coefficients and their asymptotic standard errors |

The following standard postestimation commands are also available:
Command Description

```
estat summarize summary statistics for the estimation sample
estimates cataloging estimation results
etable table of estimation results
```

estat summarize is not allowed if the binomial() option was specified in exlogistic.

## estat

## Description for estat

estat predict computes a predicted probability (or linear predictor), its asymptotic standard error, and its exact confidence interval for 1 observation. Predictions are carried out by estimating the constant coefficient after shifting the independent variables and conditioned variables by the values specified in the at () option or by their medians. Therefore, predictions must be done with the estimation sample in memory. If a different dataset is used or if the dataset is modified, then an error will result.
estat se reports odds ratio or coefficients and their asymptotic standard errors. The estimates are stored in the matrix $r$ (estimates).

## Menu for estat

Statistics $>$ Postestimation

## Syntax for estat

Single-observation prediction
estat predict [, pred_options]
Report ORs or coefficients and their asymptotic standard errors

```
estat se [, coef]
```

pred_options
Description
pr probability; the default
$\mathrm{xb} \quad$ linear predictor
at (atspec) use the specified values for the indepvars and condvars()
level(\#)
memory (\#[b|k|m|g])
set confidence level for the predicted value; default is level(95)
set limit on memory usage; default is memory ( 10 m )
[no]log
collect is allowed with estat predict; see [U] 11.1.10 Prefix commands.
These statistics are available only for the estimation sample.

## Options for estat predict

pr , the default, calculates the probability.
xb calculates the linear predictor.
at $($ varname $=\#[[$ varname $=\#][\ldots]])$ specifies values to use in computing the predicted value. Here varname is one of the independent variables, indepvars, or the conditioned variables, condvars(). The default is to use the median of each independent and conditioned variable.
level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals.
memory (\# $[\mathrm{b}|\mathrm{k}| \mathrm{m} \mid \mathrm{g}]$ ) sets a limit on the amount of memory estat predict can use when generating the conditional distribution of the constant parameter sufficient statistic. The default is memory ( 10 m ), where m stands for megabyte, or $1,048,576$ bytes. The following are also available: b stands for byte; k stands for kilobyte, which is equal to 1,024 bytes; and g stands for gigabyte, which is equal to 1,024 megabytes. The minimum setting allowed is 1 m and the maximum is 512 m or 0.5 g , but do not attempt to use more memory than is available on your computer. Also see Remarks and examples in [R] exlogistic for details on enumerating the conditional distribution.
log and nolog specify whether to display the enumeration log, which shows the progress of enumerating the distribution of the observed successes conditioned on the independent variables shifted by the values specified in at () (or by their medians). See Methods and formulas in $[R]$ exlogistic for details of the computations. The enumeration log is displayed by default unless you used set iterlog off to suppress it; see set iterlog in [R] set iter.

## Option for estat se

coef requests that the estimated coefficients and their asymptotic standard errors be reported. The default is to report the odds ratios and their asymptotic standard errors.

## Remarks and examples

Predictions must be done using the estimation sample. This is because the prediction is really an estimated constant coefficient (the intercept) after shifting the independent variables and conditioned variables by the values specified in at () or by their medians. The justification for this approach can be seen by rewriting the model as

$$
\log \left(\frac{\pi_{i}}{1-\pi_{i}}\right)=\left(\alpha+\mathbf{x}_{0} \boldsymbol{\beta}\right)+\left(\mathbf{x}_{i}-\mathbf{x}_{0}\right) \boldsymbol{\beta}
$$

where $\mathbf{x}_{0}$ are the specified values for the indepvars (Mehta and Patel 1995). Because the estimation of the constant term is required, this technique is not appropriate for stratified models that used the group() option.

## > Example 1

To demonstrate, we return to the example 2 in $[R]$ exlogistic using data from a prospective study of perinatal infection and HIV-1. Here there was an investigation into whether the blood serum levels of CD4 and CD8 measured in infants at 6 months of age might predict their development of HIV infection. The blood serum levels are coded as ordinal values 0 , 1 , and 2 . These data are used by Mehta and Patel (1995) as an exposition of exact logistic.

```
. use https://www.stata-press.com/data/r18/hiv_n
(Prospective study of perinatal infection of HIV-1; binomial form)
. exlogistic hiv ib2.cd4 ib2.cd8, binomial(n) test(probability)
    (output omitted)
```



Because we did not specify values by using the at () option, the median values of the indepvars are used for the prediction. By default, medians are used instead of means because we want to use values that are observed in the dataset. If the means of the indicators for levels of factor variables ( $0 . \mathrm{cd} 4-1 . \mathrm{cd} 8$ ) were used, we would have created floating point variables in $(0,1)$ that not only do not properly represent the indicator variables but also would be a source of computational inefficiency in generating the conditional distribution. Because the MUE is computed for the predicted value, there is no standard error estimate.

From the example discussions in [R] exlogistic, the infants at highest risk are those with a CD4 level of 0 and a CD8 level of 2 . Below, we use the at () option to make a prediction at these blood serum levels.

```
. estat predict, at(cd4=0 cd8=2) nolog
note: 2b.cd8 is a base level; setting all other levels of cd8 to zero implies
    2b.cd8 = 1.
note: CMLE estimate for _cons is +inf; computing MUE.
Predicted value at 0.cd4 = 1, 1.cd4 = 0, 0.cd8 = 0, 1.cd8 = 0
```

| hiv | Predicted | Std. err. | [95\% conf. interval] |  |
| ---: | ---: | :---: | ---: | ---: |
| Probability | $0.9063 *$ | N/A | 0.4637 | 1.0000 |

(*) identifies median unbiased estimates (MUE) ; because an MUE is computed, there is no SE estimate

## Stored results

estat predict stores the following in $r()$ :
Scalars
$r$ (imue) $\quad 1$ if $r$ (pred) is an MUE and 0 if a CMLE
$r$ (pred) estimated probability or the linear effect
$r(\mathrm{se}) \quad$ asymptotic standard error of $r(p r e d)$
Macros
r (estimate) prediction type: pr or xb
$r$ (level) confidence level
Matrices
$r(\mathrm{ci}) \quad$ confidence interval
$\mathrm{r}(\mathrm{x}) \quad$ indepvars and condvars() values

## Reference

Mehta, C. R., and N. R. Patel. 1995. Exact logistic regression: Theory and examples. Statistics in Medicine 14: 2143-2160. https://doi.org/10.1002/sim.4780141908.

## Also see

[R] exlogistic - Exact logistic regression
[U] 20 Estimation and postestimation commands

## Title

expoisson - Exact Poisson regression

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

expoisson fits an exact Poisson regression model, which produces more accurate inference in small samples than standard maximum-likelihood-based Poisson regression. For stratified data, expoisson conditions on the number of events in each stratum and is an alternative to fixed-effects Poisson regression.

## Quick start

Exact Poisson regression of y on $\mathrm{x} 1, \mathrm{x} 2$, and x 3
expoisson y x1 x2 x3
Add exposure variable evar
expoisson y x1 x2 x3, exposure(evar)
Same as above, but condition on values of $x 3$ to save time and memory
expoisson y x1 x2, exposure(evar) condvars(x3)
Same as above, and allow more memory for computing the conditional distribution of sufficient statistics
expoisson y x1 x2, exposure(evar) condvars(x3) memory (100m)
Report incidence-rate ratios rather than coefficients
expoisson y x1 x2 x3, irr
Report conditional scores tests
expoisson y x1 x2 x3, test(score)
Fit a model with strata identified by svar

```
expoisson y x1 x2 x3, group(svar)
```


## Menu

Statistics $>$ Exact statistics $>$ Exact Poisson regression

## Syntax

expoisson depvar indepvars [if] [in] [weight] [, options]
options
Model
condvars ( varlist $_{c}$ )
group (varname)
exposure ( varname $_{e}$ )
offset (varname ${ }_{o}$ )
Options
memory (\#[b|k|m|g]) saving (filename [, replace])

Reporting
level(\#)
irr
test (testopt)
mue ( varlist $_{m}$ )
midp
[no]log
display_options
coeflegend

Description
condition on variables in varlist $_{c}$
groups or strata are stratified by unique values of varname include $\ln \left(\right.$ varname $\left._{e}\right)$ in model with coefficient constrained to 1 include varname $_{o}$ in model with coefficient constrained to 1
set limit on memory usage; default is memory ( 25 m ) save the joint conditional distribution to filename
set confidence level; default is level (95) report incidence-rate ratios
report $p$-value for observed sufficient statistic, conditional scores test, or conditional probabilities test
compute the median unbiased estimates for varlist $_{m}$ use the mid- $p$-value rule
display or suppress the enumeration log; default is to display control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
display legend instead of statistics
indepvars, varlist ${ }_{c}$, and varlist $_{m}$ may contain factor variables; see [U] 11.4.3 Factor variables.
by, collect, and statsby are allowed; see [U] 11.1.10 Prefix commands.
fweights are allowed; see [U] 11.1.6 weight.
coeflegend does not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

[^24]exposure (varname ${ }_{e}$ ), offset (varname ${ }_{o}$ ); see [R] Estimation options.

Options
memory ( $\#[\mathrm{~b}|\mathrm{k}| \mathrm{m} \mid \mathrm{g}]$ ) sets a limit on the amount of memory expoisson can use when computing the conditional distribution of the parameter sufficient statistics. The default is memory ( 25 m ), where m stands for megabyte, or $1,048,576$ bytes. The following are also available: b stands for byte; k stands for kilobyte, which is equal to 1,024 bytes; and g stands for gigabyte, which is equal to 1,024 megabytes. The minimum setting allowed is 1 m and the maximum is 2048 m or 2 g , but do not attempt to use more memory than is available on your computer. Also see the first technical note under example 3 on counting the conditional distribution.
saving (filename [, replace]) saves the joint conditional distribution for each independent variable specified in indepvars. There is one file for each variable, and it is named using the prefix filename with the variable name appended. For example, saving (mydata) with an independent variable named $X$ would generate a data file named mydata_X.dta. Use replace to replace an existing file. Each file contains the conditional distribution for one of the independent variables specified in indepvars conditioned on all other indepvars and those variables specified in condvars(). There are two variables in each data file: the feasible sufficient statistics for the variable's parameter and their associated weights. The weights variable is named $\mathrm{w}_{\mathrm{L}}$.

## Reporting

level(\#); see [R] Estimation options. The level() option will not work on replay because confidence intervals are based on estimator-specific enumerations. To change the confidence level, you must refit the model.
irr reports estimated coefficients transformed to incidence-rate ratios, that is, $\exp (\beta)$ rather than $\beta$. Standard errors and confidence intervals are similarly transformed. This option affects how results are displayed, not how they are estimated or stored. irr may be specified at estimation or when replaying previously estimated results.
test (sufficient |score|probability) reports the $p$-value associated with the observed sufficient statistics, the conditional scores tests, or the conditional probabilities tests, respectively. The default is test (sufficient). All the statistics are computed at estimation time regardless of which is specified. Each statistic may thus also be displayed when replaying results after estimation without having to refit the model; see $[\mathrm{R}]$ expoisson postestimation.
mue ( varlist $_{m}$ ) specifies that median unbiased estimates (MUEs) be reported for the specified variables. By default, the conditional maximum likelihood estimates (CMLEs) are reported, except for those parameters for which the CMLEs are infinite. Specify mue (_all) if you want MUEs for all the indepvars.
midp instructs expoisson to use the mid- $p$-value rule when computing the mUEs, $p$-values, and confidence intervals. This adjustment is for the discreteness of the distribution and halves the value of the discrete probability of the observed statistic before adding it to the $p$-value. The mid- $p$-value rule cannot be applied to MUEs whose corresponding parameter CMLE is infinite.
log and nolog specify whether to display the enumeration log, which shows the progress of computing the conditional distribution of the sufficient statistics. The enumeration $\log$ is displayed by default unless you used set iterlog off to suppress it; see set iterlog in [R] set iter.
display_options: noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\% fmt), pformat (\% fmt), and sformat (\% fit); see [R] Estimation options.

Note that the maximum widths for cformat(), pformat(), and sformat() differ from those widths listed in $[\mathrm{R}]$ Estimation options. The maximum width for each format is 9 for expoisson.

The following option is available with expoisson but is not shown in the dialog box: coeflegend; see [R] Estimation options.

## Remarks and examples

Exact Poisson regression estimates the model parameters by using the conditional distributions of the parameters' sufficient statistics, and the resulting parameter estimates are known as CMLEs. Exact Poisson regression is a small-sample alternative to the maximum-likelihood Poisson model. See [R] poisson and [XT] xtpoisson to obtain maximum likelihood estimates (MLEs) for the Poisson model and the fixed-effects Poisson model.

Let $Y_{i}$ denote a Poisson random variable where we observe the outcome $Y_{i}=y_{i}, i=1, \ldots, n$. Associated with each independent observation is a $1 \times p$ vector of covariates, $\mathbf{x}_{i}$. We will denote $\mu_{i}=E\left[Y_{i} \mid \mathbf{x}_{i}\right]$ and use the log-linear model to model the relationship between $Y_{i}$ and $\mathbf{x}_{i}$,

$$
\log \left(\mu_{i}\right)=\theta+\mathbf{x}_{i} \boldsymbol{\beta}
$$

where the constant term, $\theta$, and the $p \times 1$ vector of regression parameters, $\boldsymbol{\beta}$, are unknown. The probability of observing $Y_{i}=y_{i}, i=1, \ldots, n$, is

$$
\operatorname{Pr}(\mathbf{Y}=\mathbf{y})=\prod_{i=1}^{n} \frac{\mu_{i}^{y_{i}} e^{-\mu_{i}}}{y_{i}!}
$$

where $\mathbf{Y}=\left(Y_{1}, \ldots, Y_{n}\right)$ and $\mathbf{y}=\left(y_{1}, \ldots, y_{n}\right)$. The MLEs for $\theta$ and $\boldsymbol{\beta}$ maximize the $\log$ of this function.

The sufficient statistics for $\theta$ and $\beta_{j}, j=1, \ldots, p$, are $M=\sum_{i=1}^{n} Y_{i}$ and $T_{j}=\sum_{i=1}^{n} Y_{i} x_{i j}$, respectively, and we observe $M=m$ and $T_{j}=t_{j}$. expoisson tallies the conditional distribution for each $T_{j}$, given the other sufficient statistics $T_{l}=t_{l}, l \neq j$ and $M=m$. Denote one of these values to be $t_{j}^{(k)}, k=1, \ldots, N$, with weight $w_{k}$ that accounts for all the generated $\mathbf{Y}$ vectors that give rise to $t_{j}^{(k)}$. The conditional probability of observing $T_{j}=t_{j}$ has the form

$$
\begin{equation*}
\operatorname{Pr}\left(T_{j}=t_{j} \mid T_{l}=t_{l}, l \neq j, M=m\right)=\frac{w e^{t_{j} \beta_{j}}}{\sum_{k} w_{k} e^{t_{j}^{(k)} \beta_{j}}} \tag{1}
\end{equation*}
$$

where the sum is over the subset of $\mathbf{T}$ vectors such that $\left(T_{1}^{(k)}=t_{1}, \ldots, T_{j}^{(k)}=t_{j}^{(k)}, \ldots, T_{p}^{(k)}=t_{p}\right)$ and $w$ is the weight associated with the observed $\mathbf{t}$. The CMLE for $\beta_{j}$ maximizes the $\log$ of this function.

Specifying nuisance variables in condvars() prevents expoisson from estimating their associated regression coefficients. These variables are still conditional variables when tallying the conditional distribution for the variables in indepvars.

Inferences from MLEs rely on asymptotics, and if your sample size is small, these inferences may not be valid. On the other hand, inferences from the CMLEs are exact in that they use the conditional distribution of the sufficient statistics outlined above.

For small datasets, the dependent variable can be completely determined by the data. Here the MLEs and the CMLEs are unbounded. When this occurs, expoisson will compute the MUE, the regression estimate that places the observed sufficient statistic at the median of the conditional distribution.

See $[R]$ exlogistic for a more thorough discussion of exact estimation and related statistics.

## $>$ Example 1

Armitage, Berry, and Matthews (2002, 499-501) fit a log-linear model to data containing the number of cerebrovascular accidents experienced by 41 men during a fixed period, each of whom had recovered from a previous cerebrovascular accident and was hypertensive. Sixteen men received treatment, and in the original data, there are three age groups ( $40-49,50-59, \geq 60$ ), but we pool the first two age groups to simplify the example. Armitage, Berry, and Matthews point out that this was not a controlled trial, but the data are useful to inquire whether there is evidence of fewer accidents for the treatment group and if age may be an important factor. The dependent variable count contains the number of accidents, variable treat is an indicator for the treatment group ( $1=$ treatment, $0=$ control), and variable age is an indicator for the age group ( $0=40-59 ; 1=\geq 60$ ).

First, we load the dataset, list it, and tabulate the cerebrovascular accident counts by treatment and age group.

```
. use https://www.stata-press.com/data/r18/cerebacc
(Cerebrovascular accidents in hypotensive-treated and control groups)
. list
```

|  | treat | count | age |
| :--- | ---: | ---: | ---: |
| 1. | Control | 0 | $40 / 59$ |
| 2. | Control | 0 | $>=60$ |
| 3. | Control | 1 | $40 / 59$ |
| 4. | Control | 1 | $>=60$ |
| 5. | Control | 2 | $40 / 59$ |
| 6. | Control | 2 | $>=60$ |
| 7. | Control | 3 | $40 / 59$ |


| 35. | Treatment | 0 | $40 / 59$ |
| :--- | :--- | :--- | :--- |
| 36. | Treatment | 0 | $40 / 59$ |
| 37. | Treatment | 0 | $40 / 59$ |
| 38. | Treatment | 0 | $40 / 59$ |
| 39. | Treatment | 1 | $40 / 59$ |
| 40. | Treatment | 1 | $40 / 59$ |
| 41. | Treatment | 1 | $40 / 59$ |


| Hypotensiv e drug treatment | $\begin{array}{r} \text { Age } \\ 40 / 59 \end{array}$ | $>=60$ | Total |
| :---: | :---: | :---: | :---: |
| Control | 15 | 10 | 25 |
| Treatment | 4 | 0 | 4 |
| Total | 19 | 10 | 29 |

Next, we estimate the CMLE with expoisson and, for comparison, the MLE with poisson.

| . expoisson count i.treat i.age |  |
| :--- | ---: |
| Estimating: 1.treat |  |
| Enumerating sample-space combinations: |  |
| Observation 1: Enumerations $=$ | 11 |
| Observation 2: Enumerations $=$ | 11 |
| Observation 3: Enumerations $=$ | 11 |
| (output omitted) |  |
| Observation 39: Enumerations $=$ | 410 |
| Observation 40: Enumerations $=$ | 410 |
| Observation 41: Enumerations $=$ | 30 |
| Estimating: 1.age |  |
| Enumerating sample-space combinations: |  |
| Observation 1: Enumerations $=$ | 5 |
| Observation 2: Enumerations $=$ | 15 |
| Observation 3: Enumerations $=$ | 15 |
| (output omitted) |  |
| Observation 39: Enumerations $=$ | 455 |
| Observation 40: Enumerations $=$ | 455 |
| Observation 41: Enumerations $=$ | 30 |

Exact Poisson regression

$$
\text { Number of obs }=41
$$

| count | Coefficient | Suff. | $2 * \operatorname{Pr}($ Suff. $)$ | [95\% conf. interval] |  |
| ---: | :---: | :---: | ---: | ---: | ---: |
| treat <br> Treatment | -1.594306 | 4 | 0.0026 | -3.005089 | -.4701708 |
| age <br> $>=60$ | -.5112067 | 10 | 0.2794 | -1.416179 | .3429232 |


| . poisson count i.treat i.age, nolog |  |
| :--- | :--- |
| Poisson regression | Number of obs $=$ |
|  | LR chi2 $(2)$ |
|  | Prob $>$ chi2 $=10.64$ |
| Log likelihood $=-38.97981$ | Pseudo R2 |


| count | Coefficient | Std. err. | $z$ | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| :---: | ---: | :--- | :--- | :--- | :--- | :--- |
| treat |  |  |  |  |  |  |
| Treatment | -1.594306 | .5573614 | -2.86 | 0.004 | -2.686714 | -.5018975 |
| age |  |  |  |  |  |  |
| $>=60$ | -.5112067 | .4043525 | -1.26 | 0.206 | -1.303723 | .2813096 |
| _cons | .233344 | .2556594 | 0.91 | 0.361 | -.2677391 | .7344271 |

expoisson generates an enumeration log for each independent variable in indepvars. The conditional distribution of the parameter sufficient statistic is tallied for each independent variable. The conditional distribution for treat, for example, has 30 records containing the weights, $w_{k}$, and feasible sufficient statistics, $t_{\text {treat }}^{(k)}$. In essence, the set of points $\left(w_{k}, t_{\text {treat }}^{(k)}\right), k=1, \ldots, 30$, tallied by expoisson now become the data to estimate the regression coefficient for treat, using (1) as the likelihood. Remember that 1 of the $30\left(w_{k}, t_{\text {treat }}^{(k)}\right)$ must contain the observed sufficient statistic, $t_{\text {treat }}=\sum_{i=1}^{41}$ treat $_{i} \times$ count $_{i}=4$, and its relative position in the sorted set of points (sorted by $t_{\text {treat }}^{(k)}$ ) is how the sufficient-statistic $p$-value is computed. This algorithm is repeated for the age variable.

The regression coefficients for treat and age are numerically identical for both Poisson models. Both models provide evidence that the treatment reduces the rate of cerebrovascular accidents, $\approx e^{-1.59} \approx 0.204$, or a reduction of about $80 \%$. There is no evidence that age plays a role in the rate of accidents.

The results based on the sufficient statistic provide stronger evidence that treatment reduces the rate of cerebrovascular accidents than the corresponding asymptotic statistics. However, the exact confidence intervals are wider than their asymptotic counterparts.

## $>$ Example 2

Agresti $(2013,129)$ used the data from Laird and Olivier (1981) to demonstrate the Poisson model for modeling rates. The data consist of patient survival after heart valve replacement operations. The sample consists of 109 patients that are classified by type of heart valve (aortic, mitral) and by age $(<55, \geq 55)$. Follow-up observations cover lengths from 3 to 97 months, and the time at risk, or exposure, is stored in the variable TAR. The response is whether the subject died. First, we take a look at the data and then estimate the incidence rates (IRs) with expoisson and poisson.

```
. use https://www.stata-press.com/data/r18/heartvalve
(Heart valve replacement data)
. list
\begin{tabular}{|rcrr|}
\hline age & valve & deaths & TAR \\
\hline\(<55\) & Aortic & 4 & 1259 \\
\(<55\) & Mitral & 1 & 2082 \\
\(>=55\) & Aortic & 7 & 1417 \\
\(>=55\) & Mitral & 9 & 1647 \\
\hline
\end{tabular}
```

The age variable is coded 0 for age $<55$ and 1 for age $\geq 55$, and the valve variable is coded 0 for the aortic valve and 1 for the mitral valve. The total number of deaths, $M=21$, is small enough that enumerating the conditional distributions for age and valve type is feasible and asymptotic inferences associated with standard maximum-likelihood Poisson regression may be questionable.
. expoisson deaths i.age i.valve, exposure(TAR) irr
Estimating: 1.age
Enumerating sample-space combinations:
Observation 1: Enumerations = 11
Observation 2: Enumerations $=11$
Observation 3: Enumerations $=132$
Observation 4: Enumerations $=22$
Estimating: 1.valve
Enumerating sample-space combinations:
Observation 1: Enumerations = 17
Observation 2: Enumerations = 17
Observation 3: Enumerations $=102$
Observation 4: Enumerations $=22$
Exact Poisson regression
Number of obs $=4$

| deaths | IRR | Suff. | $2 * \operatorname{Pr}($ Suff. $)$ | [95\% conf. interval] |  |
| :---: | ---: | :---: | ---: | ---: | ---: |
| age <br> $>=55$ <br> valve <br> Mitral <br> $\ln (T A R)$ | 3.390401 | 16 | 0.0194 | 1.182297 | 11.86935 |


| Poisson regression | Number of obs | $=\quad 4$ |
| :---: | :---: | :---: |
|  | LR chi2(2) | $=7.62$ |
|  | Prob > chi2 | $=0.0222$ |
| Log likelihood = -8.1747285 | Pseudo R2 | $=0.3178$ |


| deaths | IRR | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| :---: | ---: | ---: | ---: | ---: | ---: | ---: |
| age <br> $>=55$ | 3.390401 | 1.741967 | 2.38 | 0.017 | 1.238537 | 9.280965 |
| valve |  |  |  |  |  |  |
| Mitral |  |  |  |  |  |  |
| _cons |  |  |  |  |  |  |
| ln(TAR) | .7190197 | .3150492 | -0.75 | 0.452 | .3046311 | 1.6971 |

Note: _cons estimates baseline incidence rate.
The CMLE and the MLE are numerically identical. We have strong evidence that the death rate for the older age group is higher than the younger age group, specifically 3.4 times higher ( $p=0.017$ ). This means that for every death in the younger group each month, we would expect about three deaths in the older group. The IR estimate for valve type is approximately 0.72 , but we do not have enough evidence to claim that it is different from one. The exact Poisson confidence intervals are a bit wider than the asymptotic confidence intervals.

You can use ir (see [R] Epitab) to estimate IRs and exact confidence intervals for one covariate, and we compare these confidence intervals with those from expoisson, where we estimate the IR by using age only.


Exact Poisson regression
Number of obs $=4$

| deaths | IRR | Suff. | $2 * \operatorname{Pr}($ Suff. $)$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| age | 3.489295 | 16 | 0.0099 | 1.324926 | 10.64922 |
| $\ln (T A R)$ | 1 | (exposure) |  |  |  |

Note: Mid-p-value computed for the probabilities and CIs.
Both ir and expoisson give identical IRs and $p$-values. Both report the two-sided exact $p$-value by using the mid- $p$-value rule that accounts for the discreteness in the distribution by subtracting $p_{1 / 2}=$ $\operatorname{Pr}(T=t) / 2$ from $p_{l}=\operatorname{Pr}(T \leq t)$ and $p_{g}=\operatorname{Pr}(T \geq t)$, computing $2 \times \min \left(p_{l}-p_{1 / 2}, p_{g}-p_{1 / 2}\right)$. By default, expoisson will not use the mid- $p$-value rule (when you exclude the midp option), and here the two-sided exact $p$-value would be $2 \times \min \left(p_{l}, p_{g}\right)=0.0158$. The confidence intervals differ because expoisson uses the mid- $p$-value rule when computing the confidence intervals, yet ir does not. You can verify this by executing expoisson without the midp option for this example; you will get the same confidence intervals as ir.

You can replay expoisson to view the conditional scores test or the conditional probabilities test by using the test() option.

| Exact Poisson regression |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: |
| deaths | IRR | Score | Pr $>=$ Score | [95\% conf | interval] |
| age | 3.489295 | 6.76528 | 0.0113 | 1.324926 | 10.64922 |
| $\ln (\mathrm{TAR})$ |  | (exposure) |  |  |  |

[^25]All the statistics for expoisson are defined in Methods and formulas of [R] exlogistic. Apart from enumerating the conditional distributions for the logistic and Poisson sufficient statistics, computationally, the primary difference between exlogistic and expoisson is the weighting values in the likelihood for the parameter sufficient statistics.

## > Example 3

In this example, we fabricate data that will demonstrate the difference between the CMLE and the MUE when the CMLE is not infinite. A difference in these estimates will be more pronounced when the probability of the coefficient sufficient statistic is skewed when plotted as a function of the regression coefficient.

```
. clear
. input y x
                y x
    1. 0 2
    2. 1 1
    3. 10
    4. 0 0
    5. 0 . }
    6. 1 . 5
    7. 2 . 01
    8. 3 . 001
    9. 4.0001
    10. end
. expoisson y x, test(score)
Enumerating sample-space combinations:
Observation 1: Enumerations = 13
Observation 2: Enumerations = 91
Observation 3: Enumerations = 169
Observation 4: Enumerations = 169
Observation 5: Enumerations = 313
Observation 6: Enumerations = 313
Observation 7: Enumerations = 1469
Observation 8: Enumerations = 5525
Observation 9: Enumerations = 5479
Exact Poisson regression
```

                                    Number of obs \(=9\)
    | y | Coefficient | Score | Pr>=Score | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| x | -1.534468 | 2.955316 | 0.0810 | -3.761718 | .0485548 |

. expoisson $y$ x, test(score) mue(x) nolog
Exact Poisson regression
Number of obs $=9$

| y | Coefficient | Score | Pr>=Score | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| x | $-1.309268 *$ | 2.955316 | 0.0810 | -3.761718 | .0485548 |

(*) median unbiased estimates (MUE)
We observe $\left(x_{i}, y_{i}\right), i=1, \ldots, 9$. If we condition on $m=\sum_{i=1}^{9} y_{i}=12$, the conditional distribution of $T_{x}=\sum_{i} Y_{i} x_{i}$ has a size of 5,479 elements. For each entry in this enumeration, a realization of $Y_{i}=y_{i}^{(k)}, k=1, \ldots, 5,479$, is generated such that $\sum_{i} y_{i}^{(k)}=12$. One of these realizations produces the observed $t_{x}=\sum_{i} y_{i} x_{i} \approx 1.5234$.

Below is a graphical display comparing the CMLE with the MUE. We plot $\operatorname{Pr}\left(T_{x}=t_{x} \mid M=12, \beta_{x}\right)$ versus $\beta_{x},-6 \leq \beta_{x} \leq 1$, in the upper panel and the cumulative probabilities, $\operatorname{Pr}\left(T_{x} \leq t_{x} \mid M=\right.$ $\left.12, \beta_{x}\right)$ and $\operatorname{Pr}\left(T_{x} \geq t_{x} \mid M=12, \beta_{x}\right)$, in the lower panel.


The location of the CMLE, indicated by the dashed line, is at the mode of the probability profile, and the MUE, indicated by the dotted line, is to the right of the mode. If we solve for the $\beta_{x}^{(u)}$ and $\beta_{x}^{(l)}$ such that $\operatorname{Pr}\left(T_{x} \leq t_{x} \mid M=12, \beta_{x}^{(u)}\right)=1 / 2$ and $\operatorname{Pr}\left(T_{x} \geq t_{x} \mid M=12, \beta_{x}^{(l)}\right)=1 / 2$, the MUE is $\left(\beta_{x}^{(u)}+\beta_{x}^{(l)}\right) / 2$. As you can see in the lower panel, the MUE cuts through the intersection of these cumulative probability profiles.

## - Technical note

The memory () option limits the amount of memory that expoisson will consume when computing the conditional distribution of the parameter sufficient statistics. memory () is independent of the data maximum memory setting (see set max_memory in [D] memory), and it is possible for expoisson to exceed the memory limit specified in set max_memory without terminating. By default, a log is provided that displays the number of enumerations (the size of the conditional distribution) after processing each observation. Typically, you will see the number of enumerations increase, and then at some point they will decrease as the multivariate shift algorithm (Hirji, Mehta, and Patel 1987) determines that some of the enumerations cannot achieve the observed sufficient statistics of the conditioning variables. When the algorithm is complete, however, it is necessary to store the conditional distribution of the parameter sufficient statistics as a dataset. It is possible, therefore, to get a memory error when the algorithm has completed if there is not enough memory to store the conditional distribution.

## - Technical note

Computing the conditional distributions and reported statistics requires data sorting and numerical comparisons. If there is at least one single-precision variable specified in the model, expoisson will make comparisons with a relative precision of $2^{-5}$. Otherwise, a relative precision of $2^{-11}$ is used. Be careful if you use recast to promote a single-precision variable to double precision (see [D] recast). You might try listing the data in full precision (maybe $\% 20.15 \mathrm{~g}$; see [D] format) to make sure that this is really what you want. See [D] Data types for information on precision of numeric storage types.

## Stored results

expoisson stores the following in e():
Scalars
e(N)
e(k_groups)
e(relative_weight)
e(sum_y)
e(k_indvars)
e(k_condvars)
e(midp)
e(eps)
Macros
e(cmd)
e(cmdline)
e(title)
e(depvar)
e(indvars)
e(condvars)
e(groupvar)
e(exposure)
e(offset)
e(level)
e(wtype)
e(wexp)
$e($ datasignature)
e(datasignaturevars)
e(properties)
e(estat_cmd)
$e$ (marginsnotok)
number of observations
number of groups
relative weight for the observed e(sufficient) and e(condvars)
sum of depvar
number of independent variables
number of conditioning variables
mid- $p$-value rule indicator
relative difference tolerance
expoisson
command as typed
title in estimation output
name of dependent variable
independent variables
conditional variables
group variable
exposure variable
linear offset variable
confidence level
weight type
weight expression
the checksum
variables used in calculation of checksum
b
program used to implement estat
predictions disallowed by margins

```
Matrices
    e(b)
    e(mue_indicators)
    e(se)
    e(ci)
    e(sum_y_groups)
    e(N_g)
    e(sufficient)
    e(p_sufficient)
    e(scoretest)
    e(p_scoretest)
    e(probtest)
    e(p_probtest)
```

Functions
e(sample)
coefficient vector
indicator for elements of e(b) estimated using MUE instead of CMLE $e(b)$ standard errors (CMLEs only)
matrix of e(level) confidence intervals for e(b)
sum of e(depvar) for each group
number of observations in each group
sufficient statistics for e(b)
$p$-value for e(sufficient)
conditional scores tests for indepvars
$p$-values for e(scoretest)
conditional probabilities tests for indepvars
$p$-value for e(probtest)
marks estimation sample

## Methods and formulas

Let $\left\{Y_{1}, Y_{2}, \ldots, Y_{n}\right\}$ be a set of $n$ independent Poisson random variables. For each $i=1, \ldots, n$, we observe $Y_{i}=y_{i} \geq 0$, and associated with each observation is the covariate row vector of length $p, \mathbf{x}_{i}=\left(x_{i 1}, \ldots, x_{i p}\right)$. Denote $\boldsymbol{\beta}=\left(\beta_{1}, \ldots, \beta_{p}\right)^{T}$ to be the column vector of regression parameters and $\theta$ to be the constant. The sufficient statistic for $\beta_{j}$ is $T_{j}=\sum_{i=1}^{n} Y_{i} x_{i j}, j=1, \ldots, p$, and for $\theta$ is $M=\sum_{i=1}^{n} Y_{i}$. We observe $T_{j}=t_{j}, t_{j}=\sum_{i=1}^{n} y_{i} x_{i j}$, and $M=m, m=\sum_{i=1}^{n} y_{i}$. Let $\kappa_{i}$ be the exposure for the $i$ th observation. Then the probability of observing $\left(Y_{1}=y_{1}, Y_{2}=y_{2}, \ldots, Y_{n}=y_{n}\right)$ is

$$
\operatorname{Pr}\left(Y_{1}=y_{1}, \ldots, Y_{n}=y_{n} \mid \boldsymbol{\beta}, \mathbf{X}, \boldsymbol{\kappa}\right)=\frac{\exp (m \theta+\mathbf{t} \boldsymbol{\beta})}{\exp \left\{\sum_{i=1}^{n} \kappa_{i} \exp \left(\theta+\mathbf{x}_{i} \boldsymbol{\beta}\right)\right\}} \prod_{i=1}^{n} \frac{\kappa_{i}^{y_{i}}}{y_{i}!}
$$

where $\mathbf{t}=\left(t_{1}, \ldots, t_{p}\right), \mathbf{X}=\left(\mathbf{x}_{1}^{T}, \ldots, \mathbf{x}_{n}^{T}\right)^{T}$, and $\kappa=\left(\kappa_{1}, \ldots, \kappa_{n}\right)^{T}$.
The joint distribution of the sufficient statistics $(\mathbf{T}, M)$ is obtained by summing over all possible sequences $Y_{1} \geq 0, \ldots, Y_{n} \geq 0$ such that $\mathbf{T}=\mathbf{t}$ and $M=m$. This probability function is

$$
\operatorname{Pr}\left(T_{1}=t_{1}, \ldots, T_{p}=t_{p}, M=m \mid \boldsymbol{\beta}, \mathbf{X}, \boldsymbol{\kappa}\right)=\frac{\exp (m \theta+\mathbf{t} \boldsymbol{\beta})}{\exp \left\{\sum_{i=1}^{n} \kappa_{i} \exp \left(\theta+\mathbf{x}_{i} \boldsymbol{\beta}\right)\right\}}\left(\sum_{\mathbf{u}} \prod_{i=1}^{n} \frac{\kappa_{i}^{u_{i}}}{u_{i}!}\right)
$$

where the sum $\sum_{\mathbf{u}}$ is over all nonnegative vectors $\mathbf{u}$ of length $n$ such that $\sum_{i=1}^{n} u_{i}=m$ and $\sum_{i=1}^{n} u_{i} \mathbf{x}_{i}=\mathbf{t}$.

## Conditional distribution

Without loss of generality, we will restrict our discussion to the conditional distribution of the sufficient statistic for $\beta_{1}, T_{1}$. If we condition on observing $M=m$ and $T_{2}=t_{2}, \ldots, T_{p}=t_{p}$, the probability function of ( $\left.T_{1} \mid \beta_{1}, T_{2}=t_{2}, \ldots, T_{p}=t_{p}, M=m\right)$ is

$$
\begin{equation*}
\operatorname{Pr}\left(T_{1}=t_{1} \mid \beta_{1}, T_{2}=t_{2}, \ldots, T_{p}=t_{p}, M=m\right)=\frac{\left(\sum_{\mathbf{u}} \prod_{i=1}^{n} \frac{\kappa_{i}^{u_{i}}}{u_{i}!}\right) e^{t_{1} \beta_{1}}}{\sum_{\mathbf{v}}\left(\prod_{i=1}^{n} \frac{\kappa_{i}^{v_{i}}}{v_{i}!}\right) e^{\beta_{1} \sum_{i} v_{i} x_{i 1}}} \tag{2}
\end{equation*}
$$

where the sum $\sum_{\mathbf{u}}$ is over all nonnegative vectors $\mathbf{u}$ of length $n$ such that $\sum_{i=1}^{n} u_{i}=m$ and $\sum_{i=1}^{n} u_{i} \mathbf{x}_{i}=\mathbf{t}$, and the sum $\sum_{\mathbf{v}}$ is over all nonnegative vectors $\mathbf{v}$ of length $n$ such that $\sum_{i=1}^{n} v_{i}=m$, $\sum_{i=1}^{n=1} v_{i} x_{i 2}=t_{2}, \ldots, \sum_{i=1}^{n} v_{i} x_{i p}=t_{p}$. The CMLE for $\beta_{1}$ is the value that maximizes the log of (1). This optimization task is carried out by ml (see [R] ml ), using the conditional distribution of $\left(T_{1} \mid T_{2}=t_{2}, \ldots, T_{p}=t_{p}, M=m\right)$ as a dataset. This dataset consists of the feasible values and weights for $T_{1}$,

$$
\left\{\left(s_{1}, \prod_{i=1}^{n} \frac{\kappa_{i}^{v_{i}}}{v_{i}!}\right): \sum_{i=1}^{n} v_{i}=m, \sum_{i=1}^{n} v_{i} x_{i 1}=s_{1}, \sum_{i=1}^{n} v_{i} x_{i 2}=t_{2}, \ldots, \sum_{i=1}^{n} v_{i} x_{i p}=t_{p}\right\}
$$

Computing the CMLE, MUE, confidence intervals, conditional hypothesis tests, and sufficient statistic $p$-values is discussed in Methods and formulas of [R] exlogistic. The only difference between the two techniques is the use of the weights; that is, the weights for exact logistic are the combinatorial coefficients, $c(\mathbf{t}, m)$, in (1) of Methods and formulas in [R] exlogistic. expoisson and exlogistic use the same ml likelihood evaluator to compute the CMLEs as well as the same ado-programs and Mata functions to compute the MUEs and estimate statistics.

## References

Agresti, A. 2013. Categorical Data Analysis. 3rd ed. Hoboken, NJ: Wiley.
Armitage, P., G. Berry, and J. N. S. Matthews. 2002. Statistical Methods in Medical Research. 4th ed. Oxford: Blackwell.

Cox, D. R., and E. J. Snell. 1989. Analysis of Binary Data. 2nd ed. London: Chapman and Hall.
Hirji, K. F., C. R. Mehta, and N. R. Patel. 1987. Computing distributions for exact logistic regression. Journal of the American Statistical Association 82: 1110-1117. https://doi.org/10.2307/2289388.
Laird, N. M., and D. Olivier. 1981. Covariance analysis of censored survival data using log-linear analysis techniques. Journal of the American Statistical Association 76: 231-240. https://doi.org/10.2307/2287816.

## Also see

$[R]$ expoisson postestimation - Postestimation tools for expoisson
[R] poisson - Poisson regression
[XT] xtpoisson - Fixed-effects, random-effects, and population-averaged Poisson models
[U] 20 Estimation and postestimation commands

## Title

expoisson postestimation - Postestimation tools for expoisson

Postestimation commands estat Remarks and examples Also see

## Postestimation commands

The following postestimation command is of special interest after expoisson:

| Command | Description |
| :--- | :--- |
| estat se | report coefficients or IRRs and their asymptotic standard errors |

The following standard postestimation commands are also available:
Command
Description
estat summarize
summary statistics for the estimation sample
estimates
cataloging estimation results
etable
table of estimation results

## estat

## Description for estat

estat se reports regression coefficients or incidence-rate asymptotic standard errors. The estimates are stored in the matrix $r$ (estimates).

## Menu for estat

Statistics $>$ Postestimation

## Syntax for estat

```
estat se [, irr]
```


## Option for estat

irr requests that the incidence-rate ratios and their asymptotic standard errors be reported. The default is to report the coefficients and their asymptotic standard errors.

## Remarks and examples

## Example 1

To demonstrate estat se after expoisson, we use the British physicians smoking data.

```
. use https://www.stata-press.com/data/r18/smokes
(Cigarette smoking and lung cancer among British physicians (45-49 years))
. expoisson cases smokes, exposure(peryrs) irr nolog
```

Exact Poisson regression
Number of obs $=7$

| cases | IRR | Suff. | $2 * \operatorname{Pr}($ Suff. $)$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| smokes | 1.077718 | 797.4 | 0.0000 | 1.04552 | 1.111866 |
| $\ln$ (peryrs) | 1 | (exposure) |  |  |  |

. estat se, irr

| cases | IRR | Std. err. |
| ---: | ---: | ---: |
| smokes | 1.077718 | .0168547 |

## Also see

[R] expoisson - Exact Poisson regression
[U] 20 Estimation and postestimation commands

## Title

fp - Fractional polynomial regression

Description
Options
Acknowledgment

Quick start
Remarks and examples
References

Menu
Stored results Also see

Syntax
Methods and formulas

## Description

fp <term>: est_cmd fits models with the "best"-fitting fractional polynomial substituted for <term> wherever it appears in est_cmd. fp <weight>: regress mpg <weight> foreign would fit a regression model of mpg on a fractional polynomial in weight and (linear) foreign.

By specifying option $f p()$, you may set the exact powers to be used. Otherwise, a search through all possible fractional polynomials up to the degree set by dimension() with the powers set by powers() is performed.
fp without arguments redisplays the previous estimation results, just as typing est_cmd would. You can type either one. fp will include a fractional polynomial comparison table.
fp generate creates fractional polynomial power variables for a given set of powers. For instance, $f p$ <weight>: regress mpg <weight> foreign might produce the fractional polynomial weight $^{(-2,-1)}$ and store weight ${ }^{-2}$ in weight_1 and weight ${ }^{-1}$ in weight_2. Typing fp generate weight^ $(-2-1)$ would allow you to create the same variables in another dataset.

See [R] mfp for multivariable fractional polynomial models.

## Quick start

Fit models with fractional polynomials
Find optimal second-degree fractional polynomial of $x 1$ in regression of $y$ on $x 2$ and $x 3$
fp <x1>: regress y <x1> x2 x3
Same as above, but search only powers of $-1,-0.5,1$, and 2 .
fp <x1>, power (-1 -. 5 1 2) : regress y <x1> x2 x3
Same as above, but allow search to include third-degree fractional polynomials
fp <x1>, power(-1-. 5 1 2) dimension(3): regress y <x1> x2 x3
Fit model including $\mathrm{x} 1^{-2}$ and $\mathrm{x} 1^{2}$ without performing search
fp <x1>, fp(-2 2): regress y <x1> x2 x3
Rescale x 1 to nonextreme positive values when computing fractional polynomials
fp <x1>, scale: regress y <x1> x2 x3
Same as above, and center fractional polynomial of $x 1$ at its scaled mean
fp <x1>, center scale: regress y <x1> x2 x3
Set fractional polynomial to zero for nonpositive values of $x 1$
fp <x1>, zero: regress y <x1> x2 x3
Same as above, and include an indicator variable in the model for nonpositive values of x 1
fp <x1>, catzero: regress y <x1> x2 x3

Create variables corresponding to fractional polynomial powers
Generate $\mathrm{x} 1 \_1$ and $\mathrm{x} 1 \_2$ corresponding to $\mathrm{x} 1^{-2}$ and $\mathrm{x} 1^{2}$
fp generate $\mathrm{x} 1^{\wedge}\left(\begin{array}{ll}-2 & 2\end{array}\right)$
Same as above, but generate fractional polynomial variables with automatic scaling and centering
fp generate $\mathrm{x} 1^{\wedge}(-22)$, center scale

Note: In the above examples, regress could be replaced with any estimation command allowing the fp prefix.

## Menu

fp
Statistics $>$ Linear models and related $>$ Fractional polynomials $>$ Fractional polynomial regression

## fp generate

Statistics $>$ Linear models and related $>$ Fractional polynomials $>$ Create fractional polynomial variables

## Syntax

## Estimation

```
fp <term> [, est_options]: est_cmd
```

Specify that fractional powers of varname be calculated during estimation
fp <term>(varname) [, est_options]: est_cmd

Replay estimation results
fp [, replay_options]

Create specified fractional polynomial power variables fp generate [type][newvar =] varname^(numlist) [if] [in] [, gen_options]
est_cmd may be almost any estimation command that stores the e(ll) result. To confirm whether fp works with a specific est_cmd, see the documentation for that est_cmd. est_cmd may not contain other prefix commands; see [U] 11.1.10 Prefix commands.
Instances of <term> (with the angle brackets) that occur within est_cmd are replaced in est_cmd by a varlist containing the fractional powers of the variable term. These variables will be named term_1, term_2, ....
fp performs est_cmd with this substitution, fitting a fractional polynomial regression in term.

Main
powers (\# \# ... \#) powers to be searched; default is powers ( $-2-1-.50 .5123$ ) dimension(\#) maximum degree of fractional polynomial; default is dimension(2)
fp(\# \# ... \#)
Options
classic
replace
all
scale(\#_a \#_b)
scale
center (\#_c)
center
zero
catzero

Reporting
replay_options
perform automatic scaling and centering and omit comparison table
replace existing fractional polynomial power variables named term_1, term_2, ...
generate term_1, term_2, ... in all observations; default is in observations if esample()
use $($ term $+a) / b$; default is to use variable term as is
specify $a$ and $b$ automatically
report centered-on- $c$ results; default is uncentered results
specify $c$ to be the mean of (scaled) term
set term_1, term_2, .. to zero if scaled term $\leq 0$; default is to issue an error message
same as zero and include term_0 $=($ term $\leq 0)$ among fractional polynomial power variables
specify how results are displayed

Description
replay_options
Reporting
nocompare
reporting_options
do not display model-comparison test results
any options allowed by est_cmd for replaying estimation results
gen_options
Description
Main
replace
scale
center (\#_c)
center
zero
catzero
scale $\left(\#-a \# \_b\right) \quad$ use $($ term $+a) / b$; default is to use variable term as is
replace existing fractional polynomial power variables named term_1, term_2, ...
specify $a$ and $b$ automatically
report centered-on- $c$ results; default is uncentered results
specify $c$ to be the mean of (scaled) term
set term_1, term_2, .. to zero if scaled term $\leq 0$; default is to issue an error message
same as zero and include term_0 $=($ term $\leq 0)$ among fractional polynomial power variables
collect is allowed with $f p$ and $f p$ generate; see [U] 11.1.10 Prefix commands.

## Options

Options are presented under the following headings:
Options for $f p$
Options for fp generate

## Options for fp

$\qquad$ Main
powers (\# \# ... \#) specifies that a search be performed and details about the search provided. powers() works with the dimension() option; see below. The default is powers ( $-2-1-.50$ .5123 ).
dimension(\#) specifies the maximum degree of the fractional polynomial to be searched. The default is dimension(2).

If the defaults for both powers() and dimension() are used, then the fractional polynomial could be any of the following 44 possibilities:

$$
\begin{gathered}
\text { term }^{(-2)} \\
\text { term }^{(-1)} \\
\vdots \\
\text { term }^{(3)} \\
\text { term }^{(-2)}, \text { term }^{(-2)} \\
\text { term }^{(-2)}, \text { term }^{(-1)} \\
\vdots \\
\text { term }^{(-2)}, \text { term }^{(3)} \\
\text { term }^{(-1)}, \text { term }^{(-2)} \\
\vdots \\
\text { term }^{(3)}, \text { term }^{(3)}
\end{gathered}
$$

fp(\# \# ... \#) specifies that no search be performed and that the fractional polynomial specified be used. fp() is an alternative to powers() and dimension().

## Options

classic performs automatic scaling and centering and omits the comparison table. Specifying classic is equivalent to specifying scale, center, and nocompare.
replace replaces existing fractional polynomial power variables named term_1, term_2, ....
all specifies that term_1, term_2, .. be filled in for all observations in the dataset rather than just for those in e(sample).
scale (\#_a \#_b) specifies that term be scaled in the way specified, namely, that $($ term $+a) / b$ be calculated. All values of the scaled term are required to be greater than zero unless you specify options zero or catzero. Values should not be too large or too close to zero, because by default, cubic powers and squared reciprocal powers will be considered. When scale $(a b)$ is specified, values in the variable term are not modified; fp merely remembers to scale the values whenever powers are calculated.

You will probably not use scale $(a b)$ for values of $a$ and $b$ that you create yourself, although you could. It is usually easier just to generate a scaled variable. For instance, if term is age, and age in your data is required to be greater than or equal to 20 , you might generate an age 5 variable, for use as term:

```
. generate age5 = (age-19)/5
```

scale ( $a b$ ) is useful when you previously fit a model using automatic scaling (option scale) in one dataset and now want to create the fractional polynomials in another. In the first dataset, fp with scale added notes to the dataset concerning the values of $a$ and $b$. You can see them by typing

```
. notes
```

You can then use fp generate, scale ( $a b$ ) in the second dataset.
The default is to use term as it is used in calculating fractional powers; thus, term's values are required to be greater than zero unless you specify options zero or catzero. Values should not be too large, because by default, cubic powers will be considered.
scale specifies that term be scaled to be greater than zero and not too large in calculating fractional powers. See Scaling for more details. When scale is specified, values in the variable term are not modified; fp merely remembers to scale the values whenever powers are calculated.
center (\#_c) reports results for the fractional polynomial in (scaled) term, centered on $c$. The default is to perform no centering.
term $^{\left(p_{1}, p_{2}, \ldots, p_{m}\right)}-c^{\left(p_{1}, p_{2}, \ldots, p_{m}\right)}$ is reported. This makes the constant coefficient (intercept) easier to interpret. See Centering for more details.
center performs center $(c)$, where $c$ is the mean of (scaled) term.
zero and catzero specify how nonpositive values of term are to be handled. By default, nonpositive values of term are not allowed, because we will be calculating natural logarithms and fractional powers of term. Thus, an error message is issued.
zero sets the fractional polynomial value to zero for nonpositive values of (scaled) term.
catzero sets the fractional polynomial value to zero for nonpositive values of (scaled) term and includes a dummy variable indicating where nonpositive values of (scaled) term appear in the model.

## Reporting

nocompare suppresses display of the comparison tests.
reporting_options are any options allowed by est_cmd for replaying estimation results.

## Options for $\mathfrak{f p}$ generate

Main
replace replaces existing fractional polynomial power variables named term_1, term_2, ...
scale(\#_a \#_b) specifies that term be scaled in the way specified, namely, that (term+a) $/ b$ be calculated. All values of the scaled term are required to be greater than zero unless you specify options zero or catzero. Values should not be too large or too close to zero, because by default, cubic powers and squared reciprocal powers will be considered. When scale ( $a b$ ) is specified, values in the variable term are not modified; fp merely remembers to scale the values whenever powers are calculated.

You will probably not use scale $(a b)$ for values of $a$ and $b$ that you create yourself, although you could. It is usually easier just to generate a scaled variable. For instance, if term is age, and age in your data is required to be greater than or equal to 20 , you might generate an age 5 variable, for use as term:

```
. generate age5 = (age-19)/5
```

scale ( $a b$ ) is useful when you previously fit a model using automatic scaling (option scale) in one dataset and now want to create the fractional polynomials in another. In the first dataset, fp with scale added notes to the dataset concerning the values of $a$ and $b$. You can see them by typing

```
. notes
```

You can then use fp generate, $\operatorname{scale}(a b)$ in the second dataset.
The default is to use term as it is used in calculating fractional powers; thus, term's values are required to be greater than zero unless you specify options zero or catzero. Values should not be too large, because by default, cubic powers will be considered.
scale specifies that term be scaled to be greater than zero and not too large in calculating fractional powers. See Scaling for more details. When scale is specified, values in the variable term are not modified; fp merely remembers to scale the values whenever powers are calculated.
center (\#_c) reports results for the fractional polynomial in (scaled) term, centered on $c$. The default is to perform no centering.
term ${ }^{\left(p_{1}, p_{2}, \ldots, p_{m}\right)}-c^{\left(p_{1}, p_{2}, \ldots, p_{m}\right)}$ is reported. This makes the constant coefficient (intercept) easier to interpret. See Centering for more details.
center performs center $(c)$, where $c$ is the mean of (scaled) term.
zero and catzero specify how nonpositive values of term are to be handled. By default, nonpositive values of term are not allowed, because we will be calculating natural logarithms and fractional powers of term. Thus, an error message is issued.
zero sets the fractional polynomial value to zero for nonpositive values of (scaled) term.
catzero sets the fractional polynomial value to zero for nonpositive values of (scaled) term and includes a dummy variable indicating where nonpositive values of (scaled) term appear in the model.

## Remarks and examples

Remarks are presented under the following headings:
Fractional polynomial regression
Scaling
Centering
Examples

## Fractional polynomial regression

Regression models based on fractional polynomial functions of a continuous covariate are described by Royston and Altman (1994).

Fractional polynomials increase the flexibility afforded by the family of conventional polynomial models. Although polynomials are popular in data analysis, linear and quadratic functions are limited in their range of curve shapes, whereas cubic and higher-order curves often produce undesirable artifacts such as edge effects and waves.

Fractional polynomials differ from regular polynomials in that 1) they allow logarithms, 2) they allow noninteger powers, and 3) they allow powers to be repeated.

We will write a fractional polynomial in $x$ as

$$
x^{\left(p_{1}, p_{2}, \ldots, p_{m}\right)^{\prime}} \boldsymbol{\beta}
$$

We will write $x^{(p)}$ to mean a regular power except that $x^{(0)}$ is to be interpreted as meaning $\ln (x)$ rather than $x^{(0)}=1$.

Then if there are no repeated powers in $\left(p_{1}, p_{2}, \ldots, p_{m}\right)$,

$$
x^{\left(p_{1}, p_{2}, \ldots, p_{m}\right) \prime} \boldsymbol{\beta}=\beta_{0}+\beta_{1} x^{\left(p_{1}\right)}+\beta_{2} x^{\left(p_{2}\right)}+\cdots+\beta_{m} x^{\left(p_{m}\right)}
$$

Powers are allowed to repeat in fractional polynomials. Each time a power repeats, it is multiplied by another $\ln (x)$. As an extreme case, consider the fractional polynomial with all-repeated powers, say, $m$ of them,

$$
x^{(p, p, \ldots, p) \prime} \boldsymbol{\beta}=\beta_{0}+\beta_{1} x^{(p)}+\beta_{2} x^{(p)} \ln (x)+\cdots+\beta_{m} x^{(p)}\{\ln (x)\}^{m-1}
$$

Thus, the fractional polynomial $x^{(0,0,2) \prime} \boldsymbol{\beta}$ would be

$$
\begin{aligned}
x^{(0,0,2) \prime} \boldsymbol{\beta} & =\beta_{0}+\beta_{1} x^{(0)}+\beta_{2} x^{(0)} \ln (x)+\beta_{3} x^{(2)} \\
& =\beta_{0}+\beta_{1} \ln (x)+\beta_{2}\{\ln (x)\}^{2}+\beta_{3} x^{2}
\end{aligned}
$$

With this definition, we can obtain a much wider range of shapes than can be obtained with regular polynomials. The following graphs appeared in Royston and Sauerbrei (2008, sec. 4.5). The first graph shows the shapes of differing fractional polynomials.


The second graph shows some of the curve shapes available with different $\boldsymbol{\beta}$ s for the degree-2 fractional polynomial, $x^{(-2,2)}$.


In modeling a fractional polynomial, Royston and Sauerbrei (2008) recommend choosing powers from among $\{-2,-1,-0.5,0,0.5,1,2,3\}$. By default, fp chooses powers from this set, but other powers can be explicitly specified in the powers () option.
fp <term> : est_cmd fits models with the terms of the best-fitting fractional polynomial substituted for <term> wherever it appears in est_cmd. We will demonstrate with auto.dta, which contains repair records and other information about a variety of vehicles in 1978.

We use $f p$ to find the best fractional polynomial in automobile weight (lbs.) (weight) for the linear regression of miles per gallon ( mpg ) on weight and an indicator of whether the vehicle is foreign (foreign).

By default, fp will fit degree-2 fractional polynomial (FP2) models and choose the fractional powers from the set $\{-2,-1,-0.5,0,0.5,1,2,3\}$. Because car weight is measured in pounds and will have a cubic transformation applied to it, we shrink it to a smaller scale before estimation by dividing by 1,000 .

We modify the existing weight variable for conciseness and to facilitate the comparison of tables. When applying a data transformation in practice, rather than modifying the existing variables, you should create new variables that hold the transformed values.

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. replace weight = weight/1000
variable weight was int now float
(74 real changes made)
. fp <weight>: regress mpg <weight> foreign
(fitting 44 models)
( . . . 10% . . . 20% . . . . 30% . . . . 40% . . . 50% . . . . 60% . . . . 70% . . . . 80% . . . . 90% . . . . 100%)
```

Fractional polynomial comparisons:

| weight | Test <br> df | Deviance | Residual <br> std. dev. | Deviance <br> diff. | P | Powers |
| ---: | ---: | ---: | ---: | ---: | ---: | :--- |
| omitted | 4 | 456.347 | 5.356 | 75.216 | 0.000 |  |
| linear | 3 | 388.366 | 3.407 | 7.236 | 0.082 | 1 |
| m $=1$ | 2 | 381.806 | 3.259 | 0.675 | 0.733 | -.5 |
| $\mathrm{~m}=2$ | 0 | 381.131 | 3.268 | 0.000 | -- | $-2-2$ |

Note: Test df is degrees of freedom, and $\mathbf{P}=\mathrm{P}>\mathrm{F}$ is sig. level for tests comparing models vs. model with $m=2$ based on deviance difference, F (df , 68) .

| Source | SS | df | MS | Number of obs | = | 74 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | $\mathrm{F}(3,70)$ | = | 52.95 |
| Model | 1696.05949 | 3 | 565.353163 | Prob > F | = | 0.0000 |
| Residual | 747.399969 | 70 | 10.6771424 | R -squared | = | 0.6941 |
|  |  |  |  | Adj R-squared | = | 0.6810 |
| Total | 2443.45946 | 73 | 33.4720474 | Root MSE | = | 3.2676 |


| mpg | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :--- | :---: | :---: | ---: | ---: |
| weight_1 | 15.88527 | 20.60329 | 0.77 | 0.443 | -25.20669 | 56.97724 |
| weight_2 | 127.9349 | 47.53106 | 2.69 | 0.009 | 33.13723 | 222.7326 |
| foreign | -2.222515 | 1.053782 | -2.11 | 0.039 | -4.324218 | -.1208131 |
| _cons | 3.705981 | 3.367949 | 1.10 | 0.275 | -3.011182 | 10.42314 |

fp begins by showing the model-comparison table. This table shows the best fractional polynomial model of weight for each examined degree, $m$, which is obtained by searching through all possible power combinations. The row labeled omitted describes the null model, which entirely omits weight from the model. A separate row is provided for the model with a linear function of weight because it is often the default when including a predictor in the model.

The fractional powers of the models are shown in the Powers column. An estimate of the residual standard error is given in the Residual std. dev. column. The model deviance, which we define as twice the negative log likelihood, is given in the Deviance column. The Deviance diff. column reports the difference in deviance compared with the model with the lowest deviance, which is always the model with the highest-degree fractional polynomial.

The Test df column displays the degrees of freedom used when testing a model's fit against the fit of the model with the lowest deviance. For normal error models such as linear regression, a partial $F$ test is performed, and Test df is the numerator degrees of freedom of the $F$ test. In other settings, a likelihood-ratio test is performed, and Test df is the degrees of freedom of the $\chi^{2}$ statistic. In both cases, the $p$-value for the test is reported in column $P$.

Under robust variance estimation and some other cases (see [R] Irtest), the likelihood-ratio test cannot be performed. When the likelihood-ratio test cannot be performed on the model specified in est_cmd, fp still reports the model-comparison table, but the comparison tests are not performed.
fp reports the "best" model as the model with the lowest deviance; however, users may choose a more efficient model based on the comparison table. They may choose the lowest degree model that the partial $F$ test (or likelihood-ratio test) fails to reject in favor of the lowest deviance model.

After the comparison table, the results of the estimation command for the lowest deviance model are shown. Here the best model has terms weight ${ }^{(-2,-2)}$. However, based on the model-comparison table, we can reject the model without weight and the linear model at the 0.1 significance level. We fail to reject the $m=1$ model at any reasonable level. We will choose the FP1 model, which includes weight ${ }^{(-.5)}$.

We use $f p$ again to estimate the parameters for this model. We use the $f p()$ option to specify what powers we want to use; this option specifies that we do not want to perform a search for the best powers. We also specify the replace option to overwrite the previously created fractional polynomial power variables.

```
. fp <weight>, fp(-.5) replace: regress mpg <weight> foreign
-> regress mpg weight_1 foreign
```

| Source | SS | df | MS | Number of obs | = | 74 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | F (2, 71) | = | 79.51 |
| Model | 1689.20865 | 2 | 844.604325 | Prob > F | = | 0.0000 |
| Residual | 754.25081 | 71 | 10.6232508 | R -squared | = | 0.6913 |
|  |  |  |  | Adj R-squared | = | 0.6826 |
| Total | 2443.45946 | 73 | 33.4720474 | Root MSE | = | 3.2593 |


| mpg | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :--- | :---: | :---: | ---: | ---: |
| weight_1 | 66.89665 | 6.021749 | 11.11 | 0.000 | 54.88963 | 78.90368 |
| foreign | -2.095622 | 1.043513 | -2.01 | 0.048 | -4.176329 | -.0149157 |
| _cons | -17.58651 | 3.397992 | -5.18 | 0.000 | -24.36192 | -10.8111 |

Alternatively, we can use fp generate to create the fractional polynomial variable corresponding to weight ${ }^{(-.5)}$ and then use regress. We store weight ${ }^{(-.5)}$ in the new variable wgt_nsqrt.

```
. fp generate wgt_nsqrt=weight^(-.5)
```

. regress mpg wgt_nsqrt foreign

| Source | SS | df | MS | Number of obs | = | 74 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | $\mathrm{F}(2,71)$ | = | 79.51 |
| Model | 1689.20874 | 2 | 844.604371 | Prob > F | = | 0.0000 |
| Residual | 754.250718 | 71 | 10.6232495 | R -squared | = | 0.6913 |
|  |  |  |  | Adj R-squared | = | 0.6826 |
| Total | 2443.45946 | 73 | 33.4720474 | Root MSE | = | 3.2593 |


| mpg | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :--- | :---: | :---: | ---: | ---: |
| wgt_nsqrt_1 | 66.89665 | 6.021748 | 11.11 | 0.000 | 54.88963 | 78.90368 |
| foreign | -2.095622 | 1.043513 | -2.01 | 0.048 | -4.176328 | -.0149155 |
| _cons | -17.58651 | 3.397991 | -5.18 | 0.000 | -24.36191 | -10.8111 |

## Scaling

Fractional polynomials are defined only for positive term variables. By default, fp will assume that the variable $x$ is positive and attempt to compute fractional powers of $x$. If the positive value assumption is incorrect, an error will be reported and estimation will not be performed.

If the values of the variable are too large or too small, the reported results of $f p$ may be difficult to interpret. By default, cubic powers and squared reciprocal powers will be considered in the search for the best fractional polynomial in term.

We can scale the variable $x$ to 1) make it positive and 2) ensure its magnitude is not too large or too small.

Suppose you have data on hospital patients with age as a fractional polynomial variable of interest. age is required to be greater than or equal to 20 , so you might generate an age 5 variable by typing

```
. generate age5 = (age-19)/5
```

A unit change in age5 is equivalent to a five-year change in age, and the minimum value of age 5 is $1 / 5$ instead of 20 .

In the automobile example of Fractional polynomial regression, our term variable was automobile weight (lbs.). Cars weigh in the thousands of pounds, so cubing their weight figures results in large numbers. We prevented this from being a problem by shrinking the weight by 1,000 ; that is, we typed

```
. replace weight = weight/1000
```

Calendar year is another type of variable that can have a problematically large magnitude. We can shrink this by dividing by 10 , making a unit change correspond to a decade.

```
. generate decade = calendar_year/10
```

You may also have a variable that measures deviation from zero. Perhaps x has already been demeaned and is symmetrically about zero. The fractional polynomial in $x$ will be undefined for half of its domain. We can shift the location of $x$, making it positive by subtracting its minimum and adding a small number to it. Suppose $x$ ranges from -4 to 4 ; we could use
. generate newx $=x+5$
Rescaling ourselves provides easily communicated results. We can tell exactly how the scaling was performed and how it should be performed in similar applications.

Alternatively, fp can scale the fractional polynomial variable so that its values are positive and the magnitude of the values are not too large. This can be done automatically or by directly specifying the scaling values.

Scaling can be automatically performed with $f p$ by specifying the scale option. If term has nonpositive values, the minimum value of term is subtracted from each observation of term. In this case, the counting interval, the minimum distance between the sorted values of term, is also added to each observation of term.

After adjusting the location of term so that its minimum value is positive, creating term*, automatic scaling will divide each observation of term by a power of ten. The exponent of this scaling factor is given by

$$
\begin{aligned}
p & =\log _{10}\left\{\max \left(\text { term }^{*}\right)-\min \left(\text { term }^{*}\right)\right\} \\
p^{*} & =\operatorname{sign}(p) \text { floor }(|p|)
\end{aligned}
$$

Rather than letting fp automatically choose the scaling of term, you may specify adjustment and scale factors $a$ and $b$ by using the scale $(a b)$ option. Fractional powers are then calculated using the $($ term $+a) / b$ values.

When scale or scale $(a b)$ is specified, values in the variable term are not modified; fp merely remembers to scale the values whenever powers are calculated.

In addition to fp , both scale and scale $(a b)$ may be used with fp generate.
You will probably not use scale $(a b)$ with fp for values of $a$ and $b$ that you create yourself, although you could. As we demonstrated earlier, it is usually easier just to generate a scaled variable.
scale ( $a b$ ) is useful when you previously fit a model using scale in one dataset and now want to create the fractional polynomials in another. In the first dataset, fp with scale added notes to the dataset concerning the values of $a$ and $b$. You can see them by typing

```
. notes
```

You can then use fp generate, scale ( $a b$ ) in the second dataset.
When you apply the scaling rules of a previously fit model to new data with the scale ( $a b$ ) option, it is possible that the scaled term may have nonpositive values. $f p$ will be unable to calculate the fractional powers of the term in this case and will issue an error.

The options zero and catzero cause $f p$ and $f p$ generate to output zero values for each fractional polynomial variable when the input (scaled) fractional polynomial variable is nonpositive. Specifying catzero causes a dummy variable indicating nonpositive values of the (scaled) fractional polynomial variable to be included in the model. A detailed example of the use of catzero and zero is shown in example 3 below.

Using the scaling options, we can fit our previous model again using the auto.dta. We specify scale (01000) so that $f p$ will shrink the magnitude of weight in estimating the regression. This is done for demonstration purposes because our scaling rule is simple. As mentioned before, in practice, you would probably only use scale $(a b)$ when applying the scaling rules from a previous analysis. Allowing fp to scale does have the advantage of not altering the original variable, weight.

```
. use https://www.stata-press.com/data/r18/auto, clear
(1978 automobile data)
```

. fp <weight>, fp(-.5) scale(0 1000) : regress mpg <weight> foreign
-> regress mpg weight_1 foreign

| Source | SS | df | MS | Number of obs | = | 74 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | $\mathrm{F}(2,71)$ | = | 79.51 |
| Model | 1689.20861 | 2 | 844.604307 | Prob > F | = | 0.0000 |
| Residual | 754.250846 | 71 | 10.6232514 | R -squared | = | 0.6913 |
|  |  |  |  | Adj R-squared | = | 0.6826 |
| Total | 2443.45946 | 73 | 33.4720474 | Root MSE | = | 3.2593 |


| mpg | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | $[95 \%$ conf. interval] |  |
| ---: | ---: | :--- | :---: | :---: | ---: | ---: |
| weight_1 | 66.89665 | 6.021749 | 11.11 | 0.000 | 54.88963 | 78.90368 |
| foreign | -2.095622 | 1.043513 | -2.01 | 0.048 | -4.176329 | -.0149159 |
| _cons | -17.58651 | 3.397992 | -5.18 | 0.000 | -24.36192 | -10.81111 |

The scaling is clearly indicated in the variable notes for the generated variable weight_1.

```
. notes weight_1
weight_1:
    1. fp term 1 of }\mp@subsup{x}{}{\wedge}(-.5), where x is weight scaled.
    2. Scaling was user specified: x = (weight+a)/b where a=0 and b=1000
    3. Fractional polynomial variables created by fp <weight>, fp(-.5)
        scale(0 1000): regress mpg <weight> foreign
    4. To re-create the fractional polynomial variables, for instance, in
        another dataset, type fp gen double weight`(-.5), scale(0 1000)
```


## Centering

The fractional polynomial of term, centered on $c$ is

$$
\left(\operatorname{term}^{\left(p_{1}, \ldots, p_{m}\right)}-c^{\left(p_{1}, \ldots, p_{m}\right)}\right)^{\prime} \boldsymbol{\beta}
$$

The intercept of a centered fractional polynomial can be interpreted as the effect at zero for all the covariates. When we center the fractional polynomial terms using $c$, the intercept is now interpreted as the effect at term $=c$ and zero values for the other covariates.

Suppose we wanted to center the fractional polynomial of $x$ with powers $(0,0,2)$ at $x=c$.

$$
\begin{aligned}
& \left(x^{(0,0,2)}-c^{(0,0,2)}\right)^{\prime} \boldsymbol{\beta} \\
& =\beta_{0}+\beta_{1}\left(x^{(0)}-c^{(0)}\right)+\beta_{2}\left\{x^{(0)} \ln (x)-c^{(0)} \ln (c)\right\}+\beta_{3}\left(x^{(2)}-c^{(2)}\right) \\
& =\beta_{0}+\beta_{1}\{\ln (x)-\ln (c)\}+\beta_{2}\left[\{\ln (x)\}^{2}-\{\ln (c)\}^{2}\right]+\beta_{3}\left(x^{2}-c^{2}\right)
\end{aligned}
$$

When center is specified, fp centers based on the sample mean of (scaled) term. A previously chosen value for centering, $c$, may also be specified in center ( $c$ ). This would be done when applying the results of a previous model fitting to a new dataset.

The center and center (c) options may be used in fp or fp generate.
Returning to the model of mileage per gallon based on automobile weight and foreign origin, we refit the model with the fractional polynomial of weight centered at its scaled mean.

```
. use https://www.stata-press.com/data/r18/auto, clear
(1978 automobile data)
. fp <weight>, fp(-.5) scale(0 1000) center: regress mpg <weight> foreign
-> regress mpg weight_1 foreign
```

| Source | SS | df | MS | Number of obs | = | 74 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | $\mathrm{F}(2,71)$ | = | 79.51 |
| Model | 1689.20861 | 2 | 844.604307 | Prob > F | = | 0.0000 |
| Residual | 754.250846 | 71 | 10.6232514 | R-squared | = | 0.6913 |
|  |  |  |  | Adj R-squared | = | 0.6826 |
| Total | 2443.45946 | 73 | 33.4720474 | Root MSE | = | 3.2593 |


| mpg | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | $[95 \%$ conf. interval] |  |
| ---: | ---: | :--- | :---: | :---: | ---: | ---: |
| weight_1 | 66.89665 | 6.021749 | 11.11 | 0.000 | 54.88963 | 78.90368 |
| foreign | -2.095622 | 1.043513 | -2.01 | 0.048 | -4.176329 | -.0149159 |
| _cons | 20.91163 | .4624143 | 45.22 | 0.000 | 19.9896 | 21.83366 |

Note that the coefficients for weight_1 and foreign do not change. Only the intercept _cons changes. It can be interpreted as the estimated average miles per gallon of an American-made car of average weight.

## Examples

> Example 1: Linear regression
Consider the serum immunoglobulin G (IgG) dataset from Isaacs et al. (1983), which consists of 298 independent observations in young children. The dependent variable sqrtigg is the square root of the IgG concentration, and the independent variable age is the age of each child. (Preliminary Box-Cox analysis shows that a square root transformation removes the skewness in IgG.)

The aim is to find a model that accurately predicts the mean of sqrtigg given age. We use $f p$ to find the best FP2 model (the default option). We specify center for automatic centering. The age of each child is small in magnitude and positive, so we do not use the scaling options of $f p$ or scale ourselves.

```
. use https://www.stata-press.com/data/r18/igg, clear
(Immunoglobulin in children)
. fp <age>, scale center: regress sqrtigg <age>
(fitting 44 models)
(. . . . 10% . . . 20% . . . . 30% . . . 40% . . . . 50% . . . 60% . . . . 70% . . . . 80% . . . . 90% . . . . 100%)
```

Fractional polynomial comparisons:

| age | Test <br> df | Deviance | Residual <br> std. dev. | Deviance <br> diff. | P | Powers |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| omitted | 4 | 427.539 | 0.497 | 108.090 | 0.000 |  |
| linear | 3 | 337.561 | 0.428 | 18.113 | 0.000 | 1 |
| $m=1$ | 2 | 327.436 | 0.421 | 7.987 | 0.020 | 0 |
| $\mathrm{~m}=2$ | 0 | 319.448 | 0.416 | 0.000 | -- | -22 |

Note: Test df is degrees of freedom, and $\mathbf{P}=\mathrm{P}>\mathrm{F}$ is sig. level for tests comparing models vs. model with $m=2$ based on deviance difference, F (df, 293) .

| Source | SS | df | MS | Number of obs | = | 298 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | F (2, 295) |  | 64.49 |
| Model | 22.2846976 | 2 | 11.1423488 | Prob > F | = | 0.0000 |
| Residual | 50.9676492 | 295 | . 172771692 | R -squared | = | 0.3042 |
|  |  |  |  | Adj R-squared |  | 0.2995 |
| Total | 73.2523469 | 297 | . 246640898 | Root MSE |  | . 41566 |


| sqrtigg | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| age_1 | -.1562156 | .027416 | -5.70 | 0.000 | -.2101713 | -.10226 |
| age_2 | .0148405 | .0027767 | 5.34 | 0.000 | .0093757 | .0203052 |
| _cons | 2.283145 | .0305739 | 74.68 | 0.000 | 2.222974 | 2.343315 |

The new variables created by fp contain the best-fitting fractional polynomial powers of age, as centered by $f p$. For example, age_1 is centered by subtracting the mean of age raised to the power -2 .

The variables created by $f p$ and $f p$ generate are centered or scaled as specified by the user, which is reflected in the estimated regression coefficients and intercept. Centering does have its advantages (see Centering earlier in this entry). By default, $f p$ will not perform scaling or centering. For a more detailed discussion, see Royston and Sauerbrei (2008, sec. 4.11).

The fitted curve has an asymmetric $S$ shape. The best model has powers $(-2,2)$ and deviance 319.448. We reject lesser degree models: the null, linear, and natural $\log$ power models at the 0.05 level. As many as 44 models have been fit in the search for the best powers. Now let's look at
models of degree $\leq 4$. The highest allowed degree is specified in dimension(). We overwrite the previously generated fractional polynomial power variables by including replace.

```
. fp <age>, dimension(4) center replace: regress sqrtigg <age>
(fitting 494 models)
(. . . . 10% . . . 20% . . . . 30% . . . 40% . . . . 50% . . . . 60% . . . . 70% . . . . 80% . . . . 90% . . . . 100%%)
Fractional polynomial comparisons:
```

| age | Test <br> df | Deviance | Residual <br> std. dev. | Deviance <br> diff. | P | Powers |
| ---: | ---: | ---: | ---: | ---: | ---: | :--- |
| omitted | 8 | 427.539 | 0.497 | 109.795 | 0.000 |  |
| linear | 7 | 337.561 | 0.428 | 19.818 | 0.007 | 1 |
| $m=1$ | 6 | 327.436 | 0.421 | 9.692 | 0.149 | 0 |
| $m=2$ | 4 | 319.448 | 0.416 | 1.705 | 0.798 | -22 |
| $m=3$ | 2 | 319.275 | 0.416 | 1.532 | 0.476 | -211 |
| $m=4$ | 0 | 317.744 | 0.416 | 0.000 | -- | 0333 |

Note: Test df is degrees of freedom, and $\mathbf{P}=P>F$ is sig. level for tests comparing models vs. model with $m=4$ based on deviance difference, F(df, 289).

| Source | SS | df | MS | Number of obs | = | 298 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | F (4, 293) | = | 32.63 |
| Model | 22.5754541 | 4 | 5.64386353 | Prob > F | = | 0.0000 |
| Residual | 50.6768927 | 293 | . 172958678 | R -squared | = | 0.3082 |
|  |  |  |  | Adj R-squared | = | 0.2987 |
| Total | 73.2523469 | 297 | . 246640898 | Root MSE | = | . 41588 |


| sqrtigg | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :---: | ---: | :---: | ---: | ---: |
| age_1 | .8761824 | .1898721 | 4.61 | 0.000 | .5024962 | 1.249869 |
| age_2 | -.1922029 | .0684934 | -2.81 | 0.005 | -.3270044 | -.0574015 |
| age_3 | .2043794 | .074947 | 2.73 | 0.007 | .0568767 | .3518821 |
| age_4 | -.0560067 | .0212969 | -2.63 | 0.009 | -.097921 | -.0140924 |
| _cons | 2.238735 | .0482705 | 46.38 | 0.000 | 2.143734 | 2.333736 |

It appears that the FP4 model is not significantly different from the other fractional polynomial models (at the 0.05 level).

Let's compare the curve shape from the $m=2$ model with that from a conventional quartic polynomial whose fit turns out to be significantly better than a cubic (not shown). We use the ability of $f p$ both to generate the required powers of age, namely, $(1,2,3,4)$ for the quartic and $(-2,2)$ for the second-degree fractional polynomial, and to fit the model. The fp() option is used to specify the powers. We use predict to obtain the fitted values of each regression. We fit both models with fp and graph the resulting curves with twoway scatter.

```
fp <age>, center fp(1 2 3 4) replace: regress sqrtigg <age>
-> regress sqrtigg age_1 age_2 age_3 age_4
```

| Source | SS | df | MS | Number of obs | = | 298 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | F (4, 293) | = | 32.65 |
| Model | 22.5835458 | 4 | 5.64588646 | Prob > F | = | 0.0000 |
| Residual | 50.668801 | 293 | . 172931061 | R -squared | = | 0.3083 |
|  |  |  |  | Adj R-squared |  | 0.2989 |
| Total | 73.2523469 | 297 | . 246640898 | Root MSE | $=$ | . 41585 |


| sqrtigg | Coefficient | Std. err. | t | $\mathrm{P}>\mid \mathrm{tI}$ | [95\% conf. interval] |  |
| ---: | ---: | :---: | ---: | :---: | ---: | ---: |
| age_1 | 2.047831 | .4595962 | 4.46 | 0.000 | 1.143302 | 2.952359 |
| age_2 | -1.058902 | .2822803 | -3.75 | 0.000 | -1.614456 | -.5033479 |
| age_3 | .2284917 | .0667591 | 3.42 | 0.001 | .0971037 | .3598798 |
| age_4 | -.0168534 | .0053321 | -3.16 | 0.002 | -.0273475 | -.0063594 |
| _cons | 2.240012 | .0480157 | 46.65 | 0.000 | 2.145512 | 2.334511 |

- predict fit1
(option xb assumed; fitted values)
. label variable fit1 "Quartic"
. fp <age>, center fp(-2 2) replace: regress sqrtigg <age>
-> regress sqrtigg age_1 age_2

| Source | SS | df | MS | Number of obs | = | 298 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | F (2, 295) | = | 64.49 |
| Model | 22.2846976 | 2 | 11.1423488 | Prob > F | = | 0.0000 |
| Residual | 50.9676492 | 295 | . 172771692 | R -squared | = | 0.3042 |
|  |  |  |  | Adj R-squared | = | 0.2995 |
| Total | 73.2523469 | 297 | . 246640898 | Root MSE | = | . 41566 |


| sqrtigg | Coefficient | Std. err. | t | $\mathrm{P}>\mid \mathrm{tl}$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| age_1 | -.1562156 | .027416 | -5.70 | 0.000 | -.2101713 | -.10226 |
| age_2 | .0148405 | .0027767 | 5.34 | 0.000 | .0093757 | .0203052 |
| _cons | 2.283145 | .0305739 | 74.68 | 0.000 | 2.222974 | 2.343315 |

[^26]. label variable fit2 "FP 2"
. scatter sqrtigg fit1 fit2 age, c(. l l) m(o i i) msize (small)
> lpattern(. __.) ytitle("Square root of IgG") xtitle("Age (years)")


The quartic curve has an unsatisfactory wavy appearance that is implausible for the known behavior of IgG, the serum level of which increases throughout early life. The fractional polynomial curve (FP2) increases monotonically and is therefore biologically the more plausible curve. The two models have approximately the same deviance.

## > Example 2: Cox regression

Data from Smith et al. (1992) contain times to complete healing of leg ulcers in a randomized, controlled clinical trial of two treatments in 192 elderly patients. Several covariates were available, of which an important one is mthson, the number of months since the recorded onset of the ulcer. This time is recorded in whole months, not fractions of a month; therefore, some zero values are recorded.

Because the response variable is time to an event of interest and some (in fact, about one-half) of the times are censored, using Cox regression to analyze the data is appropriate. We consider fractional polynomials in mthson, adjusting for four other covariates: age; ulcarea, the area of tissue initially affected by the ulcer; deepppg, a binary variable indicating the presence or absence of deep vein involvement; and treat, a binary variable indicating treatment type.

We fit fractional polynomials of degrees 1 and 2 with fp . We specify scale to perform automatic scaling on mthson. This makes it positive and ensures that its magnitude is not too large. (See Scaling for more details.) The display option nohr is specified before the colon so that the coefficients and not the hazard ratios are displayed.

The center option is specified to obtain automatic centering. age and ulcarea are also demeaned by using summarize and then subtracting the returned result $r$ (mean).

In Cox regression, there is no constant term, so we cannot see the effects of centering in the table of regression estimates. The effects would be present if we were to graph the baseline hazard or survival function because these functions are defined with all predictors set equal to 0 .

In these graphs, we will see the estimated baseline hazard or survival function under no deep vein involvement or treatment and under mean age, ulcer area, and number of months since the recorded onset of the ulcer.
. use https://www.stata-press.com/data/r18/legulcer2, clear
(Leg ulcer clinical trial)
. stset ttevent, fail(healed)
Survival-time data settings
Failure event: healed!=0 \& healed<.
Observed time interval: ( 0 , ttevent]
Exit on or before: failure


Fractional polynomial comparisons:


The best-fitting fractional polynomial of degree 2 has powers $(0.5,0.5)$ and deviance 736.709 . However, this model does not fit significantly better than the fractional polynomial of degree 1 (at the 0.05 level), which has power -0.5 and deviance 738.969. We prefer the model with $m=1$.

| . fp <mthson>, replace center scale nohr fp <br> > deepppg treat <br> -> stcox mthson_1 age ulcarea deepppg treat |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Cox regression with Breslow method for ties |  |  |  |  |  |  |
| ```No. of subjects = 192 No. of failures = 92 Time at risk = 13,825``` |  |  |  |  | Number of obs $=192$ |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  | LR chi2(5) | $=106.33$ |
| Log likelihoo | $=-369.48426$ |  |  |  | Prob > chi2 | $=0.0000$ |
| _t | Coefficient | Std. err. | z | $P>\|z\|$ | [95\% conf | interval] |
| mthson_1 | . 1985592 | . 0493922 | 4.02 | 0.000 | . 1017523 | . 2953662 |
| age | -. 02691 | . 0087875 | -3.06 | 0.002 | -. 0441331 | -. 0096868 |
| ulcarea | -. 0017416 | . 0003482 | -5.00 | 0.000 | -. 0024241 | -. 0010591 |
| deepppg | -. 5740759 | . 2185134 | -2.63 | 0.009 | -1.002354 | -. 1457975 |
| treat | -. 1798575 | . 2175726 | -0.83 | 0.408 | -. 6062921 | . 246577 |

The hazard for healing is much higher for patients whose ulcer is of recent onset than for those who have had an ulcer for many months.

A more appropriate analysis of this dataset, if one wanted to model all the predictors, possibly with fractional polynomial functions, would be to use $m f p$; see $[R] \mathbf{m f p}$.

## > Example 3: Logistic regression

The zero option permits fitting a fractional polynomial model to the positive values of a covariate, taking nonpositive values as zero. An application is the assessment of the effect of cigarette smoking as a risk factor. Whitehall 1 is an epidemiological study, which was examined in Royston and Sauerbrei (2008), of 18,403 male British Civil Servants employed in London. We examine the data collected in Whitehall 1 and use logistic regression to model the odds of death based on a fractional polynomial in the number of cigarettes smoked.

Nonsmokers may be qualitatively different from smokers, so the effect of smoking (regarded as a continuous variable) may not be continuous between zero cigarettes and one cigarette. To allow for this possibility, we model the risk as a constant for the nonsmokers and as a fractional polynomial function of the number of cigarettes for the smokers, adjusted for age.

The dependent variable all 10 is an indicator of whether the individual passed away in the 10 years under study. cigs is the number of cigarettes consumed per day. After loading the data, we demean age and create a dummy variable, nonsmoker. We then use $f p$ to fit the model.

```
. use https://www.stata-press.com/data/r18/smoking, clear
(Smoking and mortality data)
. quietly sum age
. replace age = age - r(mean)
variable age was byte now float
(17,260 real changes made)
. generate byte nonsmoker = cond(cigs==0, 1, 0) if cigs < .
. fp <cigs>, zero: logit all10 <cigs> nonsmoker age
(fitting 44 models)
( . . . 10% . . . 20% . . . . 30% . . . 40% . . . 50% . . . . 60% . . . . 70% . . . . 80% . . . . 90% . . . . 100%%)
```

Fractional polynomial comparisons:

| cigs | Test <br> df | Deviance | Deviance <br> diff. | P | Powers |
| ---: | ---: | ---: | ---: | ---: | :--- |
| omitted | 4 | 9990.804 | 46.096 | 0.000 |  |
| linear | 3 | 9958.801 | 14.093 | 0.003 | 1 |
| m $=1$ | 2 | 9946.603 | 1.895 | 0.388 | 0 |
| $\mathrm{~m}=2$ | 0 | 9944.708 | 0.000 | -- | $-1-1$ |

Note: Test df is degrees of freedom, and $\mathbf{P}=\mathrm{P}>\mathrm{chi2}$ is sig. level for tests comparing models vs. model with $m=2$ based on deviance difference, chi2.

Logistic regression

Log likelihood $=-4972.3539$

Number of obs $=17,260$
LR chi2(4) $=1029.03$
Prob > chi2 $=0.0000$
Pseudo R2 $=0.0938$

| all10 | Coefficient | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| cigs_1 | -1.285867 | .3358483 | -3.83 | 0.000 | -1.944117 | -.6276162 |
| cigs_2 | -1.982424 | .572109 | -3.47 | 0.001 | -3.103736 | -.8611106 |
| nonsmoker | -1.223749 | .1119583 | -10.93 | 0.000 | -1.443183 | -1.004315 |
| age | .1194541 | .0045818 | 26.07 | 0.000 | .1104739 | .1284343 |
| _cons | -1.591489 | .1052078 | -15.13 | 0.000 | -1.797693 | -1.385286 |

Omission of the zero option would cause $f p$ to halt with an error message because nonpositive covariate values (for example, values of cigs) are invalid unless the scale option is specified.

A closely related approach involves the catzero option. Here we no longer need to have nonsmoker in the model, because $f p$ creates its own dummy variable cigs_0 to indicate whether the individual does not smoke on that day.

```
. fp <cigs>, catzero replace: logit all10 <cigs> age
(fitting 44 models)
( . . . 10% . . . 20% . . . . 30% . . . 40% . . . 50% . . . . 60% . . . . 70% . . . . 80% . . . 90% . . . 100%)
```

Fractional polynomial comparisons:

| cigs | Test <br> df | Deviance | Deviance <br> diff. | P | Powers |
| ---: | ---: | ---: | ---: | ---: | :--- |
| omitted | 5 | 10175.75 | 231.047 | 0.000 |  |
| linear | 3 | 9958.80 | 14.093 | 0.003 | 1 |
| m $=1$ | 2 | 9946.60 | 1.895 | 0.388 | 0 |
| $\mathrm{~m}=2$ | 0 | 9944.71 | 0.000 | -- | $-1-1$ |

Note: Test df is degrees of freedom, and $\mathbf{P}=\mathrm{P}>$ chi2 is sig. level
for tests comparing models vs. model with $m=2$ based on deviance difference, chi2.
Logistic regression $\quad$ Number of obs $=17,260$
LR chi2(4) $=1029.03$
Prob > chi2 $=0.0000$
Pseudo R2 $=0.0938$

| all10 | Coefficient | Std. err. | $z$ | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| cigs_0 | -1.223749 | .1119583 | -10.93 | 0.000 | -1.443183 | -1.004315 |
| cigs_1 | -1.285867 | .3358483 | -3.83 | 0.000 | -1.944117 | -.6276162 |
| cigs_2 | -1.982424 | .572109 | -3.47 | 0.001 | -3.103736 | -.8611106 |
| age | .1194541 | .0045818 | 26.07 | 0.000 | .1104739 | .1284343 |
| _cons | -1.591489 | .1052078 | -15.13 | 0.000 | -1.797693 | -1.385286 |

Under both approaches, the comparison table suggests that we can accept the FP1 model instead of the FP2 model. We estimate the parameters of the accepted model-that is, the one that uses the natural logarithm of cigs-with fp.

| . fp <cigs>, catzero replace fp(0): logit all10 <cigs> age |  |
| :--- | :--- |
| -> logit allio cigs_0 cigs_1 age |  |
| Logistic regression | Number of obs $=17,260$ |
|  | LR chi2 (3) $=1027.13$ |
|  | Prob $>$ chi2 $=0.0000$ |
| Log likelihood $=-4973.3016$ | Pseudo R2 $=0.0936$ |


| all10 | Coefficient | Std. err. | $z$ | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| cigs_0 | .1883732 | .1553093 | 1.21 | 0.225 | -.1160274 | .4927738 |
| cigs_1 | .3469842 | .0543552 | 6.38 | 0.000 | .2404499 | .4535185 |
| age | .1194976 | .0045818 | 26.08 | 0.000 | .1105174 | .1284778 |
| _cons | -3.003767 | .1514909 | -19.83 | 0.000 | -3.300683 | -2.70685 |

The high $p$-value for cigs_0 in the output indicates that we cannot reject that there is no extra effect at zero for nonsmokers.

## Stored results

In addition to the results that est_cmd stores, fp stores the following in e():
Scalars
e(fp_dimension)
e(fp_center_mean)
e(fp_scale_a)
e(fp_scale_b)
e(fp_compare_df2)
Macros
e(fp_cmd)
e(fp_cmdline)
e(fp_variable)
e(fp_terms)
e(fp_gen_cmdline)
e(fp_catzero)
e(fp_zero)
e(fp_compare_type)
Matrices
e(fp_fp)
e(fp_compare)
e(fp_compare_stat)
e(fp_compare_df1)
e(fp_compare_fp)
e(fp_compare_length)
e(fp_powers)
degree of fractional polynomial
value used for centering or .
value used for scaling or .
value used for scaling or .
denominator degree of freedom in $F$ test
fp, search(): or fp, powers():
full $f p$ command as typed
fractional polynomial variable
generated $f p$ variables
fp generate command to re-create e(fp_terms) variables
catzero, if specified
zero, if specified
F or chi2
powers used in fractional polynomial
results of model comparisons
$F$ test statistics
chi2 degrees of freedom or numerator degrees of freedom of $F$ test powers of comparison models encoded string for display of row titles powers that are searched
fp generate stores the following in $r()$ :
Scalars
$r$ (fp_center_mean) value used for centering or .
$r\left(f p-s c a l e \_a\right) \quad$ value used for scaling or .
$r\left(f p \_s c a l e \_b\right) \quad$ value used for scaling or .
Macros
$r$ (fp_cmdline)
$r$ (fp_variable)
$r$ (fp_terms)
r(fp_catzero)
r(fp_zero)
full $f p$ generate command as typed
fractional polynomial variable
generated $f p$ variables
catzero, if specified
zero, if specified
Matrices
$r\left(f p \_f p\right) \quad$ powers used in fractional polynomial

## Methods and formulas

The general definition of a fractional polynomial, accommodating possible repeated powers, may be written for functions $H_{1}(x), \ldots, H_{m}(x)$ of $x>0$ as

$$
\beta_{0}+\sum_{j=1}^{m} \beta_{j} H_{j}(x)
$$

where $H_{1}(x)=x^{\left(p_{1}\right)}$ and for $j=2, \ldots, m$,

$$
H_{j}(x)= \begin{cases}x^{\left(p_{j}\right)} & \text { if } p_{j} \neq p_{j-1} \\ H_{j-1}(x) \ln (x) & \text { if } p_{j}=p_{j-1}\end{cases}
$$

For example, a fractional polynomial of degree 3 with powers $(1,3,3)$ has $H_{1}(x)=x, H_{2}(x)=x^{3}$, and $H_{3}(x)=x^{3} \ln (x)$ and equals $\beta_{0}+\beta_{1} x+\beta_{2} x^{3}+\beta_{3} x^{3} \ln (x)$.

We can express a fractional polynomial in vector notation by using $\mathbf{H}(x)=\left[H_{1}(x), \ldots, H_{d}(x)\right]^{\prime}$. We define $x^{\left(p_{1}, p_{2}, \ldots, p_{m}\right)}=\left[\mathbf{H}(x)^{\prime}, 1\right]^{\prime}$. Under this notation, we can write

$$
x^{(1,3,3) \prime} \boldsymbol{\beta}=\beta_{0}+\beta_{1} x+\beta_{2} x^{3}+\beta_{3} x^{3} \ln (x)
$$

The fractional polynomial may be centered so that the intercept can be more easily interpreted. When centering the fractional polynomial of $x$ at $c$, we subtract $c^{\left(p_{1}, p_{2}, \ldots, p_{m}\right)}$ from $x^{\left(p_{1}, p_{2}, \ldots, p_{m}\right)}$, where $c^{\left(p_{1}, p_{2}, \ldots, p_{d}\right)}=\left[\mathbf{H}(x)^{\prime}, 0\right]^{\prime}$. The centered fractional polynomial is

$$
\left(x^{\left(p_{1}, \ldots, p_{d}\right)}-c^{\left(p_{1}, \ldots, p_{d}\right)}\right)^{\prime} \boldsymbol{\beta}
$$

The definition may be extended to allow $x \leq 0$ values. For these values, the fractional polynomial is equal to the intercept $\beta_{0}$ or equal to a zero-offset term $\alpha_{0}$ plus the intercept $\beta_{0}$.

The deviance $D$ of a model is defined as -2 times its maximized log likelihood. For normal error models, we use the formula

$$
D=n\left(1-\bar{l}+\ln \frac{2 \pi \mathrm{RSS}}{n}\right)
$$

where $n$ is the sample size, $\bar{l}$ is the mean of the lognormalized weights ( $\bar{l}=0$ if the weights are all equal), and RSS is the residual sum of squares as fit by regress.

When fp is used to search for the best combination of powers, it reports a table comparing fractional polynomial models of degree $k<m$ with the degree $m$ fractional polynomial model, which will have the lowest deviance. The comparison table also includes the linear model, in which <term> is not raised to a power, and the null model, in which <term> is omitted.

The Test df column of the model comparison table does not correspond to the model degrees of freedom for the individual models but rather to the degrees of freedom of a test comparing that model with the model with the lowest deviance. For normal error models, this is the numerator degrees of freedom of a partial $F$ test; for other models, it is the degrees of freedom of the likelihood-ratio $\chi^{2}$ test. When calculating the test degrees of freedom, the command accounts for the two types of parameters that are being estimated by fp : coefficients $\left(\beta_{j}\right)$ and powers. Because the powers in a fractional polynomial are chosen from a finite set rather than from the entire real line, the degrees of freedom defined in this way are approximate and generally yield somewhat conservative tests (Royston and Altman 1994).

The $p$-values reported by fp are calculated differently for normal error models than for other models. Let $D_{k}$ and $D_{m}$ be the deviances of the models with degrees $k$ and $m$, respectively. For normal error models, a variance ratio $F$ is calculated as

$$
F=\frac{d_{2}}{d_{1}}\left\{\exp \left(\frac{D_{k}-D_{m}}{n}\right)-1\right\}
$$

where $d_{1}$ is the numerator df , the number of additional parameters estimated by the degree $m$ model over the degree $k$ model. $d_{2}$ is the denominator degrees of freedom and equals the residual degrees of freedom of the degree $m$ model minus the number of powers estimated, $m$. The $p$-value is obtained by referring $F$ to an $F$ distribution on $\left(d_{1}, d_{2}\right) \mathrm{df}$.

For nonnormal models, the $p$-value is obtained by referring $D_{k}-D_{m}$ to a $\chi^{2}$ distribution on $d_{1}$ degrees of freedom, with $d_{1}$ defined as above.

## Acknowledgment

We thank Patrick Royston of the MrC Clinical Trials Unit, London, and coauthor of the Stata Press book Flexible Parametric Survival Analysis Using Stata: Beyond the Cox Model for writing fracpoly and fracgen, the commands on which $f p$ and $f p$ generate are based. We also thank Professor Royston for his advice on and review of the new $f p$ commands.

## References

Isaacs, D., D. G. Altman, C. E. Tidmarsh, H. B. Valman, and A. D. Webster. 1983. Serum immunoglobulin concentrations in preschool children measured by laser nephelometry: Reference ranges for $\operatorname{IgG}, \operatorname{IgA}, \mathrm{IgM}$. Journal of Clinical Pathology 36: 1193-1196. https://doi.org/10.1136/jcp.36.10.1193.
Libois, F., and V. Verardi. 2013. Semiparametric fixed-effects estimator. Stata Journal 13: 329-336.
Royston, P., and D. G. Altman. 1994. Regression using fractional polynomials of continuous covariates: Parsimonious parametric modelling. Journal of the Royal Statistical Society, Series C 43: 429-467. https://doi.org/10.2307/2986270.
Royston, P., and W. Sauerbrei. 2008. Multivariable Model-Building: A Pragmatic Approach to Regression Analysis Based on Fractional Polynomials for Modelling Continuous Variables. Chichester, UK: Wiley.

Smith, J. M., C. J. Dore, A. Charlett, and J. D. Lewis. 1992. A randomized trial of Biofilm dressing for venous leg ulcers. Phlebology 7: 108-113. https://doi.org/10.1177/026835559200700307.

## Also see

[R] fp postestimation - Postestimation tools for fp
[R] mfp - Multivariable fractional polynomial models
[U] 20 Estimation and postestimation commands

## Title

fp postestimation - Postestimation tools for fp

| Postestimation commands | predict | margins |
| :--- | :--- | :--- |
| fp plot and fp predict | Remarks and examples | Methods and formulas |
| Acknowledgment | Reference | Also see |

## Postestimation commands

The following postestimation commands are of special interest after $f p$ :
Command Description
fp plot component-plus-residual plot from most recently fit fractional polynomial model
fp predict create variable containing prediction or SEs of fractional polynomials

The following standard postestimation commands are also available if available after est_cmd:

| Command | Description |
| :---: | :---: |
| contrast | contrasts and ANOVA-style joint tests of estimates |
| estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estimates | cataloging estimation results |
| etable | table of estimation results |
| forecast | dynamic forecasts and simulations |
| hausman | Hausman's specification test |
| lincom | point estimates, standard errors, testing, and inference for linear combinations of coefficients |
| linktest | link test for model specification |
| lrtest | likelihood-ratio test |
| margins | marginal means, predictive margins, marginal effects, and average marginal effects |
| marginsplot | graph the results from margins (profile plots, interaction plots, etc.) |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients |
| predict | predictions, residuals, influence statistics, and other diagnostic measures |
| predictnl | point estimates, standard errors, testing, and inference for generalized predictions |
| pwcompare | pairwise comparisons of estimates |
| suest | seemingly unrelated estimation |
| test | Wald tests of simple and composite linear hypotheses |
| testnl | Wald tests of nonlinear hypotheses |

## predict

The behavior of predict following fp is determined by est_cmd. See the corresponding est_cmd postestimation entry for available predict options.

Also see information on fp predict below.

## margins

The behavior of margins following fp is determined by est_cmd. See the corresponding est_cmd postestimation entry for available margins options.

## fp plot and fp predict

## Description for fp plot and fp predict

fp plot produces a component-plus-residual plot. The fractional polynomial comprises the component, and the residual is specified by the user in residuals (). The component-plus-residuals are plotted against the fractional polynomial variable. If you only want to plot the component fit, without residuals, you would specify residuals(none).
fp predict generates the fractional polynomial or the standard error of the fractional polynomial. The fractional polynomial prediction is equivalent to the fitted values prediction given by predict, xb , with the covariates other than the fractional polynomial variable set to zero. The standard error may be quite large if the range of the other covariates is far from zero. In this situation, the covariates would be centered and their range would include, or come close to including, zero.

These postestimation commands can be used only when the fractional polynomial variables do not interact with other variables in the specification of est_cmd. See [U] 11.4.3 Factor variables for more information about interactions.

## Menu for fp plot and fp predict

## fp plot

Statistics $>$ Linear models and related $>$ Fractional polynomials $>$ Component-plus-residual plot

## fp predict

Statistics $>$ Linear models and related $>$ Fractional polynomials $>$ Fractional polynomial prediction

## Syntax for fp plot and fp predict

Component-plus-residual plot for most recently fit fractional polynomial model
fp plot [if] [in], residuals(res_option) [graph_options]

Create variable containing the prediction or SEs of fractional polynomials
fp predict [type] newvar [if] [in] [, predict_options]

## graph_options

Description

## Main

*residuals(res_option)
residual option name to use in predict after est_cmd, or residuals (none) if residuals are not to be graphed
equation (eqno)
specify equation
level(\#)
set confidence level; default is level (95)
Plot
plotopts(scatter_options) affect rendition of the component-plus-residual scatter points
Fitted line
lineopts (cline_options) affect rendition of the fitted line
Cl plot
ciopts (area_options) affect rendition of the confidence bands
Add plots
addplot (plot) add other plots to the generated graph
Y axis, X axis, Titles, Legend, Overall
twoway_options
any options other than by () documented in [G-3] twoway_options
${ }^{*}$ residuals(res_option) is required.
predict_options
Description
Main
fp calculate the fractional polynomial; the default
stdp
equation(eqno)
calculate the standard error of the fractional polynomial specify equation

## Options for fp plot

Main
residuals (res_option) specifies what type of residuals to plot in the component-plus-residual plot. res_option is the same option that would be specified to predict after est_cmd. Residuals can be omitted from the plot by specifying residuals (none). residuals () is required.
equation (eqno) is relevant only when you have previously fit a multiple-equation model in est_cmd. It specifies the equation to which you are referring.
equation(\#1) would mean that the calculation is to be made for the first equation, equation(\#2) would mean the second, and so on. You could also refer to the equations by their names: equation(income) would refer to the equation named income, and equation(hours) would refer to the equation named hours.
If you do not specify equation(), the results are the same as if you specified equation(\#1). level (\#); see [R] Estimation options.
$\qquad$ [G-2] graph twoway scatter.
$\qquad$ Fitted line
lineopts(cline_options) affects the rendition of the fitted line; see [G-3] cline_options.
$\qquad$
ciopts(area_options) affects the rendition of the confidence bands; see [G-3] area_options.
$\sqrt{\text { Add plots }}$
addplot (plot) provides a way to add other plots to the generated graph. See [G-3] addplot_option.
$\qquad$ Y axis, X axis, Titles, Legend, Overall
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

## Options for fp predict

$\qquad$ Main
fp calculates the fractional polynomial, the linear prediction with other variables set to zero. This is the default.
stdp calculates the standard error of the fractional polynomial.
equation (eqno) is relevant only when you have previously fit a multiple-equation model in est_cmd. It specifies the equation to which you are referring.
equation(\#1) would mean that the calculation is to be made for the first equation, equation(\#2) would mean the second, and so on. You could also refer to the equations by their names: equation(income) would refer to the equation named income, and equation(hours) would refer to the equation named hours.
If you do not specify equation(), the results are the same as if you specified equation(\#1).

## Remarks and examples

After a model is fit using $f p$, the estimated fractional polynomial may be of interest. This is the linear combination of the fractional polynomial terms and the constant intercept using the model coefficients estimated by fp . It is equivalent to the fitted values prediction given by predict, xb , with the covariates and the fractional polynomial variable set to zero. When these other covariates have been centered, the prediction is made at the centering values of the covariates.

A component-plus-residual plot is generated by fp plot. The fractional polynomial comprises the component, and the residual is specified by the user in residuals(). The residuals() option takes the same argument that would be supplied to predict after est_cmd to obtain the desired type of residuals. If you only want to plot the component fit, without residuals, you would specify residuals(none).
fp predict generates the fractional polynomial. If the stdp option is specified, the standard error of the fractional polynomial is generated instead. This standard error may be quite large if the range of the other covariates is far from zero. In this situation, the covariates would be centered and their range would include, or come close to including, zero.

These postestimation commands can be used only when the fractional polynomial terms do not interact with other variables in the specification of est_cmd. See [U] 11.4.3 Factor variables for more information about interactions.

## Examples

$>$ Example 1: fp plot after linear regression
In example 1 of [R] fp, we modeled the mean of the square root of a child's serum immunoglobulin $\mathrm{G}(\mathrm{IgG})$ level as a fractional polynomial function of the child's age. An FP2 model with powers $(-2,2)$ is chosen.

We load the data and then fit the model with $f p$. Then, we use $f p$ plot to draw the component-plus-residual plot. A $95 \%$ confidence interval is produced for the fractional polynomial in age (the component). The residuals prediction option for regress is specified in the residuals() option in $f p$ plot so that the residuals are rendered.

```
. use https://www.stata-press.com/data/r18/igg
(Immunoglobulin in children)
. fp <age>, scale center: regress sqrtigg <age>
    (output omitted)
. fp plot, residuals(residuals)
```



## > Example 2: fp plot after Cox regression

In example 2 of $[\mathrm{R}] \mathbf{f p}$, we modeled the time to complete healing of leg ulcers for 192 elderly patients using a Cox regression. A one-degree fractional polynomial in mthson, the number of months since the onset of the ulcer, is used as a predictor in the regression. The power -0.5 is used for mthson. Other covariates are age (age), ulcer area (ulcarea), treatment type, and a binary indicator of deep vein involvement (deepppg).

We load the data and then demean ulcer area and age. Then, we fit the model with $f p$ and draw the component-plus-residual plot with fp plot. mgale is specified in the residuals() option to obtain martingale residuals. See [ST] stcox postestimation for more details.

```
. use https://www.stata-press.com/data/r18/legulcer2, clear
(Leg ulcer clinical trial)
. quietly stset ttevent, failure(healed)
. quietly summarize age
. replace age = age - r(mean)
variable age was byte now float
(192 real changes made)
. quietly summarize ulcarea
. replace ulcarea = ulcarea - r(mean)
variable ulcarea was int now float
(192 real changes made)
. fp <mthson>, replace center scale nohr fp(-.5): stcox <mthson> age ulcarea
> deepppg treat
    (output omitted)
fp plot, residuals(mgale)
```



## Example 3: fp plot and fp predict after logistic regression

In example 3 of [R] fp, we used logistic regression to model the odds of death for male civil servants in Britain conditional on cigarette consumption. The dependent variable all10 is an indicator of whether the individual passed away in the 10 years under study.

Nonsmokers may be qualitatively different from smokers, so the effect of smoking (regarded as a continuous variable) may not be continuous between zero cigarettes and one cigarette. To allow for this possibility, we model the risk as constant intercept for the nonsmokers and as a fractional polynomial function of the number of cigarettes for the smokers, cigs, adjusted for age. An FP1 model with power 0 is chosen.

We load the data and demean age. Then, we fit the model using $f p$ and graph the fit of the model and $95 \%$ confidence interval using fp plot. Only the component fit is graphed by specifying residuals(none).

```
. use https://www.stata-press.com/data/r18/smoking, clear
(Smoking and mortality data)
. quietly summarize age
. replace age = age - r(mean)
variable age was byte now float
(17,260 real changes made)
. fp <cigs>, catzero replace fp(0): logit all10 <cigs> age
-> logit all10 cigs_0 cigs_1 age
Logistic regression Number of obs = 17,260
                                    LR chi2(3) = 1027.13
                                    Prob > chi2 = 0.0000
                                    Pseudo R2 = 0.0936
```

| all10 | Coefficient | Std. err. | $z$ | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | :---: | ---: | :---: | ---: | ---: |
| cigs_0 | .1883732 | .1553093 | 1.21 | 0.225 | -.1160274 | .4927738 |
| cigs_1 | .3469842 | .0543552 | 6.38 | 0.000 | .2404499 | .4535185 |
| age | .1194976 | .0045818 | 26.08 | 0.000 | .1105174 | .1284778 |
| _cons | -3.003767 | .1514909 | -19.83 | 0.000 | -3.300683 | -2.70685 |

[^27]

We see a small spike at zero for cigs because of the effect of cigs_0 on the fractional polynomial; however, the high $p$-value for cigs_0 in the model output indicates that we cannot reject that there is no extra effect at zero for nonsmokers.

We can also use fp predict to predict the fractional polynomial for nonsmokers and the mean of age. This is the value at the spike. We store the result in fpO . We see it is equivalent to the sum of the constant intercept estimate and the estimate of the cigs_0 coefficient.

```
. fp predict fpO if cigs == 0
(7,157 missing values generated)
. summarize fp0
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline fp0 & 10,103 & -2.815393 & & 0 & -2.815393
\end{tabular}
. display _b[cigs_0]+_b[_cons]
-2.8153935
```


## Methods and formulas

Let the data consist of triplets $\left(y_{i}, x_{i}, \mathbf{z}_{i}\right), i=1, \ldots, n$, where $\mathbf{z}_{i}$ is the vector of covariates for the $i$ th observation and $x_{i}$ is the fractional polynomial variable.
fp predict calculates the fractional polynomial at the centering value $x_{0}, \widehat{\eta}_{i}=\left(x_{i}{ }^{\left(p_{1}, \ldots, p_{d}\right)}-\right.$ $\left.x_{0}{ }^{\left(p_{1}, \ldots, p_{m}\right)}\right)^{\prime} \widehat{\boldsymbol{\beta}}$. This is equivalent to the linear predictor of the model at $\mathbf{z}_{i}=\mathbf{0}$. The standard error is calculated from the variance-covariance matrix of $\widehat{\boldsymbol{\beta}}$, ignoring estimation of the powers. When $x_{i} \leq 0, \mathbf{H}\left(x_{i}\right)$, and thus $x_{i}{ }^{\left(p_{1}, \ldots, p_{m}\right)}$, is either undefined or zero. A zero offset term, $\alpha_{0}$, may be added to $\widehat{\eta}_{i}$ for these nonpositive $x_{i}$ values.

The values $\widehat{\eta}_{i}$ represent the behavior of the fractional polynomial model for $x$ at fixed values $\mathbf{z}=\mathbf{0}$ of the (centered) covariates. The $i$ th component-plus-residual is defined as $\widehat{\eta}_{i}+d_{i}$, where $d_{i}$ is the residual for the $i$ th observation. The definition of $d_{i}$ will change according to the type of model used and the preference of the user. fp plot plots $\widehat{\eta}_{i}+d_{i}$ versus $x_{i}$, overlaying $\widehat{\eta}_{i}$ and its confidence interval.

## Acknowledgment

We thank Patrick Royston of the MRC Clinical Trials Unit, London, and coauthor of the Stata Press book Flexible Parametric Survival Analysis Using Stata: Beyond the Cox Model for writing fracplot and fracpred, the commands on which fp plot and fp predict are based. We also thank Professor Royston for his advice on and review of $f p$ plot and fp predict.

## Reference

Royston, P. 2017. Model selection for univariable fractional polynomials. Stata Journal 17: 619-629.

## Also see

[R] fp - Fractional polynomial regression
[U] 20 Estimation and postestimation commands

## Title

fracreg - Fractional response regression

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

fracreg fits a fractional response model for a dependent variable that is greater than or equal to 0 and less than or equal to 1 . It uses a probit, logit, or heteroskedastic probit model for the conditional mean. These models are often used for outcomes such as rates, proportions, and fractional data.

## Quick start

Fractional probit model for y with values between 0 and 1 on continuous variable x 1
fracreg probit y x1
Same as above, but use logit distribution
fracreg logit y x1
Fractional probit model for y on x 1 and use x 2 to model the variance of y
fracreg probit $y$ x1, het(x2)

## Menu

Statistics $>$ Fractional outcomes $>$ Fractional regression

## Syntax

Syntax for fractional probit regression
fracreg probit depvar [indepvars] [if] [in] [weight] [, options]
Syntax for fractional logistic regression
fracreg logit depvar [indepvars] [if] [in] [weight] [, options]

Syntax for fractional heteroskedastic probit regression

$$
\begin{aligned}
& \text { fracreg probit depvar }[\text { indepvars }][\text { if }][\text { in }][\text { weight }], \\
& \operatorname{het}\left(\text { varlist }\left[, \text { offset }\left(\text { varname }_{o}\right)\right]\right)[\text { options }]
\end{aligned}
$$

options
Description
Model
noconstant
offset (varname)
constraints (constraints)
*het(varlist $\left[\right.$, offset $\left(\right.$ varname $\left.\left._{o}\right)\right]$
suppress constant term
include varname in model with coefficient constrained to 1 apply specified linear constraints
independent variables to model the variance and optional offset variable with fracreg probit
vcetype may be robust, cluster clustvar, bootstrap, or jackknife

Reporting
level(\#)
or
nocnsreport
display_options
set confidence level; default is level (95)
report odds ratios; only valid with fracreg logit do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
control the maximization process; seldom used
do not display the coefficient table; seldom used
keep collinear variables
display legend instead of statistics
*het () may be used only with fracreg probit to compute fractional heteroskedastic probit regression. indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
depvar and indepvars may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bayes, bootstrap, by, collect, fp, jackknife, mi estimate, rolling, statsby, and svy are allowed; see
[U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: fracreg.
vce (bootstrap) and vce (jackknife) are not allowed with the mi estimate prefix; see [MI] mi estimate.
Weights are not allowed with the bootstrap prefix; see $[R]$ bootstrap.
vce(), nocoef, and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
nocoef, collinear, and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

$\qquad$ Model
noconstant, offset (varname), constraints (constraints); see [R] Estimation options. het (varlist $\left[\right.$, offset $\left(\right.$ varname $\left.\left._{o}\right)\right]$ ) specifies the independent variables and, optionally, the offset variable in the variance function. het() may only be used with fracreg probit to compute fractional heteroskedastic probit regression.
offset ( varname $_{o}$ ) specifies that selection offset varname ${ }_{o}$ be included in the model with the coefficient constrained to be 1 .

## SE/Robust

vce (vcetype) specifies the type of standard error reported, which includes types that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
level(\#); see [R] Estimation options.
or reports the estimated coefficients transformed to odds ratios, that is, $e^{b}$ rather than $b$. Standard errors and confidence intervals are similarly transformed. This option affects how results are displayed, not how they are estimated or may be specified at estimation or when replaying previously estimated results. This option may only be used with fracreg logit.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate(\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from (init_specs); see [R] Maximize. These options are seldom used.

The following options are available with fracreg but are not shown in the dialog box:
nocoef specifies that the coefficient table not be displayed. This option is sometimes used by programmers but is of no use interactively.
collinear, coeflegend; see [R] Estimation options.

## Remarks and examples

Fractional response data may occur when the outcome of interest is measured as a fraction, for example, a patient's oxygen saturation or Gini coefficient values. These data are also often observed when proportions are generated from aggregated binary outcomes. For example, rather than having data on whether individual students passed an exam, we might simply have data on the proportion of students in each school that passed.

These models are appropriate when you have a dependent variable that takes values between 0 and 1 and may also be equal to 0 or 1 , denoted for conciseness with the notation $[0,1]$. If the dependent variable takes only values between 0 and 1, betareg might be a valid alternative. betareg provides more flexibility in the distribution of the mean of the dependent variable but is misspecified if the dependent variable is equal to 0 or 1 . See $[R]$ betareg for more information.

These models have been applied to various topics. For example, Papke and Wooldridge (1996) studied the participation rates of employees in firms' 401(k) retirement plans. Papke and Wooldridge (2008) also evaluated an education policy by studying the pass rates for an exam administered to fourth grade Michigan students over time.

The models fit by fracreg are quasilikelihood estimators like the generalized linear models described in $[R]$ glm. Fractional regression is a model of the mean of the dependent variable $y$ conditional on covariates $\mathbf{x}$, which we denote by $\mu_{\mathbf{x}}$. Because $y$ is in $[0,1]$, we must ensure that $\mu_{\mathbf{x}}$ is also in $[0,1]$. We do this by using a probit, logit, or heteroskedastic probit model for $\mu_{\mathbf{x}}$.

The key insight from quasilikelihood estimation is that you do not need to know the true distribution of the entire model to obtain consistent parameter estimates. In fact, the only information that you need is the correct specification of the conditional mean.

This means that the true model does not need to be, for example, a probit. If the true model is a probit, then fitting a probit regression via maximum likelihood gives you consistent parameter estimates and asymptotically efficient standard errors.

By contrast, if the conditional mean of the model is the same as the conditional mean of a probit but the model is not a probit, the point estimates are consistent, but the standard errors are not asymptotically efficient. The standard errors are not efficient, because no assumptions about the distribution of the unobserved components in the model are made. Thus fracreg uses robust standard errors by default.

For further discussion on quasilikelihood estimation in the context of fractional regression, please see Papke and Wooldridge (1996) and Wooldridge (2010).

## > Example 1: Fractional probit model of rates

In this example, we look at the expected participation rate in $401(\mathrm{k})$ plans for a cross-section of firms. Participation rate (prate) is defined as the fraction of eligible employees in a firm that participate in a $401(\mathrm{k})$ plan. We use summarize to see the range of the participation rate.

```
. use https://www.stata-press.com/data/r18/401k
(Firm-level data on 401k participation)
```

- summarize prate

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| prate | 4,075 | .840607 | .1874841 | .0036364 | 1 |

The variable has values between 0 and 1 but also has at least 1 firm for which the participation rate is exactly 1 .

As in Papke and Wooldridge (1996), we surmise that the expected participation rate depends on the matching rate of employee $401(\mathrm{k})$ contributions (mrate), the natural $\log$ of the total number of employees (ltotemp), the age of the plan (age), and whether the $401(\mathrm{k})$ plan is the only retirement plan offered by the employer (sole). We include ltotemp and age, along with their squares, using factor-variable notation; see [U] 11.4.3 Factor variables.

If we believe that the functional form of the expected participation rate is a cumulative normal density, we may use fracreg probit.
. fracreg probit prate mrate c.ltotemp\#\#c.ltotemp c.age\#\#c.age i.sole
Iteration 0: Log pseudolikelihood $=-1769.6832$
Iteration 1: Log pseudolikelihood $=-1675.2763$
Iteration 2: Log pseudolikelihood $=-1674.6234$
Iteration 3: Log pseudolikelihood $=-1674.6232$

Iteration 4: Log pseudolikelihood $=-1674.6232$$\quad$|  |
| :--- |
| Fractional probit regression |

| prate | Coefficient | Robust <br> std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| mrate <br> ltotemp | -.5859715 | .0387616 | 15.12 | 0.000 | .5100002 | .6619429 |
| c.ltotemp\# |  |  |  |  |  |  |
| c.ltotemp | .03102767 | .0615052 | -9.92 | 0.000 | -.7308246 | -.4897288 |
| age | .0273266 | .0031926 | 8.56 | 0.000 | .0210691 | .033584 |
| c.age\#c.age | -.0003159 | .0000875 | -3.61 | 0.000 | -.0004874 | -.0001443 |
| sole |  |  |  |  |  |  |
| Only plan |  |  |  |  |  |  |
| _cons | .0683196 | .0272091 | 2.51 | 0.012 | .0149908 | .1216484 |

Like those obtained from probit, the parameters provide the sign of the marginal effect of the covariates on the outcome, but the magnitude is difficult to interpret. We can use margins to estimate conditional or population-averaged effects; see example 2. The standard errors are robust by default because the true data-generating process need not be a probit, even though we use the probit likelihood to obtain our parameter estimates.

## > Example 2: Changing the distribution of the conditional mean

Continuing with example 1, we may instead believe that the expected participation rate follows a fractional logistic response. In this case, we should use fractional logistic regression instead of fractional probit regression to obtain consistent estimates of the parameters of the conditional mean.

| Iteration 0: Log pseudolikelihood = -1983.8372 |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Iteration 1: Log pseudolikelihood $=-1682.4496$ |  |  |  |  |  |  |
| Iteration 2: Log pseudolikelihood = -1673.6458 |  |  |  |  |  |  |
| Iteration 3: Log pseudolikelihood $=-1673.5566$ |  |  |  |  |  |  |
| Iteration 4: Log pseudolikelihood = -1673.5566 |  |  |  |  |  |  |
| Fractional logistic regression |  |  |  |  | Number of | $=4,075$ |
|  |  |  |  |  | Wald chi2 | 817.73 |
|  |  |  |  |  | Prob > chi2 | $=0.0000$ |
| Log pseudolikelihood $=-1673.5566$ |  |  |  |  | Pseudo R2 | $=0.0638$ |
| prate | Coefficient | Robust <br> std. err. | z | $P>\|z\|$ | [95\% conf | interval] |
| mrate | 1.143516 | . 074748 | 15.30 | 0.000 | . 9970125 | 1.290019 |
| ltotemp | -1.103275 | . 1130667 | -9.76 | 0.000 | -1.324882 | -. 8816687 |
| c.ltotemp\# |  |  |  |  |  |  |
| age | . 0512643 | . 0059399 | 8.63 | 0.000 | . 0396223 | . 0629064 |
| c.age\#c.age | -. 0005891 | . 0001645 | -3.58 | 0.000 | -. 0009114 | -. 0002667 |
| sole |  |  |  |  |  |  |
| Only plan | . 1137479 | . 0507762 | 2.24 | 0.025 | . 0142284 | . 2132674 |
| _cons | 5.747761 | . 4294386 | 13.38 | 0.000 | 4.906077 | 6.589445 |

Like those obtained from logit, the parameters provide the sign of the marginal effect of the covariates on the outcome, but the magnitude is again difficult to interpret. As with fracreg probit in example 1, we would use margins to obtain the marginal effects or other predictions of interest.
> Example 3: Odds ratios from a fractional logit model
When the conditional mean of our outcome is interpretable as a probability, it is possible to adopt an odds-ratio interpretation of the results of a fractional logit model. In example 2, this is plausible because expected participation rates can be viewed as estimates of the probability of participation. We obtain the odds ratios by specifying the option or.

```
. fracreg logit prate mrate c.ltotemp##c.ltotemp c.age##c.age i.sole, or
Iteration 0: Log pseudolikelihood = -1983.8372
Iteration 1: Log pseudolikelihood = -1682.4496
Iteration 2: Log pseudolikelihood = -1673.6458
Iteration 3: Log pseudolikelihood = -1673.5566
Iteration 4: Log pseudolikelihood = -1673.5566
Fractional logistic regression Number of obs = 4,075
Wald chi2(6) = 817.73
Prob > chi2 = 0.0000
Pseudo R2 = 0.0638
```

| prate | Odds ratio | Robust std. err. | z | $P>\|z\|$ | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| mrate | 3.137781 | . 2345429 | 15.30 | 0.000 | 2.710173 | 3.632857 |
| ltotemp | . 3317826 | . 0375136 | -9.76 | 0.000 | . 2658343 | . 4140913 |
| c.ltotemp\# <br> c.ltotemp | 1.058209 | . 0077125 | 7.76 | 0.000 | 1.043201 | 1.073434 |
| age | 1.052601 | . 0062524 | 8.63 | 0.000 | 1.040418 | 1.064927 |
| c.age\#c.age | . 9994111 | . 0001644 | -3.58 | 0.000 | . 999089 | . 9997333 |
| sole | 1.12047 | 0568932 | 2.24 | 0. 025 | 1.01433 | 1.237716 |
| _cons | 313.4879 | 134.6238 | 13.38 | 0.000 | 135.1083 | 727.3771 |

```
Note: _cons estimates baseline odds.
```

Among other things, we see that if the $401(\mathrm{k})$ is the only plan offered by the employer, then the odds of an employee participating increase by a factor of 1.12 . We can also see that if the matching rate goes from 0 to $1: 1$ (exactly matching employee contributions) or from $1: 1$ to $2: 1$ (doubling employee contributions), then the odds of participating increase by 3.1.

The use of an odds-ratio interpretation is not appropriate if the conditional mean cannot be viewed as a probability. For example, if the fractional outcome were a Gini coefficient, we could not interpret the expected values of our outcomes as probabilities. The Gini coefficient is a measure of inequality between zero and one and cannot be interpreted as a probability. In this case, using the odds-ratio option would not be sensible.

## Stored results

fracreg stores the following in e():
Scalars

```
e(N)
e(k)
e(k_eq)
e(k_eq_model)
e(k_dv)
e(df_m)
e(r2_p)
e(ll)
e(ll_0)
e(N_clust)
e(chi2)
e(p)
e(rank)
e(ic)
e(rc)
e(converged)
```

Macros

```
e(cmd)
e(cmdline)
e(estimator)
e(depvar)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(offset)
e(chi2type)
e(vce)
e(vcetype)
e(opt)
e(which)
e(ml_method)
e(user)
e(technique)
e(properties)
e(estat_cmd)
e(predict)
e(marginsnotok)
e(asbalanced)
e(asobserved)
```

Matrices
e(b)
$e(m n s)$
e(Cns)
e(ilog)
e(gradient)
e(V)
e(V_modelbased)

Functions

```
e(sample)
```

number of observations
number of parameters
number of equations in $e(b)$
number of equations in overall model test
number of dependent variables
model degrees of freedom
pseudo- $R^{2}$
log likelihood
log likelihood, constant-only model
number of clusters
$\chi^{2}$
$p$-value for model test
rank of e(V)
number of iterations
return code
1 if converged, 0 otherwise

## fracreg

command as typed
model for conditional mean; logit, probit, or hetprobit
name of dependent variable
weight type
weight expression
title in estimation output
name of cluster variable
offset
Wald; type of model $\chi^{2}$ test
vcetype specified in vce()
title used to label Std. err.
type of optimization
max or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement estat
program used to implement predict
predictions disallowed by margins
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
vector of means of the independent variables
constraints matrix
iteration $\log$ (up to 20 iterations)
gradient vector
variance-covariance matrix of the estimators
model-based variance
marks estimation sample

In addition to the above, the following is stored in r() :
Matrices
$r$ (table)
matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

The log-likelihood function for fractional models is of the form

$$
\ln L=\sum_{j=1}^{N} w_{j} y_{j} \ln \left\{G\left(\mathbf{x}_{j}^{\prime} \boldsymbol{\beta}\right)\right\}+w_{j}\left(1-y_{j}\right) \ln \left\{1-G\left(\mathbf{x}_{j}^{\prime} \boldsymbol{\beta}\right)\right\}
$$

where $N$ is the sample size, $y_{j}$ is the dependent variable, $w_{j}$ denotes the optional weights, $\ln L$ is maximized, as described in [R] Maximize, and $G(\cdot)$ can be

| Model | Functional form for $G\left(\mathbf{x}_{j}^{\prime} \boldsymbol{\beta}\right)$ |
| :---: | :---: |
| probit | $\Phi\left(\mathbf{x}_{j}^{\prime} \boldsymbol{\beta}\right)$ |
| logit | $\exp \left(\mathbf{x}_{j}^{\prime} \boldsymbol{\beta}\right) /\left\{1+\exp \left(\mathbf{x}_{j}^{\prime} \boldsymbol{\beta}\right)\right\}$ |
| hetprobit | $\Phi\left\{\mathbf{x}_{j}^{\prime} \boldsymbol{\beta} / \exp \left(\mathbf{z}_{j}^{\prime} \boldsymbol{\gamma}\right)\right\}$ |

where $\mathbf{x}_{j}$ are the covariates for individual $j, \mathbf{z}_{j}$ are the covariates used to model the variance of the outcome for the heteroskedastic probit model, and $\Phi$ is the standard normal cumulative density function.

## References

Gray, L. A., and M. Hernández-Alava. 2018. A command for fitting mixture regression models for bounded dependent variables using the beta distribution. Stata Journal 18: 51-75.
Papke, L. E., and J. M. Wooldridge. 1996. Econometric methods for fractional response variables with an application to $401(\mathrm{k})$ plan participation rates. Journal of Applied Econometrics 11: 619-632. https://doi.org/10.1002/(SICI) 1099-1255(199611)11:6〈619::AID-JAE418〉3.0.CO;2-1.
——. 2008. Panel data methods for fractional response variables with an application to test pass rates. Journal of Econometrics 145: 121-133. https://doi.org/10.1016/j.jeconom.2008.05.009.
Wooldridge, J. M. 2010. Econometric Analysis of Cross Section and Panel Data. 2nd ed. Cambridge, MA: MIT Press.
Wulff, J. N. 2019. Generalized two-part fractional regression with cmp. Stata Journal 19: 375-389.
Xu, J., and J. S. Long. 2005. Confidence intervals for predicted outcomes in regression models for categorical outcomes. Stata Journal 5: 537-559.

## Also see

[R] fracreg postestimation - Postestimation tools for fracreg
[R] betareg - Beta regression
[R] glm - Generalized linear models
[R] ivfprobit - Fractional probit model with continuous endogenous covariates
[BAYES] bayes: fracreg - Bayesian fractional response regression
[MI] Estimation - Estimation commands for use with mi estimate
[SVY] svy estimation - Estimation commands for survey data
[U] 20 Estimation and postestimation commands

## Title

fracreg postestimation - Postestimation tools for fracreg

| Postestimation commands | predict |
| :--- | :--- |
| Remarks and examples | Also see |

## Postestimation commands

The following standard postestimation commands are available after fracreg:

| Command | Description |
| :--- | :--- |
| contrast | contrasts and ANOVA-style joint tests of estimates <br> estat ic |
| Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian in- <br> formation criteria (AIC, CAIC, AICc, and BIC) |  |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estat (svy) | postestimation statistics for survey data <br> cataloging estimation results |
| estimates | table of estimation results |
| etable | dynamic forecasts and simulations |
| forecast | Hausman's specification test |
| lincom | point estimates, standard errors, testing, and inference for linear combinations |
| margins | marginal means, predictive margins, marginal effects, and average marginal effects |
| marginsplot | graph the results from margins (profile plots, interaction plots, etc.) |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations |
| of coefficients |  |

[^28]
## predict

## Description for predict

predict creates a new variable containing predictions such as conditional means, linear predictions, standard errors, and equation-level scores.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

```
predict [type] newvar [if] [in] [, statistic nooffset]
predict [type] stub* [if] [in], scores
```

statistic Description

Main
$\mathrm{cm} \quad$ conditional mean; the default
$\mathrm{xb} \quad$ linear prediction
sigma standard deviation of the error term (for het())
stdp standard error of the linear prediction

## Options for predict

$\qquad$ Main
cm , the default, calculates the conditional mean of the outcome.
xb calculates the linear prediction.
sigma calculates the standard deviation of the error term. It is available only when het () is specified. stdp calculates the standard error of the linear prediction.
nooffset is relevant only if you specified offset (varname). It modifies the calculations made by predict so that they ignore the offset variable; the linear prediction is treated as $\mathbf{x}_{j} \mathbf{b}$ rather than as $\mathbf{x}_{j} \mathbf{b}+$ offset $_{j}$.
scores calculates the equation-level scores. In the case of fracreg probit and fracreg logit, $\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)$ is calculated, and if the option het() was specified with fracreg probit, then $\partial \ln L / \partial\left(\mathbf{z}_{j} \gamma\right)$ is also calculated.

## margins

## Description for margins

margins estimates margins of response for conditional means and linear predictions.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [点redict(statistic ...)...] [options]
```

statistic Description
$\mathrm{cm} \quad$ conditional mean; the default
$x b \quad$ linear prediction
sigma standard deviation of the error term (for het())
stdp not allowed with margins
scores not allowed with margins

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

## Remarks and examples

Remarks are presented under the following headings:
Obtaining predicted values
Performing hypothesis tests

## Obtaining predicted values

Once you have fit a model using fracreg, you can obtain the conditional mean of the fractional response by using the predict command for both the estimation sample and other samples; see [U] 20 Estimation and postestimation commands and [R] predict.

When you use the fractional probit estimator, fracreg probit, with the option het (), there is an additional statistic available, sigma. With the sigma option, predict calculates the predicted standard deviation, $\sigma_{j}=\exp \left(\mathbf{z}_{j} \gamma\right)$.

## Performing hypothesis tests

## > Example 1: Conditional means

In example 1 of [R] fracreg, we fit a fractional probit model to see how participation rate (prate) in 401(k) plans is affected by the matching rate of employer contributions (mrate). To obtain the predicted conditional means, we use predict and do not specify the default cm option.

```
. use https://www.stata-press.com/data/r18/401k
(Firm-level data on 401k participation)
. fracreg probit prate mrate c.ltotemp##c.ltotemp c.age##c.age i.sole
Iteration 0: Log pseudolikelihood = -1769.6832
Iteration 1: Log pseudolikelihood = -1675.2763
Iteration 2: Log pseudolikelihood = -1674.6234
Iteration 3: Log pseudolikelihood = -1674.6232
Iteration 4: Log pseudolikelihood = -1674.6232
Fractional probit regression Number of obs = 4,075
Wald chi2(6) = 815.88
Prob > chi2 = 0.0000
Pseudo R2 = 0.0632
```

| prate | Coefficient | Robust |  |  |  |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| std. err. | z | P>\|z| | [95\% conf. interval] |  |  |  |
| mrate | .5859715 | .0387616 | 15.12 | 0.000 | .5100002 | .6619429 |
| ltotemp | -.6102767 | .0615052 | -9.92 | 0.000 | -.7308246 | -.4897288 |
| c.ltotemp\# |  |  |  |  |  |  |
| c.ltotemp | .0313576 | .003975 | 7.89 | 0.000 | .0235667 | .0391484 |
| age | .0273266 | .0031926 | 8.56 | 0.000 | .0210691 | .033584 |
| c.age\#c.age | -.0003159 | .0000875 | -3.61 | 0.000 | -.0004874 | -.0001443 |
| sole |  |  |  |  |  |  |
| Only plan | .0683196 | .0272091 | 2.51 | 0.012 | .0149908 | .1216484 |
| _cons | 3.25991 | .2323929 | 14.03 | 0.000 | 2.804429 | 3.715392 |

```
- predict mpart
(option cm assumed)
```

We can then summarize these conditional mean estimates (cmean) over the population to get the population average conditional mean participation rate in our sample.

```
. summarize mpart
```

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| mpart | 4,075 | .8405767 | .0828094 | .6251739 | .9964518 |

The average of the conditional mean of participation rate in our sample is $84 \%$ with a range between $62.5 \%$ and $99.6 \%$.

## > Example 2: Average marginal effects

In example 2 of $[\mathrm{R}]$ fracreg, we used the outcome variable and covariates of example 1 but instead of fitting a fractional probit regression, we fit a fractional logit. Using margins, we explore the average marginal effect of mrate on prate for both specifications.

Below, we use margins after fracreg logit with the option post to post the average marginal effects as estimates. We then store our results with the name logit. We do the same with our probit estimates.

| . fracreg logit prate mrate c.ltotemp\#\#c.ltotemp c.age\#\#c.age i.sole, or |  |  |  | c.age\#\#c.age i.sole, or |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Iteration 0: Log pseudolikelihood $=-1983.8372$ |  |  |  |  |  |  |
| Iteration 1: Log pseudolikelihood $=-1682.4496$ |  |  |  |  |  |  |
| Iteration 2: Log pseudolikelihood $=-1673.6458$ |  |  |  |  |  |  |
| Iteration 3: Log pseudolikelihood $=-1673.5566$ |  |  |  |  |  |  |
| Iteration 4: Log pseudolikelihood $=-1673.5566$ |  |  |  |  |  |  |
| Fractional logistic regression |  |  |  |  | Number of | $=4,075$ |
|  |  |  |  |  | Wald chi2 ${ }^{\text {(6) }}$ | $=817.73$ |
|  |  |  |  |  | Prob > chi2 | $=0.0000$ |
| Log pseudolikelihood $=-1673.5566$ |  |  |  |  | Pseudo R2 | $=0.0638$ |
| prate | Odds ratio | Robust std. err. | z | $P>\|z\|$ | [95\% conf | interval] |
| mrate | 3.137781 | . 2345429 | 15.30 | 0.000 | 2.710173 | 3.632857 |
| ltotemp | . 3317826 | . 0375136 | -9.76 | 0.000 | . 2658343 | . 4140913 |
| c. Itotemp\# |  |  |  |  |  |  |
| age | 1.052601 | . 0062524 | 8.63 | 0.000 | 1.040418 | 1.064927 |
| c.age\#c.age | . 9994111 | . 0001644 | -3.58 | 0.000 | . 999089 | . 9997333 |
| sole |  |  |  |  |  |  |
| Only plan | 1.12047 | . 0568932 | 2.24 | 0.025 | 1.01433 | 1.237716 |
| _cons | 313.4879 | 134.6238 | 13.38 | 0.000 | 135.1083 | 727.3771 |

Note: _cons estimates baseline odds.
. margins, dydx(mrate) post
Average marginal effects Number of obs $=4,075$
Model VCE: Robust
Expression: Conditional mean of prate, predict()
dy/dx wrt: mrate

|  | Delta-method |  |  |  |  |  |
| ---: | ---: | ---: | :---: | :---: | :---: | :---: | :---: |
|  | dy/dx | std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| mrate | .1450106 | .0094558 | 15.34 | 0.000 | .1264776 | .1635436 |

```
. estimates store logit
```

The marginal effects from fracreg logit suggest that a small change in the matching rate of employers can increase participation by more than $14 \%$.

```
. quietly fracreg probit prate mrate c.ltotemp##c.ltotemp c.age##c.age i.sole
. margins, dydx(mrate) post
Average marginal effects Number of obs = 4,075
Model VCE: Robust
Expression: Conditional mean of prate, predict()
dy/dx wrt: mrate
```

|  | Delta-method |  |  |  |  |  |
| ---: | ---: | ---: | :---: | :---: | :---: | :---: |
|  | dy/dx | std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| mrate | .1335505 | .0087385 | 15.28 | 0.000 | .1164233 | .1506776 |

```
. estimates store probit
```

For the probit model, a change in the matching rate increases participation by more than $13 \%$.
Because we stored our margins results as estimates, we can now produce a table showing both the logit and probit results.
. estimates table logit probit, se

| Variable | logit | probit |
| ---: | :--- | :--- |
| mrate | .14501059 | .13355046 |
|  | .00945578 | .00873852 |

Legend: b/se
As indicated by the standard errors in the table, both average marginal effects are significant. The difference between the two estimates is approximately one percentage point. This relatively small difference is consistent with the intuition that marginal effects obtained from probit and logit conditional means give us analogous results.
> Example 3: Average marginal effects for different levels of participation
We can also use margins to find the expected participation rate for various levels of employer matching. Using our probit model, we obtain the following by typing

```
. quietly fracreg probit prate mrate c.ltotemp##c.ltotemp c.age##c.age i.sole
. margins, at(mrate=(0(.2)2))
Predictive margins Number of obs = 4,075
Model VCE: Robust
Expression: Conditional mean of prate, predict()
1._at: mrate = 0
2._at: mrate = .2
3._at: mrate = .4
4._at: mrate = . }
5._at: mrate = . 8
6._at: mrate = 1
7._at: mrate = 1.2
8._at: mrate = 1.4
9._at: mrate = 1.6
10._at: mrate = 1.8
11._at: mrate = 2
```

|  | Delta-method <br> std. err. |  |  |  |  | $\mathbf{z}$ |
| :---: | ---: | ---: | ---: | ---: | ---: | ---: |
|  | Margin | P>\|z| | [95\% conf. interval] |  |  |  |
| at |  |  |  |  |  |  |
| 1 | .780858 | .0052738 | 148.06 | 0.000 | .7705216 | .7911944 |
| 2 | .8128364 | .003441 | 236.22 | 0.000 | .8060923 | .8195806 |
| 3 | .8417642 | .002672 | 315.03 | 0.000 | .8365271 | .8470013 |
| 4 | .8675979 | .0029882 | 290.34 | 0.000 | .8617412 | .8734547 |
| 5 | .8903734 | .0036591 | 243.33 | 0.000 | .8832018 | .8975451 |
| 6 | .9101957 | .0042293 | 215.21 | 0.000 | .9019065 | .9184849 |
| 7 | .9272265 | .0045767 | 202.60 | 0.000 | .9182563 | .9361966 |
| 8 | .9416712 | .004694 | 200.61 | 0.000 | .9324711 | .9508712 |
| 9 | .9537652 | .0046115 | 206.82 | 0.000 | .9447268 | .9628035 |
| 10 | .9637608 | .004372 | 220.44 | 0.000 | .9551919 | .9723298 |
| 11 | .9719159 | .0040207 | 241.73 | 0.000 | .9640355 | .9797964 |

Going from no matching to equal matching changes the participation rate from $78 \%$ to $91 \%$, and double matching moves participation all the way to $97.2 \%$.

We can also see these results in a graph by using marginsplot.
. marginsplot

$$
\text { Predictive margins with } 95 \% \text { Cls }
$$



## Also see

[R] fracreg - Fractional response regression
[U] 20 Estimation and postestimation commands

## Title

frontier - Stochastic frontier models

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

frontier fits stochastic production or cost frontier models; the default is a production frontier model. It provides estimators for the parameters of a linear model with a disturbance that is assumed to be a mixture of two components, which have a strictly nonnegative and symmetric distribution, respectively. frontier can fit models in which the nonnegative distribution component (a measurement of inefficiency) is assumed to be from a half-normal, exponential, or truncated-normal distribution. See Kumbhakar and Lovell (2000) for a detailed introduction to frontier analysis.

## Quick start

Cobb-Douglas production frontier model of $\operatorname{lny} 1$ as a function of $\ln x 1$ and $\ln x 2$ frontier lny1 lnx1 lnx2

Same as above, but use exponential instead of half-normal distribution for the inefficiency term frontier lny1 lnx1 lnx2, distribution(exponential)

Include $x 3$ as an explanatory variable in the idiosyncratic error variance function frontier $\operatorname{lny} 1 \ln x 1 \ln x 2$, vhet( $x 3$ )

Same as above, and include $x 4$ as an explanatory variable in the technical inefficiency variance function

```
frontier lny1 lnx1 lnx2, vhet(x3) uhet(x4)
```

Conditional mean model with the mean modeled as a linear function of x3
frontier lny1 lnx1 lnx2, distribution(tnormal) cm(x3)
Cost frontier model of y 2 as a function of $\ln x 1$ and $\ln x 2$
frontier y2 lnx1 lnx2, distribution(tnormal) cost

## Menu

Statistics $>$ Linear models and related $>$ Frontier models

## Syntax

frontier depvar [indepvars] [if] [in] [weight] [, options]
options
Description
Model
noconstant
distribution(hnormal)
distribution(exponential)
distribution(tnormal)
ufrom (matrix)
cm (varlist [, noconstant])
suppress constant term
half-normal distribution for the inefficiency term
exponential distribution for the inefficiency term
truncated-normal distribution for the inefficiency term specify untransformed log likelihood; only with d(tnormal)
fit conditional mean model; only with d(tnormal); use noconstant to suppress constant term

Model 2
constraints (constraints) apply specified linear constraints
uhet (varlist [, noconstant]) explanatory variables for technical inefficiency variance function; use noconstant to suppress constant term vhet(varlist [, noconstant])
cost
SE/Robust

* vce (vcetype)

Reporting
level(\#)
nocnsreport
display_options
vcetype may be oim, robust, cluster clustvar, opg, bootstrap, or jackknife
explanatory variables for idiosyncratic error variance function; use noconstant to suppress constant term fit cost frontier model; default is production frontier model
set confidence level; default is level(95)
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

## Maximization

maximize_options
collinear
coeflegend
control the maximization process; seldom used
keep collinear variables
display legend instead of statistics
*vce(robust) and vce(cluster clustvar) may not be specified with distribution(tnormal).
indepvars and varlist may contain factor variables; see [U] 11.4.3 Factor variables.
bootstrap, by, collect, fp, jackknife, rolling, and statsby are allowed; see [U] 11.1.10 Prefix commands.
Weights are not allowed with the bootstrap prefix; see [R] bootstrap.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

## Model <br> noconstant; see [R] Estimation options.

distribution (distname) specifies the distribution for the inefficiency term as half-normal (hnormal), exponential, or truncated-normal (tnormal). The default is hnormal.
ufrom (matrix) specifies a $1 \times K$ matrix of untransformed starting values when the distribution is truncated-normal (tnormal). frontier can estimate the parameters of the model by maximizing either the $\log$ likelihood or a transformed log likelihood (see Methods and formulas). frontier automatically transforms the starting values before passing them on to the transformed log likelihood. The matrix must have the same number of columns as there are parameters to estimate.
cm (varlist [, noconstant]) may be used only with distribution(tnormal). Here frontier will fit a conditional mean model in which the mean of the truncated-normal distribution is modeled as a linear function of the set of covariates specified in varlist. Specifying noconstant suppresses the constant in the mean function.

## Model 2

constraints (constraints); see [R] Estimation options.
By default, when fitting the truncated-normal model or the conditional mean model, frontier maximizes a transformed log likelihood. When constraints are applied, frontier will maximize the untransformed $\log$ likelihood with constraints defined in the untransformed metric.
uhet (varlist [, noconstant ]) specifies that the technical inefficiency component is heteroskedastic, with the variance function depending on a linear combination of varlist ${ }_{u}$. Specifying noconstant suppresses the constant term from the variance function. This option may not be specified with distribution(tnormal).
vhet (varlist [, noconstant]) specifies that the idiosyncratic error component is heteroskedastic, with the variance function depending on a linear combination of varlist $t_{v}$. Specifying noconstant suppresses the constant term from the variance function. This option may not be specified with distribution(tnormal).
cost specifies that frontier fit a cost frontier model.

## SE/Robust

vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
vce(robust) and vce(cluster clustvar) may not be specified with distribution(tnormal).

[^29]maximize_options: difficult, technique (algorithm_spec), iterate(\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance (\#), nonrtolerance, and from(init_specs); see [R] Maximize. These options are seldom used.

Setting the optimization type to technique (bhhh) resets the default vcetype to vce (opg).

The following options are available with frontier but are not shown in the dialog box:
collinear, coeflegend; see [R] Estimation options.

## Remarks and examples

Stochastic production frontier models were introduced by Aigner, Lovell, and Schmidt (1977) and Meeusen and van den Broeck (1977). Since then, stochastic frontier models have become a popular subfield in econometrics. Kumbhakar and Lovell (2000) provide a good introduction.
frontier fits three stochastic frontier models with distinct parameterizations of the inefficiency term and can fit stochastic production or cost frontier models.

Let's review the nature of the stochastic frontier problem. Suppose that a producer has a production function $f\left(\mathbf{z}_{i}, \boldsymbol{\beta}\right)$. In a world without error or inefficiency, the $i$ th firm would produce

$$
q_{i}=f\left(\mathbf{z}_{i}, \boldsymbol{\beta}\right)
$$

Stochastic frontier analysis assumes that each firm potentially produces less than it might due to a degree of inefficiency. Specifically,

$$
q_{i}=f\left(\mathbf{z}_{i}, \boldsymbol{\beta}\right) \xi_{i}
$$

where $\xi_{i}$ is the level of efficiency for firm $i ; \xi_{i}$ must be in the interval $(0,1]$. If $\xi_{i}=1$, the firm is achieving the optimal output with the technology embodied in the production function $f\left(\mathbf{z}_{i}, \boldsymbol{\beta}\right)$. When $\xi_{i}<1$, the firm is not making the most of the inputs $\mathbf{z}_{i}$ given the technology embodied in the production function $f\left(\mathbf{z}_{i}, \boldsymbol{\beta}\right)$. Because the output is assumed to be strictly positive (that is, $q_{i}>0$ ), the degree of technical efficiency is assumed to be strictly positive (that is, $\xi_{i}>0$ ).

Output is also assumed to be subject to random shocks, implying that

$$
q_{i}=f\left(\mathbf{z}_{i}, \boldsymbol{\beta}\right) \xi_{i} \exp \left(v_{i}\right)
$$

Taking the natural $\log$ of both sides yields

$$
\ln \left(q_{i}\right)=\ln \left\{f\left(\mathbf{z}_{i}, \boldsymbol{\beta}\right)\right\}+\ln \left(\xi_{i}\right)+v_{i}
$$

Assuming that there are $k$ inputs and that the production function is linear in logs, defining $u_{i}=-\ln \left(\xi_{i}\right)$ yields

$$
\begin{equation*}
\ln \left(q_{i}\right)=\beta_{0}+\sum_{j=1}^{k} \beta_{j} \ln \left(z_{j i}\right)+v_{i}-u_{i} \tag{1}
\end{equation*}
$$

Because $u_{i}$ is subtracted from $\ln \left(q_{i}\right)$, restricting $u_{i} \geq 0$ implies that $0<\xi_{i} \leq 1$, as specified above.

Kumbhakar and Lovell (2000) provide a detailed version of the above derivation, and they show that performing an analogous derivation in the dual cost function problem allows us to specify the problem as

$$
\begin{equation*}
\ln \left(c_{i}\right)=\beta_{0}+\beta_{q} \ln \left(q_{i}\right)+\sum_{j=1}^{k} \beta_{j} \ln \left(p_{j i}\right)+v_{i}+u_{i} \tag{2}
\end{equation*}
$$

where $q_{i}$ is output, $z_{j i}$ are input quantities, $c_{i}$ is cost, and the $p_{j i}$ are input prices.
Intuitively, the inefficiency effect is required to lower output or raise expenditure, depending on the specification.

## - Technical note

The model that frontier actually fits is of the form

$$
y_{i}=\beta_{0}+\sum_{j=1}^{k} \beta_{j} x_{j i}+v_{i}-s u_{i}
$$

where

$$
s=\left\{\begin{aligned}
1, & \text { for production functions } \\
-1, & \text { for cost functions }
\end{aligned}\right.
$$

so, in the context of the discussion above, $y_{i}=\ln \left(q_{i}\right)$, and $x_{j i}=\ln \left(z_{j i}\right)$ for a production function; and for a cost function, $y_{i}=\ln \left(c_{i}\right)$, and the $x_{j i}$ are the $\ln \left(p_{j i}\right)$ and $\ln \left(q_{i}\right)$. You must take the natural logarithm of the data before fitting a stochastic frontier production or cost model. frontier performs no transformations on the data.

Different specifications of the $u_{i}$ and the $v_{i}$ terms give rise to distinct models. frontier provides estimators for the parameters of three basic models in which the idiosyncratic component, $v_{i}$, is assumed to be independently $N\left(0, \sigma_{v}\right)$ distributed over the observations. The basic models differ in their specification of the inefficiency term, $u_{i}$, as follows:
exponential: the $u_{i}$ are independently exponentially distributed with variance $\sigma_{u}^{2}$ hnormal: the $u_{i}$ are independently half-normally $N^{+}\left(0, \sigma_{u}^{2}\right)$ distributed tnormal: the $u_{i}$ are independently $N^{+}\left(\mu, \sigma_{u}^{2}\right)$ distributed with truncation point at 0
For half-normal or exponential distributions, frontier can fit models with heteroskedastic error components, conditional on a set of covariates. For a truncated-normal distribution, frontier can also fit a conditional mean model in which the mean is modeled as a linear function of a set of covariates.

## > Example 1: The half-normal and the exponential models

For our first example, we demonstrate the half-normal and exponential models by reproducing a study found in Greene (2003, 505), which uses data originally published in Zellner and Revankar (1969). In this study of the transportation equipment manufacturing industry, observations on value added, capital, and labor are used to estimate a Cobb-Douglas production function. The variable lnv is the log-transformed value added, lnk is the log-transformed capital, and $\operatorname{lnl}$ is the log-transformed labor. OLS estimates are compared with those from stochastic frontier models using both the half-normal and exponential distribution for the inefficiency term.
. use https://www.stata-press.com/data/r18/greene9
(Transportation equipment manufacturing industry)
. regress lnv lnk lnl

| Source | SS | df | MS |
| ---: | :---: | ---: | :---: |
| Model | 44.1727741 | 2 | 22.086387 |
| Residual | 1.22225984 | 22 | .055557265 |
| Total | 45.3950339 | 24 | 1.89145975 |


| Number of obs | $=$ | 25 |
| :--- | :--- | ---: |
| $\mathrm{~F}(2,22)$ | $=$ | 397.54 |
| Prob $>$ F | $=$ | 0.0000 |
| R-squared | $=$ | 0.9731 |
| Adj R-squared | $=$ | 0.9706 |
| Root MSE | $=$ | .23571 |


| lnv | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | :---: | :---: |
| lnk | .2454281 | .1068574 | 2.30 | 0.032 | .0238193 | .4670368 |
| lnl | .805183 | .1263336 | 6.37 | 0.000 | .5431831 | 1.067183 |
| _cons | 1.844416 | .2335928 | 7.90 | 0.000 | 1.359974 | 2.328858 |

. frontier lnv lnk lnl
Iteration 0: Log likelihood $=2.3357572$
Iteration 1: Log likelihood $=2.4673009$
Iteration 2: Log likelihood $=2.4695125$
Iteration 3: Log likelihood $=2.4695222$
Iteration 4: Log likelihood $=2.4695222$
Stoc. frontier normal/half-normal model Number of obs $=\quad 25$
Wald chi2(2) $=743.71$
Prob > chi2 $=0.0000$

| lnv | Coefficient | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| lnk | .2585478 | .098764 | 2.62 | 0.009 | .0649738 | .4521218 |
| lnl | .7802451 | .1199399 | 6.51 | 0.000 | .5451672 | 1.015323 |
| _cons | 2.081135 | .281641 | 7.39 | 0.000 | 1.529128 | 2.633141 |
| /lnsig2v | -3.48401 | .6195353 | -5.62 | 0.000 | -4.698277 | -2.269743 |
| /lnsig2u | -3.014599 | 1.11694 | -2.70 | 0.007 | -5.203761 | -.8254368 |
| sigma_v | .1751688 | .0542616 |  |  | .0954514 | .3214633 |
| sigma_u | .2215073 | .1237052 |  |  | .074134 | .6618486 |
| sigma2 | .0797496 | .0426989 |  |  | -.0039388 | .163438 |
| lambda | 1.264536 | .1678684 |  |  | .9355204 | 1.593552 |

LR test of sigma_u=0: chibar2(01) $=0.43$
Prob >= chibar2 $=0.256$

- predict double u_h, u
. frontier lnv lnk lnl, distribution(exponential)
Iteration 0: Log likelihood $=2.7270659$
Iteration 1: Log likelihood $=2.8551532$
Iteration 2: Log likelihood = 2.8604815
Iteration 3: Log likelihood $=2.8604897$
Iteration 4: Log likelihood $=2.8604897$
Stoc. frontier normal/exponential model
Log likelihood $=2.8604897$

Number of obs = 25
Wald chi2(2) $=845.68$
Prob > chi2 $=0.0000$

| lnv | Coefficient | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | :---: | :---: |
| lnk | .2624859 | .0919988 | 2.85 | 0.004 | .0821717 | .4428002 |
| lnl | .7703795 | .1109569 | 6.94 | 0.000 | .5529079 | .9878511 |
| _cons | 2.069242 | .2356159 | 8.78 | 0.000 | 1.607444 | 2.531041 |
| /lnsig2v | -3.527598 | .4486176 | -7.86 | 0.000 | -4.406873 | -2.648324 |
| /lnsig2u | -4.002457 | .9274575 | -4.32 | 0.000 | -5.820241 | -2.184674 |
| sigma_v | .1713925 | .0384448 |  |  | .1104231 | .2660258 |
| sigma_u | .1351691 | .0626818 |  |  | .0544692 | .3354317 |
| sigma2 | .0476461 | .0157921 |  |  | .016694 | .0785981 |
| lambda | .7886525 | .087684 |  |  | .616795 | .9605101 |

LR test of sigma_u=0: chibar2 $(01)=1.21 \quad$ Prob $>=$ chibar2 $=0.135$
. predict double u_e, u
. list state u_h u_e

| state | u_h | u_e |
| ---: | ---: | ---: |
| Alabama | .2011338 | .14592865 |
| California | .14480966 | .0972165 |
| Connecticut | .1903485 | .13478797 |
| Florida | .51753139 | .5903303 |
| Georgia | .10397912 | .07140994 |
| Illinois | .12126696 | .0830415 |
| Indiana | .21128212 | .15450664 |
| Iowa | .24933153 | .20073081 |
| Kansas | .10099517 | .06857629 |
| Kentucky | .05626919 | .04152443 |
| Louisiana | .20332731 | .15066405 |
| Maine | .22263164 | .17245793 |
| Maryland | .13534062 | .09245501 |
| Massachusetts | .15636999 | .10932923 |
| Michigan | .15809566 | .10756915 |
| Missouri | .10288047 | .0704146 |
| NewJersey | .09584337 | .06587986 |
| NewYork | .27787793 | .22249416 |
| Ohio | .22914231 | .16981857 |
| Pennsylvania | .1500667 | .10302905 |
| Texas | .20297875 | .14552218 |
| Virginia | .14000132 | .09676078 |
| Washington | .11047581 | .07533251 |
| WestVirginia | .15561392 | .11236153 |
| Wisconsin | .14067066 | .0970861 |

The parameter estimates and the estimates of the inefficiency terms closely match those published in Greene (2003, 505), but the standard errors of the parameter estimates are estimated differently (see the technical note below).

The output from frontier includes estimates of the standard deviations of the two error components, $\sigma_{v}$ and $\sigma_{u}$, which are labeled sigma_v and sigma_u, respectively. In the log likelihood, they are parameterized as $\ln \sigma_{v}^{2}$ and $\ln \sigma_{u}^{2}$, and these estimates are labeled $/ \ln s i g 2 \mathrm{v}$ and $/ \operatorname{lnsig} 2 \mathrm{u}$ in the output. frontier also reports two other useful parameterizations. The estimate of the total error variance, $\sigma_{S}^{2}=\sigma_{v}^{2}+\sigma_{u}^{2}$, is labeled sigma2, and the estimate of the ratio of the standard deviation of the inefficiency component to the standard deviation of the idiosyncratic component, $\lambda=\sigma_{u} / \sigma_{v}$, is labeled lambda.

At the bottom of the output, frontier reports the results of a test that there is no technical inefficiency component in the model. This is a test of the null hypothesis $H_{0}: \sigma_{u}^{2}=0$ against the alternative hypotheses $H_{1}: \sigma_{u}^{2}>0$. If the null hypothesis is true, the stochastic frontier model reduces to an OLS model with normal errors. However, because the test lies on the boundary of the parameter space of $\sigma_{u}^{2}$, the standard likelihood-ratio test is not valid, and a one-sided generalized likelihood-ratio test must be constructed; see Gutierrez, Carter, and Drukker (2001). For this example, the output shows LR $=0.43$ with a $p$-value of 0.256 for the half-normal model and $\mathrm{LR}=1.21$ with a $p$-value of 0.135 for the exponential model. There are several possible reasons for the failure to reject the null hypothesis, but the fact that the test is based on an asymptotic distribution and the sample size was 25 is certainly a leading candidate among those possibilities.

## Technical note

frontier maximizes the log-likelihood function of a stochastic frontier model by using the Newton-Raphson method, and the estimated variance-covariance matrix is calculated as the inverse of the negative Hessian (matrix of second partial derivatives); see $[\mathrm{R}] \mathrm{ml}$. When comparing the results with those published using other software, be aware of the difference in the optimization methods, which may result in different, yet asymptotically equivalent, variance estimates.
> Example 2: Models with heteroskedasticity
Often the error terms may not have constant variance. frontier allows you to model heteroskedasticity in either error term as a linear function of a set of covariates. The variance of either the technical inefficiency or the idiosyncratic component may be modeled as

$$
\sigma_{i}^{2}=\exp \left(\mathbf{w}_{i} \boldsymbol{\delta}\right)
$$

The default constant included in $\mathbf{w}_{i}$ may be suppressed by appending a noconstant option to the list of covariates. Also, you can simultaneously specify covariates for both $\sigma_{u_{i}}$ and $\sigma_{v_{i}}$.

In this example, we use a sample of 756 observations of fictional firms producing a manufactured good by using capital and labor. The firms are hypothesized to use a constant returns-to-scale technology, but the sizes of the firms differ. Believing that this size variation will introduce heteroskedasticity into the idiosyncratic error term, we estimate the parameters of a Cobb-Douglas production function. To do this, we use a conditional heteroskedastic half-normal model, with the size of the firm as an explanatory variable in the variance function for the idiosyncratic error. We also perform a test of the hypothesis that the firms use a constant returns-to-scale technology.
. use https://www.stata-press.com/data/r18/frontier1, clear
. frontier lnoutput lnlabor lncapital, vhet(size)
Iteration 0: Log likelihood $=-1508.3692$
Iteration 1: Log likelihood $=-1501.583$
Iteration 2: Log likelihood $=-1500.3942$
Iteration 3: Log likelihood $=-1500.3794$
Iteration 4: Log likelihood $=-1500.3794$
Stoc. frontier normal/half-normal model
Log likelihood $=-1500.3794$

| Number of obs | $=$ | 756 |
| :--- | :--- | ---: |
| Wald chi2 (2) | $=$ | 9.68 |
| Prob $>$ chi2 | $=$ | 0.0079 |


| lnoutput | Coefficient | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| lnoutput <br> lnlabor | .7090933 | .2349374 | 3.02 | 0.003 | .2486244 | 1.169562 |
| lncapital |  |  |  |  |  |  |
| _cons | 1.252199 | 3.14656 | 0.40 | 0.691 | -4.914946 | 7.419344 |
| lnsig2v |  |  |  |  |  |  |
| size | -.0016951 | .0004748 | -3.57 | 0.000 | -.0026256 | -.0007645 |
| _cons | 3.156091 | .9265826 | 3.41 | 0.001 | 1.340023 | 4.97216 |
| lnsig2u |  |  |  |  |  |  |
| _cons | 1.947487 | .1017653 | 19.14 | 0.000 | 1.748031 | 2.146943 |
| sigma_u | 2.647838 | .134729 |  |  | 2.396514 | 2.925518 |

```
. test _b[lnlabor] + _b[lncapital] = 1
    ( 1) [lnoutput]lnlabor + [lnoutput]lncapital = 1
    chi2( 1) = 0.03
    Prob > chi2 = 0.8622
```

The output above indicates that the variance of the idiosyncratic error term is a function of firm size. Also, we failed to reject the hypothesis that the firms use a constant returns-to-scale technology.

## Technical note

In small samples, the conditional heteroskedastic estimators will lack precision for the variance parameters and may fail to converge altogether.
> Example 3: The truncated-normal model
Let's turn our attention to the truncated-normal model. Once again, we will use fictional data. For this example, we have 1,231 observations on the quantity of output, the total cost of production for each firm, the prices that each firm paid for labor and capital services, and a categorical variable measuring the quality of each firm's management. After taking the natural logarithm of the costs (lncost), prices (lnp_k and lnp_l), and output (lnout), we fit a stochastic cost frontier model and specify the distribution for the inefficiency term to be truncated normal.

```
. use https://www.stata-press.com/data/r18/frontier2
. frontier lncost lnp_k lnp_l lnout, distribution(tnormal) cost
Iteration 0: Log likelihood = -2386.9523
Iteration 1: Log likelihood = -2386.5146
Iteration 2: Log likelihood = -2386.2704
Iteration 3: Log likelihood = -2386.2504
Iteration 4: Log likelihood = -2386.2493
Iteration 5: Log likelihood = -2386.2493
```

Stoc. frontier normal/truncated-normal model
Log likelihood $=-2386.2493$

| Number of obs | $=1,231$ |
| ---: | :--- |
| Wald chi2 $(3)$ | $=8.82$ |
| Prob $>$ chi2 | $=0.0318$ |


| lncost | Coefficient | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | :---: | ---: |
| lnp_k | .3410717 | .2363861 | 1.44 | 0.149 | -.1222366 | .80438 |
| lnp_1 | .6608628 | .4951499 | 1.33 | 0.182 | -.3096131 | 1.631339 |
| lnout | .7528653 | .3468968 | 2.17 | 0.030 | .0729601 | 1.432771 |
| _cons | 2.602609 | 1.083004 | 2.40 | 0.016 | .4799595 | 4.725259 |
| /mu | 1.095705 | .881517 | 1.24 | 0.214 | -.632037 | 2.823446 |
| /lnsigma2 | 1.5534 | .1873464 | 8.29 | 0.000 | 1.186208 | 1.920592 |
| /lgtgamma | 1.257862 | .2589522 | 4.86 | 0.000 | .7503255 | 1.765399 |
| sigma2 | 4.727518 | .8856833 |  |  | 3.274641 | 6.825001 |
| gamma | .7786579 | .0446303 |  |  | .6792496 | .8538846 |
| sigma_u2 | 3.681119 | .7503408 |  |  | 2.210478 | 5.15176 |
| sigma_v2 | 1.046399 | .2660035 |  |  | .5250413 | 1.567756 |

HO: No inefficiency component: z = 5.595
Prob >= z = 0.0000

In addition to the coefficients, the output reports estimates for several parameters. sigma_v2 is the estimate of $\sigma_{v}^{2}$. sigma_u2 is the estimate of $\sigma_{u}^{2}$. gamma is the estimate of $\gamma=\sigma_{u}^{2} / \sigma_{S}^{2}$. sigma2 is the estimate of $\sigma_{S}^{2}=\sigma_{v}^{2}+\sigma_{u}^{2}$. Because $\gamma$ must be between 0 and 1 , the optimization is parameterized in terms of the logit of $\gamma$, and this estimate is reported as lgtgamma. Because $\sigma_{S}^{2}$ must be positive, the optimization is parameterized in terms of $\ln \left(\sigma_{S}^{2}\right)$, whose estimate is reported as lnsigma2. Finally, mu is the estimate of $\mu$, the mean of the truncated-normal distribution.

In the output above, the generalized log-likelihood test for the presence of the inefficiency term has been replaced with a test based on the third moment of the OLS residuals. When $\mu=0$ and $\sigma_{u}=0$, the truncated-normal model reduces to a linear regression model with normally distributed errors. However, the distribution of the test statistic under the null hypothesis is not well established, because it becomes impossible to evaluate the $\log$ likelihood as $\sigma_{u}$ approaches zero, prohibiting the use of the likelihood-ratio test.

However, Coelli (1995) noted that the presence of an inefficiency term would negatively skew the residuals from an OLS regression. By identifying negative skewness in the residuals with the presence of an inefficiency term, Coelli derived a one-sided test for the presence of the inefficiency term. The results of this test are given at the bottom of the output. For this example, the null hypothesis of no inefficiency component is rejected.

In the example below, we fit a truncated model and detect a statistically significant inefficiency term in the model. We might question whether the inefficiency term is identically distributed over all firms or whether there might be heterogeneity across firms. frontier provides an extension to the truncated normal model by allowing the mean of the inefficiency term to be modeled as a linear function of a set of covariates. In our dataset, we have a categorical variable that measures the quality of a firm's management. We refit the model, including the cm () option, specifying a set of
binary indicator variables representing the different categories of the quality-measurement variable as covariates.
. frontier lncost lnp_k lnp_l lnout, distribution(tnormal) cm(i.quality) cost
Iteration 0: Log likelihood $=-2386.9523$
Iteration 1: Log likelihood $=-2384.936$
Iteration 2: Log likelihood $=-2382.3942$
Iteration 3: Log likelihood $=-2382.324$
Iteration 4: Log likelihood $=-2382.3233$
Iteration 5: Log likelihood $=-2382.3233$
Stoc. frontier normal/truncated-normal model

Log likelihood $=-2382.3233$$\quad$ Number of obs $=1,231$

| lncost | Coefficient | Std. err. | z | $P>\|z\|$ | [95\% con | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Incost |  |  |  |  |  |  |
| lnp_k | . 3611204 | . 2359749 | 1.53 | 0.126 | -. 1013819 | . 8236227 |
| lnp_l | . 680446 | . 4934935 | 1.38 | 0.168 | -. 2867835 | 1.647675 |
| lnout | . 7605533 | . 3466102 | 2.19 | 0.028 | . 0812098 | 1.439897 |
| _cons | 2.550769 | 1.078911 | 2.36 | 0.018 | . 4361417 | 4.665396 |
| mu |  |  |  |  |  |  |
| quality |  |  |  |  |  |  |
| 2 | . 5056067 | . 3382907 | 1.49 | 0.135 | -. 1574309 | 1.168644 |
| 3 | . 783223 | . 376807 | 2.08 | 0.038 | . 0446947 | 1.521751 |
| 4 | . 5577511 | . 3355061 | 1.66 | 0.096 | -. 0998288 | 1.215331 |
| 5 | . 6792882 | . 3428073 | 1.98 | 0.048 | . 0073981 | 1.351178 |
| _cons | . 6014025 | . 990167 | 0.61 | 0.544 | -1.339289 | 2.542094 |
| /lnsigma2 | 1.541784 | . 1790926 | 8.61 | 0.000 | 1.190769 | 1.892799 |
| /lgtgamma | 1.242302 | . 2588968 | 4.80 | 0.000 | . 734874 | 1.749731 |
| sigma2 | 4.67292 | . 8368852 |  |  | 3.289611 | 6.637923 |
| gamma | . 7759645 | . 0450075 |  |  | . 6758739 | . 8519189 |
| sigma_u2 | 3.62602 | . 7139576 |  |  | 2.226689 | 5.025351 |
| sigma_v2 | 1.0469 | . 2583469 |  |  | . 5405491 | 1.553251 |

The conditional mean model was developed in the context of panel-data estimators, and we can apply frontier's conditional mean model to panel data.

## Stored results

frontier stores the following in e():
Scalars

```
e(N)
e(df_m)
e(k)
e(k_eq)
e(k_eq_model)
e(k_dv)
e(chi2)
e(1l)
e(ll_c)
e(z)
e(sigma_u)
e(sigma_v)
e(p)
e(chi2_c)
e(p_z)
e(rank)
e(ic)
e(rc)
e(converged)
```

Macros
e(cmd)
e(cmdline)
e(depvar)
$e$ (function)
e(wtype)
e(wexp)
e(title)
e(chi2type)
e(dist)
$e$ (het)
e(u_hetvar)
$e$ (v_hetvar)
e(vce)
e(vcetype)
e(opt)
$e$ (which)
e(ml_method)
e(user)
e(technique)
e(properties)
e(predict)
e(asbalanced)
e(asobserved)
Matrices
e(b)
e(Cns)
e(ilog)
e(gradient)
e(V)
e(V_modelbased)

Functions
e(sample)
number of observations
model degrees of freedom
number of parameters
number of equations in $e(b)$
number of equations in overall model test
number of dependent variables
$\chi^{2}$
log likelihood
log likelihood for $H_{0}: \sigma_{u}=0$
test for negative skewness of OLS residuals
standard deviation of technical inefficiency
standard deviation of $v_{i}$
$p$-value for model test
LR test statistic
$p$-value for z
rank of e(v)
number of iterations
return code
1 if converged, 0 otherwise
frontier
command as typed
name of dependent variable
production or cost
weight type
weight expression
title in estimation output
Wald; type of model $\chi^{2}$ test
distribution assumption for $u_{i}$
heteroskedastic components
varlist in uhet()
varlist in vhet()
vcetype specified in vce()
title used to label Std. err.
type of optimization
max or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement predict
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
constraints matrix
iteration $\log$ (up to 20 iterations)
gradient vector
variance-covariance matrix of the estimators
model-based variance
marks estimation sample

In addition to the above, the following is stored in r() :
Matrices
$r$ (table)
matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

Consider an equation of the form

$$
y_{i}=\mathbf{x}_{i} \boldsymbol{\beta}+v_{i}-s u_{i}
$$

where $y_{i}$ is the dependent variable, $\mathbf{x}_{i}$ is a $1 \times k$ vector of observations on the independent variables included as indent covariates, $\boldsymbol{\beta}$ is a $k \times 1$ vector of coefficients, and

$$
s=\left\{\begin{aligned}
1, & \text { for production functions } \\
-1, & \text { for cost functions }
\end{aligned}\right.
$$

The log-likelihood functions are as follows.
Normal/half-normal model:

$$
\ln L=\sum_{i=1}^{N}\left\{\frac{1}{2} \ln \left(\frac{2}{\pi}\right)-\ln \sigma_{S}+\ln \Phi\left(-\frac{s \epsilon_{i} \lambda}{\sigma_{S}}\right)-\frac{\epsilon_{i}^{2}}{2 \sigma_{S}^{2}}\right\}
$$

Normal/exponential model:

$$
\ln L=\sum_{i=1}^{N}\left\{-\ln \sigma_{u}+\frac{\sigma_{v}^{2}}{2 \sigma_{u}^{2}}+\ln \Phi\left(\frac{-s \epsilon_{i}-\frac{\sigma_{v}^{2}}{\sigma_{u}}}{\sigma_{v}}\right)+\frac{s \epsilon_{i}}{\sigma_{u}}\right\}
$$

Normal/truncated-normal model:

$$
\begin{aligned}
\ln L= & \sum_{i=1}^{N}\left\{-\frac{1}{2} \ln (2 \pi)-\ln \sigma_{S}-\ln \Phi\left(\frac{\mu}{\sigma_{S} \sqrt{\gamma}}\right)\right. \\
& \left.+\ln \Phi\left[\frac{(1-\gamma) \mu-s \gamma \epsilon_{i}}{\left\{\sigma_{S}^{2} \gamma(1-\gamma)\right\}^{1 / 2}}\right]-\frac{1}{2}\left(\frac{\epsilon_{i}+s \mu}{\sigma_{S}}\right)^{2}\right\}
\end{aligned}
$$

where $\sigma_{S}=\left(\sigma_{u}^{2}+\sigma_{v}^{2}\right)^{1 / 2}, \lambda=\sigma_{u} / \sigma_{v}, \gamma=\sigma_{u}^{2} / \sigma_{S}^{2}, \epsilon_{i}=y_{i}-\mathbf{x}_{i} \boldsymbol{\beta}$, and $\Phi()$ is the cumulative distribution function of the standard normal distribution.

To obtain estimation for $u_{i}$, you can use either the mean or the mode of the conditional distribution $f(u \mid \epsilon)$.

$$
\begin{aligned}
& E\left(u_{i} \mid \epsilon_{i}\right)=\mu_{* i}+\sigma_{*}\left\{\frac{\phi\left(-\mu_{* i} / \sigma_{*}\right)}{\Phi\left(\mu_{* i} / \sigma_{*}\right)}\right\} \\
& M\left(u_{i} \mid \epsilon_{i}\right)= \begin{cases}\mu_{* i} & \text { if } \mu_{* i} \geq 0 \\
0 & \text { otherwise }\end{cases}
\end{aligned}
$$

Then, the technical efficiency $(s=1)$ or cost efficiency $(s=-1)$ will be estimated by

$$
\begin{aligned}
E_{i} & =E\left\{\exp \left(-s u_{i}\right) \mid \epsilon_{i}\right\} \\
& =\left\{\frac{1-\Phi\left(s \sigma_{*}-\mu_{* i} / \sigma_{*}\right)}{1-\Phi\left(-\mu_{* i} / \sigma_{*}\right)}\right\} \exp \left(-s \mu_{* i}+\frac{1}{2} \sigma_{*}^{2}\right)
\end{aligned}
$$

where $\mu_{* i}$ and $\sigma_{*}$ are defined for the normal/half-normal model as

$$
\begin{aligned}
\mu_{* i} & =-s \epsilon_{i} \sigma_{u}^{2} / \sigma_{S}^{2} \\
\sigma_{*} & =\sigma_{u} \sigma_{v} / \sigma_{S}
\end{aligned}
$$

for the normal/exponential model as

$$
\begin{aligned}
\mu_{* i} & =-s \epsilon_{i}-\sigma_{v}^{2} / \sigma_{u} \\
\sigma_{*} & =\sigma_{v}
\end{aligned}
$$

and for the normal/truncated-normal model as

$$
\begin{aligned}
\mu_{* i} & =\frac{-s \epsilon_{i} \sigma_{u}^{2}+\mu \sigma_{v}^{2}}{\sigma_{S}^{2}} \\
\sigma_{*} & =\sigma_{u} \sigma_{v} / \sigma_{S}
\end{aligned}
$$

In the half-normal and exponential models, when heteroskedasticity is assumed, the standard deviations, $\sigma_{u}$ or $\sigma_{v}$, will be replaced in the above equations by

$$
\sigma_{i}^{2}=\exp \left(\mathbf{w}_{i} \boldsymbol{\delta}\right)
$$

where $\mathbf{w}$ is the vector of explanatory variables in the variance function.
In the conditional mean model, the mean parameter of the truncated normal distribution, $\mu$, is modeled as a linear combination of the set of covariates, $\mathbf{w}$.

$$
\mu=\mathbf{w}_{i} \boldsymbol{\delta}
$$

Therefore, the log-likelihood function can be rewritten as

$$
\begin{aligned}
\ln L= & \sum_{i=1}^{N}\left[-\frac{1}{2} \ln (2 \pi)-\ln \sigma_{S}-\ln \Phi\left(\frac{\mathbf{w}_{i} \boldsymbol{\delta}}{\sqrt{\sigma_{S}^{2} \gamma}}\right)\right. \\
& \left.+\ln \Phi\left\{\frac{(1-\gamma) \mathbf{w}_{i} \boldsymbol{\delta}-s \gamma \epsilon_{i}}{\sqrt{\sigma_{S}^{2} \gamma(1-\gamma)}}\right\}-\frac{1}{2}\left(\frac{\epsilon_{i}+s \mathbf{w}_{i} \boldsymbol{\delta}}{\sigma_{S}}\right)^{2}\right]
\end{aligned}
$$

The $z$ test reported in the output of the truncated-normal model is a third-moment test developed by Coelli (1995) as an extension of a test previously developed by Pagan and Hall (1983). Coelli shows that under the null of normally distributed errors, the statistic

$$
z=\frac{m_{3}}{\left(\frac{6 m_{2}^{3}}{N}\right)^{1 / 2}}
$$

has a standard normal distribution, where $m_{3}$ is the third moment from the OLS regression. Because the residuals are either negatively skewed (production function) or positively skewed (cost function), a one-sided $p$-value is used.

## References

Aigner, D. J., C. A. K. Lovell, and P. Schmidt. 1977. Formulation and estimation of stochastic frontier production function models. Journal of Econometrics 6: 21-37. https://doi.org/10.1016/0304-4076(77)90052-5.
Badunenko, O., and P. Mozharovskyi. 2016. Nonparametric frontier analysis using Stata. Stata Journal 16: 550-589.
Badunenko, O., and H. Tauchmann. 2019. Simar and Wilson two-stage efficiency analysis for Stata. Stata Journal 19: 950-988.
Belotti, F., S. Daidone, G. Ilardi, and V. Atella. 2013. Stochastic frontier analysis using Stata. Stata Journal 13: 719-758.

Caudill, S. B., J. M. Ford, and D. M. Gropper. 1995. Frontier estimation and firm-specific inefficiency measures in the presence of heteroscedasticity. Journal of Business and Economic Statistics 13: 105-111. https://doi.org/10.2307/1392525.
Cococcioni, M., M. Grazzi, L. Li, and F. Ponchio. 2022. A toolbox for measuring heterogeneity and efficiency using zonotopes. Stata Journal 22: 25-59.

Coelli, T. J. 1995. Estimators and hypothesis tests for a stochastic frontier function: A Monte Carlo analysis. Journal of Productivity Analysis 6: 247-268. https://doi.org/10.1007/BF01076978.
Fé, E., and R. Hofler. 2020. sfcount: Command for count-data stochastic frontiers and underreported and overreported counts. Stata Journal 20: 532-547.

Greene, W. H. 2003. Econometric Analysis. 5th ed. Upper Saddle River, NJ: Prentice Hall.
Gutierrez, R. G., S. L. Carter, and D. M. Drukker. 2001. sg160: On boundary-value likelihood-ratio tests. Stata Technical Bulletin 60: 15-18. Reprinted in Stata Technical Bulletin Reprints, vol. 10, pp. 269-273. College Station, TX: Stata Press.

Hu, Y., G. Huang, and Y. Sasaki. 2023. robustpf: A command for robust estimation of production functions. Stata Journal 23: 86-96.
Karakaplan, M. U. 2017. Fitting endogenous stochastic frontier models in Stata. Stata Journal 17: 39-55.
Kumbhakar, S. C., and C. A. K. Lovell. 2000. Stochastic Frontier Analysis. Cambridge: Cambridge University Press.
Kumbhakar, S. C., H.-J. Wang, and A. P. Horncastle. 2015. A Practitioner's Guide to Stochastic Frontier Analysis Using Stata. New York: Cambridge University Press.
Lian, Y., C. Liu, and C. F. Parmeter. 2023. Two-tier stochastic frontier analysis using Stata. Stata Journal 23: 197-229.

Meeusen, W., and J. van den Broeck. 1977. Efficiency estimation from Cobb-Douglas production functions with composed error. International Economic Review 18: 435-444. https://doi.org/10.2307/2525757.

Pagan, A. R., and A. D. Hall. 1983. Diagnostic tests as residual analysis. Econometric Reviews 2: 159-218. https://doi.org/10.1080/07311768308800039.
Petrin, A. K., B. P. Poi, and J. A. Levinsohn. 2004. Production function estimation in Stata using inputs to control for unobservables. Stata Journal 4: 113-123.

Pitblado, J. S., B. P. Poi, and W. W. Gould. 2024. Maximum Likelihood Estimation with Stata. 5th ed. College Station, TX: Stata Press.
Stevenson, R. E. 1980. Likelihood functions for generalized stochastic frontier estimation. Journal of Econometrics 13: 57-66. https://doi.org/10.1016/0304-4076(80)90042-1.

Tauchmann, H. 2012. Partial frontier efficiency analysis. Stata Journal 12: 461-478.
Wang, D., K. Du, and N. Zhang. 2022. Measuring technical efficiency and total factor productivity change with undesirable outputs in Stata. Stata Journal 22: 103-124.

Zellner, A., and N. S. Revankar. 1969. Generalized production functions. Review of Economic Studies 36: 241-250. https://doi.org/10.2307/2296840.

## Also see

[R] frontier postestimation - Postestimation tools for frontier
$[\mathrm{R}]$ regress - Linear regression
[XT] xtfrontier - Stochastic frontier models for panel data
[U] 20 Estimation and postestimation commands

## Title

frontier postestimation - Postestimation tools for frontier

| Postestimation commands | predict margins Remarks and examples |
| :--- | :--- |
| Reference | Also see |

## Postestimation commands

The following postestimation commands are available after frontier:

| Command | Description |
| :---: | :---: |
| contrast | contrasts and ANOVA-style joint tests of estimates |
| estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estimates | cataloging estimation results |
| etable | table of estimation results |
| hausman | Hausman's specification test |
| lincom | point estimates, standard errors, testing, and inference for linear combinations of coefficients |
| linktest | link test for model specification |
| lrtest | likelihood-ratio test |
| margins | marginal means, predictive margins, marginal effects, and average marginal effects |
| marginsplot | graph the results from margins (profile plots, interaction plots, etc.) |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients |
| predict | predictions and their SEs, technical efficiency, etc. |
| predictnl | point estimates, standard errors, testing, and inference for generalized predictions |
| pwcompare | pairwise comparisons of estimates |
| test | Wald tests of simple and composite linear hypotheses |
| testnl | Wald tests of nonlinear hypotheses |

## predict

## Description for predict

predict creates a new variable containing predictions such as linear predictions, standard errors, and estimates of technical efficiency.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

```
predict [type] newvar [if] [in] [, statistic]
predict [type]stub* [if] [in], scores
```

statistic Description

## Main

$\mathrm{xb} \quad$ linear prediction; the default
stdp standard error of the prediction
$\mathrm{u} \quad$ estimates of minus the natural $\log$ of the technical efficiency via $E\left(u_{i} \mid \epsilon_{i}\right)$
m
te estimates of minus the natural $\log$ of the technical efficiency via $M\left(u_{i} \mid \epsilon_{i}\right)$ estimates of the technical efficiency via $E\left\{\exp \left(-s u_{i}\right) \mid \epsilon_{i}\right\}$
$s=\left\{\begin{aligned} 1, & \text { for production functions } \\ -1, & \text { for cost functions }\end{aligned}\right.$
These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

## Options for predict

Main
xb , the default, calculates the linear prediction.
stdp calculates the standard error of the linear prediction.
u produces estimates of minus the natural $\log$ of the technical efficiency via $E\left(u_{i} \mid \epsilon_{i}\right)$.
m produces estimates of minus the natural $\log$ of the technical efficiency via $M\left(u_{i} \mid \epsilon_{i}\right)$. te produces estimates of the technical efficiency via $E\left\{\exp \left(-s u_{i}\right) \mid \epsilon_{i}\right\}$.
scores calculates equation-level score variables.
The first new variable will contain $\partial \ln L / \partial\left(\mathbf{x}_{i} \boldsymbol{\beta}\right)$.
The second new variable will contain $\partial \ln L / \partial(\ln s i g 2 v)$.
The third new variable will contain $\partial \ln L / \partial(\operatorname{lnsig} 2 \mathrm{u})$.
scores may not be specified after estimation with option distribution(tnormal).

## margins

## Description for margins

margins estimates margins of response for linear predictions.

## Menu for margins

Statistics > Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [options]
```

statistic
Description
xb
linear prediction; the default
stdp not allowed with margins
u not allowed with margins
m not allowed with margins
te not allowed with margins

Statistics not allowed with margins are functions of stochastic quantities other than e(b). For the full syntax, see [R] margins.

## Remarks and examples

> Example 1
In example 2 of $[\mathrm{R}]$ frontier, we modeled heteroskedasticity by specifying the vhet () option. We would like to compare the predicted efficiency in that case with respect to a model specification without accounting for the presence of heteroskedasticity in the error term. Kumbhakar and Lovell (2000, 117) show that failing to account for heteroskedasticity associated with firm size may lead to bias in the estimation of the technical efficiency. By incorrectly assuming homoskedasticity, the estimates for relatively small firms would be biased upward, while the estimates for relatively large firms would be biased downward. Let's refit the model and use the te option of predict:

```
. use https://www.stata-press.com/data/r18/frontier1
. frontier lnoutput lnlabor lncapital, vhet(size)
    (output omitted)
. predict te_vhet, te
```

Next, we fit the model assuming homoskedasticity and then again predict the technical efficiency with the te option of predict:

```
. frontier lnoutput lnlabor lncapital
    (output omitted)
    predict te, te
```

The graph below shows the estimates for technical efficiency for the smaller and larger firms. The technical efficiency tends to be smaller for smaller firms when the model specification accounts for the presence of heteroskedasticity, whereas the predictions for the technical efficiency tends to be smaller for larger firms assuming homoskedasticity. These results agree with the theoretical statement in Kumbhakar and Lovell (2000) because the firm size was actually relevant to model heteroskedasticity in the idiosyncratic component of the error term.

Predicted technical efficiency for smaller and larger firms


## > Example 2

We also test in example 2 of $[R]$ frontier whether the firms use constant returns to scale. We can use lincom as an alternative to perform an equivalent test based on the normal distribution.

```
. use https://www.stata-press.com/data/r18/frontier1, clear
. frontier lnoutput lnlabor lncapital, vhet(size)
    (output omitted)
. lincom _b[lnlabor] + _b[lncapital]-1
( 1) [lnoutput]lnlabor + [lnoutput]lncapital = 1
```

| lnoutput | Coefficient | Std. err. | $z$ | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | :---: | ---: | ---: | ---: |
| $(1)$ | .1022278 | .5888511 | 0.17 | 0.862 | -1.051899 | 1.256355 |

The $p$-value is exactly the same as the one we obtained with the test command in example 2 of $[R]$ frontier. However, notice that by using lincom, we obtained an estimate of the deviation from the constant returns-to-scale assumption, which is not significantly different from zero in this case.

## Reference

Kumbhakar, S. C., and C. A. K. Lovell. 2000. Stochastic Frontier Analysis. Cambridge: Cambridge University Press.

## Also see

[R] frontier - Stochastic frontier models
[U] 20 Estimation and postestimation commands

## Title

fvrevar - Factor-variables operator programming command

| Description | Quick start | Syntax | Options |
| :--- | :--- | :--- | :--- |
| Remarks and examples | Stored results | Also see |  |

## Description

fvrevar creates a variable list that includes equivalent, temporary variables in place of the factor variables, interactions, or time-series-operated variables in varlist. The resulting variable list can be used by commands that do not otherwise support factor variables or time-series-operated variables. The resulting list also could be used in a program to speed execution at the cost of using more memory.

## Quick start

Create temporary indicator variables for the levels of categorical variable a and store names in r(varlist)
fvrevar i.a
Create temporary variables corresponding to the levels of $a, b$, and their interaction
fvrevar i.a\#\#i.b
Same as above, and create a temporary variable for the lag of x using tsset data fvrevar i.a\#\#i.b L.x

Return the list of unoperated variables ( $\mathrm{a}, \mathrm{b}$, and x ) in r (varlist)
fvrevar i.a\#\#i.b L.x, list
Create new variables $\mathrm{a} \_1, \mathrm{a} \_2, \ldots$, corresponding to the levels of a
fvrevar i.a, stub(a_)
Create new variables $a b \_1, a b \_2, \ldots$, corresponding to the levels of the interaction between $a \operatorname{and} b$ fvrevar i.a\#i.b, stub(ab_)

## Syntax

```
fvrevar [varlist][if][in] [, substitute tsonly list stub(stub)]
```

You must tsset your data before using fvrevar if varlist contains time-series operators; see [TS] tsset. collect is allowed; see [U] 11.1.10 Prefix commands.

## Options

substitute specifies that equivalent, temporary variables be substituted for any factor variables, interactions, or time-series-operated variables in varlist. substitute is the default action taken by fvrevar; you do not need to specify the option.
tsonly specifies that equivalent, temporary variables be substituted for only the time-series-operated variables in varlist.
list specifies that all factor-variable operators and time-series operators be removed from varlist and the resulting list of base variables be returned in $r$ (varlist). No new variables are created with this option.
stub (stub) specifies that fvrevar generate named variables instead of temporary variables. The new variables will be named stub\#.

## Remarks and examples

fvrevar might create no new variables, one new variable, or many new variables, depending on the number of factor variables, interactions, and time-series operators appearing in varlist. Any new variables created are temporary. The new, equivalent varlist is returned in $r$ (varlist). The new varlist corresponds one to one with the original varlist.
$>$ Example 1
Typing

```
. use https://www.stata-press.com/data/r18/auto2
. fvrevar i.rep78 mpg turn
```

creates five temporary variables corresponding to the levels of rep78. No new variables are created for variables mpg and turn because they do not contain factor-variable or time-series operators.

The resulting variable list is

```
. display "'r(varlist)'"
__000000 __000001 __000002 __000003 __000004 mpg turn
```

(Your temporary variable names may be different, but that is of no consequence.)
Temporary variables automatically vanish when the program concludes.

## > Example 2

Suppose we want to create temporary variables for specific levels of a factor variable. To do this, we can use the parenthesis notation of factor-variable syntax.

```
. fvrevar i (2,3)bn.rep78 mpg
```

creates two temporary variables corresponding to levels 2 and 3 of rep78. Notice that we specified that neither level 2 nor 3 be set as the base level by using the bn notation. If we did not specify bn, level 2 would have been treated as the base level.

The resulting variable list is

```
. display "'r(varlist)'"
__000005 __000002 mpg
```

We can see the results by listing the new variables alongside the original value of rep78.

```
. list rep78 'r(varlist)' in 1/5
```

| rep78 | _-000005 | _-000002 | mpg |
| ---: | ---: | ---: | ---: |
| Average | 0 | 1 | 22 |
| Average | 0 | 1 | 17 |
| $\cdot$ | . | . | 22 |
| Average | 0 | 1 | 20 |
| Good | 0 | 0 | 15 |

If we had needed only the base-variable names, we could have specified

```
. fvrevar i(2,3)bn.rep78 mpg, list
. display "'r(varlist)'"
mpg rep78
```

The order of the list will probably differ from that of the original list; base variables are listed only once.

## > Example 3

Now let's assume we have a varlist containing both an interaction and time-series-operated variables. If we want to create temporary variables for the entire equivalent varlist, we can specify fvrevar with no options.

```
. generate t = _n
. tsset t
    time variable: t, 1 to 74
        delta: 1 unit
. fvrevar c.turn#i(2,3).rep78 L.mpg
```

The resulting variable list is

```
. display "'r(varlist)'"
__000006 __000007 __000008
```

If we want to create temporary variables only for the time-series-operated variables, we can specify the tsonly option.

```
. fvrevar c.turn#i(2,3).rep78 L.mpg, tsonly
```

The resulting variable list is
. display "'r(varlist)'"
2.rep78\#c.turn 3.rep78\#c.turn __000008

Notice that fvrevar returned the expanded factor-variable list with the tsonly option.

- Technical note
fvrevar, substitute avoids creating duplicate variables. Consider
. fvrevar i.rep78 turn mpg i.rep78
i.rep78 appears twice in the varlist. fvrevar will create only one set of new variables for the five levels of rep78 and will use these new variables once in the resulting r(varlist). Moreover, fvrevar will do this even across multiple calls:
. fvrevar i.rep78 turn mpg
. fvrevar i.rep78
i.rep78 appears in two separate calls. At the first call, fvrevar creates five temporary variables corresponding to the five levels of rep78. At the second call, fvrevar remembers what it has done and uses the same temporary variables for i.rep78.


## Stored results

fvrevar stores the following in $r()$ :
Macros
$r$ (varlist) the modified variable list or list of base-variable names

## Also see

[TS] tsrevar - Time-series operator programming command
[ P ] fvexpand - Expand factor varlists
[P] syntax - Parse Stata syntax
[P] unab - Unabbreviate variable list
[U] 11 Language syntax
[U] 11.4.4 Time-series varlists
[U] 18 Programming Stata

## Title

fvset - Declare factor-variable settings

Description Quick start Syntax Options Remarks and examples
Stored results

## Description

fvset base, fvset design, and fvset clear manage factor-variable settings, which identify the base level and specify how to accumulate statistics over levels. fvset base declares the base level for each specified variable; the default for factor variables without a declared base level is the lowest value. fvset design specifies how the margins command is to accumulate over the levels of a factor variable. fvset clear removes factor-variable settings for each variable in varlist. fvset clear _all removes all factor-variable settings from all variables.
fvset report reports the current factor-variable settings for each variable in varlist. fvset without arguments is a synonym for fvset report.

## Quick start

Set the base category of categorical variable a1 to 3
fvset base 3 a1
Set the base category of a2, a3, and a4 to each variable's largest observed value
fvset base last a2 a3 a4
Set the base category of a5 to the most frequent category
fvset base frequent a5
Set a6 to have no base category
fvset base none a6
Restore the default base category (first) for a5
fvset base default a5
Specify that margins should treat a2 as though it is balanced
fvset design asbalanced a2
Clear factor-variable settings for a2 to a4
fvset clear a2-a4
List factor-variable settings for all factor variables
fvset report

## Syntax

Declare base settings
fvset base base_spec varlist

Declare design settings
fvset design design_spec varlist

Clear the current settings
fvset clear varlist

Report the current settings
fvset report [varlist] [, base(base_spec) design(design_spec)]

| base_spec | Description |
| :--- | :--- |
| default | default base |
| first | lowest level value; the default |
| last | highest level value |
| frequent | most frequent level value |
| none | no base |
| $\#$ | nonnegative integer value |

design_spec Description
default default design
asbalanced accumulate using $1 / k, k=$ number of levels
asobserved accumulate using observed relative frequencies; the default
collect is allowed; see [U] 11.1.10 Prefix commands.

## Options

base (base_spec) restricts fvset report to report only the factor-variable settings for variables with the specified base_spec.
design (design_spec) restricts fvset report to report only the factor-variable settings for variables with the specified design_spec.

## Remarks and examples

## - Example 1

Using auto2.dta, we include factor variable i.rep78 in a regression:

```
. use https://www.stata-press.com/data/r18/auto2
(1978 automobile data)
. regress mpg i.rep78, baselevels
```

| Source | SS | df | MS |
| ---: | :---: | ---: | :---: |
| Model | 549.415777 | 4 | 137.353944 |
| Residual | 1790.78712 | 64 | 27.9810488 |
| Total | 2340.2029 | 68 | 34.4147485 |


| Number of obs | $=$ | 69 |
| :--- | :--- | ---: |
| $\mathrm{~F}(4,64)$ | $=$ | 4.91 |
| Prob $>$ F | $=$ | 0.0016 |
| R-squared | $=$ | 0.2348 |
| Adj R-squared | $=$ | 0.1869 |
| Root MSE | $=$ | 5.2897 |


| mpg | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :--- | :---: | :---: | :---: | :---: |
| rep78 |  |  |  |  |  |  |
| Poor | 0 | (base) |  |  |  |  |
| Fair | -1.875 | 4.181884 | -0.45 | 0.655 | -10.22927 | 6.479274 |
| Average | -1.566667 | 3.863059 | -0.41 | 0.686 | -9.284014 | 6.150681 |
| Good | .6666667 | 3.942718 | 0.17 | 0.866 | -7.209818 | 8.543152 |
| Excellent | 6.363636 | 4.066234 | 1.56 | 0.123 | -1.759599 | 14.48687 |
|  |  | 21 | 3.740391 | 5.61 | 0.000 | 13.52771 |
| _cons |  |  |  | 28.47229 |  |  |

We specified the baselevels option so that the base level would be included in the output. By default, the first level is the base level. We can change the base level to 2 :
. fvset base 2 rep78
. regress mpg i.rep78, baselevels

| Source | SS | df | MS |
| ---: | :---: | ---: | :---: |
| Model | 549.415777 | 4 | 137.353944 |
| Residual | 1790.78712 | 64 | 27.9810488 |
| Total | 2340.2029 | 68 | 34.4147485 |


| Number of obs | $=$ | 69 |
| :--- | :--- | ---: |
| F(4, 64) | $=$ | 4.91 |
| Prob > F | $=$ | 0.0016 |
| R-squared | $=$ | 0.2348 |
| Adj R-squared | $=$ | 0.1869 |
| Root MSE | $=$ | 5.2897 |


| mpg | Coefficient | Std. err. | t | $P>\|t\|$ | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| rep78 |  |  |  |  |  |  |
| Poor | 1.875 | 4.181884 | 0.45 | 0.655 | -6.479274 | 10.22927 |
| Fair | 0 | (base) |  |  |  |  |
| Average | . 3083333 | 2.104836 | 0.15 | 0.884 | -3.896559 | 4.513226 |
| Good | 2.541667 | 2.247695 | 1.13 | 0.262 | -1.948621 | 7.031954 |
| Excellent | 8.238636 | 2.457918 | 3.35 | 0.001 | 3.32838 | 13.14889 |
| _cons | 19.125 | 1.870195 | 10.23 | 0.000 | 15.38886 | 22.86114 |

Let's set rep78 to have no base level and fit a cell-means regression:
. fvset base none rep78
. regress mpg i.rep78, noconstant

| Source | SS | df | MS | Number of obs | = | 69 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | F (5, 64) | = | 227.47 |
| Model | 31824.2129 | 5 | 6364.84258 | Prob > F | = | 0.0000 |
| Residual | 1790.78712 | 64 | 27.9810488 | R -squared | = | 0.9467 |
|  |  |  |  | Adj R-squared | = | 0.9426 |
| Total | 33615 | 69 | 487.173913 | Root MSE | = | 5.2897 |


| mpg | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :--- | ---: | ---: | ---: | ---: |
| rep78 |  |  |  |  |  |  |
| Poor | 21 | 3.740391 | 5.61 | 0.000 | 13.52771 | 28.47229 |
| Fair | 19.125 | 1.870195 | 10.23 | 0.000 | 15.38886 | 22.86114 |
| Average | 19.43333 | .9657648 | 20.12 | 0.000 | 17.504 | 21.36267 |
| Good | 21.66667 | 1.246797 | 17.38 | 0.000 | 19.1759 | 24.15743 |
| Excellent | 27.36364 | 1.594908 | 17.16 | 0.000 | 24.17744 | 30.54983 |

## Example 2

By default, margins assumes that factor variables are to be treated asobserved and accumulates a margin by using the observed relative frequencies of the factor levels or the sum of the weights if weights have been specified.

| Source | SS | df | MS | Number of obs F (1, 72) |  | = | 74 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Model | 378.153515 | 1 | $378.153515$ |  | Prob > F |  | 0.0005 |
| Residual | 2065.30594 | 72 | 28.6848048 | $R$-squared <br> Adj R-squared |  |  | 0.1548 |
|  |  |  |  |  |  |  | 0.1430 |
| Total | 2443.45946 | 73 | 33.4720474 | Root MSE |  | = | 5.3558 |
| mpg | Coefficient | Std. err. | t | $P>\|t\|$ | tl [95\% con |  | interval] |
| foreign |  |  |  |  |  |  |  |
| Foreign | 4.945804 | 1.362162 | 3.63 | 0.001 | 012.230384 |  | 7.661225 |
| _cons | 19.82692 | . 7427186 | 26.70 | 0.000 | 0018.34634 |  | 21.30751 |

. margins
Predictive margins $\quad$ Number of obs $=74$
Model VCE: OLS
Expression: Linear prediction, predict()

|  | Delta-method |  |  |  |  |  |
| :--- | ---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | Margin | std. err. | t | $\mathrm{P}>\mid \mathrm{tI}$ | [95\% conf. interval] |  |
| _cons | 21.2973 | .6226014 | 34.21 | 0.000 | 20.05616 | 22.53843 |

Let's set foreign to always accumulate using equal relative frequencies:

```
. fvset design asbalanced foreign
```

. regress mpg i.foreign

| Source | SS | df | MS | Number of obs | = | 74 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | $\mathrm{F}(1,72)$ | = | 13.18 |
| Model | 378.153515 | 1 | 378.153515 | Prob > F | = | 0.0005 |
| Residual | 2065.30594 | 72 | 28.6848048 | R -squared | = | 0.1548 |
|  |  |  |  | Adj R-squared | = | 0.1430 |
| Total | 2443.45946 | 73 | 33.4720474 | Root MSE | = | 5.3558 |


| mpg | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :--- | ---: | :--- | ---: | :--- |
| foreign |  |  |  |  |  |  |
| Foreign | 4.945804 | 1.362162 | 3.63 | 0.001 | 2.230384 | 7.661225 |
| _cons | 19.82692 | .7427186 | 26.70 | 0.000 | 18.34634 | 21.30751 |

. margins
Adjusted predictions $\quad$ Number of obs $=74$
Model VCE: OLS
Expression: Linear prediction, predict()
At: foreign (asbalanced)

|  | Delta-method |  |  |  |  |  |
| :--- | ---: | :---: | :---: | :---: | :---: | :---: |
|  | Margin | std. err. | $t$ | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| _cons | 22.29983 | .6810811 | 32.74 | 0.000 | 20.94211 | 23.65754 |

Suppose that we issued the fvset design command earlier in our session and that we cannot remember which variables we set as asbalanced. We can retrieve this information by using the fvset report command:

- fvset report, design(asbalanced)

| Variable | Base | Design |
| :--- | :--- | :--- |
| foreign |  | asbalanced |

## - Technical note

margins is aware of a factor variable's design setting only through the estimation results it is working with. The design setting is stored by the estimation command; thus changing the design setting between the estimation command and margins will have no effect. For example, the output from the following two calls to margins yields the same results:


|  | Delta-method |  |  |  |  |  |
| ---: | ---: | :---: | :---: | :---: | :---: | :---: |
|  | Margin | std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| _cons | 21.2973 | .6226014 | 34.21 | 0.000 | 20.05616 | 22.53843 |

```
. fvset design asbalanced foreign
. margins
Predictive margins Number of obs = 74
Model VCE: OLS
Expression: Linear prediction, predict()
```

|  | Delta-method |  |  |  |  |  |
| :--- | ---: | :---: | :---: | :---: | :---: | :---: |
|  | Margin | std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| _cons | 21.2973 | .6226014 | 34.21 | 0.000 | 20.05616 | 22.53843 |

## Stored results

fvset stores the following in $r()$ :
Macros

```
r(varlist) varlist
    r(baselist) base setting for each variable in varlist
    r(designlist) design setting for each variable in varlist
```


## Title

gllamm - Generalized linear and latent mixed models
Description Remarks and examples References Also see

## Description

GLLAMM stands for generalized linear latent and mixed models, and gllamm is a Stata command for fitting such models written by Sophia Rabe-Hesketh (University of California-Berkeley) as part of joint work with Anders Skrondal (Norwegian Institute of Public Health) and Andrew Pickles (King's College London).

## Remarks and examples

Generalized linear latent and mixed models are a class of multilevel latent variable models, where a latent variable is a factor or a random effect (intercept or coefficient), or a disturbance (residual). The gllamm command for fitting such models is not an official command of Stata; it has been independently developed by highly regarded authors and is itself highly regarded. You can learn more about gllamm by visiting http://www.gllamm.org.
gllamm is available from the Statistical Software Components (SSC) Archive. To install, type
. ssc describe gllamm
. ssc install gllamm
If you later wish to uninstall gllamm, type ado uninstall gllamm.

## References

Miranda, A., and S. Rabe-Hesketh. 2006. Maximum likelihood estimation of endogenous switching and sample selection models for binary, ordinal, and count variables. Stata Journal 6: 285-308.
Rabe-Hesketh, S., A. Pickles, and C. Taylor. 2000. sg129: Generalized linear latent and mixed models. Stata Technical Bulletin 53: 47-57. Reprinted in Stata Technical Bulletin Reprints, vol. 9, pp. 293-307. College Station, TX: Stata Press.

Rabe-Hesketh, S., and A. Skrondal. 2022. Multilevel and Longitudinal Modeling Using Stata. 4th ed. College Station, TX: Stata Press.

Rabe-Hesketh, S., A. Skrondal, and A. Pickles. 2002. Reliable estimation of generalized linear mixed models using adaptive quadrature. Stata Journal 2: 1-21.
——. 2003. Maximum likelihood estimation of generalized linear models with covariate measurement error. Stata Journal 3: 386-411.

Skrondal, A., and S. Rabe-Hesketh. 2004. Generalized Latent Variable Modeling: Multilevel, Longitudinal, and Structural Equation Models. Boca Raton, FL: Chapman and Hall/CRC.

Zheng, X., and S. Rabe-Hesketh. 2007. Estimating parameters of dichotomous and ordinal item response models with gllamm. Stata Journal 7: 313-333.

The references above are restricted to works by the primary authors of gllamm. There are many other books and articles that use or discuss gllamm; see http://www.gllamm.org/pub.html for a list.

## Also see

[ME] meglm - Multilevel mixed-effects generalized linear models
[ME] mixed - Multilevel mixed-effects linear regression
[SEM] Intro 2 - Learning the language: Path diagrams and command language
[SEM] Intro 5 - Tour of models

## Title

glm - Generalized linear models

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| Acknowledgments | References | Also see |  |

## Description

glm fits generalized linear models. It can fit models by using either IRLS (maximum quasilikelihood) or Newton-Raphson (maximum likelihood) optimization, which is the default.

See [U] 27 Overview of Stata estimation commands for a description of all of Stata's estimation commands, several of which fit models that can also be fit using glm.

## Quick start

Model of $y$ as a function of $x$ when $y$ is a proportion glm y x, family(binomial)

Logit model of $y$ events occurring in 15 trials as a function of $x$
glm y x, family (binomial 15) link(logit)
Probit model of $y$ events as a function of $x$ using grouped data with group sizes $n$
glm y x, family (binomial n) link(probit)
Model of discrete $y$ with user-defined family myfamily and link mylink
glm y x, family (myfamily) link(mylink)
Bootstrap standard errors in a model of $y$ as a function of $x$ with a gamma family and log link glm y x, family (gamma) link(log) vce(bootstrap)

## Menu

Statistics $>$ Generalized linear models $>$ Generalized linear models (GLM)

## Syntax

glm depvar [indepvars] [if] [in] [weight] [, options]
options Description

## Model

family (familyname) distribution of depvar; default is family (gaussian)
link(linkname) link function; default is canonical link for family() specified
Model 2
noconstant
exposure (varname)
offset (varname)
constraints (constraints)
asis
mu(varname)
init (varname)
suppress constant term
include $\ln$ (varname) in model with coefficient constrained to 1 include varname in model with coefficient constrained to 1 apply specified linear constraints
retain perfect predictor variables
use varname as the initial estimate for the mean of depvar synonym for mu (varname)

SE/Robust
vce (vcetype)
vfactor(\#)
disp(\#)
scale(x2|dev|\#)
vcetype may be oim, robust, cluster clustvar, eim, opg,
bootstrap, jackknife, hac kernel, jackknife1, or unbiased
multiply variance matrix by scalar \#
quasilikelihood multiplier
set the scale parameter
Reporting
level(\#)
eform
nocnsreport
display_options
set confidence level; default is level (95)
report exponentiated coefficients
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

Maximization
ml
irls
maximize_options
fisher (\#)
search
noheader
notable
nodisplay
collinear
coeflegend
use maximum likelihood optimization; the default use iterated, reweighted least-squares optimization of the deviance control the maximization process; seldom used use the Fisher scoring Hessian or expected information matrix (EIM) search for good starting values
suppress header table from above coefficient table suppress coefficient table
suppress the output; iteration $\log$ is still displayed
keep collinear variables
display legend instead of statistics

## familyname

Description

## gaussian

igaussian
binomial $\left[\right.$ varname $_{N} \mid \#_{N}$ ]
poisson
nbinomial $\left[\#_{k} \mid \mathrm{ml}\right]$
gamma

Gaussian (normal) inverse Gaussian Bernoulli/binomial Poisson negative binomial gamma
linkname
identity
log
logit
probit
cloglog
power \#
opower \#
nbinomial
loglog
$\log \mathrm{c}$

## Description

## identity

$\log$
logit
probit
cloglog
power
odds power
negative binomial
$\log -\log$
log-complement
indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
depvar and indepvars may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bayes, bootstrap, by, collect, fmm, fp, jackknife, mfp, mi estimate, nestreg, rolling, statsby, stepwise, and svy are allowed; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: glm and [FMM] fmm: glm.
vce(bootstrap), vce(jackknife), and vce(jackknife1) are not allowed with the mi estimate prefix; see [MI] mi estimate.
Weights are not allowed with the bootstrap prefix; see [R] bootstrap.
aweights are not allowed with the jackknife prefix; see $[R]$ jackknife.
vce(), vfactor(), disp(), scale(), irls, fisher(), noheader, notable, nodisplay, and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, aweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
noheader, notable, nodisplay, collinear, and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

$\qquad$ Model
family (familyname) specifies the distribution of depvar; family (gaussian) is the default. link (linkname) specifies the link function; the default is the canonical link for the family() specified (except for family (nbinomial)).

Model 2
noconstant, exposure(varname), offset(varname), constraints(constraints); see [R] Estimation options. constraints (constraints) is not allowed with irls.
asis forces retention of perfect predictor variables and their associated, perfectly predicted observations and may produce instabilities in maximization; see $[\mathrm{R}]$ probit. This option is allowed only with option family (binomial) with a denominator of 1.
mu(varname) specifies varname as the initial estimate for the mean of depvar. This option can be useful with models that experience convergence difficulties, such as family(binomial) models with power or odds-power links. init (varname) is a synonym.

## SE/Robust

vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
In addition to the standard vcetypes, glm allows the following alternatives:
vce (eim) specifies that the EIM estimate of variance be used.
vce (jackknife1) specifies that the one-step jackknife estimate of variance be used.
vce (hac kernel [\#]) specifies that a heteroskedasticity- and autocorrelation-consistent (HAC) variance estimate be used. HAC refers to the general form for combining weighted matrices to form the variance estimate. There are three kernels built into glm. kernel is a user-written program or one of

```
nwest | gallant| anderson
```

\# specifies the number of lags. If \# is not specified, $N-2$ is assumed. If you wish to specify vce(hac...), you must tsset your data before calling glm.
vce (unbiased) specifies that the unbiased sandwich estimate of variance be used.
vfactor (\#) specifies a scalar by which to multiply the resulting variance matrix. This option allows you to match output with other packages, which may apply degrees of freedom or other small-sample corrections to estimates of variance.
disp(\#) multiplies the variance of depvar by \# and divides the deviance by \#. The resulting distributions are members of the quasilikelihood family. This option is allowed only with option irls.
scale( $\mathrm{x} 2|\mathrm{dev}| \#$ ) overrides the default scale parameter. This option is allowed only with Hessian (information matrix) variance estimates.
By default, scale(1) is assumed for the discrete distributions (binomial, Poisson, and negative binomial), and scale(x2) is assumed for the continuous distributions (Gaussian, gamma, and inverse Gaussian).
scale ( x 2 ) specifies that the scale parameter be set to the Pearson $\chi^{2}$ (or generalized $\chi^{2}$ ) statistic divided by the residual degrees of freedom, which is recommended by McCullagh and Nelder (1989) as a good general choice for continuous distributions.
scale (dev) sets the scale parameter to the deviance divided by the residual degrees of freedom. This option provides an alternative to scale (x2) for continuous distributions and overdispersed or underdispersed discrete distributions. This option is allowed only with option irls.
scale(\#) sets the scale parameter to \#. For example, using scale(1) in family (gamma) models results in exponential-errors regression. Additional use of link(log) rather than the default link (power -1) for family (gamma) essentially reproduces Stata's streg, dist (exp) nohr command (see [ST] streg) if all the observations are uncensored.
level (\#); see [R] Estimation options.
eform displays the exponentiated coefficients and corresponding standard errors and confidence intervals. For family(binomial) link(logit) (that is, logistic regression), exponentiation results are odds ratios; for family (nbinomial) link(log) (that is, negative binomial regression) and for family (poisson) link(log) (that is, Poisson regression), exponentiated coefficients are incidence-rate ratios.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\% fmt), sformat (\% $\% m t$ ), and nolstretch; see [R] Estimation options.
$\qquad$ Maximization
ml requests that optimization be carried out using Stata's ml commands and is the default.
irls requests iterated, reweighted least-squares (IRLS) optimization of the deviance instead of NewtonRaphson optimization of the log likelihood. If the irls option is not specified, the optimization is carried out using Stata's ml commands, in which case all options of ml maximize are also available.
maximize_options: difficult, technique (algorithm_spec), iterate (\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from(init_specs); see [R] Maximize. These options are seldom used.

Setting the optimization method to technique (bhhh) resets the default vcetype to vce (opg).
If option irls is specified, only maximize_options iterate(), nolog, trace, and ltolerance() are allowed. With irls specified, the convergence criterion is satisfied when the absolute change in deviance from one iteration to the next is less than or equal to ltolerance(), where ltolerance ( $1 \mathrm{e}-6$ ) is the default.
fisher (\#) specifies the number of Newton-Raphson steps that should use the Fisher scoring Hessian or EIM before switching to the observed information matrix (OIM). This option is useful only for Newton-Raphson optimization (and not when using irls).
search specifies that the command search for good starting values. This option is useful only for Newton-Raphson optimization (and not when using irls).

The following options are available with glm but are not shown in the dialog box: noheader suppresses the header information from the output. The coefficient table is still displayed. notable suppresses the table of coefficients from the output. The header information is still displayed. nodisplay suppresses the output. The iteration $\log$ is still displayed.
collinear, coeflegend; see [R] Estimation options. collinear is not allowed with irls.

## Remarks and examples

Remarks are presented under the following headings:

## General use

Variance estimators
User-defined functions

## General use

glm fits generalized linear models of $y$ with covariates $\mathbf{x}$ :

$$
g\{E(y)\}=\mathbf{x} \boldsymbol{\beta}, \quad y \sim F
$$

$g()$ is called the link function, and $F$ is the distributional family. Substituting various definitions for $g()$ and $F$ results in a surprising array of models. For instance, if $y$ is distributed as Gaussian (normal) and $g()$ is the identity function, we have

$$
E(y)=\mathbf{x} \boldsymbol{\beta}, \quad y \sim \text { Normal }
$$

or linear regression. If $g()$ is the logit function and $y$ is distributed as Bernoulli, we have

$$
\operatorname{logit}\{E(y)\}=\mathbf{x} \boldsymbol{\beta}, \quad y \sim \text { Bernoulli }
$$

or logistic regression. If $g()$ is the natural $\log$ function and $y$ is distributed as Poisson, we have

$$
\ln \{E(y)\}=\mathbf{x} \boldsymbol{\beta}, \quad y \sim \text { Poisson }
$$

or Poisson regression, also known as the log-linear model. Other combinations are possible.
Although glm can be used to perform linear regression (and, in fact, does so by default), this regression should be viewed as an instructional feature; regress produces such estimates more quickly, and many postestimation commands are available to explore the adequacy of the fit; see [ $R$ ] regress and [ $R$ ] regress postestimation.

In any case, you specify the link function by using the link() option and specify the distributional family by using family(). The available link functions are

| Link function | glm option |
| :--- | :--- |
| identity | link(identity) |
| log | link(log) |
| logit | link(logit) |
| probit | link(probit) |
| complementary log-log | link(cloglog) |
| odds power | link(opower \#) |
| power | link(power \#) |
| negative binomial | link(nbinomial) |
| log-log | link(loglog) |
| log-complement | link(logc) |

Define $\mu=E(y)$ and $\eta=g(\mu)$, meaning that $g(\cdot)$ maps $E(y)$ to $\eta=\mathbf{x} \boldsymbol{\beta}+$ offset.

Link functions are defined as follows:
identity is defined as $\eta=g(\mu)=\mu$.
$\log$ is defined as $\eta=\ln (\mu)$.
logit is defined as $\eta=\ln \{\mu /(1-\mu)\}$, the natural $\log$ of the odds.
probit is defined as $\eta=\Phi^{-1}(\mu)$, where $\Phi^{-1}()$ is the inverse Gaussian cumulative.
cloglog is defined as $\eta=\ln \{-\ln (1-\mu)\}$.
opower is defined as $\eta=\left[\{\mu /(1-\mu)\}^{n}-1\right] / n$, the power of the odds. The function is generalized so that link (opower 0 ) is equivalent to link(logit), the natural log of the odds. power is defined as $\eta=\mu^{n}$. Specifying link(power 1) is equivalent to specifying link(identity). The power function is generalized so that $\mu^{0} \equiv \ln (\mu)$. Thus, link(power 0 ) is equivalent to $\operatorname{link}(\log )$. Negative powers are, of course, allowed.
nbinomial is defined as $\eta=\ln \{\mu /(\mu+k)\}$, where $k=1$ if family (nbinomial) is specified, $k=\#_{k}$ if family (nbinomial $\#_{k}$ ) is specified, and $k$ is estimated via maximum likelihood if family (nbinomial ml) is specified.
$\log l o g$ is defined as $\eta=-\ln \{-\ln (\mu)\}$.
loge is defined as $\eta=\ln (1-\mu)$.
The available distributional families are

| Family | glm option |
| :--- | :--- |
| Gaussian (normal) | family (gaussian) |
| inverse Gaussian | family (igaussian) |
| Bernoulli/binomial | family (binomial) |
| Poisson | family (poisson) |
| negative binomial | family (nbinomial) |
| gamma | family (gamma) |

family(normal) is a synonym for family (gaussian).
The binomial distribution can be specified as 1) family (binomial), 2) family (binomial $\#_{N}$ ), or 3) family (binomial varname ${ }_{N}$ ). In case $2, \#_{N}$ is the value of the binomial denominator $N$, the number of trials. Specifying family (binomial 1) is the same as specifying family (binomial). In case 3, varname $N_{N}$ is the variable containing the binomial denominator, allowing the number of trials to vary across observations.

The negative binomial distribution can be specified as 1) family(nbinomial), 2) family (nbinomial $\#_{k}$ ), or 3) family (nbinomial ml ). Omitting $\#_{k}$ is equivalent to specifying family (nbinomial 1). In case 3, the value of $\#_{k}$ is estimated via maximum likelihood. The value $\#_{k}$ enters the variance and deviance functions. Typical values range between 0.01 and 2 ; see the technical note below.

You do not have to specify both family() and link(); the default link() is the canonical link for the specified family() (except for nbinomial):

| Family | Default link |
| :--- | :--- |
| family (gaussian) | link(identity) |
| family (igaussian) | link(power -2) |
| family (binomial) | link(logit) |
| family (poisson) | link(log) |
| family (nbinomial) | link(log) |
| family (gamma) | link(power -1) |

If you specify both family() and link(), not all combinations make sense. You may choose from the following combinations:

|  | identity | $\log$ | logit | probit | cloglog | power | opower | nbinomial | $\operatorname{loglog}$ | $\operatorname{logc}$ |
| :--- | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Gaussian | x | x |  |  |  | x |  |  |  |  |
| inverse Gaussian | x | x |  |  |  | x |  |  |  |  |
| binomial | x | x | x | x | x | x | x |  | x | x |
| Poisson | x | x |  |  |  | x |  |  |  |  |
| negative binomial | x | x |  |  |  | x |  | x |  |  |
| gamma | x | x |  |  | x |  |  |  |  |  |

## Technical note

Some family() and link() combinations result in models already fit by Stata. These are

| family () | link() | Options | Equivalent Stata command |
| :---: | :---: | :---: | :---: |
| gaussian | identity | nothing $\mid$ irls ${ }^{\text {irls }}$ vce (oim) | regress |
| gaussian | identity | ```t(var) vce(hac nwest #) vfactor(#v)``` | newey, t (var) lag(\#) (see note 1) |
| binomial | cloglog | nothing \|irls vce(oim) | cloglog (see note 2) |
| binomial | probit | nothing \|irls vce(oim) | probit (see note 2) |
| binomial | logit | nothing $\mid$ irls $\mid$ irls vce(oim) | logit or logistic (see note 3) |
| poisson | log | nothing\|irls|irls vce(oim) | poisson (see note 3) |
| nbinomial | log | nothing\|irls vce(oim) | nbreg (see note 4) |
| gamma | log | scale(1) | streg, dist (exp) nohr (see note 5) |

## Notes:

1. The variance factor $\#_{v}$ should be set to $n /(n-k)$, where $n$ is the number of observations and $k$ the number of regressors. If the number of regressors is not specified, the estimated standard errors will, as a result, differ by this factor.
2. Because the link is not the canonical link for the binomial family, you must specify the vee (oim) option if using irls to get equivalent standard errors. If irls is used without vce(oim), the regression coefficients will be the same but the standard errors will be only asymptotically equivalent. If no options are specified (nothing), glm will optimize using Newton-Raphson, making it equivalent to the other Stata command.

See $[R]$ cloglog and $[R]$ probit for more details about these commands.
3. Because the canonical link is being used, the standard errors will be equivalent whether the EIM or the OIM estimator of variance is used.
4. Family negative binomial, log-link models-also known as negative binomial regression models - are used for data with an overdispersed Poisson distribution. Although glm can be used to fit such models, using Stata's maximum likelihood nbreg command is probably better. In the GLM approach, you specify family (nbinomial $\#_{k}$ ) and then search for a $\#_{k}$ that results in the deviance-based dispersion being 1 . You can also specify family (nbinomial ml) to estimate $\#_{k}$ via maximum likelihood, which will report the same value returned from nbreg. However, nbreg also reports a confidence interval for it; see [R] nbreg and Rogers (1993). Of course, glm allows links other than log, and for those links, including the canonical nbinomial link, you will need to use glm.
5. glm can be used to estimate parameters from exponential regressions, but this method requires specifying scale(1). However, censoring is not available. Censored exponential regression may be modeled using glm with family (poisson). The $\log$ of the original response is entered into a Poisson model as an offset, whereas the new response is the censor variable. The result of such modeling is identical to the log relative hazard parameterization of streg, dist(exp) nohr. See [ST] streg for details about the streg command.
In general, where there is overlap between a capability of glm and that of some other Stata command, we recommend using the other Stata command. Our recommendation is not because of some inferiority of the GLM approach. Rather, those other commands, by being specialized, provide options and ancillary commands that are missing in the broader glm framework. Nevertheless, glm does produce the same answers where it should.
Special note. When equivalence is expected, for some datasets, you may still see very slight differences in the results, most often only in the later digits of the standard errors. When you compare glm output to an equivalent Stata command, these tiny discrepancies arise for many reasons:
a. glm uses a general methodology for starting values, whereas the equivalent Stata command may be more specialized in its treatment of starting values.
b. When using a canonical link, glm, irls should be equivalent to the maximum likelihood method of the equivalent Stata command, yet the convergence criterion is different (one is for deviance, the other for log likelihood). These discrepancies are easily resolved by adjusting one convergence criterion to correspond to the other.
c. When both glm and the equivalent Stata command use Newton-Raphson, small differences may still occur if the Stata command has a different default convergence criterion from that of glm. Adjusting the convergence criterion will resolve the difference. See [R] ml and [R] Maximize for more details.

## $>$ Example 1

In example 1 of $[\mathrm{R}]$ logistic, we fit a model based on data from a study of risk factors associated with low birthweight (Hosmer, Lemeshow, and Sturdivant 2013, 24). We can replicate the estimation by using glm:
. use https://www.stata-press.com/data/r18/lbw
(Hosmer \& Lemeshow data)
. glm low age lwt i.race smoke ptl ht ui, family (binomial) link(logit)
Iteration 0: Log likelihood $=-101.0213$
Iteration 1: Log likelihood $=-100.72519$
Iteration 2: Log likelihood $=$-100.724
Iteration 3: Log likelihood $=-100.724$
Generalized linear models Number of obs $=\quad 189$

| Optimization $: ~ M L ~$ | Residual df | $=$ | 180 |
| :--- | :--- | :--- | :--- |

Deviance $=201.4479911 \quad(1 / \mathrm{df})$ Deviance $=1.119156$

Pearson $=182.0233425 \quad(1 / d f)$ Pearson $=1.011241$
Variance function: $V(u)=u *(1-u)$
[Bernoulli]
Link function : $g(u)=\ln (u /(1-u))$
[Logit]
AIC $=1.1611$
Log likelihood $=-100.7239956 \quad$ BIC $=-742.0665$

|  | OIM |  |  |  |  |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| low | Coefficient | std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| age | -.0271003 | .0364504 | -0.74 | 0.457 | -.0985418 | .0443412 |
| lwt | -.0151508 | .0069259 | -2.19 | 0.029 | -.0287253 | -.0015763 |
| race |  |  |  |  |  |  |
| Black | 1.262647 | .5264101 | 2.40 | 0.016 | .2309024 | 2.294392 |
| Other | .8620792 | .4391532 | 1.96 | 0.050 | .0013548 | 1.722804 |
| smoke | .9233448 | .4008266 | 2.30 | 0.021 | .137739 | 1.708951 |
| ptl | .5418366 | .346249 | 1.56 | 0.118 | -.136799 | 1.220472 |
| ht | 1.832518 | .6916292 | 2.65 | 0.008 | .4769494 | 3.188086 |
| ui | .7585135 | .4593768 | 1.65 | 0.099 | -.1418484 | 1.658875 |
| _cons | .4612239 | 1.20459 | 0.38 | 0.702 | -1.899729 | 2.822176 |

glm, by default, presents coefficient estimates, whereas logistic presents the exponentiated coefficients - the odds ratios. glm's eform option reports exponentiated coefficients, and glm, like Stata's other estimation commands, replays results.

| Generalized linear models | Number of obs |  | 189 |
| :---: | :---: | :---: | :---: |
| Optimization : ML | Residual df |  | 180 |
|  | Scale parameter |  | 1 |
| Deviance $\quad=201.4479911$ | (1/df) Deviance | = | 1.119156 |
| Pearson $=182.0233425$ | (1/df) Pearson | = | 1.011241 |
| Variance function: $V(u)=u *(1-u)$ <br> Link function : $g(u)=\ln (u /(1-u))$ | [Bernoulli] |  |  |
|  | [Logit] |  |  |
|  | AIC | = | 1.1611 |
| Log likelihood = -100.7239956 | BIC |  | -742.0665 |


|  | OIM |  |  |  |  |  |
| ---: | ---: | ---: | :---: | :---: | :---: | ---: |
| low | Odds ratio | std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| age | .9732636 | .0354759 | -0.74 | 0.457 | .9061578 | 1.045339 |
| lwt | .9849634 | .0068217 | -2.19 | 0.029 | .9716834 | .9984249 |
| race |  |  |  |  |  |  |
| Black | 3.534767 | 1.860737 | 2.40 | 0.016 | 1.259736 | 9.918406 |
| Other | 2.368079 | 1.039949 | 1.96 | 0.050 | 1.001356 | 5.600207 |
|  |  |  |  |  |  |  |
| smoke | 2.517698 | 1.00916 | 2.30 | 0.021 | 1.147676 | 5.523162 |
| ptl | 1.719161 | .5952579 | 1.56 | 0.118 | .8721455 | 3.388787 |
| ht | 6.249602 | 4.322408 | 2.65 | 0.008 | 1.611152 | 24.24199 |
| ui | 2.1351 | .9808153 | 1.65 | 0.099 | .8677528 | 5.2534 |
| _cons | 1.586014 | 1.910496 | 0.38 | 0.702 | .1496092 | 16.8134 |

Note: _cons estimates baseline odds.
These results are the same as those reported in example 1 of $[R]$ logistic.
Included in the output header are values for the Akaike (1973) information criterion (AIC) and the Bayesian information criterion (BIC) (Raftery 1995). Both are measures of model fit adjusted for the number of parameters that can be compared across models. In both cases, a smaller value generally indicates a better model fit. AIC is based on the log likelihood and thus is available only when Newton-Raphson optimization is used. BIC is based on the deviance and thus is always available.

## Technical note

The values for AIC and BIC reported in the output after glm are different from those reported by estat ic:

- estat ic

Akaike's information criterion and Bayesian information criterion

| Model | N | ll(null) | ll (model) | df | AIC | BIC |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| . | 189 |  | . | -100.724 | 9 | 219.448 |

Note: BIC uses $N=$ number of observations. See [R] IC note.
There are various definitions of these information criteria (IC) in the literature; glm and estat ic use different definitions. glm bases its computation of the BIC on deviance, whereas estat ic uses the likelihood. Both glm and estat ic use the likelihood to compute the AIC; however, the AIC from estat ic is equal to $N$, the number of observations, times the AIC from glm. Refer to Methods and formulas in this entry and $[\mathrm{R}]$ estat ic for the references and formulas used by glm and estat ic, respectively, to compute AIC and BIC. Inferences based on comparison of IC values reported by glm
for different GLM models will be equivalent to those based on comparison of IC values reported by estat ic after glm.

## > Example 2

We use data from an early insecticide experiment, given in Pregibon (1980). The variables are ldose, the log dose of insecticide; $n$, the number of flour beetles subjected to each dose; and $r$, the number killed.

```
. use https://www.stata-press.com/data/r18/ldose
. list, sep(4)
```

| ldose | n | r |
| :---: | ---: | ---: |
| 1.6907 | 59 | 6 |
| 1.7242 | 60 | 13 |
| 1.7552 | 62 | 18 |
| 1.7842 | 56 | 28 |
| 1.8113 | 63 | 52 |
| 1.8369 | 59 | 53 |
| 1.861 | 62 | 61 |
| 1.8839 | 60 | 60 |

The aim of the analysis is to estimate a dose-response relationship between $p$, the proportion killed, and $X$, the $\log$ dose.

As a first attempt, we will formulate the model as a linear logistic regression of $p$ on ldose; that is, we will take the logit of $p$ and represent the dose-response curve as a straight line in $X$ :

$$
\ln \{p /(1-p)\}=\beta_{0}+\beta_{1} X
$$

Because the data are grouped, we cannot use Stata's logistic command to fit the model. Instead, we will fit the model by using glm:

| Iteration 0: Log likelihood $=-18.824848$ |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Iteration 1: Log likelihood $=-18.715271$ |  |  |  |  |  |  |
| Iteration 2: Log likelihood $=-18.715123$ |  |  |  |  |  |  |
| Iteration 3: Log likelihood $=-18.715123$ |  |  |  |  |  |  |
| Generalized linear models |  |  |  | Numb | of obs | 8 |
| Optimization : ML |  |  |  | Resi | al df | 6 |
|  |  |  |  | Sca | parameter | 1 |
| Deviance $=11.23220702$ |  |  |  | (1/d | Deviance | 1.872035 |
| Pearson $=10.0267936$ |  |  |  | (1/d | Pearson | 1.671132 |
| Variance function: $V(\mathrm{u})=\mathrm{u} *(1-\mathrm{u} / \mathrm{n})$ |  |  |  | [Binomial] |  |  |
| Link function : $\mathrm{g}(\mathrm{u})=\ln (\mathrm{u} /(\mathrm{n}-\mathrm{u})$ ) |  |  |  | [Logit] |  |  |
|  |  |  |  | AIC |  | 5.178781 |
| Log likelihood = -18.71512262 |  |  |  | BIC |  | -1.244442 |
| r | Coefficient | $\begin{gathered} \text { OIM } \\ \text { std. err. } \end{gathered}$ | z | $P>\|z\|$ | [95\% conf | interval] |
| ldose | 34.27034 | 2.912141 | 11.77 | 0.000 | 28.56265 | 39.97803 |
| _cons | -60.71747 | 5.180713 | -11.72 | 0.000 | -70.87149 | -50.56346 |

We specified family (binomial $n$ ), meaning that variable $n$ contains the denominator.
An alternative model, which gives asymmetric sigmoid curves for $p$, involves the complementary $\log -\log$, or cloglog, function:

$$
\ln \{-\ln (1-p)\}=\beta_{0}+\beta_{1} X
$$

We fit this model by using glm:

```
. glm r ldose, family(binomial n) link(cloglog)
Iteration 0: Log likelihood = -14.883594
Iteration 1: Log likelihood = -14.822264
Iteration 2: Log likelihood = -14.822228
Iteration 3: Log likelihood = -14.822228
Generalized linear models Number of obs = 8
Optimization : ML
Deviance = 3.446418004
Pearson = 3.294675153
Variance function: V(u) = u*(1-u/n)
Link function : g(u) = ln}(-\operatorname{ln}(1-u/n)
Log likelihood = -14.82222811 BIC = -9.030231
```

|  | OIM |  |  |  |  |  |
| ---: | ---: | :---: | ---: | ---: | ---: | ---: |
| r | Coefficient | std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ldose | 22.04118 | 1.793089 | 12.29 | 0.000 | 18.52679 | 25.55557 |
| _cons | -39.57232 | 3.229047 | -12.26 | 0.000 | -45.90114 | -33.24351 |

The cloglog model is preferred; the deviance for the logistic model, 11.23, is much higher than the deviance for the cloglog model, 3.45. This change also is evident by comparing log likelihoods, or equivalently, AIC values.

This example also shows the advantage of the glm command-we can vary assumptions easily. Note the minor difference in what we typed to obtain the logistic and cloglog models:

```
. glm r ldose, family(binomial n) link(logit)
. glm r ldose, family(binomial n) link(cloglog)
```

If we were performing this work for ourselves, we would have typed the commands in a more abbreviated form:

```
. glm r ldose, f(b n) l(l)
. glm r ldose, f(b n) l(cl)
```


## - Technical note

Factor variables may be used with glm. Say that, in the example above, we had ldose, the $\log$ dose of insecticide; $n$, the number of flour beetles subjected to each dose; and $r$, the number killed-all as before-except that now we have results for three different kinds of beetles. Our hypothetical data include beetle, which contains the values 1 ("Destructive flour"), 2 ("Red flour"), and 3 ("Mealworm").
. use https://www.stata-press.com/data/r18/beetle
. list, sep(0)

| beetle | ldose | n | r |
| :--- | ---: | ---: | ---: |
| Destructive flour | 1.6907 | 59 | 6 |
| Destructive flour | 1.7242 | 60 | 13 |
| Destructive flour | 1.7552 | 62 | 18 |
| Destructive flour | 1.7842 | 56 | 28 |
| Destructive flour | 1.8113 | 63 | 52 |
| (output omitted ) |  |  |  |
| Mealworm | 1.861 | 64 | 23 |
| Mealworm | 1.8839 | 58 | 22 |

Let's assume that, at first, we wish merely to add a shift factor for the type of beetle. We could type

| Iteration 0: Log likelihood = -79.012269 |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Iteration 1: Log likelihood $=-76.94951$ |  |  |  |  |  |  |
| Iteration 2: Log likelihood $=-76.945645$ |  |  |  |  |  |  |
| Iteration 3: Log likelihood = -76.945645 |  |  |  |  |  |  |
| Generalized linear models |  |  |  | Numb | of obs | 24 |
| Optimization : ML |  |  |  | Resi | al df | 20 |
|  |  |  |  | Scal | parameter | 1 |
| Deviance $\quad=73.76505595$ |  |  |  | (1/d | Deviance | 3.688253 |
| Pearson $=71.8901173$ |  |  |  | (1/d | Pearson | 3.594506 |
| Variance function: $V(u)=u *(1-u / n)$ |  |  |  | [Binomial] |  |  |
| Link function : $\mathrm{g}(\mathrm{u})=\ln (-\ln (1-$ |  |  |  | [Complementary log-log] |  |  |
|  |  |  |  | AIC |  | 6.74547 |
| Log likelihood = -76.94564525 |  |  |  | BIC |  | 10.20398 |
| r | Coefficient | $\begin{gathered} \text { OIM } \\ \text { std. err. } \end{gathered}$ | z | $P>\|z\|$ | [95\% conf | interval] |
| beetle |  |  |  |  |  |  |
| Red flour | -. 0910396 | . 1076132 | -0.85 | 0.398 | -. 3019576 | . 1198783 |
| Mealworm | -1.836058 | . 1307125 | -14.05 | 0.000 | -2.09225 | -1.579867 |
| ldose | 19.41558 | . 9954265 | 19.50 | 0.000 | 17.46458 | 21.36658 |
| _cons | -34.84602 | 1.79333 | -19.43 | 0.000 | -38.36089 | -31.33116 |

We find strong evidence that the insecticide works differently on the mealworm. We now check whether the curve is merely shifted or also differently sloped:

| Iteration 0: Log likelihood $=-67.270188$ |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Iteration 1: Log likelihood $=-65.149316$ |  |  |  |  |  |  |
| Iteration 2: Log likelihood $=-65.147978$ |  |  |  |  |  |  |
| Iteration 3: Log likelihood = -65.147978 |  |  |  |  |  |  |
| Generalized linear models |  |  |  | Number of obs $=\quad 24$ |  |  |
| Optimization : ML |  |  |  | Residual df |  | 18 |
|  |  |  |  | Scale | rameter = | 1 |
| Deviance $\quad=50.16972096$ |  |  |  | (1/df) Deviance $=2.787207$ |  |  |
| Pearson $\quad=49.28422567$ |  |  |  | (1/df) Pearson $=2.738013$ |  |  |
| Variance function: $V(\mathrm{u})=\mathrm{u} *(1-\mathrm{u} / \mathrm{n})$ |  |  |  | [Binomial] |  |  |
| Link function | $: g(u)=\ln (-\ln (1-\mathrm{l} / \mathrm{n}))$ |  |  | [Complementary log-log] |  |  |
|  |  |  |  | AIC | = | 5.928998 |
| Log likelihood | $=-65.14797776$ |  |  | BIC |  | -7.035248 |
|  | Coefficient | $\begin{gathered} \text { OIM } \\ \text { std. err. } \end{gathered}$ | z | $P>\|z\|$ | [95\% con | interval] |
| beetle |  |  |  |  |  |  |
| Red flour | -. 79933 | 4.470882 | -0.18 | 0.858 | -9.562098 | 7.963438 |
| Mealworm | 17.78741 | 4.586429 | 3.88 | 0.000 | 8.798172 | 26.77664 |
| ldose | 22.04118 | 1.793089 | 12.29 | 0.000 | 18.52679 | 25.55557 |
| beetle\#c.ldose |  |  |  |  |  |  |
| Red flour | . 3838708 | 2.478477 | 0.15 | 0.877 | -4.473855 | 5.241596 |
| Mealworm | -10.726 | 2.526412 | -4.25 | 0.000 | -15.67768 | -5.774321 |
| _cons | -39.57232 | 3.229047 | -12.26 | 0.000 | -45.90114 | -33.24351 |

We find that the (complementary $\log -\log$ ) dose-response curve for the mealworm has roughly half the slope of that for the destructive flour beetle.

See [U] 26 Working with categorical data and factor variables; what is said there concerning linear regression is applicable to any GLM model.

## Variance estimators

glm offers many variance options and gives different types of standard errors when used in various combinations. We highlight some of them here, but for a full explanation, see Hardin and Hilbe (2018).

## > Example 3

Continuing with our flour beetle data, we rerun the most recently displayed model, this time requesting estimation via IRLS.

```
. use https://www.stata-press.com/data/r18/beetle
. glm r beetle##c.ldose, f(bin n) l(cloglog) ltol(1e-13) irls
Iteration 1: Deviance = 54.41414
Iteration 2: Deviance = 50.19424
Iteration 3: Deviance = 50.16973
    (output omitted)
\begin{tabular}{llll} 
Generalized linear models & Number of obs & \(=\) & 24 \\
Optimization & : MQL Fisher scoring & (IRLS EIM) & Residual df \\
& \(=50.16972096\) & Scale parameter \(=\) & 18 \\
Deviance & \(=49.28422528\) & \((1 / d f)\) Deviance & \(=\) \\
Pearson & (1/df) Pearson & \(=\) & 2.787207 \\
Variance function: & \(\mathrm{V}(\mathrm{u})=\mathrm{u} *(1-\mathrm{u} / \mathrm{n})\) & [Binomial] & \\
Link function \(: ~\) & \(\mathrm{~g}(\mathrm{u})=\ln (-\ln (1-\mathrm{u} / \mathrm{n}))\) & [Complementary log-log] & \\
& & BIC & \(=-7.035248\)
\end{tabular}
```

| $r$ | Coefficient | std. err. | $z$ | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| beetle <br> Red flour <br> Mealworm | -.79933 | 4.586649 | -0.17 | 0.862 | -9.788997 | 8.190337 |
| ldose | 22.04118 | 1.79741 | 4.624834 | 3.85 | 0.000 | 8.7229 | 26.85192

Note our use of the ltol() option, which, although unrelated to our discussion on variance estimation, was used so that the regression coefficients would match those of the previous Newton-Raphson (NR) fit.

Because IRLS uses the EIM for optimization, the variance estimate is also based on EIM. If we want optimization via IRLS but the variance estimate based on OIM, we specify glm, irls vce (oim):
glm r beetle\#\#c.ldose, f(b) l ( cl ) ltol(1e-15) irls vce(oim) noheader nolog

| r | Coefficient | $\begin{gathered} \text { OIM } \\ \text { std. err. } \end{gathered}$ | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| beetle |  |  |  |  |  |  |
| Red flour | -. 79933 | 4.470882 | -0.18 | 0.858 | -9.562098 | 7.963438 |
| Mealworm | 17.78741 | 4.586429 | 3.88 | 0.000 | 8.798172 | 26.77664 |
| ldose | 22.04118 | 1.793089 | 12.29 | 0.000 | 18.52679 | 25.55557 |
| beetle\#c.ldose |  |  |  |  |  |  |
| Red flour | . 3838708 | 2.478477 | 0.15 | 0.877 | -4.473855 | 5.241596 |
| Mealworm | -10.726 | 2.526412 | -4.25 | 0.000 | -15.67768 | -5.774321 |
| _cons | -39.57232 | 3.229047 | -12.26 | 0.000 | -45.90114 | -33.24351 |

This approach is identical to NR except for the convergence path. Because the cloglog link is not the canonical link for the binomial family, EIM and OIM produce different results. Both estimators, however, are asymptotically equivalent.

Going back to NR, we can also specify vce(robust) to get the Huber/White/sandwich estimator of variance:

| r | Coefficient | Robust std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| beetle |  |  |  |  |  |  |
| Red flour | -. 79933 | 5.733049 | -0.14 | 0.889 | -12.0359 | 10.43724 |
| Mealworm | 17.78741 | 5.158477 | 3.45 | 0.001 | 7.676977 | 27.89784 |
| ldose | 22.04118 | . 8998551 | 24.49 | 0.000 | 20.27749 | 23.80486 |
| beetle\#c.ldose |  |  |  |  |  |  |
| Red flour | . 3838708 | 3.174427 | 0.12 | 0.904 | -5.837892 | 6.605633 |
| Mealworm | -10.726 | 2.800606 | -3.83 | 0.000 | -16.21508 | -5. 236912 |
| _cons | -39.57232 | 1.621306 | -24.41 | 0.000 | -42.75003 | -36.39462 |

The sandwich estimator gets its name from the form of the calculation-it is the multiplication of three matrices, with the outer two matrices (the "bread") set to the OIM variance matrix. When irls is used along with vce (robust), the EIM variance matrix is instead used as the bread. Using a result from McCullagh and Nelder (1989), Newson (1999) points out that the EIM and OIM variance matrices are equivalent under the canonical link. Thus if irls is specified with the canonical link, the resulting variance is labeled "Robust". When the noncanonical link for the family is used, which is the case in the example below, the EIM and OIM variance matrices differ, so the resulting variance is labeled "Semirobust".

```
. glm r beetle##c.ldose, f(b n) l(cl) irls ltol(1e-15) vce(robust) noheader
> nolog
```

| r | Coefficient | Semirobust std. err. | z | $P>\|z\|$ | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| beetle |  |  |  |  |  |  |
| Red flour | -. 79933 | 6.288963 | -0.13 | 0.899 | -13.12547 | 11.52681 |
| Mealworm | 17.78741 | 5.255307 | 3.38 | 0.001 | 7.487194 | 28.08762 |
| ldose | 22.04118 | . 9061566 | 24.32 | 0.000 | 20.26514 | 23.81721 |
| beetle\#c.ldose |  |  |  |  |  |  |
| Red flour | . 3838708 | 3.489723 | 0.11 | 0.912 | -6.455861 | 7.223603 |
| Mealworm | -10.726 | 2.855897 | -3.76 | 0.000 | -16.32345 | -5.128542 |
| _cons | -39.57232 | 1.632544 | -24.24 | 0.000 | -42.77205 | -36.3726 |

The outer product of the gradient ( OPG ) estimate of variance is one that avoids the calculation of second derivatives. It is equivalent to the "middle" part of the sandwich estimate of variance and can be specified by using glm, vce (opg), regardless of whether NR or IRLS optimization is used.

| r | Coefficient | $\begin{gathered} \text { OPG } \\ \text { std. err. } \end{gathered}$ | z | $P>\|z\|$ | [95\% conf. | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| beetle |  |  |  |  |  |  |
| Red flour | -. 79933 | 6.664045 | -0.12 | 0.905 | -13.86062 | 12.26196 |
| Mealworm | 17.78741 | 6.838505 | 2.60 | 0.009 | 4.384183 | 31.19063 |
| ldose | 22.04118 | 3.572983 | 6.17 | 0.000 | 15.03826 | 29.0441 |
| beetle\#c.ldose |  |  |  |  |  |  |
| Red flour | . 3838708 | 3.700192 | 0.10 | 0.917 | -6.868372 | 7.636114 |
| Mealworm | -10.726 | 3.796448 | -2.83 | 0.005 | -18.1669 | -3.285097 |
| _cons | -39.57232 | 6.433101 | -6.15 | 0.000 | -52.18097 | -26.96368 |

The OPG estimate of variance is a component of the BHHH (Berndt et al. 1974) optimization technique. This method of optimization is also available with glm with the technique() option; however, the technique() option is not allowed with the irls option.

## > Example 4

The Newey-West (1987) estimator of variance is a sandwich estimator with the "middle" of the sandwich modified to account for possible autocorrelation between the observations. These estimators are a generalization of those given by the Stata command newey for linear regression. See [TS] newey for more details.

For example, consider the dataset given in [TS] newey, which has time-series measurements on usr and idle. We want to perform a linear regression with Newey-West standard errors.
. use https://www.stata-press.com/data/r18/idle2
. list usr idle time

| usr | idle | time |
| :---: | :---: | :---: |
| 0 | 100 | 1 |
| 0 | 100 | 2 |
| 0 | 97 | 3 |
| 1 | 98 | 4 |
| 2 | 94 | 5 |$|$| (output omitted) |  |  |
| :---: | :---: | :---: |
| 1 | 98 | 29 |
| 1 | 98 | 30 |

Examining Methods and formulas of [TS] newey, we see that the variance estimate is multiplied by a correction factor of $n /(n-k)$, where $k$ is the number of regressors. glm, vce (hac ...) does not make this correction, so to get the same standard errors, we must use the vfactor() option within glm to make the correction manually.

| display 30/28 |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| $1.0714286$ <br> . tsset time |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
| ```Time variable: time, 1 to 30 Delta: 1 unit``` |  |  |  |  |  |  |
| . glm usr idle, vce(hac nwest 3) vfactor(1.0714286) |  |  |  |  |  |  |
| Iteration 0: Log likelihood = -71.743396 |  |  |  |  |  |  |
| Generalized linear models |  |  |  | Num | of obs | 30 |
| Optimization | : ML |  |  | Res | al df | 28 |
|  |  |  |  | Scal | parameter | 7.493297 |
| Deviance $\quad=209.8123165$ |  |  |  | (1/d | Deviance | 7.493297 |
| Pearson $\quad=209.8123165$ |  |  |  | (1/d | Pearson | 7.493297 |
| Variance function: V u$)=1$ |  |  |  | [Gau | ian] |  |
| Link function : $\mathrm{g}(\mathrm{u})=\mathrm{u}$ |  |  |  | [Id | ity] |  |
| HAC kernel (lags) : Newey-West (3) |  |  |  |  |  |  |
|  |  |  |  | AIC |  | 4.916226 |
| Log likelihood | $=-71.74339627$ |  |  | BIC |  | 114.5788 |
| usr | Coefficient | $\begin{gathered} \text { HAC } \\ \text { std. err. } \end{gathered}$ | z | $P>\|z\|$ | [95\% con | interval] |
| idle | -. 2281501 | . 0690928 | -3.30 | 0.001 | -. 3635694 | -. 0927307 |
| _cons | 23.13483 | 6.327033 | 3.66 | 0.000 | 10.73407 | 35.53558 |

The glm command above reproduces the results given in [TS] newey. We may now generalize this output to models other than simple linear regression and to different kernel weights.

glm also offers variance estimators based on the bootstrap (resampling your data with replacement) and the jackknife (refitting the model with each observation left out in succession). Also included is the one-step jackknife estimate, which, instead of performing full reestimation when each observation is omitted, calculates a one-step NR estimate, with the full data regression coefficients as starting values.


See Hardin and Hilbe (2018) for a full discussion of the variance options that go with glm and, in particular, of how the different variance estimators are modified when vce (cluster clustvar) is specified. Finally, not all variance options are supported with all types of weights. See help glm for a current table of the variance options that are supported with the different weights.

## User-defined functions

glm may be called with a community-contributed link function, variance (family) function, NeweyWest kernel-weight function, or any combination of the three.

## Syntax of link functions

```
program progname
    version 18.0 // (or version 18.5 for StataNow)
    args todo eta mu return
    if 'todo' == -1 {
            /* Set global macros for output */
            global SGLM_lt "title for link function"
            global SGLM_lf "subtitle showing link definition"
            exit
    }
    if 'todo' == 0 {
            /* set }\eta=g(\mu) *
            /* Intermediate calculations go here */
            generate double 'eta' = ...
            exit
    }
    if 'todo' == 1 {
            /* set }\mu=\mp@subsup{g}{}{-1}(\eta)*
            /* Intermediate calculations go here */
            generate double 'mu' = ...
            exit
    }
    if 'todo' == 2 {
            /* set return = \partial\mu/\partial\eta */
            /* Intermediate calculations go here */
            generate double 'return' = ...
            exit
    }
    if 'todo' == 3 {
            /* set return = \partial}\mp@subsup{}{2}{\mu}\mu/\partial\mp@subsup{\eta}{}{2}**
            /* Intermediate calculations go here */
            generate double 'return' = ...
            exit
}
display as error "Unknown call to glm link function"
exit 198
end
```


## Syntax of variance functions

```
program progname
    version 18.0 // (or version 18.5 for StataNow)
    args todo eta mu return
    if 'todo' == -1 {
        /* Set global macros for output */
        /* Also check that depvar is in proper range */
        /* Note: For this call, eta contains indicator for whether each obs. is in est. sample */
        global SGLM_vt "title for variance function"
        global SGLM_vf "subtitle showing function definition"
        global SGLM_mu "program to call to enforce boundary conditions on }\mu\mathrm{ "
        exit
    }
    if 'todo' == 0 {
        /* set }\eta\mathrm{ to initial value. */
        /* Intermediate calculations go here */
        generate double 'eta' = ...
        exit
    }
    if 'todo' == 1 {
            /* set return = V( }\mu)*
            /* Intermediate calculations go here */
            generate double 'return' = ...
            exit
    }
    if 'todo' == 2 {
            /* set return = \partialV ( }\mu)/\partial\mu *
            /* Intermediate calculations go here */
            generate double 'return' = ...
            exit
}
if 'todo' == 3 {
            /* set return = squared deviance (per observation) */
            /* Intermediate calculations go here */
            generate double 'return' = ...
            exit
}
if 'todo' == 4 {
            /* set return = Anscombe residual */
            /* Intermediate calculations go here */
            generate double 'return' = ...
            exit
}
if 'todo' == 5 {
            /* set return = log likelihood */
            /* Intermediate calculations go here */
            generate double 'return' = ...
            exit
}
if 'todo' == 6 {
    /* set return = adjustment for deviance residuals */
    /* Intermediate calculations go here */
    generate double 'return' = ...
    exit
}
display as error "Unknown call to glm variance function"
exit 198
end
```


## Syntax of Newey-West kernel-weight functions

```
program progname, rclass
    version 18.0 // (or version 18.5 for StataNow)
    args G j
    /* G is the maximum lag */
    /* j is the current lag */
    /* Intermediate calculations go here */
    return scalar wt = computed weight
    return local setype "Newey-West"
    return local sewtype "name of kernel"
end
```


## Global macros available for community-contributed programs

| Global macro | Description |
| :--- | :--- |
| SGLM_V | program name of variance (family) evaluator |
| SGLM_L | program name of link evaluator |
| dependent variable name |  |
| SGLM_y | binomial denominator |
| SGLM_m | negative binomial $k$ |
| SGLM_a | power if power() or opower() is used, or |
| SGLM_p | an argument from a user-specified link function |
|  | indicator; set to one if scale is equal to one |
| SGLM_s1 | value of scale parameter |

## > Example 5

Suppose that we wish to perform Poisson regression with a log-link function. Although this regression is already possible with standard glm, we will write our own version for illustrative purposes.

Because we want a $\log \operatorname{lin} k, \eta=g(\mu)=\ln (\mu)$, and for a Poisson family the variance function is $V(\mu)=\mu$.

The Poisson density is given by

$$
f\left(y_{i}\right)=\frac{e^{-\exp \left(\mu_{i}\right)} e^{\mu_{i} y_{i}}}{y_{i}!}
$$

resulting in a log likelihood of

$$
L=\sum_{i=1}^{n}\left\{-e^{\mu_{i}}+\mu_{i} y_{i}-\ln \left(y_{i}!\right)\right\}
$$

The squared deviance of the $i$ th observation for the Poisson family is given by

$$
d_{i}^{2}= \begin{cases}2 \widehat{\mu}_{i} & \text { if } y_{i}=0 \\ 2\left\{y_{i} \ln \left(y_{i} / \widehat{\mu}_{i}\right)-\left(y_{i}-\widehat{\mu}_{i}\right)\right\} & \text { otherwise }\end{cases}
$$

We now have enough information to write our own Poisson-log glm module. We create the file mylog.ado, which contains

```
program mylog
    version 18.0 // (or version 18.5 for StataNow)
    args todo eta mu return
    if 'todo' == -1 \{
            global SGLM_lt "My Log" // Titles for output
                global SGLM_lf "ln(u)"
                exit
    \}
    if 'todo' == 0 \{
        gen double 'eta' \(=\ln (' m u ') \quad / / \quad \eta=\ln (\mu)\)
        exit
    \}
    if 'todo' == 1 \{
        gen double 'mu' \(=\exp (' e t a ') \quad / / \mu=\exp (\eta)\)
        exit
    \}
    if 'todo' == 2 \{
        gen double 'return' \(=\) 'mu' \(\quad / / \partial \mu / \partial \eta=\exp (\eta)=\mu\)
        exit
    \}
    if 'todo' == 3 \{
            gen double 'return' \(=\) 'mu' \(\quad / / \quad \partial^{2} \mu / \partial \eta^{2}=\exp (\eta)=\mu\)
            exit
        \}
    di as error "Unknown call to glm link function"
    exit 198
end
```

and we create the file mypois.ado, which contains

```
program mypois
    version 18.0 // (or version 18.5 for StataNow)
    args todo eta mu return
    if 'todo' == -1 {
        local y "$SGLM_y"
        local touse "'eta'" // 'eta' marks estimation sample here
        capture assert 'y'>=0 if 'touse' // check range of y
        if _rc {
            di as error '"dependent variable 'y' has negative values"'
            exit 499
            }
            global SGLM_vt "My Poisson" // Titles for output
            global SGLM_vf "u"
            global SGLM_mu "glim_mu 0 ." // see note 1
            exit
    }
    if 'todo' == 0 { // Initialization of \eta; see note 2
            gen double 'eta' = ln('mu')
            exit
    }
```

```
if 'todo' == 1 {
    gen double 'return' = 'mu' // V( }\mu)=
    exit
}
if 'todo' == 2 { // \partial V (\mu)/\partial\mu
    gen byte 'return' = 1
    exit
}
if 'todo' == 3 { // squared deviance, defined above
    local y "$SGLM_y"
    if "'y'" == "" {
            local y "'e(depvar)'"
    }
    gen double 'return' = cond('y'==0, 2*'mu', /*
    */ 2*('y'*ln('y'/'mu')-('y'-'mu')))
    exit
}
if 'todo' == 4 { // Anscombe residual; see note 3
    local y "$SGLM_y"
    if "'y'" == "" {
                local y "'e(depvar)'"
    }
    gen double 'return' = 1.5*('y'^(2/3)-'mu'^(2/3)) / 'mu'^(1/6)
    exit
}
if 'todo' == 5 { // log likelihood; see note 4
    local y "$SGLM_y"
    if "'y'" == "" {
            local y "'e(depvar)'"
    }
    gen double 'return' = -'mu'+'y'*ln('mu')-lngamma('y'+1)
    exit
}
if 'todo' == 6 { // adjustment to residual; see note 5
    gen double 'return' = 1/(6*sqrt('mu'))
    exit
}
di as error "Unknown call to glm variance function"
error }19
```

end

Notes:

1. glim_mu is a Stata program that will, at each iteration, bring $\widehat{\mu}$ back into its plausible range, should it stray out of it. Here glim_mu is called with the arguments zero and missing, meaning that zero is the lower bound of $\widehat{\mu}$ and there exists no upper bound-such is the case for Poisson models.
2. Here the initial value of $\eta$ is easy because we intend to fit this model with our user-defined log link. In general, however, the initialization may need to vary according to the link to obtain convergence. If so, the global macro SGLM_L is used to determine which link is being utilized.
3. The Anscombe formula is given here because we know it. If we were not interested in Anscombe residuals, we could merely set 'return' to missing. Also, the local macro y is set either to SGLM_y if it is in current estimation or to e (depvar) if this function is being accessed by predict.
4. If we were not interested in ML estimation, we could omit this code entirely and just leave an exit statement in its place. Similarly, if we were not interested in deviance or IRLS optimization, we could set 'return' in the deviance portion of the code ('todo'==3) to missing.
5. This code defines the term to be added to the predicted residuals if the adjusted option is specified. Again, if we were not interested, we could set 'return' to missing.
We can now test our Poisson-log module by running it on the airline data presented in $[\mathrm{R}]$ poisson.
```
. use https://www.stata-press.com/data/r18/airline
```

. list airline injuries n XYZowned

|  | airline | injuries | n | XYZowned |
| :--- | ---: | ---: | ---: | ---: |
|  | 1. | 1 | 11 | 0.0950 |
| 2. | 2 | 7 | 0.1920 | 1 |
| 3. | 3 | 7 | 0.0750 | 0 |
| 4. | 4 | 19 | 0.2078 | 0 |
| 5. | 5 | 9 | 0.1382 | 0 |
| 6. | 6 | 4 | 0.0540 | 0 |
| 7. | 7 | 3 | 0.1292 | 1 |
| 8. | 8 | 1 | 0.0503 | 0 |
| 9. | 9 | 3 | 0.0629 | 0 |

- generate $\operatorname{lnN}=\ln (n)$
. glm injuries XYZowned $\operatorname{lnN}, f(m y p o i s) ~ l(m y l o g) ~ s c a l e(1) ~$
Iteration 0: Log likelihood $=-22.557572$
Iteration 1: Log likelihood $=-22.332861$
Iteration 2: Log likelihood $=-22.332276$
Iteration 3: Log likelihood $=-22.332276$


|  | OIM |  |  |  |  |  |
| ---: | ---: | :---: | :---: | :---: | ---: | ---: | ---: |
| injuries | Coefficient | std. err. | $\mathbf{z}$ | P>\|z| | [95\% conf. interval] |  |
| XYZowned | .6840668 | .3895877 | 1.76 | 0.079 | -.0795111 | 1.447645 |
| lnN | 1.424169 | .3725155 | 3.82 | 0.000 | .6940517 | 2.154286 |
| _cons | 4.863891 | .7090501 | 6.86 | 0.000 | 3.474178 | 6.253603 |

(Standard errors scaled using dispersion equal to square root of 1.)
These are precisely the results given in $[R]$ poisson and are those that would have been given had we run glm, family(poisson) link(log). The only minor adjustment we needed to make was to specify the scale(1) option. If scale() is left unspecified, glm assumes scale(1) for discrete distributions and scale(x2) for continuous ones. By default, glm assumes that any user-defined family is continuous because it has no way of checking. Thus, we needed to specify scale(1) because our model is discrete.

Because we were careful in defining the squared deviance, we could have fit this model with IRLS. Because $\log$ is the canonical link for the Poisson family, we would not only get the same regression coefficients but also the same standard errors.

## > Example 6

Suppose now that we wish to use our log link (mylog.ado) with glm's binomial family. This task requires some modification because our current function is not equipped to deal with the binomial denominator, which we are allowed to specify. This denominator is accessible to our link function through the global macro SGLM_m. We now make the modifications and store them in mylog2.ado.

```
program mylog2 // <-- changed
    version 18.0 // (or version 18.5 for StataNow)
    args todo eta mu return
    if 'todo' == -1 {
        global SGLM_lt "My Log, Version 2" // <-- changed
        if "$SGLM_m" == "1" { // <-- changed
            global SGLM_lf "ln(u)" // <-- changed
        } (u)
        else global SGLM_lf "ln(u/$SGLM_m)"
        exit
    }
    if 'todo' == 0 {
        gen double 'eta' = ln('mu'/$SGLM_m) // <-- changed
        exit
    }
    if 'todo' == 1 {
        gen double 'mu' = $SGLM_m*exp('eta') // <-- changed
        exit
    }
    if 'todo' == 2 {
        gen double 'return' = 'mu'
        exit
    }
    if 'todo' == 3 {
        gen double 'return' = 'mu'
        exit
    }
    di as error "Unknown call to glm link function"
    exit 198
end
```

We can now run our new log link with glm's binomial family. Using the flour beetle data from earlier, we have

```
. use https://www.stata-press.com/data/r18/beetle, clear
. glm r ldose, f(bin n) l(mylog2) irls
Iteration 1: Deviance = 2212.108
Iteration 2: Deviance = 452.9352
Iteration 3: Deviance = 429.95
Iteration 4: Deviance = 429.2745
Iteration 5: Deviance = 429.2192
Iteration 6: Deviance = 429.2082
Iteration 7: Deviance = 429.2061
Iteration 8: Deviance = 429.2057
Iteration 9: Deviance = 429.2056
Iteration 10: Deviance = 429.2056
Iteration 11: Deviance = 429.2056
Iteration 12: Deviance = 429.2056
```



```
\begin{tabular}{r|rcrrrr}
\hline\(r\) & Coefficient & std. err. & EIM & P>|z| & [95\% conf. interval] \\
\hline ldose & 8.478908 & .4702808 & 18.03 & 0.000 & 7.557175 & 9.400642 \\
_cons & -16.11006 & .8723167 & -18.47 & 0.000 & -17.81977 & -14.40035 \\
\hline
\end{tabular}
```

For a more detailed discussion on user-defined functions, and for an example of a user-defined Newey-West kernel weight, see Hardin and Hilbe (2018).

John Ashworth Nelder (1924-2010) was born in Somerset, England. He studied mathematics and statistics at Cambridge and worked as a statistician at the National Vegetable Research Station and then Rothamsted Experimental Station. In retirement, he was actively affiliated with Imperial College London. Nelder was especially well known for his contributions to the theory of linear models and to statistical computing. He was the principal architect of generalized and hierarchical generalized linear models and of the programs GenStat and GLIM.

Robert William Maclagan Wedderburn (1947-1975) was born in Edinburgh and studied mathematics and statistics at Cambridge. At Rothamsted Experimental Station, he developed the theory of generalized linear models with Nelder and originated the concept of quasilikelihood. He died of anaphylactic shock from an insect bite on a canal holiday.

## Stored results

$\mathrm{glm}, \mathrm{ml}$ stores the following in e() :
Scalars

```
e(N)
e(k)
e(k_eq)
e(k_eq_model)
e(k_dv)
e(df_m)
e(df)
e(phi)
e(aic)
e(bic)
e(11)
e(N_clust)
e(chi2)
e(p)
e(deviance)
e(deviance_s)
e(deviance_p)
e(deviance_ps)
e(dispers)
e(dispers_s)
e(dispers_p)
e(dispers_ps)
e(nbml)
e(vf)
e(power)
e(rank)
e(ic)
e(rc)
e(converged)
```

Macros
e(cmd)
e(cmdline)
e(depvar)
e(varfunc)
$e$ (varfunct)
$e$ (varfuncf)
e(link)
e(linkt)
e(linkf)
$e(m)$
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(offset)
e(chi2type)
$e$ (cons)
e(hac_kernel)
e(hac_lag)
e(vce)
e(vcetype)
e(opt)
e(opt1)
e(opt2)
$e$ (which)
e(ml_method)
e(user)
e(technique)
number of observations
number of parameters
number of equations in $e(b)$
number of equations in overall model test
number of dependent variables
model degrees of freedom
residual degrees of freedom
scale parameter
model AIC
model BIC
$\log$ likelihood, if NR
number of clusters
$\chi^{2}$
$p$-value for model test
deviance
scaled deviance
Pearson deviance
scaled Pearson deviance
dispersion
scaled dispersion
Pearson dispersion
scaled Pearson dispersion
1 if negative binomial parameter estimated via ML, 0 otherwise
factor set by vfactor(), 1 if not set
power set by link(power \#) or link(opower \#)
rank of e(V)
number of iterations
return code
1 if converged, 0 otherwise
glm
command as typed
name of dependent variable
program to calculate variance function
variance title
variance function
program to calculate link function
link title
link function
number of binomial trials
weight type
weight expression
title in estimation output
name of cluster variable
linear offset variable
Wald; type of model $\chi^{2}$ test
noconstant, if specified
HAC kernel
HAC lag
vcetype specified in vce()
title used to label Std. err.
ml or irls
optimization title, line 1
optimization title, line 2
max or min; whether optimizer is to perform maximization or minimization type of ml method
name of likelihood-evaluator program
maximization technique

```
e(properties) b V
e(predict) program used to implement predict
e(marginsok) predictions allowed by margins
e(marginsnotok) predictions disallowed by margins
e(asbalanced) factor variables fvset as asbalanced
e(asobserved) factor variables fvset as asobserved
e(b)
e(Cns)
e(ilog) iteration log (up to 20 iterations)
e(gradient)
e(V)
e(V_modelbased)
e(sample) marks estimation sample
```

Matrices
Functions

In addition to the above, the following is stored in r() :

```
Matrices
```

    \(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values,
    and confidence intervals
    Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.
glm, irls stores the following in e():
Scalars
e(N)
e(k)
e(k_eq_model)
e(df_m)
e(df)
e(phi)
e(disp)
e(bic)
e(N_clust)
e(deviance)
e(deviance_s)
e(deviance_p)
e(deviance_ps)
e(dispers)
e(dispers_s)
e(dispers_p)
e(dispers_ps)
$e(n b m l)$
e(vf)
e(power)
number of observations
number of parameters
number of equations in overall model test
model degrees of freedom
residual degrees of freedom
scale parameter
dispersion parameter
model BIC
number of clusters
deviance
scaled deviance
Pearson deviance
scaled Pearson deviance
dispersion
scaled dispersion
Pearson dispersion
scaled Pearson dispersion
1 if negative binomial parameter estimated via ML, 0 otherwise
factor set by vfactor(), 1 if not set
power set by link(power \#) or link(opower \#)
rank of e(V)
return code
Macros
e(cmd)
e(cmdline)
e(depvar)
e(varfunc)
e(varfunct)
e(varfuncf)
e(link)
e(linkt)
e(linkf)
$e(m)$

```
e(wtype)
e(wexp)
e(clustvar)
e(offset)
e(cons)
e(hac_kernel)
e(hac_lag)
e(vce)
e(vcetype)
        e(opt)
e(opt1)
e(opt2)
e(properties)
e(predict)
e(marginsok)
e(marginsnotok)
e(asbalanced)
e(asobserved)
Matrices
    e(b)
    e(V)
    e(V_modelbased)
Functions
    e(sample)
```

    weight type
        weight expression
        name of cluster variable
    linear offset variable
noconstant, if specified
HAC kernel
HAC lag
vcetype specified in vce()
title used to label Std. err.
ml or irls
optimization title, line 1
optimization title, line 2
b V
program used to implement predict
predictions allowed by margins
predictions disallowed by margins
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
variance-covariance matrix of the estimators
model-based variance
marks estimation sample

In addition to the above, the following is stored in r() :

```
Matrices
```

    \(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values,
    and confidence intervals
    Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

The canonical reference on GLM is McCullagh and Nelder (1989). The term "generalized linear model" is from Nelder and Wedderburn (1972). Many people use the acronym GLIM for GLM models because of the classic GLM software tool GLIM, by Baker and Nelder (1985). See Dobson and Barnett (2018) for a concise introduction and overview. See Rabe-Hesketh and Everitt (2007) for more examples of GLM using Stata. Hoffmann (2004) focuses on applying generalized linear models, using real-world datasets, along with interpreting computer output, which for the most part is obtained using Stata.

This discussion highlights the details of parameter estimation and predicted statistics. For a more detailed treatment, and for information on variance estimation, see Hardin and Hilbe (2018). glm supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.
glm obtains results by IRLS, as described in McCullagh and Nelder (1989), or by maximum likelihood using Newton-Raphson. The implementation here, however, allows user-specified weights, which we denote as $v_{j}$ for the $j$ th observation. Let $M$ be the number of "observations" ignoring weights. Define

$$
w_{j}= \begin{cases}1 & \text { if no weights are specified } \\ v_{j} & \text { if } \mathrm{f} w e i g h t s \text { or iweights are specified } \\ M v_{j} /\left(\sum_{k} v_{k}\right) & \text { if aweights or pweights are specified }\end{cases}
$$

The number of observations is then $N=\sum_{j} w_{j}$ if fweights are specified and $N=M$ otherwise. Each IRLS step is performed by regress using $w_{j}$ as the weights.
Let $d_{j}^{2}$ denote the squared deviance residual for the $j$ th observation:
For the Gaussian family, $d_{j}^{2}=\left(y_{j}-\widehat{\mu}_{j}\right)^{2}$.
For the Bernoulli family (binomial with denominator 1),

$$
d_{j}^{2}= \begin{cases}-2 \ln \left(1-\widehat{\mu}_{j}\right) & \text { if } y_{j}=0 \\ -2 \ln \left(\widehat{\mu}_{j}\right) & \text { otherwise }\end{cases}
$$

For the binomial family with denominator $m_{j}$,

$$
d_{j}^{2}= \begin{cases}2 y_{j} \ln \left(y_{j} / \widehat{\mu}_{j}\right)+2\left(m_{j}-y_{j}\right) \ln \left\{\left(m_{j}-y_{j}\right) /\left(m_{j}-\widehat{\mu}_{j}\right)\right\} & \text { if } 0<y_{j}<m_{j} \\ 2 m_{j} \ln \left\{m_{j} /\left(m_{j}-\widehat{\mu}_{j}\right)\right\} & \text { if } y_{j}=0 \\ 2 y_{j} \ln \left(y_{j} / \widehat{\mu}_{j}\right) & \text { if } y_{j}=m_{j}\end{cases}
$$

For the Poisson family,

$$
d_{j}^{2}= \begin{cases}2 \widehat{\mu}_{j} & \text { if } y_{j}=0 \\ 2\left\{y_{j} \ln \left(y_{j} / \widehat{\mu}_{j}\right)-\left(y_{j}-\widehat{\mu}_{j}\right)\right\} & \text { otherwise }\end{cases}
$$

For the gamma family, $d_{j}^{2}=-2\left\{\ln \left(y_{j} / \widehat{\mu}_{j}\right)-\left(y_{j}-\widehat{\mu}_{j}\right) / \widehat{\mu}_{j}\right\}$.
For the inverse Gaussian, $d_{j}^{2}=\left(y_{j}-\widehat{\mu}_{j}\right)^{2} /\left(\widehat{\mu}_{j}^{2} y_{j}\right)$.
For the negative binomial,

$$
d_{j}^{2}= \begin{cases}2 \ln \left(1+k \widehat{\mu}_{j}\right) / k & \text { if } y_{j}=0 \\ 2 y_{j} \ln \left(y_{j} / \widehat{\mu}_{j}\right)-2\left\{\left(1+k y_{j}\right) / k\right\} \ln \left\{\left(1+k y_{j}\right) /\left(1+k \widehat{\mu}_{j}\right)\right\} & \text { otherwise }\end{cases}
$$

Let $\phi=1$ if the scale parameter is set to one; otherwise, define $\phi=\widehat{\phi}_{0}(n-k) / n$, where $\widehat{\phi}_{0}$ is the estimated scale parameter and $k$ is the number of covariates in the model (including intercept).

Let $\ln L_{j}$ denote the $\log$ likelihood for the $j$ th observation:
For the Gaussian family,

$$
\ln L_{j}=-\frac{1}{2}\left[\left\{\frac{\left(y_{j}-\widehat{\mu}_{j}\right)^{2}}{\phi}\right\}+\ln (2 \pi \phi)\right]
$$

For the binomial family with denominator $m_{j}$ (Bernoulli if all $m_{j}=1$ ),

$$
\ln L_{j}=\phi \times \begin{cases}\ln \left\{\Gamma\left(m_{j}+1\right)\right\}-\ln \left\{\Gamma\left(y_{j}+1\right)\right\}-\ln \left\{\Gamma\left(m_{j}-y_{j}+1\right)\right\} & \text { if } 0<y_{j}<m_{j} \\ \quad+\left(m_{j}-y_{j}\right) \ln \left(1-\widehat{\mu}_{j} / m_{j}\right)+y_{j} \ln \left(\widehat{\mu}_{j} / m_{j}\right) & \\ m_{j} \ln \left(1-\widehat{\mu}_{j} / m_{j}\right) & \text { if } y_{j}=0 \\ m_{j} \ln \left(\widehat{\mu}_{j} / m_{j}\right) & \text { if } y_{j}=m_{j}\end{cases}
$$

For the Poisson family,

$$
\ln L_{j}=\phi\left[y_{j} \ln \left(\widehat{\mu}_{j}\right)-\widehat{\mu}_{j}-\ln \left\{\Gamma\left(y_{j}+1\right)\right\}\right]
$$

For the gamma family, $\ln L_{j}=-y_{j} / \widehat{\mu}_{j}+\ln \left(1 / \widehat{\mu}_{j}\right)$.
For the inverse Gaussian,

$$
\ln L_{j}=-\frac{1}{2}\left\{\frac{\left(y_{j}-\widehat{\mu}_{j}\right)^{2}}{y_{j} \widehat{\mu}_{j}^{2}}+3 \ln \left(y_{j}\right)+\ln (2 \pi)\right\}
$$

For the negative binomial (let $m=1 / k$ ),

$$
\begin{aligned}
\ln L_{j}= & \phi\left[\ln \left\{\Gamma\left(m+y_{j}\right)\right\}-\ln \left\{\Gamma\left(y_{j}+1\right)\right\}-\ln \{\Gamma(m)\}\right. \\
& \left.-m \ln \left(1+\widehat{\mu}_{j} / m\right)+y_{j} \ln \left\{\widehat{\mu}_{j} /\left(\widehat{\mu}_{j}+m\right)\right\}\right]
\end{aligned}
$$

The overall deviance reported by glm is $D^{2}=\sum_{j} w_{j} d_{j}^{2}$. The dispersion of the deviance is $D^{2}$ divided by the residual degrees of freedom.
The Akaike information criterion (AIC) and Bayesian information criterion (BIC) are given by

$$
\begin{aligned}
& \mathrm{AIC}=\frac{-2 \ln L+2 k}{N} \\
& \mathrm{BIC}=D^{2}-(N-k) \ln (N)
\end{aligned}
$$

where $\ln L=\sum_{j} w_{j} \ln L_{j}$ is the overall $\log$ likelihood.
The Pearson deviance reported by glm is $\sum_{j} w_{j} r_{j}^{2}$. The corresponding Pearson dispersion is the Pearson deviance divided by the residual degrees of freedom. glm also calculates the scaled versions of all of these quantities by dividing by the estimated scale parameter.

## Acknowledgments

glm was written by James Hardin of the Arnold School of Public Health at the University of South Carolina and Joseph Hilbe (1944-2017) of Arizona State University, the coauthors of the Stata Press book Generalized Linear Models and Extensions. The previous version of this routine was written by Patrick Royston of the MRC Clinical Trials Unit, London, and coauthor of the Stata Press book Flexible Parametric Survival Analysis Using Stata: Beyond the Cox Model. The original version of this routine was published in Royston (1994). Royston's work, in turn, was based on a prior implementation by Joseph Hilbe, first published in Hilbe (1993). Roger Newson wrote an early implementation (Newson 1999) of robust variance estimates for GLM. Parts of this entry are excerpts from Hardin and Hilbe (2018).

## References

Akaike, H. 1973. Information theory and an extension of the maximum likelihood principle. In Second International Symposium on Information Theory, ed. B. N. Petrov and F. Csaki, 267-281. Budapest: Akailseoniai-Kiudo.

Anscombe, F. J. 1953. Contribution of discussion paper by H. Hotelling "New light on the correlation coefficient and its transforms". Journal of the Royal Statistical Society, Series B 15: 229-230. https://doi.org/10.1111/j.25176161.1953.tb00136.x.

Baker, R. J., and J. A. Nelder. 1985. The Generalized Linear Interactive Modelling System, Release 3.77. Oxford: Numerical Algorithms Group.
Basu, A. 2005. Extended generalized linear models: Simultaneous estimation of flexible link and variance functions. Stata Journal 5: 501-516.

Berndt, E. K., B. H. Hall, R. E. Hall, and J. A. Hausman. 1974. Estimation and inference in nonlinear structural models. Annals of Economic and Social Measurement 3/4: 653-665.

Cummings, P. 2009. Methods for estimating adjusted risk ratios. Stata Journal 9: 175-196.
Discacciati, A., and M. Bottai. 2017. Instantaneous geometric rates via generalized linear models. Stata Journal 17: 358-371.

Dobson, A. J., and A. G. Barnett. 2018. An Introduction to Generalized Linear Models. 4th ed. Boca Raton, FL: Chapman and Hall/CRC.

Hardin, J. W., and J. M. Hilbe. 2018. Generalized Linear Models and Extensions. 4th ed. College Station, TX: Stata Press.

Hilbe, J. M. 1993. sg16: Generalized linear models. Stata Technical Bulletin 11: 20-28. Reprinted in Stata Technical Bulletin Reprints, vol. 2, pp. 149-159. College Station, TX: Stata Press.
—_. 2009. Logistic Regression Models. Boca Raton, FL: Chapman and Hall/CRC.
_—. 2014. Modeling Count Data. New York: Cambridge University Press.
Hoffmann, J. P. 2004. Generalized Linear Models: An Applied Approach. Boston: Pearson.
Hosmer, D. W., Jr., S. A. Lemeshow, and R. X. Sturdivant. 2013. Applied Logistic Regression. 3rd ed. Hoboken, NJ: Wiley.

McCullagh, P., and J. A. Nelder. 1989. Generalized Linear Models. 2nd ed. London: Chapman and Hall/CRC.
Nelder, J. A. 1975. Robert William MacLagan Wedderburn, 1947-1975. Journal of the Royal Statistical Society, Series A 138: 587.

Nelder, J. A., and R. W. M. Wedderburn. 1972. Generalized linear models. Journal of the Royal Statistical Society, Series A 135: 370-384. https://doi.org/10.2307/2344614.
Newey, W. K., and K. D. West. 1987. A simple, positive semi-definite, heteroskedasticity and autocorrelation consistent covariance matrix. Econometrica 55: 703-708. https://doi.org/10.2307/1913610.

Newson, R. B. 1999. sg114: rglm—Robust variance estimates for generalized linear models. Stata Technical Bulletin 50: 27-33. Reprinted in Stata Technical Bulletin Reprints, vol. 9, pp. 181-190. College Station, TX: Stata Press.
——. 2004. Generalized power calculations for generalized linear models and more. Stata Journal 4: 379-401.
Newson, R. B., and M. Falcaro. 2023. Robit regression in Stata. Stata Journal 23: 658-682.
Orsini, N., R. Bellocco, and P. C. Sjölander. 2013. Doubly robust estimation in generalized linear models. Stata Journal 13: 185-205.

Parner, E. T., and P. K. Andersen. 2010. Regression analysis of censored data using pseudo-observations. Stata Journal 10: 408-422.

Pregibon, D. 1980. Goodness of link tests for generalized linear models. Journal of the Royal Statistical Society, Series C 29: 15-24.
https://doi.org/10.2307/2346405.
Rabe-Hesketh, S., and B. S. Everitt. 2007. A Handbook of Statistical Analyses Using Stata. 4th ed. Boca Raton, FL: Chapman and Hall/CRC.

Rabe-Hesketh, S., A. Skrondal, and A. Pickles. 2002. Reliable estimation of generalized linear mixed models using adaptive quadrature. Stata Journal 2: 1-21.
Raftery, A. E. 1995. Bayesian model selection in social research. In Vol. 25 of Sociological Methodology, ed. P. V. Marsden, 111-163. Oxford: Blackwell.

Rogers, W. H. 1993. sg16.4: Comparison of nbreg and glm for negative binomial. Stata Technical Bulletin 16: 7. Reprinted in Stata Technical Bulletin Reprints, vol. 3, pp. 82-84. College Station, TX: Stata Press.
Royston, P. 1994. sg22: Generalized linear models: Revision of glm. Stata Technical Bulletin 18: 6-11. Reprinted in Stata Technical Bulletin Reprints, vol. 3, pp. 112-121. College Station, TX: Stata Press.

Sasieni, P. D. 2012. Age-period-cohort models in Stata. Stata Journal 12: 45-60.
Schonlau, M. 2005. Boosted regression (boosting): An introductory tutorial and a Stata plugin. Stata Journal 5: 330-354.

Senn, S. J. 2003. A conversation with John Nelder. Statistical Science 18: 118-131.
https://doi.org/10.1214/ss/1056397489.

Williams, R. 2010. Fitting heterogeneous choice models with oglm. Stata Journal 10: 540-567.

## Also see <br> [R] glm postestimation - Postestimation tools for glm <br> [R] cloglog - Complementary log-log regression <br> [ R$]$ logistic - Logistic regression, reporting odds ratios <br> [R] nbreg - Negative binomial regression <br> [R] poisson - Poisson regression <br> [R] regress - Linear regression <br> [BAYES] bayes: glm - Bayesian generalized linear models <br> [FMM] fmm: glm - Finite mixtures of generalized linear regression models <br> [ME] meglm - Multilevel mixed-effects generalized linear models <br> [MI] Estimation - Estimation commands for use with mi estimate <br> [SVY] svy estimation - Estimation commands for survey data <br> [XT] xtgee - GEE population-averaged panel-data models <br> [U] 20 Estimation and postestimation commands

## Title

glm postestimation - Postestimation tools for glm

Postestimation commands
Remarks and examples
Also see
predict
Methods and formulas
margins
References

## Postestimation commands

The following postestimation commands are available after glm:

| Command | Description |
| :---: | :---: |
| contrast | contrasts and ANOVA-style joint tests of estimates |
| *estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estat (svy) | postestimation statistics for survey data |
| estimates | cataloging estimation results |
| etable | table of estimation results |
| $\dagger$ forecast | dynamic forecasts and simulations |
| $\dagger$ hausman | Hausman's specification test |
| lincom | point estimates, standard errors, testing, and inference for linear combinations of coefficients |
| linktest | link test for model specification |
| * $\dagger$ lrtest | likelihood-ratio test |
| margins | marginal means, predictive margins, marginal effects, and average marginal effects |
| marginsplot | graph the results from margins (profile plots, interaction plots, etc.) |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients |
| predict | predictions, residuals, influence statistics, and other diagnostic measures |
| predictnl | point estimates, standard errors, testing, and inference for generalized predictions |
| pwcompare | pairwise comparisons of estimates |
| suest | seemingly unrelated estimation |
| test | Wald tests of simple and composite linear hypotheses |
| testnl | Wald tests of nonlinear hypotheses |

*estat ic and lrtest are not appropriate after glm, irls.
$\dagger_{\text {forecast, hausman, and lrtest }}$ are not appropriate with svy estimation results. forecast is also not appropriate with mi estimation results.

## predict

## Description for predict

predict creates a new variable containing predictions such as expected values, linear predictions, standard errors, residuals, Cook's distance, diagonals of the "hat" matrix, weighted averages, differences between the observed and fitted outcomes, and equation-level scores.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

predict [type] newvar [if] [in] [, statistic options]
statistic Description
Main
mu expected value of $y$; the default
xb linear prediction $\eta=\mathbf{x} \widehat{\boldsymbol{\beta}}$
eta synonym of $x b$
stdp standard error of the linear prediction
anscombe Anscombe (1953) residuals
cooksd
deviance
hat
likelihood
pearson
response
score
working

Cook's distance
deviance residuals
diagonals of the "hat" matrix
a weighted average of standardized deviance and standardized Pearson residuals Pearson residuals
differences between the observed and fitted outcomes
first derivative of the $\log$ likelihood with respect to $\mathbf{x}_{j} \boldsymbol{\beta}$ working residuals

Options
nooffset
adjusted
standardized
studentized
modified
modify calculations to ignore offset variable
adjust deviance residual to speed up convergence
multiply residual by the factor $(1-h)^{-1 / 2}$
multiply residual by one over the square root of the estimated scale parameter modify denominator of residual to be a reasonable estimate of the variance of depvar

These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.
$\mathrm{mu}, \mathrm{xb}$, stdp, and score are the only statistics allowed with svy estimation results.

## Options for predict

## Main

mu , the default, specifies that predict calculate the expected value of $y$, equal to $g^{-1}(\mathbf{x} \widehat{\boldsymbol{\beta}})\left[n g^{-1}(\mathbf{x} \widehat{\boldsymbol{\beta}})\right.$ for the binomial family].
xb calculates the linear prediction $\eta=\mathbf{x} \widehat{\boldsymbol{\beta}}$.
eta is a synonym for $x b$.
stdp calculates the standard error of the linear prediction.
anscombe calculates the Anscombe (1953) residuals to produce residuals that closely follow a normal distribution.
cooksd calculates Cook's distance, which measures the aggregate change in the estimated coefficients when each observation is left out of the estimation.
deviance calculates the deviance residuals. Deviance residuals are recommended by McCullagh and Nelder (1989) and by others as having the best properties for examining the goodness of fit of a GLM. They are approximately normally distributed if the model is correct. They may be plotted against the fitted values or against a covariate to inspect the model's fit. Also see the pearson option below.
hat calculates the diagonals of the "hat" matrix, analogous to linear regression.
likelihood calculates a weighted average of standardized deviance and standardized Pearson residuals. pearson calculates the Pearson residuals. Pearson residuals often have markedly skewed distributions for nonnormal family distributions. Also see the deviance option above.
response calculates the differences between the observed and fitted outcomes.
score calculates the equation-level score, $\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)$.
working calculates the working residuals, which are response residuals weighted according to the derivative of the link function.

Options
nooffset is relevant only if you specified offset (varname) for glm. It modifies the calculations made by predict so that they ignore the offset variable; the linear prediction is treated as $\mathbf{x}_{j} \mathbf{b}$ rather than as $\mathbf{x}_{j} \mathbf{b}+$ offset $_{j}$.
adjusted adjusts the deviance residual to speed up the convergence to the limiting normal distribution. The adjustment deals with adding to the deviance residual a higher-order term that depends on the variance function family. This option is allowed only when deviance is specified.
standardized requests that the residual be multiplied by the factor $(1-h)^{-1 / 2}$, where $h$ is the diagonal of the hat matrix. This operation is done to account for the correlation between depvar and its predicted value.
studentized requests that the residual be multiplied by one over the square root of the estimated scale parameter.
modified requests that the denominator of the residual be modified to be a reasonable estimate of the variance of depvar. The base residual is multiplied by the factor $(k / w)^{-1 / 2}$, where $k$ is either one or the user-specified dispersion parameter and $w$ is the specified weight (or one if left unspecified).

## margins

## Description for margins

margins estimates margins of response for expected values and linear predictions.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...)...] [options]
```

statistic
Description
mu expected value of $y$; the default
$\mathrm{xb} \quad$ linear prediction $\eta=\mathbf{x} \widehat{\boldsymbol{\beta}}$
eta synonym for $x b$
stdp not allowed with margins
anscombe not allowed with margins
cooksd not allowed with margins
deviance not allowed with margins
hat not allowed with margins
likelihood not allowed with margins
pearson not allowed with margins
response not allowed with margins
score not allowed with margins
working not allowed with margins

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

## Remarks and examples

Remarks are presented under the following headings:
Predictions
Other postestimation commands

## Predictions

## > Example 1

After glm estimation, predict may be used to obtain various predictions based on the model. In example 2 of $[\mathrm{R}]$ glm, we mentioned that the complementary $\log -\log$ link seemed to fit the data better than the logit link. Now, we go back and obtain the fitted values and deviance residuals:

```
. use https://www.stata-press.com/data/r18/ldose
. glm r ldose, family(binomial n) link(logit)
(output omitted)
. predict mu_logit
(option mu assumed; predicted mean r)
. predict dr_logit, deviance
. quietly glm r ldose, f(binomial n) l(cloglog)
. predict mu_cl
(option mu assumed; predicted mean r)
. predict dr_cl, d
. format mu_logit dr_logit mu_cl dr_cl %9.5f
. list r mu_logit dr_logit mu_cl dr_cl, sep(4)
```

| $r$ | mu_logit | dr_logit | mu_cl | dr_cl |
| ---: | ---: | ---: | ---: | ---: |
| 6 | 3.45746 | 1.28368 | 5.58945 | 0.18057 |
| 13 | 9.84167 | 1.05969 | 11.28067 | 0.55773 |
| 18 | 22.45139 | -1.19611 | 20.95422 | -0.80330 |
| 28 | 33.89761 | -1.59412 | 30.36942 | -0.63439 |
| 52 | 50.09584 | 0.60614 | 47.77644 | 1.28883 |
| 53 | 53.29092 | -0.12716 | 54.14273 | -0.52366 |
| 61 | 59.22216 | 1.25107 | 61.11331 | -0.11878 |
| 60 | 58.74297 | 1.59398 | 59.94723 | 0.32495 |

In six of the eight cases, $\mid \mathrm{dr} \_$logit $\left|>\left|\mathrm{dr} \_\mathrm{cl}\right|\right.$. The above represents only one of the many available options for predict. See Hardin and Hilbe (2018) for a more in-depth examination.

## Other postestimation commands

- Technical note

After glm estimation, you may perform any of the postestimation commands that you would perform after any other kind of estimation in Stata; see [U] 20 Estimation and postestimation commands. Below, we test the joint significance of all the interaction terms.

```
. use https://www.stata-press.com/data/r18/beetle, clear
. glm r beetle##c.ldose, family(binomial n) link(cloglog)
(output omitted)
. testparm i.beetle beetle#c.ldose
( 1) [r]2.beetle = 0
( 2) [r]3.beetle = 0
( 3) [r]2.beetle#c.ldose = 0
( 4) [r]3.beetle#c.ldose = 0
    chi2( 4) = 249.69
    Prob > chi2 = 0.0000
```

If you wanted to print the variance-covariance matrix of the estimators, you would type estat vce.

If you use the linktest postestimation command, you must also specify the family() and link() options; see [R] linktest.

## Methods and formulas

We follow the terminology used in Methods and formulas of [R] glm.
The deviance residual calculated by predict following glm is $r_{j}^{D}=\operatorname{sign}\left(y_{j}-\widehat{\mu}_{j}\right) \sqrt{d_{j}^{2}}$.
The Pearson residual calculated by predict following glm is

$$
r_{j}^{P}=\frac{y_{j}-\widehat{\mu}_{j}}{\sqrt{V\left(\widehat{\mu}_{j}\right)}}
$$

where $V\left(\widehat{\mu}_{j}\right)$ is the family-specific variance function.

$$
V\left(\widehat{\mu}_{j}\right)= \begin{cases}\widehat{\mu}_{j}\left(1-\widehat{\mu}_{j} / m_{j}\right) & \text { if binomial or Bernoulli }\left(m_{j}=1\right) \\ \widehat{\mu}_{j}^{2} & \text { if gamma } \\ 1 & \text { if Gaussian } \\ \widehat{\mu}_{j}^{3} & \text { if inverse Gaussian } \\ \widehat{\mu}_{j}+k \widehat{\mu}_{j}^{2} & \text { if negative binomial } \\ \widehat{\mu}_{j} & \text { if Poisson }\end{cases}
$$

The response residuals are given by $r_{j}^{R}=y_{j}-\widehat{\mu}_{j}$. The working residuals are

$$
r_{j}^{W}=\left(y_{j}-\widehat{\mu}_{j}\right)\left(\frac{\partial \eta}{\partial \mu}\right)_{j}
$$

and the score residuals are

$$
r_{j}^{S}=\frac{y_{j}-\widehat{\mu}_{j}}{V\left(\widehat{\mu}_{j}\right)}\left(\frac{\partial \eta}{\partial \mu}\right)_{j}^{-1}
$$

Define $\widehat{W}=V(\widehat{\mu})$ and $X$ to be the covariate matrix. $h_{j}$, then, is the $j$ th diagonal of the hat matrix given by

$$
\widehat{H}=\widehat{W}^{1 / 2} X\left(X^{T} \widehat{W} X\right)^{-1} X^{T} \widehat{W}^{1 / 2}
$$

As a result, the likelihood residuals are given by

$$
r_{j}^{L}=\operatorname{sign}\left(y_{j}-\widehat{\mu}_{j}\right)\left\{h_{j}\left(r_{j}^{P \prime}\right)^{2}+\left(1-h_{j}\right)\left(r_{j}^{D \prime}\right)^{2}\right\}^{1 / 2}
$$

where $r_{j}^{P \prime}$ and $r_{j}^{D \prime}$ are the standardized Pearson and standardized deviance residuals, respectively. By standardized, we mean that the residual is divided by $\left\{1-h_{j}\right\}^{1 / 2}$.

Cook's distance is an overall measure of the change in the regression coefficients caused by omitting the $i$ th observation from the analysis. Computationally, Cook's distance is obtained as

$$
C_{j}=\frac{\left(r_{j}^{P \prime}\right)^{2} h_{j}}{k\left(1-h_{j}\right)}
$$

where $k$ is the number of regressors, including the constant.
Anscombe residuals are given by

$$
r_{j}^{A}=\frac{A\left(y_{j}\right)-A\left(\widehat{\mu}_{j}\right)}{A^{\prime}\left(\widehat{\mu}_{j}\right)\left\{V\left(\widehat{\mu}_{j}\right)\right\}^{1 / 2}}
$$

where

$$
A(\cdot)=\int \frac{d \mu}{V^{1 / 3}(\mu)}
$$

Deviance residuals may be adjusted (predict, adjusted) to make the following correction:

$$
r_{j}^{D a}=r_{j}^{D}+\frac{1}{6} \rho_{3}(\theta)
$$

where $\rho_{3}(\theta)$ is a family-specific correction. See Hardin and Hilbe (2018) for the exact forms of $\rho_{3}(\theta)$ for each family.

## References

Anscombe, F. J. 1953. Contribution of discussion paper by H. Hotelling "New light on the correlation coefficient and its transforms". Journal of the Royal Statistical Society, Series B 15: 229-230. https://doi.org/10.1111/j.25176161.1953.tb00136.x.

Hardin, J. W., and J. M. Hilbe. 2018. Generalized Linear Models and Extensions. 4th ed. College Station, TX: Stata Press.
McCullagh, P., and J. A. Nelder. 1989. Generalized Linear Models. 2nd ed. London: Chapman and Hall/CRC.
Newson, R. B. 2013. Attributable and unattributable risks and fractions and other scenario comparisons. Stata Journal 13: 672-698.

## Also see

[ R ] glm - Generalized linear models
$[R]$ regress postestimation - Postestimation tools for regress
[U] 20 Estimation and postestimation commands

## Title

gmm - Generalized method of moments estimation

| Description | Menu | Syntax | Options |
| :--- | :--- | :--- | :--- |
| Remarks and examples | Stored results | Methods and formulas | References |

Menu
Stored results

Syntax
Methods and formulas

## Description

gmm performs generalized method of moments (GMM) estimation. With the interactive version of the command, you enter the residual equation for each moment condition directly into the dialog box or on the command line by using substitutable expressions. The moment-evaluator program version gives you greater flexibility in exchange for increased complexity; with this version, you write a program in an ado-file that calculates the moments based on a vector of parameters passed to it.
gmm can fit both single- and multiple-equation models. It allows moment conditions of the form $E\left\{\mathbf{z}_{i} u_{i}(\beta)\right\}=\mathbf{0}$, where $\mathbf{z}_{i}$ is a vector of instruments, and $u_{i}(\boldsymbol{\beta})$ is an error term, as well as more general moment conditions of the form $E\left\{\mathbf{h}_{i}\left(\mathbf{z}_{i} ; \boldsymbol{\beta}\right)\right\}=\mathbf{0}$. gmm works with cross-sectional, time-series, and longitudinal (panel) data.

## Menu

Statistics $>$ Endogenous covariates $>$ Generalized method of moments estimation

## Syntax

Interactive version
$\operatorname{gmm}\left(\left[\right.\right.$ reqname $\left.\left._{1}:\right] \operatorname{rexp}_{1}\right)\left(\left[\right.\right.$ reqname $\left.\left._{2}:\right] \operatorname{rexp}_{2}\right) \ldots[$ if $][$ in $]$ [weight $][$, options $]$

Moment-evaluator program version

```
gmm moment_prog [if] [in] [weight], { equations(namelist)|nequations(#)}
```


reqname $_{j}$ is the $j$ th residual equation name, $\operatorname{rexp}_{j}$ is the substitutable expression for the $j$ th residual equation, and moment_prog is a moment-evaluator program.
options
Description
Model
derivative( $[$ reqname $\mid \#] /$ name $=\operatorname{dexp}_{j k}$ )
specify derivative of reqname (or \#) with respect to parameter name; can be specified more than once (interactive version only)
*twostep use two-step GMM estimator; the default

* onestep use one-step GMM estimator
*igmm use iterative GMM estimator
variables(varlist) specify variables in model
nocommonesample do not restrict estimation sample to be the same for all equations
Instruments
instruments ([reqlist: $]$ varlist $[$, noconstant $]$ )
specify instruments; can be specified more than once
xtinstruments ([reqlist: $]$ varlist, lags $\left(\#_{1} / \#_{2}\right)$ )
specify panel-style instruments; can be specified more than once
Weight matrix
wmatrix (wmtype [, independent])
specify weight matrix; wmtype may be robust, cluster clustvar, hac kernel [lags], or unadjusted
center center moments in weight-matrix computation
winitial (iwtype [, independent])
specify initial weight matrix; iwtype may be unadjusted, identity, xt xtspec, or the name of a Stata matrix

SE/Robust
vce (vcetype [, independent])
vcetype may be robust, cluster clustvar, bootstrap, jackknife, hac kernel lags, or unadjusted
quickderivatives

Reporting
level(\#)
title (string)
title2 (string)
display_options
set confidence level; default is level (95)
display string as title above the table of parameter estimates display string as subtitle
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

Optimization
from(initial_values)
$\ddagger$ igmmiterate(\#)
$\ddagger \overline{\text { igmmeps }}$ (\#)
$\ddagger_{\text {igmmweps }}$ (\#)
optimization_options
coeflegend
specify initial values for parameters
specify maximum number of iterations for iterated GMM estimator
specify \# for iterated GMM parameter convergence criterion; default is igmmeps (1e-6)
specify \# for iterated GMM weight-matrix convergence criterion; default is igmmweps (1e-6)
control the optimization process; seldom used
display legend instead of statistics
*You can specify at most one of these options.
$\ddagger$ These options may be specified only when igmm is specified.
program_options
Description
Model
evaluator_options
*hasderivatives
*haslfderivatives
$\dagger$ equations (namelist)
$\dagger$ nequations (\#)
$\ddagger$ parameters (namelist)
$\ddagger$ nparameters (\#)
additional options to be passed to the moment-evaluator program moment-evaluator program can calculate parameter-level derivatives moment-evaluator program can calculate linear-form derivatives specify residual equation names
specify number of residual equations
specify parameter names
specify number of parameters

[^30]$\operatorname{rexp}_{j}$ and $\operatorname{dexp}_{j k}$ may contain factor variables and time-series operators; see [U] 11.4.3 Factor variables and [U] 11.4.4 Time-series varlists.
bootstrap, by, collect, jackknife, rolling, and statsby are allowed; see [U] 11.1.10 Prefix commands.
Weights are not allowed with the bootstrap prefix; see $[R]$ bootstrap.
aweights are not allowed with the jackknife prefix; see $[R]$ jackknife.
aweights, fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
coeflegend does not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.
$\operatorname{rexp}_{j}$ and $\operatorname{dexp}_{j k}$ are substitutable expressions, that is, Stata expressions that also contain parameters to be estimated. The parameters are enclosed in curly braces and must satisfy the naming requirements for variables; \{beta\} is an example of a parameter. The notation \{lcname:varlist\} is allowed for linear combinations of multiple covariates and their parameters. For example, $\{x b$ : mpg price turn _cons\} defines a linear combination of the variables mpg, price, turn, and _cons (the constant term). See Substitutable expressions under Remarks and examples below.

## Options

## Model

derivative ( $[$ reqname $\mid \#] /$ name $=\operatorname{dexp}_{j k}$ ) specifies the derivative of residual equation reqname or \# with respect to parameter name. If reqname or \# is not specified, gmm assumes that the derivative applies to the first residual equation.
For a moment condition of the form $E\left\{\mathbf{z}_{j i} u_{j i}(\boldsymbol{\beta})\right\}=\mathbf{0}$, derivative $\left(j / \beta_{k}=\operatorname{dexp}{ }_{j k}\right)$ is to contain a substitutable expression for $\partial u_{j i} / \partial \beta_{k}$. If you specified m as the reqname, then for a moment condition of the form $E\left\{\mathbf{z}_{m i} u_{m i}(\boldsymbol{\beta})\right\}=\mathbf{0}$, you can specify derivative ( $\mathrm{m} / \beta_{k}=$ $\operatorname{dexp}_{m k}$ ), where $m$ is the index of $m$.
$\operatorname{dexp}_{j k}$ uses the same substitutable expression syntax as is used to specify residual equations. If you declare a linear combination in a residual equation, you provide the derivative for the linear combination; gmm then applies the chain rule for you. See Specifying derivatives under Remarks and examples below for examples.
If you do not specify the derivative() option, gmm calculates derivatives numerically. You must either specify no derivatives or specify a derivative for each of the $k$ parameters that appears in each of the $j$ residual equations unless the derivative is identically zero. You cannot specify some analytic derivatives and have gmm compute the rest numerically.
twostep, onestep, and igmm specify which estimator is to be used. You can specify at most one of these options. twostep is the default.
twostep requests the two-step GMM estimator. gmm obtains parameter estimates based on the initial weight matrix, computes a new weight matrix based on those estimates, and then reestimates the parameters based on that weight matrix.
onestep requests the one-step GMM estimator. The parameters are estimated based on an initial weight matrix, and no updating of the weight matrix is performed except when calculating the appropriate variance-covariance (VCE) matrix.
igmm requests the iterative GMM estimator. gmm obtains parameter estimates based on the initial weight matrix, computes a new weight matrix based on those estimates, reestimates the parameters based on that weight matrix, computes a new weight matrix, and so on, to convergence. Convergence is declared when the relative change in the parameter vector is less than igmmeps(), the relative change in the weight matrix is less than igmmweps(), or igmmiterate() iterations have been
completed. Hall (2005, sec. 2.4 and 3.6) mentions that there may be gains to finite-sample efficiency from using the iterative estimator.
variables (varlist) specifies the variables in the model. gmm ignores observations for which any of these variables has a missing value. If you do not specify variables (), then gmm assumes all the observations are valid and issues an error message if any residual equations evaluate to missing for any observations at the initial value of the parameter vector.
nocommonesample requests that gmm not restrict the estimation sample to be the same for all equations. By default, gmm will restrict the estimation sample to observations that are available for all equations in the model, mirroring the behavior of other multiple-equation estimators such as nlsur, sureg, or reg3. For certain models, however, different equations can have different numbers of observations. For these models, you should specify nocommonesample. See Dynamic panel-data models below for one application of this option. You cannot specify weights if you specify nocommonesample.

Instruments
instruments ([reqlist: $]$ varlist $[$, noconstant $]$ ) specifies a list of instrumental variables to be used. If you specify a single residual equation, then you do not need to specify the equations to which the instruments apply; you can omit the reqlist and simply specify instruments (varlist). By default, a constant term is included in varlist; to omit the constant term, use the noconstant suboption: instruments(varlist, noconstant).

If your model has multiple moment conditions of the form

$$
E\left\{\begin{array}{c}
\mathbf{z}_{1 i} u_{1 i}(\boldsymbol{\beta}) \\
\cdots \\
\mathbf{z}_{q i} u_{q i}(\boldsymbol{\beta})
\end{array}\right\}=\mathbf{0}
$$

then you can specify multiple corresponding residual equations. Then, specify the reqname or an reqlist to indicate the residual equations for which the list of variables is to be used as instruments if you do not want that list applied to all the residual equations. For example, you might type

```
gmm (main: rexp 1) (rexp 2) (rexp ), instruments(z1 z2) ///
    instruments(2: z3) instruments(main 3: z4)
```

Variables $z 1$ and $z 2$ will be used as instruments for all three equations, $z 3$ will be used as an instrument for the second equation, and $z 4$ will be used as an instrument for the first and third equations. Notice that we chose to supply a name for the first residual equation but not the second two, identifying each by its equation number.
varlist may contain factor variables and time-series operators; see [U] 11.4.3 Factor variables and [U] 11.4.4 Time-series varlists, respectively.
xtinstruments ([reqlist: $]$ varlist, lags $\left(\#_{1} / \#_{2}\right)$ ) is for use with panel-data models in which the set of available instruments depends on the time period. As with instruments(), you can prefix the list of variables with residual equation names or numbers to target instruments to specific equations. Unlike with instruments(), a constant term is not included in varlist. You must xtset your data before using this option; see [XT] xtset.
If you specify
gmm ..., xtinstruments( $\mathrm{x}, \operatorname{lags}(1 /).) \ldots$
then for panel $i$ and period $t$, gmm uses $x_{i, t-1}, x_{i, t-2}, \ldots, x_{i 1}$ as instruments. More generally, specifying xtinstruments ( $\mathrm{x}, \operatorname{lags}\left(\#_{1}, \#_{2}\right)$ ) uses $x_{i, t-\#_{1}}, \ldots, x_{i, t-\#_{2}}$ as instruments; setting $\#_{2}=$. requests all available lags. $\#_{1}$ and $\#_{2}$ must be zero or positive integers.
gmm automatically excludes observations for which no valid instruments are available. It does, however, include observations for which only a subset of the lags is available. For example, if you request that lags one through three be used, then gmm will include the observations for the second and third time periods even though fewer than three lags are available as instruments.

## Weight matrix

wmatrix (wmtype [, independent]) specifies the type of weight matrix to be used in conjunction with the two-step and iterated GMM estimators.
Specifying wmatrix (robust) requests a weight matrix that is appropriate when the errors are independent but not necessarily identically distributed. wmatrix (robust) is the default.

Specifying wmatrix(cluster clustvar) requests a weight matrix that accounts for arbitrary correlation among observations within clusters identified by clustvar.

Specifying wmatrix(hac kernel \#) requests a heteroskedasticity- and autocorrelation-consistent (HAC) weight matrix using the specified kernel (see below) with \# lags. The bandwidth of a kernel is equal to the number of lags plus one.

Specifying wmatrix (hac kernel opt [\#]) requests an HAC weight matrix using the specified kernel, and the lag order is selected using Newey and West's (1994) optimal lag-selection algorithm. \# is an optional tuning parameter that affects the lag order selected; see the discussion in Methods and formulas.

Specifying wmatrix (hac kernel) requests an HAC weight matrix using the specified kernel and $N-2$ lags, where $N$ is the sample size.

There are three kernels available for HAC weight matrices, and you can request each one by using the name used by statisticians or the name perhaps more familiar to economists:
bartlett or nwest requests the Bartlett (Newey-West) kernel;
parzen or gallant requests the Parzen (Gallant) kernel; and
quadraticspectral or andrews requests the quadratic spectral (Andrews) kernel.
Specifying wmatrix (unadjusted) requests a weight matrix that is suitable when the errors are homoskedastic. In some applications, the GMM estimator so constructed is known as the (nonlinear) two-stage least-squares (2SLS) estimator.

Including the independent suboption creates a weight matrix that assumes moment conditions are independent. This suboption is often used to replicate other models that can be motivated outside the GMM framework, such as the estimation of a system of equations by system-wide 2SLS. This suboption has no effect if only one residual equation is specified.
wmatrix() has no effect if onestep is also specified.
center requests that the sample moments be centered (demeaned) when computing GMM weight matrices. By default, centering is not done.
winitial (iwtype [, independent]) specifies the weight matrix to use to obtain the first-step parameter estimates.

Specifying winitial (unadjusted) requests a weight matrix that assumes the moment conditions are independent and identically distributed. This matrix is of the form $\left(\mathbf{Z}^{\prime} \mathbf{Z}\right)^{-1}$, where $\mathbf{Z}$ represents all the instruments specified in the instruments () option. To avoid a singular weight matrix, you should specify at least $q-1$ moment conditions of the form $E\left\{\mathbf{z}_{h i} u_{h i}(\boldsymbol{\beta})\right\}=\mathbf{0}$, where $q$ is the number of moment conditions, or you should specify the independent suboption.

Including the independent suboption creates a weight matrix that assumes moment conditions are independent. Elements of the weight matrix corresponding to covariances between two moment conditions are set equal to zero. This suboption has no effect if only one residual equation is specified.
winitial(unadjusted) is the default.
winitial (identity) requests that the identity matrix be used.
winitial (xt xtspec) is for use with dynamic panel-data models in which one of the residual equations is specified in first-differences form. xtspec is a string consisting of the letters "L" and "D", the length of which is equal to the number of residual equations in the model. You specify "L" for a residual equation if that residual equation is written in levels, and you specify "D" for a residual equation if it is written in first differences; xtspec is not case sensitive. When you specify this option, you can specify at most one residual equation in levels and one residual equation in first differences. See the examples listed in Dynamic panel-data models under Remarks and examples below.
winitial (matname) requests that Stata matrix matname be used. You cannot specify the independent suboption if you specify winitial(matname).

## SE/Robust

vce (vcetype [, independent]) specifies the type of standard error reported, which includes types that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
vce (unadjusted) specifies that an unadjusted (nonrobust) VCE matrix be used; this, along with the twostep option, results in the "optimal two-step GMM" estimates often discussed in textbooks.
The default vcetype is based on the wmtype specified in the wmatrix() option. If wmatrix () is specified but vce() is not, then vcetype is set equal to wmtype. To override this behavior and obtain an unadjusted (nonrobust) VCE matrix, specify vce(unadjusted).

Specifying vce(bootstrap) or vce(jackknife) results in standard errors based on the bootstrap or jackknife, respectively. See $[R]$ vec_option, $[R]$ bootstrap, and $[R]$ jackknife for more information on these VCEs.

The syntax for vcetypes other than bootstrap and jackknife is identical to those for wmatrix(). quickderivatives requests that an alternative method be used to compute the numerical derivatives for the VCE. This option has no effect if you specify the derivatives(), hasderivatives, or haslfderivatives option.

The VCE depends on a matrix of partial derivatives that gmm must compute numerically unless you supply analytic derivatives. This Jacobian matrix will be especially large if your model has many instruments, residual equations, or parameters.
By default, gmm computes each element of the Jacobian matrix individually, searching for an optimal step size each time. Although this procedure results in accurate derivatives, it is computationally taxing: gmm may have to evaluate the moments of your model five or more times for each element of the Jacobian matrix.

When you specify the quickderivatives option, gmm computes all derivatives corresponding to a parameter at once, using a fixed step size proportional to the parameter's value. This method requires just two evaluations of the model's moments to compute an entire column of the Jacobian matrix and therefore has the most impact when you specify many instruments or residual equations.

Most of the time, the two methods produce virtually identical results, but the quickderivatives method may fail if a residual equation is highly nonlinear or if instruments differ by orders of magnitude. In the rare case where you specify quickderivatives and obtain suspiciously large or small standard errors, try refitting your model without this option.

## Reporting

level (\#); see [R] Estimation options.
title(string) specifies an optional title that will be displayed just above the table of parameter estimates.
title2 (string) specifies an optional subtitle that will be displayed between the title specified in title() and the table of parameter estimates. If title2() is specified but title() is not, title2() has the same effect as title().
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.
$\sqrt{\text { Optimization }}$
from(initial_values) specifies the initial values to begin the estimation. You can specify a parameter name, its initial value, another parameter name, its initial value, and so on, or you can specify a $1 \times k$ matrix, where $k$ is the number of parameters in the model. For example, to initialize alpha to 1.23 and delta to 4.57 , you would type
gmm ..., from(alpha 1.23 delta 4.57) ...
or equivalently

```
matrix define initval = (1.23, 4.57)
gmm ..., from(initval) ...
```

Initial values declared in the from() option override any that are declared within substitutable expressions. If you specify a parameter that does not appear in your model, gmm exits with an error message. If you specify a matrix, the values must be in the same order in which the parameters are declared in your model.
igmmiterate(\#), igmmeps(\#), and igmmweps(\#) control the iterative process for the iterative GMM estimator. These options can be specified only if you also specify igmm.
igmmiterate(\#) specifies the maximum number of iterations to perform with the iterative GMM estimator. The default is the number set using set maxiter, which is 300 by default.
igmmeps (\#) specifies the convergence criterion used for successive parameter estimates when the iterative GMM estimator is used. The default is igmmeps ( $1 \mathrm{e}-6$ ). Convergence is declared when the relative difference between successive parameter estimates is less than igmmeps () and the relative difference between successive estimates of the weight matrix is less than igmmweps().
igmmweps (\#) specifies the convergence criterion used for successive estimates of the weight matrix when the iterative GMM estimator is used. The default is igmmweps ( $1 \mathrm{e}-6$ ). Convergence is declared when the relative difference between successive parameter estimates is less than igmmeps () and the relative difference between successive estimates of the weight matrix is less than igmmweps ().
optimization_options: technique(), conv_maxiter(), conv_ptol(), conv_vtol(),
conv_nrtol(), tracelevel(). technique() specifies the optimization technique to use; gn
(the default), nr , dfp , and bfgs are allowed. conv_maxiter() specifies the maximum number
of iterations; conv_ptol(), conv_vtol(), and conv_nrtol() specify the convergence criteria for the parameters, gradient, and scaled Hessian, respectively. tracelevel () allows you to obtain additional details during the iterative process. See [M-5] optimize().

The following options pertain only to the moment-evaluator program version of gmm.
$\qquad$ Model
evaluator_options refer to any options allowed by your moment_prog.
hasderivatives and haslfderivatives indicate that you have written your moment-evaluator program to compute derivatives. You may specify one or the other but not both. If you do not specify either of these options, gmm computes the derivatives numerically.
hasderivatives indicates that your moment-evaluator program computes parameter-level derivatives.
haslfderivatives indicates that your moment-evaluator program computes equation-level derivatives and is useful only when you specify the parameters of your model using the \{lcname: varlist $\}$ syntax of the parameters () option.
See Details of moment-evaluator programs below for more information.
equations (namelist) specifies the names of the residual equations in the model. If you specify both equations() and nequations(), the number of names in the former must match the number specified in the latter.
nequations(\#) specifies the number of residual equations in the model. If you do not specify names with the equations() option, gmm numbers the residual equations $1,2,3, \ldots$. If you specify both equations() and nequations(), the number of names in the former must match the number specified in the latter.
parameters (namelist) specifies the names of the parameters in the model. The names of the parameters must comply with the naming conventions of Stata's variables; see [U] 11.3 Naming conventions.

Alternatively, you can use parameter equation notation to specify linear combinations of parameters. Each linear combination is of the form \{lcname:varlist\}, where varlist is one or more variable names. Specify the system variable _cons in varlist to include a constant term. Distinguish between \{lcname: varlist\}, in which lcname identifies the linear combination, and (reqname:rexp), in which reqname identifies the residual equation. When you use linear-combination syntax, gmm prepends each element of the parameter vector passed to your evaluator program with lcname: to generate unique names.

If you specify both parameters() and nparameters(), the number of names in the former must match the number specified in the latter.
nparameters (\#) specifies the number of parameters in the model. If you do not specify names with the parameters() option, gmm names them b1, b2, ..., b\#. If you specify both parameters() and nparameters(), the number of names in the former must match the number specified in the latter.

The following option is available with gmm but is not shown in the dialog box:
coeflegend; see [R] Estimation options.

## Remarks and examples

Remarks are presented under the following headings:

Introduction<br>Substitutable expressions<br>The weight matrix and two-step estimation<br>Obtaining standard errors<br>Factor-variable coefficients in multiple residual functions<br>Parameter interpretation using margins<br>Exponential (Poisson) regression models<br>Specifying derivatives<br>Exponential regression models with panel data<br>Rational-expectations models<br>System estimators<br>Dynamic panel-data models<br>Details of moment-evaluator programs

## Introduction

The GMM estimator is a workhorse of modern econometrics and is discussed in all the leading textbooks, including Cameron and Trivedi (2005, 2022), Davidson and MacKinnon (1993), Greene (2018, 500-534), Ruud (2000), Hayashi (2000), Wooldridge (2010), Hamilton (1994), and Baum (2006). An excellent treatise on GMM with a focus on time-series applications is Hall (2005). The collection of papers by Mátyás (1999) provides both theoretical and applied aspects of GMM. Here we give a brief introduction to the methodology and emphasize how the various options of gmm are used.

The starting point for the GMM estimator is the analogy principle, which says we can estimate a parameter by replacing a population moment condition with its sample analogue. For example, the mean of an independent and identically distributed (i.i.d.) population is defined as the value $\mu$ such that the first (central) population moment is zero; that is, $\mu$ solves $E(y-\mu)=0$, where $y$ is a random draw from the population. The analogy principle tells us that to obtain an estimate, $\widehat{\mu}$, of $\mu$, we replace the population-expectations operator with its sample analogue (Manski 1988; Wooldridge 2010),

$$
E(y-\mu)=0 \longrightarrow \frac{1}{N} \sum_{i=1}^{N}\left(y_{i}-\widehat{\mu}\right)=0 \longrightarrow \widehat{\mu}=\frac{1}{N} \sum_{i=1}^{N} y_{i}
$$

where $N$ denotes sample size, and $y_{i}$ represents the $i$ th observation of $y$ in our dataset. The estimator $\widehat{\mu}$ is known as the method of moments (MM) estimator because we started with a population moment condition and then applied the analogy principle to obtain an estimator that depends on the observed data.

Ordinary least-squares (OLS) regression can also be viewed as an MM estimator. In the model

$$
y=\mathbf{x}^{\prime} \boldsymbol{\beta}+u
$$

we assume that $u$ has mean zero conditional on $\mathbf{x}: E(u \mid \mathbf{x})=0$. This conditional expectation implies the unconditional expectation $E(\mathbf{x} u)=\mathbf{0}$ because, with the law of iterated expectations,

$$
E(\mathbf{x} u)=E_{\mathbf{x}}\{E(\mathbf{x} u \mid \mathbf{x})\}=E_{\mathbf{x}}\{\mathbf{x} E(u \mid \mathbf{x})\}=\mathbf{0}
$$

(Using the law of iterated expectations to derive unconditional expectations based on conditional expectations, perhaps motivated by subject theory, is extremely common in GMM estimation.) Continuing, we see that

$$
E(\mathbf{x} u)=E\left\{\mathbf{x}\left(y-\mathbf{x}^{\prime} \boldsymbol{\beta}\right)\right\}=\mathbf{0}
$$

Applying the analogy principle, we obtain

$$
E\left\{\mathbf{x}\left(y-\mathbf{x}^{\prime} \beta\right)\right\} \longrightarrow \frac{1}{N} \sum_{i=1}^{N} \mathbf{x}_{i}\left(y_{i}-\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\right)=\mathbf{0}
$$

so that

$$
\widehat{\boldsymbol{\beta}}=\left(\sum_{i} \mathbf{x}_{i} \mathbf{x}_{i}^{\prime}\right)^{-1} \sum_{i} \mathbf{x}_{i} y_{i}
$$

which is just the more familiar formula $\widehat{\boldsymbol{\beta}}=\left(\mathbf{X}^{\prime} \mathbf{X}\right)^{-1} \mathbf{X}^{\prime} \mathbf{y}$ written with summation notation.
In both of the previous examples, the number of parameters we were estimating equaled the number of moment conditions. In the first example, we estimated one parameter, $\mu$, and had one moment condition $E(y-\mu)=0$. In the second example, the parameter vector $\boldsymbol{\beta}$ had $k$ elements, as did the vector of regressors $\mathbf{x}$, yielding $k$ moment conditions. Ignoring peculiar cases, we see that a model of $m$ equations in $m$ unknowns has a unique solution; and because the residual equations in these examples were linear, we could solve for the parameters analytically. If the moment conditions had been nonlinear, we would have had to use numerical techniques to solve for the parameters, but that is not a significant limitation with modern computers.

What if we have more moment conditions than parameters? Say we have $q$ moment conditions and $k$ parameters. A model of $q>k$ equations in $k$ unknowns does not have a unique solution. Any size- $k$ subset of the moment conditions would yield a consistent parameter estimate, though the parameter estimate would in general be different depending on which $k$ moment conditions we used.

For concreteness, let's return to our regression model,

$$
y=\mathbf{x}^{\prime} \boldsymbol{\beta}+u
$$

Now, however, we no longer wish to assume that $E(\mathbf{x} u)=\mathbf{0}$; we suspect that the error term $u$ affects one or more elements of $\mathbf{x}$. Thus, we can no longer use the OLS estimator. Suppose we have a vector $\mathbf{z}$ with the properties that $E(\mathbf{z} u)=\mathbf{0}$, that the rank of $E\left(\mathbf{z}^{\prime} \mathbf{z}\right)$ equals $q$, and that the rank of $E\left(\mathbf{z}^{\prime} \mathbf{x}\right)=k$. The first assumption simply states that $\mathbf{z}$ is not correlated with the error term. The second assumption rules out perfect collinearity among the elements of $\mathbf{z}$. The third assumption, known as the rank condition in econometrics, ensures that $\mathbf{z}$ is sufficiently correlated with $\mathbf{x}$ and that the estimator is feasible. If some elements of $\mathbf{x}$ are not correlated with $u$, then they should also appear in $\mathbf{z}$.

If $q<k$, then the rank of $E\left(\mathbf{z}^{\prime} \mathbf{x}\right)<k$, which violates the rank condition.
If $q=k$, then we can use the simpler MM estimator we already discussed; we would obtain what is sometimes called the simple instrumental-variables estimator $\widehat{\boldsymbol{\beta}}=\left(\sum_{i} \mathbf{z}_{i} \mathbf{x}_{i}^{\prime}\right)^{-1} \sum_{i} \mathbf{z}_{i} y_{i}$. The rank condition ensures that $\sum_{i} \mathbf{z}_{i} \mathbf{x}_{i}^{\prime}$ is invertible, at least in the population.

If $q>k$, the GMM estimator chooses the value, $\widehat{\boldsymbol{\beta}}$, that minimizes a quadratic function of the moment conditions. We could define

$$
\widehat{\boldsymbol{\beta}} \equiv \arg \min _{\boldsymbol{\beta}}\left\{\frac{1}{N} \sum_{i} \mathbf{z}_{i} u_{i}(\boldsymbol{\beta})\right\}^{\prime}\left\{\frac{1}{N} \sum_{i} \mathbf{z}_{i} u_{i}(\boldsymbol{\beta})\right\}
$$

where for our linear regression example $u_{i}(\boldsymbol{\beta})=y_{i}-\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}$. This estimator tries to make the moment conditions as close to zero as possible. This simple estimator, however, applies equal weight to each of the moment conditions; and as we will see later, we can obtain more efficient estimators by choosing to weight some moment conditions more highly than others.

Consider the quadratic function

$$
Q(\boldsymbol{\beta})=\left\{\frac{1}{N} \sum_{i} \mathbf{z}_{i} u_{i}(\boldsymbol{\beta})\right\}^{\prime} \mathbf{W}\left\{\frac{1}{N} \sum_{i} \mathbf{z}_{i} u_{i}(\boldsymbol{\beta})\right\}
$$

where $\mathbf{W}$ is a symmetric positive-definite matrix known as a weight matrix. Then we define the GMM estimator as

$$
\begin{equation*}
\widehat{\boldsymbol{\beta}} \equiv \arg \min _{\boldsymbol{\beta}} Q(\boldsymbol{\beta}) \tag{1}
\end{equation*}
$$

Continuing with our regression model example, if we choose

$$
\mathbf{W}=\left(\frac{1}{N} \sum_{i} \mathbf{z}_{i} \mathbf{z}_{i}^{\prime}\right)^{-1}
$$

then we obtain

$$
\begin{gathered}
\widehat{\boldsymbol{\beta}}=\left\{\left(\frac{1}{N} \sum_{i} \mathbf{x}_{i} \mathbf{z}_{i}^{\prime}\right)\left(\frac{1}{N} \sum_{i} \mathbf{z}_{i} \mathbf{z}_{i}^{\prime}\right)^{-1}\left(\frac{1}{N} \sum_{i} \mathbf{z}_{i} \mathbf{x}_{i}^{\prime}\right)\right\}^{-1} \times \\
\\
\left(\frac{1}{N} \sum_{i} \mathbf{x}_{i} \mathbf{z}_{i}^{\prime}\right)\left(\frac{1}{N} \sum_{i} \mathbf{z}_{i} \mathbf{z}_{i}^{\prime}\right)^{-1}\left(\frac{1}{N} \sum_{i} \mathbf{z}_{i} y_{i}\right)
\end{gathered}
$$

which is the well-known two-stage least-squares (2SLS) estimator. Our choice of weight matrix here was based on the assumption that $u$ was homoskedastic. A feature of GMM estimation is that by selecting different weight matrices, we can obtain estimators that can tolerate heteroskedasticity, clustering, autocorrelation, and other features of $u$. See [R] ivregress for more information about the 2SLS and linear GMM estimators.

Returning to the case where the model is "just identified", meaning that $q=k$, if we apply the GMM estimator, we will obtain the same estimate, $\widehat{\boldsymbol{\beta}}$, regardless of our choice of $\mathbf{W}$. Because $q=k$, if a unique solution exists, it will set all the sample moment conditions jointly to zero, so $\mathbf{W}$ has no impact on the value of $\boldsymbol{\beta}$ that minimizes the objective function.

We will highlight other features of the GMM estimator and the gmm command as we proceed through examples. First, though, we discuss how to specify moment conditions by using substitutable expressions.

## Substitutable expressions

To use the interactive version of gmm, you define the moment conditions by using substitutable expressions. Your moment conditions are of the form $E\left\{z_{i}^{\prime} u_{i}(\boldsymbol{\beta})\right\}=\mathbf{0}$, where $u_{i}(\boldsymbol{\beta})$ is a residual expression that depends on the parameter vector $\boldsymbol{\beta}$ as well as variables in your dataset, though we suppress expressing the variables for notational simplicity.
gmm requires you to write a substitutable expression for $u_{i}(\boldsymbol{\beta})$. This substitutable expression is the right-hand side of the model written in terms of $u$, or in the language of Stata syntax, a "residual equation". For example, suppose you want to fit the function $y=f(\mathbf{x} ; \boldsymbol{\beta})+u$. In this example, $u_{i}(\boldsymbol{\beta})=u=y-f(\mathbf{x} ; \boldsymbol{\beta})$, so you would type
gmm (y - expression for $f(\mathbf{x} ; \boldsymbol{\beta})), \ldots$
Note that we are not restricted to models with additive error terms.

In general, there are three rules to follow when defining substitutable expressions:

1. Parameters of the model are bound in curly braces: \{b0\}, \{param\}, etc. Parameter names must follow the same conventions as variable names. See [U] 11.3 Naming conventions.
2. Initial values for parameters are given by including an equal sign and the initial value inside the curly braces: $\{b 0=1\}$, $\{$ param $=3.571\}$, etc.
You can also specify initial values by using the from() option. Initial values specified in from () override whatever initial values are given within the substitutable expression. If you do not specify an initial value for a parameter, it is initialized to 0 .
3. Linear combinations of variables can be included using the notation \{lcname: varlist $\}$ : $\{\mathrm{xb}$ : mpg price weight\}, \{score: wxz\}, etc. Parameters of linear combinations are initialized to 0 .

Substitutable expressions may use any mathematical expression involving scalars and variables. See [U] 13.2 Operators and [U] 13.3 Functions for more information on expressions.

The notation $\{\mathrm{xb}: \mathrm{x} 1 \mathrm{x} 2 \mathrm{x} 3\}$ tells gmm that you want a linear combination of the variables x 1 , x 2 , and x 3 . We named this linear combination xb , so gmm will name the three parameters $\mathrm{xb}: \mathrm{x} 1$, $\mathrm{xb}: \mathrm{x} 2$, and $\mathrm{xb}: \mathrm{x} 3$, which corresponds to the three variables x 1 , x 2 , and x 3 in the xb equation. Specify _cons to include a constant term in a linear combination. Factor variables and time-series operators are allowed; see [U] 11.4.3 Factor variables and [U] 11.4.4 Time-series varlists.

Once we have declared the variables in the linear combination xb , we can refer to the linear combination in our substitutable expression by using the notation xb :. The colon is not optional; it tells gmm that you are referring to a previously declared linear combination, not an individual parameter. This shorthand notation is also handy when specifying derivatives, as we will show later.
> Example 1: OLS regression
In Introduction, we stated that OLS is an MM estimator. Say that we want to fit the model

$$
\mathrm{mpg}=\beta_{0}+\beta_{1} \text { weight }+\beta_{2} \text { length }+u
$$

where $u$ is an i.i.d. error term. Recall that the moment condition for ols regression is $E(\mathbf{x} u)=\mathbf{0}$, where $\mathbf{x}$, the list of instruments, is the same as the list of regressors in the model. Writing this in the form required for a gmm substitutable expression, we have

$$
u=\mathrm{mpg}-\beta_{0}-\beta_{1} \text { weight }-\beta_{2} \text { length }
$$

The right-hand side of the equation is the substitutable expression that we will provide to gmm. We give $\beta_{0}, \beta_{1}$, and $\beta_{2}$ the parameter names b 0 , b 1 , and b 2 and enclose them in curly braces. Because linear combinations declared in substitutable expressions do not include a constant term by default, we include our own (b0). We specify the regressors, weight and length, with their respective parameters and also in the instruments() option. gmm includes a constant term in the instrument list by default, so we do not need to add an additional term there. Thus, our command is

| (1978 automob <br> - gmm (mpg - | e data) <br> 1\}*weight | $\{b 2\} * \text { lengt }$ | $-\{b 0\})$ | instruments (weight length) |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Step 1 |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
| Iteration 1: GMM criterion $\mathrm{Q}(\mathrm{b})=2$ |  |  | 96e-20 |  |  |  |
| Iteration 2: | GMM criterio | $Q(b)=3$ | 29e-27 |  |  |  |
| Step 2 |  |  |  |  |  |  |
| Iteration 0: GMM criterion $\mathrm{Q}(\mathrm{b})=5$ |  |  | 09e-28 |  |  |  |
| Iteration 1: GMM criterion Q(b) = 7.237e-32 |  |  |  |  |  |  |
| note: model is exactly identified. |  |  |  |  |  |  |
| GMM estimation |  |  |  |  |  |  |
| Number of parameters $=3$ |  |  |  |  |  |  |
| Number of moments $=3$ |  |  |  |  |  |  |
| Initial weight matrix: Unadjusted |  |  |  | Numb | of obs | 74 |
| GMM weight matrix: Robust |  |  |  |  |  |  |
|  |  | Robust |  |  |  |  |
|  | Coefficient | std. err. | z | $P>\|z\|$ | [95\% conf | interval] |
| /b1 | -. 0038515 | . 0019472 | -1.98 | 0.048 | -. 0076678 | -. 0000351 |
| /b2 | -. 0795935 | . 0677528 | -1.17 | 0.240 | -. 2123866 | . 0531996 |
| /b0 | 47.88487 | 7.50599 | 6.38 | 0.000 | 33.1734 | 62.59634 |

Instruments for equation 1: weight length _cons
Because the number of moments equals the number of parameters we are estimating, the model is said to be "just identified" or "exactly identified". Therefore, the choice of weight matrix has no impact on the solution to (1), and the criterion function $Q(\boldsymbol{\beta})$ achieves its minimum value at 0 .

The OLS estimator is a one-step GMM estimator, but we did not bother to specify the onestep option, because the model is just identified. Doing a second step of GMM estimation affects neither the point estimates nor the standard errors, so to keep the syntax as simple as possible, we did not include the onestep option. The first step of estimation resulted in $Q(\boldsymbol{\beta})=0$ as expected, and the second step of estimation did not change the minimized value of $Q(\boldsymbol{\beta})$. (The final iterations of the first and second steps result in $Q(\boldsymbol{\beta})$ values of 0 for all practical purposes.)

When you do not specify either the wmatrix () or the vce() option, gmm reports heteroskedasticityrobust standard errors. The parameter estimates reported here match those that we would obtain from the command

```
. regress mpg weight length, vce(robust)
```

The standard errors reported by that regress command would be larger than those reported by gmm by a factor of $\operatorname{sqrt}(74 / 71)$ because regress makes a small-sample adjustment to the estimated variance matrix while gmm does not. Likewise, if we had specified the vce (unadjusted) option with our gmm command, then our standard errors would differ by a factor of $\operatorname{sqrt}(74 / 71)$ from those reported by regress without the vce(robust) option.

We could have submitted our substitutable expression using the notation for linear combinations of parameters. If we select $x b$ as the name of our parameter equation, we could type

```
. gmm (mpg - {xb: weight length _cons}), instruments(weight length)
```

and obtain identical results. With this syntax, instead of having parameters b1, b2, and b0, we would have parameters xb :weight, xb :length, and xb :_cons. Note that _cons allows you to include a constant in a linear combination, so this time, we do not have to specify a separate parameter from our varlist.

Factor variables and time-series-operated variables are allowed in the linear combinations. For example,

```
. regress mpg i.foreign i.foreign#c.weight, vce(robust)
```

produces the same results as

```
. gmm (mpg - {xb:i.foreign i.foreign#c.weight _cons}),
> instruments(i.foreign i.foreign#c.weight)
```

See [U] 11.4.3 Factor variables and [U] 11.4.4 Time-series varlists for an introduction to factor variables and time-series operators. See example 4 for an example of factor-variable syntax with gmm. See example 15 for an example using time-series-operated variables.

## > Example 2: Instrumental-variables regression

In Introduction, we mentioned that 2SLS can be viewed as a GMM estimator. In example 1 of $[R]$ ivregress, we fit by a 2SLS model of rental rates (rent) as a function of the value of owner-occupied housing (hsngval) and the percentage of the population living in urban areas (pcturban):

$$
\text { rent }=\beta_{0}+\beta_{1} \text { hsngval }+\beta_{2} \text { pcturban }+u
$$

We argued that random shocks that affect rental rates likely also affect housing values, so we treated hsngval as an endogenous variable. As additional instruments, we used family income, faminc, and three regional dummies (reg2-reg4).

To replicate the results of ivregress 2sls by using gmm, we type

| $\begin{aligned} & \text { ( use https://www.stata-press.com/data/r18/] } \\ & \text { (1980 Census housing data) } \\ & \text {. gmm (rent - \{xb:hsngval pcturban _cons\}), } \end{aligned}$ |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| > instruments (pcturban faminc reg2-reg4) vce(unadjusted) onestep |  |  |  |  |  |  |
| Step 1 |  |  |  |  |  |  |
| Iteration 0: GMM criterion Q(b) = 56115.03 |  |  |  |  |  |  |
| Iteration 1: GMM criterion Q (b) $=110.91583$ |  |  |  |  |  |  |
| Iteration 2: GMM criterion $\mathrm{Q}(\mathrm{b})=110.91583$ |  |  |  |  |  |  |
| GMM estimation |  |  |  |  |  |  |
| Number of parameters $=3$ |  |  |  |  |  |  |
| Number of moments = 6 |  |  |  |  |  |  |
| Initial weight matrix: Unadjusted |  |  |  | Numb | of obs | 50 |
|  | Coefficient | Std. err. | z | $P>\|z\|$ | [95\% con | interval] |
| hsngval | . 0022398 | . 0003284 | 6.82 | 0.000 | . 0015961 | . 0028836 |
| pcturban | . 081516 | . 2987652 | 0.27 | 0.785 | -. 5040531 | . 6670851 |
| _cons | 120.7065 | 15.22839 | 7.93 | 0.000 | 90.85942 | 150.5536 |

Instruments for equation 1: pcturban faminc reg2 reg3 reg4 _cons
We specified vce(unadjusted) so that we would obtain an unadjusted VCE matrix and our standard errors would match those reported in [R] ivregress.

Note how we specified the instruments() option. In Introduction, we mentioned that the moment conditions for the 2SLS estimator are $E(\mathbf{z} u)=\mathbf{0}$, and we mentioned that if some elements of $\mathbf{x}$ (the regressors) are not endogenous, then they should also appear in $\mathbf{z}$. In this model, we assume the regressor pcturban is exogenous, so we included it in the list of instrumental variables. Commands like ivregress, ivprobit, and ivtobit accept standard varlists, so they can deduce the exogenous regressors in the model. Because gmm accepts arbitrary functions in the form of substitutable expressions, it has no way of discerning the exogenous variables of the model on its own.

Also notice that we specified the onestep option. The 2SLS estimator is a one-step GMM estimator that is based on a weight matrix that assumes the error terms are i.i.d. Unlike the previous example, this example had more instruments than parameters, so the minimized value of $Q(\boldsymbol{\beta})$ is nonzero. We discuss the weight matrix and its relationship to two-step estimation next.

## The weight matrix and two-step estimation

Recall our definition of the GMM estimator given in (1). The estimator, $\widehat{\boldsymbol{\beta}}$, depends on the choice of the weight matrix, $\mathbf{W}$. Under relatively mild assumptions, our estimator, $\widehat{\boldsymbol{\beta}}$, is consistent regardless of the choice of $\mathbf{W}$, so how are we to decide what $\mathbf{W}$ to use? The most common solution is to use the two-step estimator, which we now describe.

A key result in Hansen's (1982) seminal paper is that if we denote by $\mathbf{S}$ the covariance matrix of the moment conditions, then the optimal (in a way we make precise later) GMM estimator is the one that uses a weight matrix equal to the inverse of the moment covariance matrix. That is, if we let $\mathbf{S}=\operatorname{Cov}(\mathbf{z} u)$, then we want to use $\mathbf{W}=\mathbf{S}^{-1}$. But how do we obtain $\mathbf{S}$ in the first place?

If we assume that the errors are i.i.d., then

$$
\operatorname{Cov}(\mathbf{z} u)=E\left(u^{2} \mathbf{z} \mathbf{z}^{\prime}\right)=\sigma^{2} E\left(\mathbf{z z}^{\prime}\right)
$$

where $\sigma^{2}$ is the variance of $u$. Because $\sigma^{2}$ is a positive scalar, we can ignore it when solving (1). Thus, we compute

$$
\begin{equation*}
\widehat{\mathbf{W}}_{1}=\left(\frac{1}{N} \sum_{i} \mathbf{z}_{i} \mathbf{z}_{i}^{\prime}\right)^{-1} \tag{2}
\end{equation*}
$$

which does not depend on any unknown model parameters. (Notice that $\widehat{\mathbf{W}}_{1}$ is the same weight matrix used in 2SLS.) Given $\widehat{\mathbf{W}}_{1}$, we can solve (1) to obtain an initial estimate, say, $\widehat{\boldsymbol{\beta}}_{1}$.

Our estimate, $\widehat{\boldsymbol{\beta}}_{1}$, is consistent, so by Slutsky's theorem, the sample residuals $\widehat{u}$ computed at this value of $\boldsymbol{\beta}$ will also be consistent. Using virtually the same arguments used to justify the Huber/Eicker/White heteroskedasticity-robust VCE, if we assume that the residuals are independent though not identically distributed, we can estimate $\mathbf{S}$ as

$$
\widehat{\mathbf{S}}=\frac{1}{N} \sum_{i} \widehat{u}_{i}^{2} \mathbf{z}_{i} \mathbf{z}_{i}^{\prime}
$$

Then, in the second step, we re-solve (1), using $\widehat{\mathbf{W}}_{2}=\widehat{\mathbf{S}}^{-1}$, which yields the two-step GMM estimate $\widehat{\beta}_{2}$. If the residuals exhibit clustering, you can specify wmatrix (cluster varname) so that gmm computes a weight matrix that does not assume the $u_{i}$ 's are independent within clusters identified by varname. You can specify wmatrix (hac ...) to obtain weight matrices that are suitable for when the $u_{i}$ 's exhibit autocorrelation as well as heteroskedasticity.

We could take the point estimates from the second round of estimation and use them to compute yet another weight matrix, $\widehat{\mathbf{W}}_{3}$, say, re-solve (1) yet again, and so on, stopping when the parameters or weight matrix do not change much from one iteration to the next. This procedure is known as the iterative GMM estimator and is obtained with the igmm option. Asymptotically, the two-step and iterative GMM estimators have the same distribution. However, Hall (2005, 90) suggests that the iterative estimator may have better finite-sample properties.

Instead of computing $\widehat{\mathbf{W}}_{1}$ as in (2), we could simply choose $\widehat{\mathbf{W}}_{1}=\mathbf{I}$, the identity matrix. The initial estimate, $\widehat{\boldsymbol{\beta}}_{1}$, would still be consistent. You can request this behavior by specifying the winitial (identity) option. However, if you specify all of your moment conditions of the form $E(\mathbf{z} u)=\mathbf{0}$, we recommend using the default winitial (unadjusted) instead; the rescaling of the moment conditions implied by using a homoskedastic initial weight matrix makes the numerical routines used to solve (1) more stable.

If you fit a model with more than one of the moment conditions of the form $E\{h(\mathbf{z} ; \boldsymbol{\beta})\}=\mathbf{0}$, then you must use winitial(identity) or winitial (unadjusted, independent). With moment conditions of that form, you do not specify a list of instruments, and gmm cannot evaluate (2)-the matrix expression in parentheses would necessarily be singular, so it cannot be inverted.

## > Example 3: Two-step linear GMM estimator

From the previous discussion and the comments in Introduction, we see that the linear 2SLS estimator is a one-step GMM estimator where we use the weight matrix defined in (2) that assumes the errors are i.i.d. If we use the 2SLS estimate of $\boldsymbol{\beta}$ to obtain the sample residuals, compute a new weight matrix based on those residuals, and then do a second step of GMM estimation, we obtain the linear two-step GMM estimator as implemented by ivregress gmm.

In example 3 of $[R]$ ivregress, we fit the model of rental rates as discussed in example 2 above. We now allow the residuals to be heteroskedastic, though we will maintain our assumption that they are independent. We type

```
. gmm (rent - {xb:hsngval pcturban _cons}), instruments(pcturban faminc reg2-reg4)
Step 1
Iteration 0: GMM criterion Q(b) = 56115.03
Iteration 1: GMM criterion Q(b) = 110.91583
Iteration 2: GMM criterion Q(b) = 110.91583
Step 2
Iteration 0: GMM criterion Q(b) = . }240608
Iteration 1: GMM criterion Q(b) = . }1367280
Iteration 2: GMM criterion Q(b) = . }1367280
GMM estimation
Number of parameters = 3
Number of moments = 6
Initial weight matrix: Unadjusted Number of obs = 50
GMM weight matrix: Robust
```

|  | Robust |  |  |  |  |  |
| ---: | ---: | :---: | :---: | :---: | :---: | ---: |
|  | Coefficient | std. err. | $z$ | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| hsngval | .0014643 | .0004473 | 3.27 | 0.001 | .0005877 | .002341 |
| pcturban | .7615482 | .2895105 | 2.63 | 0.009 | .1941181 | 1.328978 |
| _cons | 112.1227 | 10.80234 | 10.38 | 0.000 | 90.95052 | 133.2949 |

Instruments for equation 1: pcturban faminc reg2 reg3 reg4 _cons
By default, gmm computes a heteroskedasticity-robust weight matrix before the second step of estimation, though we could have specified wmatrix (robust) if we wanted to be explicit. Because we did not specify the vce() option, gmm used a heteroskedasticity-robust one. Our results match those in example 3 of $[R]$ ivregress. Moreover, the only substantive difference between this example and example 2 is that here we did not specify the onestep option, so we obtain the two-step estimates.

## Obtaining standard errors

This section is a bit more theoretical and can be skipped on first reading. However, the information is sufficiently important that you should return to this section at some point.

So far in our discussion, we have focused on point estimation without much mention of how we obtain the standard errors of the estimates. We also mentioned that if we choose $\mathbf{W}$ to be the inverse of the covariance matrix of the moment conditions, then we obtain the "optimal" GMM estimator. We elaborate those points now.

Using mostly standard statistical arguments, we can show that for the GMM estimator defined in (1), the variance of $\widehat{\boldsymbol{\beta}}$ is given by

$$
\begin{equation*}
\operatorname{Var}(\widehat{\boldsymbol{\beta}})=\frac{1}{N}\left\{\overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})^{\prime} \mathbf{W} \overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})\right\}^{-1} \overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})^{\prime} \mathbf{W} \mathbf{S W} \overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})\left\{\overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})^{\prime} \mathbf{W} \overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})\right\}^{-1} \tag{3}
\end{equation*}
$$

where

$$
\overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})=\left.\frac{1}{N} \sum_{i} \mathbf{z}_{i} \frac{\partial u_{i}}{\partial \boldsymbol{\beta}}\right|_{\boldsymbol{\beta}=\widehat{\boldsymbol{\beta}}} \quad \text { or } \quad \overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})=\left.\frac{1}{N} \sum_{i} \frac{\partial \mathbf{h}_{i}}{\partial \boldsymbol{\beta}}\right|_{\boldsymbol{\beta}=\widehat{\boldsymbol{\beta}}}
$$

as the case may be and $\mathbf{S}=E\left(\mathbf{z} u u^{\prime} \mathbf{z}^{\prime}\right)$.
Assuming the vce (unadjusted) option is not specified, gmm reports standard errors based on the robust variance matrix defined in (3). For the two-step estimator, $\mathbf{W}$ is the weight matrix requested with the wmatrix() option, and it is calculated based on the residuals obtained after the first estimation step. The second-step point estimates and residuals are obtained, and $\mathbf{S}$ is calculated based on the specification of the vce() option. For the iterated estimator, $\mathbf{W}$ is calculated based on the second-to-last round of estimation, while $\mathbf{S}$ is based on the residuals obtained after the last round of estimation. Computation of the covariance matrix for the one-step estimator is, perhaps surprisingly, more involved; we discuss the covariance matrix with the one-step estimator in the technical note at the end of this section.

If the model is exactly identified, the matrix $\overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})$ is square, and (3) simplifies to the following:

$$
\operatorname{Var}(\widehat{\boldsymbol{\beta}})=\frac{1}{N} \overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})^{-1} \mathbf{S}\left(\overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})^{\prime}\right)^{-1}
$$

If we choose the weight matrix to be the inverse of the covariance matrix of the moment conditions so that $\mathbf{W}=\mathbf{S}^{-1}$, then (3) simplifies substantially:

$$
\begin{equation*}
\operatorname{Var}(\widehat{\boldsymbol{\beta}})=\frac{1}{N}\left\{\overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})^{\prime} \mathbf{W} \overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})\right\}^{-1} \tag{4}
\end{equation*}
$$

The GMM estimator constructed using this choice of weight matrix along with the covariance matrix in (4) is known as the "optimal" GMM estimator. One can show that if in fact $\mathbf{W}=\mathbf{S}^{-1}$, then the variance in (4) is smaller than the variance in (3) of any other GMM estimator based on the same moment conditions but with a different choice of weight matrix. Thus, the optimal GMM estimator is also known as the efficient GMM estimator because it has the smallest variance of any estimator based on the given moment conditions.

To obtain standard errors from gmm based on the optimal GMM estimator, you specify the vce (unadjusted) option. We call that VCE unadjusted because we do not recompute the residuals after estimation to obtain the matrix $\mathbf{S}$ required in (3) or allow for the fact that those residuals may not be i.i.d. Some statistical packages by default report standard errors based on (4) and offer standard errors based on (3) only as an option or not at all. While the optimal GMM estimator is theoretically appealing, Cameron and Trivedi $(2005,177)$ suggest that in finite samples, it need not perform better than the GMM estimator that uses (3) to obtain standard errors.

## - Technical note

Computing the covariance matrix of the parameters after using the one-step estimator is actually a bit more complex than after using the two-step or iterative estimator. We can illustrate most of the intricacies by using linear regression with moment conditions of the form $E\left\{\mathbf{x}\left(y-\mathbf{x}^{\prime} \boldsymbol{\beta}\right)\right\}=\mathbf{0}$.

If you specify winitial (unadjusted) and vce(unadjusted), then the initial weight matrix will be computed as

$$
\begin{equation*}
\widehat{\mathbf{W}}_{1}=\left(\frac{1}{N} \sum_{i} \mathbf{x}_{i} \mathbf{x}_{i}^{\prime}\right)^{-1} \tag{5}
\end{equation*}
$$

Moreover, for linear regression, we can show that

$$
\overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})=\frac{1}{N} \sum_{i} \mathbf{x}_{i} \mathbf{x}_{i}^{\prime}
$$

so that (4) becomes

$$
\begin{align*}
\operatorname{Var}(\widehat{\boldsymbol{\beta}}) & =\frac{1}{N}\left\{\left(\frac{1}{N} \sum_{i} \mathbf{x}_{i} \mathbf{x}_{i}^{\prime}\right)\left(\frac{1}{N} \sum_{i} \mathbf{x}_{i} \mathbf{x}_{i}^{\prime}\right)^{-1}\left(\frac{1}{N} \sum_{i} \mathbf{x}_{i} \mathbf{x}_{i}^{\prime}\right)\right\}^{-1} \\
& =\left(\sum_{i} \mathbf{x}_{i} \mathbf{x}_{i}^{\prime}\right)^{-1} \\
& =\left(\mathbf{X}^{\prime} \mathbf{X}\right)^{-1} \tag{6}
\end{align*}
$$

However, we know that the nonrobust covariance matrix for the OLS estimator is actually $\widehat{\sigma}^{2}\left(\mathbf{X}^{\prime} \mathbf{X}\right)^{-1}$. What is missing from (6) is the scalar $\widehat{\sigma}^{2}$, the estimated variance of the residuals. When you use the one-step estimator and specify winitial (unadjusted), the weight matrix (5) does not include the $\widehat{\sigma}^{2}$ term because gmm does not have a consistent estimate of $\boldsymbol{\beta}$ from which it can then estimate $\sigma^{2}$. The point estimates are still correct because multiplying the weight matrix by a scalar factor does not affect the solution to the minimization problem.

To circumvent this issue, if you specify winitial (unadjusted) and vce(unadjusted), gmm uses the estimated $\widehat{\boldsymbol{\beta}}$ (which is consistent) to obtain a new unadjusted weight matrix that does include the term $\widehat{\sigma}^{2}$ so that evaluating (4) will yield correct standard errors.

If you use the two-step or iterated GMM estimator, this extra effort is not needed to obtain standard errors because the first-step (and subsequent steps') estimate of $\boldsymbol{\beta}$ is consistent and can be used to estimate $\sigma^{2}$ or some other weight matrix based on the wmatrix() option. Straightforward algebra shows that this extra effort is also not needed if you request any type of adjusted (robust) covariance matrix with the one-step estimator.

A similar issue arises when you specify winitial (identity) and vce(unadjusted) with the one-step estimator. Again the solution is to compute an unadjusted weight matrix after obtaining $\widehat{\beta}$ so that (4) provides the correct standard errors.

We have illustrated the problem and solution using a single-equation linear model. However, the problem arises whenever you use the one-step estimator with an unadjusted VCE, regardless of the number of equations, and gmm handles all the details automatically. Computation of Hansen's $J$ statistic presents an identical issue, and gmm takes care of that as well.

If you supply your own initial weight matrix by using winitial(matname), then the standard errors (as well as the $J$ statistic reported by estat overid) are based on that weight matrix. You should verify that the weight matrix you provide will yield appropriate statistics.

## Factor-variable coefficients in multiple residual functions

The long example in this section uses gmm to replicate the results produced by regress with factor variables and margins. It illustrates how to refer to the coefficients on factor variables in linear combinations in subsequent residual functions. The example also shows how to use gmm to address the two-step estimation problem, or the inconsistency of standard errors produced by two-step estimators that depend on previously estimated parameters.

## > Example 4: Means of linear combinations of factor variables

The mean of a variable when everyone in a population receives a given treatment level is known as a potential-outcome mean. We use regress and margins to estimate the potential-outcome means of a mother's smoking behavior while pregnant on the birthweight of her baby after controlling for the mother's age and an indicator for whether the mother had a prenatal visit in the first trimester. We use an extract of data from Cattaneo (2010) in which bweight is the baby's birthweight in grams, mbsmoke is a binary variable indicating whether a mother smoked while pregnant, mage is the mother's age, and prenatall is a binary variable indicating whether the mother had a prenatal visit in the first trimester.

We use regress to estimate the regression coefficients.
. use https://www.stata-press.com/data/r18/cattaneo2
(Excerpt from Cattaneo (2010) Journal of Econometrics 155: 138-154)
. regress bweight ibn.mbsmoke ibn.mbsmoke\#(c.mage i.prenatal1),
> noconstant vce(robust)
Linear regression $\quad$ Number of obs $=4,642$
$\mathrm{F}(6,4636)=27751.75$
Prob > F $=0.0000$
R-squared $=0.9726$
Root MSE $=565.08$


We used factor variables to interact mbsmoke with the other covariates to allow for separate coefficients for smoking and nonsmoking mothers.

The postestimation command margins uses the estimated regression coefficients to estimate the potential-outcome means of bweight, first assuming that no mother smoked and then assuming that all mothers smoked.

```
. margins i.mbsmoke, vce(unconditional)
Predictive margins Number of obs = 4,642
Expression: Linear prediction, predict()
```

|  | Unconditional <br> std. err. |  |  |  | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |
| :---: | ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| mbsmoke |  |  |  |  |  |  |  |
| Nonsmoker | 3407.506 | 9.346894 | 364.56 | 0.000 | 3389.181 | 3425.83 |  |
| Smoker | 3138.23 | 21.20463 | 148.00 | 0.000 | 3096.659 | 3179.801 |  |

Note that the standard errors for the estimated means account for the estimation error in the estimated coefficients used to compute them.

Before using gmm to simultaneously estimate the regression coefficients and the potential-outcome means, we demonstrate the equivalence of point estimates from gmm and regress and illustrate the two-step estimation problem. First, we use gmm to estimate just the regression coefficients. Note that we specify the factor variables in the linear combination xb : and in the instrument list.


Instruments for equation eq1: 0.mbsmoke 1.mbsmoke 0.mbsmoke\#c.mage
1.mbsmoke\#c.mage 0o.mbsmoke\#Ob.prenatal1 0.mbsmoke\#1.prenatal1
10.mbsmoke\#0b.prenatal1 1.mbsmoke\#1.prenatal1

We specified the coeflegend option to learn the names of the coefficients on the factor variables in the linear combination. As expected, the point estimates are the same as those reported by regress.

Next, we illustrate the effect of the two-step estimation problem if we calculate the potentialoutcome means by hand. We can calculate the mean when no mothers smoke by accessing these coefficients and then estimate the standard errors in the estimated means:

```
. generate mean0 = _b[xb:0.mbsmoke] + _b[xb:0.mbsmoke#c.mage]*mage
> + _b[xb:0.mbsmoke#1.prenatal1]*prenatal1
. mean mean0
```

Mean estimation $\quad$ Number of obs $=4,642$

|  | Mean | Std. err. | [95\% conf. interval] |  |
| ---: | ---: | :--- | :--- | :--- |
| mean0 | 3407.506 | 1.085503 | 3405.378 | 3409.634 |

The estimated potential-outcome mean for no mothers smoking is the same as that reported by margins, but the standard error is much smaller because mean ignores the estimation error in the coefficients. This underscores the importance of accounting for the estimation error when estimating the standard errors.

Now, we use gmm to estimate the coefficients and the potential-outcome means simultaneously.

| . gmm (eq1: bweight - \{xb:ibn.mbsmoke ibn.mbsmoke\#(c.mage i.prenatal1) \}) |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| > (eq2: \{xb:0.mbsmoke\} + \{xb:Obn.mbsmoke\#c.mage\}*mage |  |  |  |  |  |  |
| $>\quad+\{\mathrm{xb}: 0 \mathrm{bn} . \mathrm{mbsmoke} \mathrm{\# 1.prenatal1} \mathrm{\} *1.prenatal1}$ |  |  |  |  |  |  |
| > (eq3: \{xb:1.mbsmoke\} + \{xb:1.mbsmoke\#c.mage\}*mage |  |  |  |  |  |  |
| $>\quad+\{x b: 1 . m b s m o k e \# 1 . p r e n a t a l 1\} * 1 . p r e n a t a l 1-\{m 1\})$, |  |  |  |  |  |  |
| > instruments(eq1: ibn.mbsmoke ibn.mbsmoke\#(c.mage i.prenatali) |  |  |  |  |  |  |
| $>$ noconstant) |  |  |  |  |  |  |
| > instruments (eq2 eq3:) winitial (identity) onestep |  |  |  |  |  |  |
| Step 1 |  |  |  |  |  |  |
| Iteration 0: GMM criterion $\mathrm{Q}(\mathrm{b})=5.819 \mathrm{e}+09$ |  |  |  |  |  |  |
| Iteration 1: GMM criterion $\mathrm{Q}(\mathrm{b})=3.108 \mathrm{e}-13$ |  |  |  |  |  |  |
| Iteration 2: GMM criterion $\mathrm{Q}(\mathrm{b})=1.010 \mathrm{e}-22$ |  |  |  |  |  |  |
| note: model is exactly identified. |  |  |  |  |  |  |
| GMM estimation |  |  |  |  |  |  |
| Number of parameters $=8$ |  |  |  |  |  |  |
| Number of moments = 8 |  |  |  |  |  |  |
| Initial weight matrix: Identity |  |  |  | Numb | of obs | 4,642 |
|  | Coefficient | Robust std. err. | z | $P>\|z\|$ | [95\% conf | interval] |
| mbsmoke |  |  |  |  |  |  |
| Nonsmoker | 3073.201 | 48.65751 | 63.16 | 0.000 | 2977.834 | 3168.568 |
| Smoker | 3217.973 | 93.57647 | 34.39 | 0.000 | 3034.566 | 3401.379 |
| mbsmoke\# c.mage |  |  |  |  |  |  |
| Nonsmoker | 9.737189 | 1.824372 | 5.34 | 0.000 | 6.161485 | 13.31289 |
| Smoker | -4.962403 | 3.850123 | -1.29 | 0.197 | -12.5085 | 2.583699 |
| mbsmoke\# prenatal1 |  |  |  |  |  |  |
| Nonsmoker \# Yes | 95.11727 | 26.80305 | 3.55 | 0.000 | 42.58425 | 147.6503 |
| Smoker\#Yes | 64.61752 | 39.69749 | 1.63 | 0.104 | -13.18813 | 142.4232 |
| /m0 | 3407.506 | 9.340852 | 364.80 | 0.000 | 3389.198 | 3425.813 |
| /m1 | 3138.23 | 21.19093 | 148.09 | 0.000 | 3096.696 | 3179.763 |

```
Instruments for equation eq1: 0.mbsmoke 1.mbsmoke 0.mbsmoke#c.mage
    1.mbsmoke#c.mage Oo.mbsmoke#Ob.prenatal1 0.mbsmoke#1.prenatal1
    1o.mbsmoke#0b.prenatal1 1.mbsmoke#1.prenatal1
Instruments for equation eq2: _cons
Instruments for equation eq3: _cons
```

This output has five noteworthy features.

1. We specify three different residual equations. The first, eq1:, defines the moment conditions for the regression using the covariates as instruments; eq2: is the moment condition for the potential outcome when no mothers smoke; and eq3: is the moment condition for the potential outcome when all mothers smoke.
2. In eq2: and eq3:, we refer to the individual coefficients on the factor variables in the linear combination xb : by enclosing their names in curly braces.
3. The option instruments () is repeatable. We first specify that the covariates in the regression are the instruments for the residual equation eq1: and then specify that only the unit variable, also known as the constant, is an instrument for each of eq2: and eq3:.
4. The point estimates and the standard errors match those reported by regress and margins, after accounting for the small-sample adjustment performed by regress.
5. Although the point estimates match, the standard errors reported by gmm are much larger than those reported by mean because gmm takes into account that the regression coefficients are estimated.

## Parameter interpretation using margins

In the last section, we demonstrated how to use gmm to estimate potential-outcome means in a linear regression model jointly with the coefficients of the models. However, you can also estimate the potential-outcome mean, or any other predictive margins, by using the margins command after gmm. Using margins after gmm can allow more flexibility in the predictive margins that we estimate and is also more convenient. See Obtaining margins of responses in $[R]$ margins for more information about predictive margins.

## > Example 5: Predicting treatment effects after estimation

In example 4, we used gmm to estimate potential-outcome means of a mother's smoking behavior on her baby's birthweight (in grams) after controlling for age and whether she had a prenatal visit in the first trimester. Here we demonstrate how to use margins after gmm to estimate the potential-outcome means.

First, we load the data and reestimate the regression coefficients.

```
. use https://www.stata-press.com/data/r18/cattaneo2
(Excerpt from Cattaneo (2010) Journal of Econometrics 155: 138-154)
. gmm (eq1: bweight - {xb:ibn.mbsmoke ibn.mbsmoke#(c.mage i.prenatal1)}),
> instruments(eq1: ibn.mbsmoke ibn.mbsmoke#(c.mage i.prenatal1), noconstant)
> onestep
Step 1
Iteration 0: GMM criterion Q(b) = 11316945
Iteration 1: GMM criterion Q(b) = 7.143e-19
Iteration 2: GMM criterion Q(b) = 2.051e-26
note: model is exactly identified.
GMM estimation
Number of parameters = 6
Number of moments = 6
Initial weight matrix: Unadjusted Number of obs = 4,642
```

|  | Coefficient | Robust std. err. | z | $P>\|z\|$ | [95\% conf. | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| mbsmoke |  |  |  |  |  |  |
| Nonsmoker | 3073.201 | 48.65752 | 63.16 | 0.000 | 2977.834 | 3168.568 |
| Smoker | 3217.973 | 93.57647 | 34.39 | 0.000 | 3034.566 | 3401.379 |
| mbsmoke\# c.mage |  |  |  |  |  |  |
| Nonsmoker | 9.737189 | 1.824372 | 5.34 | 0.000 | 6.161484 | 13.31289 |
| Smoker | -4.962403 | 3.850123 | -1.29 | 0.197 | -12.5085 | 2.583699 |
| mbsmoke\# prenatal1 |  |  |  |  |  |  |
| Nonsmoker \# | 95.11727 | 26.80305 | 3.55 | 0.000 | 42.58425 | 147.6503 |
| Smoker\#Yes | 64.61752 | 39.69749 | 1.63 | 0.104 | -13.18813 | 142.4232 |

Instruments for equation eq1: 0.mbsmoke 1.mbsmoke 0.mbsmoke\#c.mage 1.mbsmoke\#c.mage Oo.mbsmoke\#Ob.prenatal1 0.mbsmoke\#1.prenatal1 1o.mbsmoke\#Ob.prenatal1 1.mbsmoke\#1.prenatal1

Now, we use margins to estimate the potential-outcome means. We specify vce (unconditional) to obtain standard errors for the potential-outcome means of the population rather than the sample. When we specify this option, the standard errors for the estimated means will account for the estimation error in the estimated coefficients from gmm.

| . margins i.mbsmoke, vce(unconditional) |
| :--- |
| Predictive margins |
| Expression: Linear prediction, predict() |

Our point estimates of the potential-outcome means exactly match those that appear as $\backslash \mathrm{m} 0$ and $\backslash \mathrm{m} 1$ in example 4. However, the standard errors are slightly higher because gmm and margins use different values in the denominator for the formula for the robust covariance matrix. gmm uses $N$ while margins
uses $N-1$, so the standard errors differ by a factor of $\sqrt{\{N /(N-1)\}}=\sqrt{4,642 / 4,641} \approx 1.0002$. More details about the calculation of the standard errors are provided in Marginal predictions with unconditional standard errors in the Methods and formulas.

In addition to potential-outcome means, we can use margins to estimate the average treatment effect (ATE) of the mother's smoking behavior on birthweight. We use the contrast operator r. to instruct margins to difference the potential-outcome means and estimate a treatment effect. We specify the contrast (nowald) option to suppress the Wald test that margins displays by default for contrasts.

```
. margins r.mbsmoke, vce(unconditional) contrast(nowald)
Contrasts of predictive margins Number of obs = 4,642
Expression: Linear prediction, predict()
```

|  | Unconditional <br> std. err. |  |  |  |
| :--- | :---: | :---: | :---: | :---: |
| Contrast | [95\% conf. interval] |  |  |  |
| mbsmoke | -269.2759 | 23.16069 | -314.67 | -223.8818 |

The ATE of -269.28 is interpreted as the difference between the average birthweight if all mothers in the population smoked and the average birthweight if all mothers in the population did not smoke. The average birthweight if all mothers were to smoke would be 269.28 grams less than if they did not smoke.

## Exponential (Poisson) regression models

Exponential regression models are frequently encountered in applied work. For example, they can be used as alternatives to linear regression models on log-transformed dependent variables, obviating the need for post-hoc transformations to obtain predicted values in the original metric of the dependent variable. When the dependent variable represents a discrete count variable, exponential regression models are also known as Poisson regression models; see Cameron and Trivedi (2013).

For now, we consider models of the form

$$
\begin{equation*}
y=\exp \left(\mathbf{x}^{\prime} \boldsymbol{\beta}\right)+u \tag{7}
\end{equation*}
$$

where $u$ is a zero-mean additive error term so that $E(y)=\exp \left(\mathbf{x}^{\prime} \boldsymbol{\beta}\right)$. Because the error term is additive, if $\mathbf{x}$ represents strictly exogenous regressors, then we have the population moment condition

$$
\begin{equation*}
E\left[\mathbf{x}\left\{y-\exp \left(\mathbf{x}^{\prime} \boldsymbol{\beta}\right)\right\}\right]=\mathbf{0} \tag{8}
\end{equation*}
$$

Moreover, because the number of parameters in the model is equal to the number of instruments, there is no point to using the two-step GMM estimator.

## > Example 6: Exponential regression

Cameron and Trivedi $(2022,584)$ fit a model of the number of doctor visits based on whether the patient has private insurance, whether the patient has a chronic disease, gender, and income. Here we fit that model by using gmm. To allow for potential excess dispersion, we will obtain a robust VCE matrix, which is the default for gmm anyway. We type

```
. use https://www.stata-press.com/data/r18/docvisits
. gmm (docvis - exp({xb:private chronic female income _cons})),
> instruments(private chronic female income) onestep
Step 1
Iteration 0: GMM criterion Q(b) = 16.853973
Iteration 1: GMM criterion Q(b) = 2.2706472
Iteration 2: GMM criterion Q(b) = . }1908809
Iteration 3: GMM criterion Q(b) = .00041101
Iteration 4: GMM criterion Q(b) = 3.939e-09
Iteration 5: GMM criterion Q(b) = 6.572e-19
note: model is exactly identified.
GMM estimation
Number of parameters = 5
Number of moments = 5
Initial weight matrix: Unadjusted Number of obs = 4,412
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline & Coefficient & Robust std. err. & z & \(P>|z|\) & [95\% conf. & interval] \\
\hline private & . 7986654 & . 1089891 & 7.33 & 0.000 & . 5850507 & 1.01228 \\
\hline chronic & 1.091865 & . 0559888 & 19.50 & 0.000 & . 9821291 & 1.201601 \\
\hline female & . 4925481 & . 0585298 & 8.42 & 0.000 & . 3778317 & . 6072644 \\
\hline income & . 003557 & . 0010824 & 3.29 & 0.001 & . 0014356 & . 0056784 \\
\hline _cons & -. 2297263 & . 1108607 & -2.07 & 0.038 & -. 4470093 & -. 0124434 \\
\hline
\end{tabular}
Instruments for equation 1: private chronic female income _cons
```

Our point estimates agree with those reported by Cameron and Trivedi (2022) to at least six significant digits; the small discrepancies are attributable to different optimization techniques and convergence criteria being used by gmm and poisson. The standard errors differ by a factor of sqrt(4412/4411) because gmm uses $N$ in the denominator of the formula for the robust covariance matrix, while the robust covariance matrix estimator used by poisson uses $N-1$.

## Technical note

That the GMM and maximum likelihood estimators of the exponential regression model coincide is not a general property of these two classes of estimators. The maximum likelihood estimator solves the score equations

$$
\frac{1}{N} \sum_{i=1}^{N} \frac{\partial \ln \ell_{i}}{\partial \boldsymbol{\beta}}=\mathbf{0}
$$

where $\ell_{i}$ is the likelihood for the $i$ th observation. These score equations can be viewed as the sample analogues of the population moment conditions

$$
E\left(\frac{\partial \ln \ell_{i}}{\partial \boldsymbol{\beta}}\right)=\mathbf{0}
$$

establishing that maximum likelihood estimators represent a subset of the class of GMM estimators.
For the Poisson model,

$$
\ln \ell_{i}=-\exp \left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\right)+y_{i} \mathbf{x}_{i}^{\prime} \boldsymbol{\beta}-\ln y_{i}!
$$

so the score equations are

$$
\frac{1}{N} \sum_{i=1}^{N} \mathbf{x}_{i}\left\{y_{i}-\exp \left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\right)\right\}=\mathbf{0}
$$

which are just the sample moment conditions implied by (8) that we used in the previous example. That is why our results using gmm match Cameron and Trivedi’s (2022) results using poisson.

On the other hand, an intuitive set of moment conditions to consider for GMM estimation of a probit model is

$$
E\left[\mathbf{x}\left\{y-\Phi\left(\mathbf{x}^{\prime} \boldsymbol{\beta}\right)\right\}\right]=\mathbf{0}
$$

where $\Phi()$ is the standard normal cumulative distribution function. Differentiating the likelihood function for the maximum-likelihood probit estimator, we can show that the corresponding score equations are

$$
\frac{1}{N} \sum_{i=1}^{N}\left[\mathbf{x}_{i}\left\{y_{i} \frac{\phi\left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\right)}{\Phi\left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\right)}-\left(1-y_{i}\right) \frac{\phi\left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\right)}{1-\Phi\left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\right)}\right\}\right]=\mathbf{0}
$$

where $\phi()$ is the standard normal density function. These two moment conditions are not equivalent, so the maximum likelihood and GMM probit estimators are distinct.
> Example 7: Comparison of GMM and maximum likelihood
Using the automobile dataset, we fit a probit model of foreign on gear_ratio, length, and headroom using first the score equations and then the intuitive set of GMM equations. We type

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. gmm (foreign*normalden({xb:gear_ratio length headroom _cons})/
> normal({xb:}) - (1-foreign)*normalden({xb:})/(1-normal({xb:}))),
> instruments(gear_ratio length headroom) onestep
    (output omitted)
. estimates store ml
. gmm (foreign - normal({xb:gear_ratio length headroom _cons})),
> instruments(gear_ratio length headroom) onestep
    (output omitted)
. estimates store gmm
. estimates table ml gmm, b se
\begin{tabular}{r|rr}
\hline Variable & \multicolumn{1}{|c}{ml} & \multicolumn{1}{c}{ gmm } \\
\hline gear_ratio & 2.9586277 & 2.8489213 \\
& .64042341 & .63570246 \\
length & -.02148933 & -.02056033 \\
& .01382043 & .01396954 \\
headroom & .01136927 & .02240761 \\
& .27278528 & .2849891 \\
_cons & -6.0222289 & -5.8595615 \\
& 3.5594588 & 3.5188028 \\
\hline
\end{tabular}

The coefficients on gear_ratio and length are close for the two estimators. The GMM estimate of the coefficient on headroom is twice that of the maximum likelihood estimate, though the relatively large standard errors imply that this difference is not significant. You can verify that the coefficients in the column marked "ml" match those you would obtain with probit. We have not discussed the differences among standard errors based on the various GMM and maximum-likelihood covariance matrix estimators to avoid tedious algebra. However, you can verify that the robust covariance matrix after one-step GMM estimation differs by only a finite-sample adjustment factor of \((N / N-1)\) from the robust covariance matrix reported by probit. Both the maximum likelihood and GMM probit estimators require the normality assumption, and the maximum likelihood estimator is efficient if that normality assumption is correct; therefore, in this example, there is no reason to prefer the GMM estimator.

We can modify (8) easily to allow for endogenous regressors. Suppose that \(x_{j}\) is endogenous in the sense that \(E\left(u \mid x_{j}\right) \neq 0\). Then, (8) is no longer a valid moment condition. However, suppose we have some variables other than \(\mathbf{x}\) such that \(E(u \mid \mathbf{z})=\mathbf{0}\). We can instead use the moment conditions
\[
E(\mathbf{z} u)=E\left[\mathbf{z}\left\{y-\exp \left(\mathbf{x}^{\prime} \boldsymbol{\beta}\right)\right\}\right]=\mathbf{0}
\]

As usual, if some elements of \(\mathbf{x}\) are exogenous, then they should appear in \(\mathbf{z}\) as well.

\section*{> Example 8: Exponential regression with endogenous regressors}

Returning to the model discussed in example 6, we treat income as endogenous; unobservable factors that determine a person's income may also affect the number of times a person visits a doctor. We use a person's age and race as instruments. These are valid instruments if we believe that age and race influence a person's income but do not have a direct impact on the number of doctor visits. (Whether this belief is justified is another matter; we test that belief in [R] gmm postestimation.) Because we have more instruments (seven) than parameters (five), we have an overidentified model. Therefore, the choice of weight matrix does matter. We will use the default two-step GMM estimator. In the first step, we will use a weight matrix that assumes the errors are i.i.d. In the second step, we will use a weight matrix that assumes heteroskedasticity. When you specify twostep, these are the defaults for the first- and second-step weight matrices, so we do not have to use the winitial() or wmatrix () options. We will again obtain a robust VCE, which is also the default. We type
```

. use https://www.stata-press.com/data/r18/docvisits
. gmm (docvis - exp({xb:private chronic female income _cons})),
> instruments(private chronic female age black hispanic) twostep
Step 1
Iteration 0: GMM criterion Q(b) = 16.910173
Iteration 1: GMM criterion Q(b) = .82276104
Iteration 2: GMM criterion Q(b) = . }2183203
Iteration 3: GMM criterion Q(b) = . }1268593
Iteration 4: GMM criterion Q(b) = . }1267236
Iteration 5: GMM criterion Q(b) = . }1267236
Step 2
Iteration 0: GMM criterion Q(b) = .00234641
Iteration 1: GMM criterion Q(b) = .00215957
Iteration 2: GMM criterion Q(b) = .00215911
Iteration 3: GMM criterion Q(b) = .00215911

```
```

GMM estimation

```
Number of parameters \(=5\)
Number of moments = 7
Initial weight matrix: Unadjusted \(\quad\) Number of obs \(=4,412\)
GMM weight matrix: Robust
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline & Coefficient & Robust std. err. & z & \(P>|z|\) & [95\% conf. & interval] \\
\hline private & . 535335 & . 1599039 & 3.35 & 0.001 & . 2219291 & . 8487409 \\
\hline chronic & 1.090126 & . 0617659 & 17.65 & 0.000 & . 9690668 & 1.211185 \\
\hline female & . 6636579 & . 0959884 & 6.91 & 0.000 & . 4755241 & . 8517918 \\
\hline income & . 0142855 & . 0027162 & 5.26 & 0.000 & . 0089618 & . 0196092 \\
\hline _cons & -. 5983477 & . 138433 & -4.32 & 0.000 & -. 8696713 & -. 327024 \\
\hline
\end{tabular}

Instruments for equation 1: private chronic female age black hispanic _cons
Once we control for the endogeneity of income, we find that its coefficient has quadrupled in size. Additionally, access to private insurance has less of an impact on the number of doctor visits and gender has more of an impact.

\section*{- Technical note}

Although you may be tempted to try, you cannot, as you can in a Poisson model, replace \(\mathbf{x}\) in the moment conditions for the probit (or logit) model with a vector of instruments, \(\mathbf{z}\), if you have endogenous regressors. See Wilde (2008).

Mullahy (1997) considers a slightly more complicated version of the exponential regression model that incorporates nonadditive unobserved heterogeneity. His model can be written as
\[
y_{i}=\exp \left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\right) \eta_{i}+\epsilon_{i}
\]
where \(\eta_{i}>0\) is an unobserved heterogeneity term that may be correlated with \(\mathbf{x}_{i}\). One result from his article is that instead of using the additive moment condition (8), we can use the multiplicative moment condition
\[
E\left\{\mathbf{z} \frac{y-\exp \left(\mathbf{x}^{\prime} \boldsymbol{\beta}\right)}{\exp \left(\mathbf{x}^{\prime} \boldsymbol{\beta}\right)}\right\}=E\left[\mathbf{z}\left\{y \exp \left(-\mathbf{x}^{\prime} \boldsymbol{\beta}\right)-1\right\}\right]=\mathbf{0}
\]

Windmeijer and Santos Silva (1997) discuss the use of additive versus multiplicative moment conditions with endogenous regressors and note that a set of instruments that satisfies the additive moment conditions will not also satisfy the multiplicative moment conditions. They remark that the decision about which to use is an empirical issue that can at least partially be settled by using the test of overidentifying restrictions that is implemented by estat overid after gmm to see whether the instruments for a given model are valid. See [R] gmm postestimation for information on the test of overidentifying restrictions.

\section*{Specifying derivatives}

By default, gmm calculates derivatives numerically, and the method used produces accurate results for the vast majority of applications. However, if you refit the same model repeatedly or else have the derivatives available, then gmm will run more quickly if you supply it with analytic derivatives.

When you use the interactive version of gmm, you specify derivatives using substitutable expressions in much the same way you specify the residual equations. There are three rules you must follow:
1. As with the substitutable expressions that define residual equations, you bind parameters of the model in curly braces: \{b0\}, \{param\}, etc.
2. You must specify a derivative for each parameter that appears in each residual equation. If a parameter does not appear in a residual equation, then you do not specify a derivative for that parameter in that residual equation.
3. If you declare a linear combination in an equation, then you specify a derivative with respect to that linear combination. gmm applies the chain rule to obtain the derivatives with respect to the individual parameters encompassed by that linear combination.

We illustrate with several examples.
> Example 9: Derivatives for a single-equation model
Consider a simple exponential regression model with one exogenous regressor and a constant term. We have
\[
u_{i}=y_{i}-\exp \left(\beta_{0}+\beta_{1} x_{i}\right)
\]

Now,
\[
\frac{\partial u_{i}}{\partial \beta_{0}}=-\exp \left(\beta_{0}+\beta_{1} x_{i}\right) \quad \text { and } \quad \frac{\partial u_{i}}{\partial \beta_{1}}=-x_{i} \exp \left(\beta_{0}+\beta_{1} x_{i}\right)
\]

In Stata, we type
```

. gmm (docvis - exp({b0} + {b1}*income)), instruments(income)
> deriv(/b0 = -1*exp({b0} + {b1}*income))
> deriv(/b1 = -1*income*exp({b0}+{b1}*income)) onestep
Step 1
Iteration 0: GMM criterion Q(b) = 9.1548611
Iteration 1: GMM criterion Q(b) = 3.5146131
Iteration 2: GMM criterion Q(b) = .01344695
Iteration 3: GMM criterion Q(b) = 3.690e-06
Iteration 4: GMM criterion Q(b) = 4.606e-13
Iteration 5: GMM criterion Q(b) = 1.502e-26
note: model is exactly identified.
GMM estimation
Number of parameters = 2
Number of moments = 2
Initial weight matrix: Unadjusted Number of obs = 4,412

```
\begin{tabular}{c|rcccccc}
\hline & \multicolumn{3}{|c}{ Robust } & & & \\
& Coefficient & std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline /b0 & 1.204888 & .0462355 & 26.06 & 0.000 & 1.114268 & 1.295507 \\
/b1 & .0046702 & .0009715 & 4.81 & 0.000 & .0027662 & .0065743 \\
\hline
\end{tabular}

Instruments for equation 1: income _cons

Notice how we specified the derivative() option for each parameter. We simply specified a slash, the name of the parameter, an equal sign, then a substitutable expression that represents the derivative. Because our model has only one residual equation, we do not need to specify equation numbers in the derivative() options.

When you specify a linear combination of variables, your derivative should be with respect to the entire linear combination. For example, say we have the residual equation
\[
u_{i}=y-\exp \left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}+\beta_{0}\right)
\]
for which we would type
```

.gmm (y - exp({xb: x1 x2 x3} + {b0}) ...

```

Then, in addition to the derivative \(\partial u_{i} / \partial \beta_{0}\), we are to compute and specify
\[
\frac{\partial u_{i}}{\partial \mathbf{x}_{i}^{\prime} \boldsymbol{\beta}}=-\exp \left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}+\beta_{0}\right)
\]

Using the chain rule, \(\partial u_{i} / \partial \beta_{j}=\partial u_{i} / \partial\left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\right) \times \partial\left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\right) / \partial \beta_{j}=-x_{i j} \exp \left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}+\beta_{0}\right)\). Stata does this last calculation automatically. It knows the variables in the linear combination, so all it needs is the derivative of the residual equation with respect to the linear combination. This allows you to change the variables in your linear combination without having to change the derivatives.
\(>\) Example 10: Derivatives with a linear combination
We refit the model described in the example illustrating exponential regression with endogenous regressors, now providing analytic derivatives. We type
```

. gmm (docvis - exp({xb:private chronic female income _cons})),
> instruments(private chronic female age black hispanic)
> derivative(/xb = -1*exp({xb:}))
Step 1
Iteration 0: GMM criterion Q(b) = 16.910173
Iteration 1: GMM criterion Q(b) = .82270871
Iteration 2: GMM criterion Q(b) = . 21831995
Iteration 3: GMM criterion Q(b) = . 12685934
Iteration 4: GMM criterion Q(b) = . }1267236
Iteration 5: GMM criterion Q(b) = . }1267236
Step 2
Iteration 0: GMM criterion Q(b) = .00234641
Iteration 1: GMM criterion Q(b) = .00215957
Iteration 2: GMM criterion Q(b) = .00215911
Iteration 3: GMM criterion Q(b) = .00215911
GMM estimation
Number of parameters = 5
Number of moments = 7
Initial weight matrix: Unadjusted Number of obs = 4,412
GMM weight matrix: Robust

|  | Robust <br> std. err. |  |  |  | z | $\mathrm{P}>\|\mathrm{z}\|$ |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
|  | Coefficient | [95\% conf. interval] |  |  |  |  |
| private | .535335 | .159904 | 3.35 | 0.001 | .221929 | .848741 |
| chronic | 1.090126 | .0617659 | 17.65 | 0.000 | .9690668 | 1.211185 |
| female | .6636579 | .0959885 | 6.91 | 0.000 | .475524 | .8517918 |
| income | .0142855 | .0027162 | 5.26 | 0.000 | .0089618 | .0196092 |
| _cons | -.5983477 | .138433 | -4.32 | 0.000 | -.8696714 | -.327024 |

Instruments for equation 1: private chronic female age black hispanic _cons

```

In the first derivative() option, we specified the name of the linear combination, xb , instead of an individual parameter's name. We already declared the variables of our linear combination in the substitutable expression for the residual equation, so in our substitutable expressions for the derivatives, we can use the shorthand notation \(\{x b:\}\) to refer to it.

Our point estimates are identical to those we obtained earlier. The standard errors and confidence intervals differ by only trivial amounts.

\section*{Exponential regression models with panel data}

In addition to supporting cross-sectional and time-series data, gmm also works with panel-data models. Here we illustrate gmm's panel-data capabilities by expanding our discussion of exponential regression models to allow for panel data. This also provides us the opportunity to demonstrate the moment-evaluator program version of gmm. Our discussion is based on Blundell, Griffith, and Windmeijer (2002). Also see Wooldridge (1999) for further discussion of nonlinear panel-data models.

First, we expand (7) for panel data. With individual heterogeneity term \(\eta_{i}\), we have
\[
E\left(y_{i t} \mid \mathbf{x}_{i t}, \eta_{i}\right)=\exp \left(\mathbf{x}_{i t}^{\prime} \boldsymbol{\beta}+\eta_{i}\right)=\mu_{i t} \nu_{i}
\]
where \(\mu_{i t}=\exp \left(\mathbf{x}_{i t}^{\prime} \boldsymbol{\beta}\right)\) and \(\nu_{i}=\exp \left(\eta_{i}\right)\). Note that there is no constant term in this model, because its effect cannot be disentangled from \(\nu_{i}\). With an additive idiosyncratic error term, we have the regression model
\[
y_{i t}=\mu_{i t} \nu_{i}+\epsilon_{i t}
\]

We do not impose the assumption \(E\left(\mathbf{x}_{i t} \eta_{i}\right)=\mathbf{0}\), so \(\eta_{i}\) can be considered a fixed effect in the sense that it may be correlated with the regressors.

As discussed by Blundell, Griffith, and Windmeijer (2002), if \(\mathbf{x}_{i t}\) is strictly exogenous, meaning \(E\left(\mathbf{x}_{i t} \epsilon_{i s}\right)=\mathbf{0}\) for all \(t\) and \(s\), then we can estimate the parameters of the model by using the sample moment conditions
\[
\begin{equation*}
\sum_{i} \sum_{t} \mathbf{x}_{i t}\left(y_{i t}-\mu_{i t} \frac{\bar{y}_{i}}{\bar{\mu}_{i}}\right)=\mathbf{0} \tag{9}
\end{equation*}
\]
where \(\bar{y}_{i}\) and \(\bar{\mu}_{i}\) are the means of \(y_{i t}\) and \(\mu_{i t}\) for panel \(i\), respectively. Because \(\bar{\mu}_{i}\) depends on the parameters of the model, it must be recomputed each time gmm needs to evaluate the residual equation. Therefore, we cannot use the substitutable expression version of gmm. Instead, we must use the moment-evaluator program version.

The moment-evaluator program version of gmm functions much like the function-evaluator program versions of nl and nlsur. The program you write is passed one or more variables to be filled in with the residuals evaluated at the parameter values specified in an option passed to your program. For the fixed-effects Poisson model with strictly exogenous regressors, our first crack at a function-evaluator program is
```

program gmm_poi
version 18.0 // (or version 18.5 for StataNow)
syntax varlist if, at(name)
quietly {
tempvar mu mubar ybar
generate double 'mu' = exp(x1*'at'[1,1] + x2*'at'[1,2] ///
+ x3*'at'[1,3]) 'if'
egen double 'mubar' = mean('mu') 'if', by(id)
egen double 'ybar' = mean(y) 'if', by(id)
replace 'varlist' = y - 'mu'*'ybar'/'mubar' 'if'
}

```
end

You can save your program in an ado-file named name. ado, where name is the name you use for your program; here we would save the program in the ado-file gmm_poi.ado. Alternatively, if you are working from within a do-file, you can simply define the program before calling gmm. The syntax statement declares we are expecting to receive a varlist that will contain the names of variables whose values we are to replace with the values of the residual equations and an if expression that will mark the estimation sample; because our model has one residual equation, varlist will consist of one variable. at () is a required option to our program, and it will contain the name of a matrix containing the parameter values at which we are to evaluate the residual equation. All moment-evaluator programs must accept the varlist, if condition, and at () option.

The first part of our program computes \(\mu_{i t}\). In the model we will fit shortly, we have three regressors, named \(x 1, x 2\), and \(x 3\). The 'at' vector will have three elements, one for each of those variables. Notice that we included 'if' at the end of each statement that affects variables to restrict the computations to the relevant estimation sample. The two egen statements compute \(\bar{\mu}_{i}\) and \(\bar{y}_{i}\); in the example dataset we will use shortly, the panel variable is named id, and for simplicity, we hardcoded that variable into our program as well. Finally, we compute the residual equation, which is the portion of (9) bound in parentheses.

\section*{> Example 11: Panel Poisson with strictly exogenous regressors}

To fit our model, we type
```

. use https://www.stata-press.com/data/r18/poisson1
. gmm gmm_poi, nequations(1) parameters(b1 b2 b3)
> instruments(x1 x2 x3, noconstant) vce(cluster id) onestep
Step 1
Iteration 0: GMM criterion Q(b) = 51.99142
Iteration 1: GMM criterion Q(b) = .04345191
Iteration 2: GMM criterion Q(b) = 8.720e-06
Iteration 3: GMM criterion Q(b) = 7.115e-13
Iteration 4: GMM criterion Q(b) = 5.130e-27
note: model is exactly identified.
GMM estimation
Number of parameters = 3
Number of moments = 3
Initial weight matrix: Unadjusted Number of obs = 409
(Std. err. adjusted for 45 clusters in id)

|  | Robust |  |  |  | P>\|z| | [95\% conf. interval] |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
|  | Coefficient | std. err. | z | P> |  |  |
| /b1 | 1.94866 | .1000265 | 19.48 | 0.000 | 1.752612 | 2.144709 |
| /b2 | -2.966119 | .0923592 | -32.12 | 0.000 | -3.14714 | -2.785099 |
| /b3 | 1.008634 | .1156561 | 8.72 | 0.000 | .781952 | 1.235315 |

```

Instruments for equation 1: x1 x2 x3
All three of our regressors are strictly exogenous, so they can serve as their own regressors. There is no constant term in the model (it would be unidentified), so we exclude a constant term from our list of instruments. We have one residual equation as indicated by nequations (1), and we have three parameters, named b1, b2, and b3. The order in which you declare parameters in the parameters() option determines the order in which they appear in the 'at' vector in the moment-evaluator program. We specified vce(cluster id) to obtain standard errors that allow for correlation among observations within each panel.

The program we just wrote is sufficient to fit the model to the poisson1 dataset, but if we want to fit that model to other datasets, we need to change the variable names and perhaps account for having a different number of parameters as well. Despite those limitations, if you just want to fit a single model, that program is adequate.

Next, we take advantage of the ability to specify full equation names in the parameters() option and rewrite our evaluator program so that we can more easily change the variables in our model. This approach is particularly useful if some of the residual equations are linear in the parameters because then we can use matrix score (see [P] matrix score) to evaluate those moments.

Our new evaluator program is
```

program gmm_poieq
version 18.0 // (or version 18.5 for StataNow)
syntax varlist if, at(name)
quietly {
tempvar mu mubar ybar
matrix score double 'mu' = 'at' 'if', eq(\#1)
replace 'mu' = exp('mu')
egen double 'mubar' = mean('mu') 'if', by(id)
egen double 'ybar' = mean(y) 'if', by(id)
replace 'varlist' = y - 'mu'*'ybar'/'mubar' 'if'
}

```
end

Rather than using generate to compute the temporary variable 'mu', we used matrix score to obtain the linear combination \(\mathbf{x}_{i t}^{\prime} \boldsymbol{\beta}\) and then called replace to compute \(\exp \left(\mathrm{x}_{i t}^{\prime} \boldsymbol{\beta}\right)\).

\section*{> Example 12: Panel Poisson using matrix score}

To fit our model, we type
```

. use https://www.stata-press.com/data/r18/poisson1
. gmm gmm_poieq, nequations(1) parameters({y:x1 x2 x3})
> instruments(x1 x2 x3, noconstant) vce(cluster id) onestep
Step 1
Iteration 0: GMM criterion Q(b) = 51.99142
Iteration 1: GMM criterion Q(b) = .04345191
Iteration 2: GMM criterion Q(b) = 8.720e-06
Iteration 3: GMM criterion Q(b) = 7.115e-13
Iteration 4: GMM criterion Q(b) = 5.130e-27
note: model is exactly identified.
GMM estimation
Number of parameters = 3
Number of moments = 3
Initial weight matrix: Unadjusted Number of obs = 409

```
(Std. err. adjusted for 45 clusters in id)
\begin{tabular}{r|rrrrrr}
\hline & \multicolumn{4}{c}{ Robust } & P \\
& Coefficient & std. err. & z & P>|z| & [95\% conf. interval] \\
\hline x1 & 1.94866 & .1000265 & 19.48 & 0.000 & 1.752612 & 2.144709 \\
x2 & -2.966119 & .0923592 & -32.12 & 0.000 & -3.14714 & -2.785099 \\
x3 & 1.008634 & .1156561 & 8.72 & 0.000 & .781952 & 1.235315 \\
\hline
\end{tabular}

Instruments for equation 1: x1 x2 x3
Instead of specifying simple parameter names in the parameters() option, we specified a linear combination name and the variables associated with that combination. We named our linear combination y, but you could use any valid Stata name. When we use this syntax, the rows of the coefficient table are grouped by the equation names.

Say we wanted to refit our model using just x1 and x3 as regressors. We do not need to make any changes to gmm_poieq. We just change the specification of the parameters() option:
```

. gmm gmm_poieq, nequations(1) parameters(y:x1 y:x3)
> instruments(x1 x3, noconstant) vce(cluster id) onestep

```

In this evaluator program, we have still hardcoded the name of the dependent variable. The next two examples include methods to tackle that shortcoming.

\section*{Technical note}

Say we specify the parameters() option like this:
```

. gmm ..., parameters({y1:x1 x2 _cons} {y2:_cons} {y3:x1 _cons})

```

Then, the 'at' vector passed to our program will have the following column names attached to it:
```

'at'[1,6]
y1: y1: y1: y2: y3: y3:

```

Typing
```

. matrix score double eq1 = 'at', eq(\#1)

```
is equivalent to typing
```

. generate double eq1 = x1*'at'[1,1] + x2*'at'[1,2] + 'at'[1,3]

```
with one important difference. If we change some of the variables in the parameters() option when we call gmm, matrix score will compute the correct linear combination. If we were to use the generate statement instead, then every time we wanted to change the variables in our model, we would have to modify that statement as well.

The command
. matrix score double alpha = 'at', eq(\#2)
is equivalent to
```

. scalar alpha = 'at'[1,4]

```

Thus, even if you specify linear combination and variable names in the parameters() option, you can still have scalar parameters in your model.

When past values of the idiosyncratic error term affect the value of a regressor, we say that regressor is predetermined. When one or more regressors are predetermined, sample moment condition (8) is no longer valid. However, Chamberlain (1992) shows that a simple alternative is to consider moment conditions of the form
\[
\begin{equation*}
\sum_{i} \sum_{t=2}^{T} \mathbf{x}_{i, t-1}\left(y_{i, t-1}-\mu_{i, t-1} \frac{y_{i t}}{\mu_{i t}}\right)=\mathbf{0} \tag{10}
\end{equation*}
\]

Also see Wooldridge (1997) and Windmeijer (2000) for other moment conditions that can be used with predetermined regressors.

\section*{Example 13: Panel Poisson with predetermined regressors}

Here we refit the previous model, treating all the regressors as predetermined and using the moment conditions in (10). Our moment-evaluator program is
```

program gmm_poipre
version 18.0 // (or version 18.5 for StataNow)
syntax varlist if, at(name) mylhs(varlist)
quietly {
tempvar mu
matrix score double 'mu' = 'at' 'if', eq(\#1)
replace 'mu' = exp('mu')
replace 'varlist' = L.'mylhs' - L.'mu'*'mylhs'/'mu' 'if'
}
end

```

To compute the residual equation, we used lag-operator notation so that Stata properly handles gaps in our panel dataset. We also made our program accept an additional option that we will use to pass in the dependent variable. When we specify this option in our gmm statement, it will get passed to our evaluator program because gmm will not recognize the option as one of its own. Equation (10) shows that we are to use the first lags of the regressors as instruments, so we type
```

. gmm gmm_poipre, mylhs(y) nequations(1) vce(cluster id) onestep
> parameters({y:x1 x2 x3}) instruments(L.(x1 x2 x3), noconstant)
note: 45 missing values returned for equation 1 at initial values.
Step 1
Iteration 0: GMM criterion Q(b) = 76.652367
Iteration 1: GMM criterion Q(b) = 1.9118192
Iteration 2: GMM criterion Q(b) = .06634724
Iteration 3: GMM criterion Q(b) = .0000233
Iteration 4: GMM criterion Q(b) = 2.998e-12
Iteration 5: GMM criterion Q(b) = 4.834e-26
note: model is exactly identified.
GMM estimation
Number of parameters = 3
Number of moments = 3
Initial weight matrix: Unadjusted Number of obs = 364
(Std. err. adjusted for 45 clusters in id)

|  | Coefficient | Robust std. err. | z | $P>\|z\|$ | [95\% conf. | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| x 1 | 2.088246 | . 2513626 | 8.31 | 0.000 | 1.595584 | 2.580907 |
| x 2 | -2.905504 | . 2133908 | -13.62 | 0.000 | -3.323742 | -2.487266 |
| x3 | 1.121081 | . 201654 | 5.56 | 0.000 | . 7258459 | 1.516315 |

```

Instruments for equation 1: L.x1 L.x2 L.x3
Here, like earlier with strictly exogenous regressors, the number of instruments equals the number of parameters, so there is no gain to using the two-step or iterated estimator. However, if you do have more instruments than parameters, you will most likely want to use one of those other estimators instead.

The note at the top of the output is given because we have 45 panels in our dataset. Our residual equation includes lagged terms and therefore cannot be evaluated for the first time period within each panel. Notes like this can be ignored once you know why they occurred. If you receive a note that you were not expecting, you should first investigate the cause of the note before trusting the results.

Instead of making our program accept the mylhs () option, we could have used Stata's coleq macro function to determine the dependent variable based on the column names attached to the 'at' vector; see \([\mathrm{P}]\) macro. Then, we could refit our model with a different dependent variable by changing the lcname used in the parameters() option. In the next example, we take this approach.

In the previous example, we used \(\mathbf{x}_{i, t-1}\) as instruments. A more efficient GMM estimator would also use \(\mathbf{x}_{i, t-2}, \mathbf{x}_{i, t-3}, \ldots, \mathbf{x}_{i, 1}\) as instruments in period \(t\) as well. gmm's xtinstruments() option allows you to specify instrument lists that grow as \(t\) increases. Later, we discuss the xtinstruments () option in detail in the context of linear dynamic panel-data models.

When a regressor is contemporaneously correlated with the idiosyncratic error term, we say that regressor is endogenous. Windmeijer (2000) shows that we can use the moment condition
\[
\sum_{i} \sum_{t=3}^{T} \mathbf{x}_{i, t-2}\left(\frac{y_{i t}}{\mu_{i t}}-\frac{y_{i, t-1}}{\mu_{i, t-1}}\right)
\]

Here we use the second lag of the endogenous regressor as an instrument. If a variable is strictly exogenous, it can of course serve as its own instrument.

\section*{> Example 14: Panel Poisson with endogenous regressors}

Here we refit the model, treating \(x 3\) as endogenous and \(x 1\) and \(x 2\) as strictly exogenous. Our moment-evaluator program is
```

program gmm_poiend
version 18.0 // (or version 18.5 for StataNow)
syntax varlist if, at(name)
quietly {
tempvar mu
matrix score double 'mu' = 'at' 'if', eq(\#1)
replace 'mu' = exp('mu')
local mylhs : coleq 'at'
local mylhs : word 1 of 'mylhs'
replace 'varlist' = 'mylhs'/'mu' - L.'mylhs'/L.'mu' 'if'
}
end

```

Now, we call gmm using x 1 , x 2 , and L2. x 3 as instruments:
```

. use https://www.stata-press.com/data/r18/poisson2
. gmm gmm_poiend, nequations(1) vce(cluster id) onestep
> parameters(y:x1 y:x2 y:x3) instruments(x1 x2 L2.x3, noconstant)
note: 500 missing values returned for equation 1 at initial values.
Step 1
Iteration 0: GMM criterion Q(b) = 61.832288
Iteration 1: GMM criterion Q(b) = .03402584
Iteration 2: GMM criterion Q(b) = .01101288
Iteration 3: GMM criterion Q(b) = 6.339e-06
Iteration 4: GMM criterion Q(b) = 1.620e-12
Iteration 5: GMM criterion Q(b) = 1.312e-25
note: model is exactly identified.
GMM estimation
Number of parameters = 3
Number of moments = 3
Initial weight matrix: Unadjusted Number of obs = 3,766

```
(Std. err. adjusted for 500 clusters in id)
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline & Coefficient & Robust std. err. & z & \(P>|z|\) & [95\% conf. & interval] \\
\hline x1 & 1.8141 & . 2688318 & 6.75 & 0.000 & 1.2872 & 2.341001 \\
\hline x 2 & -2.982671 & . 1086666 & -27.45 & 0.000 & -3.195653 & -2.769688 \\
\hline x3 & 4.126518 & 6.369334 & 0.65 & 0.517 & -8.357147 & 16.61018 \\
\hline
\end{tabular}

Instruments for equation 1: x1 x2 L2.x3
The note at the top of the output is given because that we have 500 panels in our dataset. As in the previous example, our residual equation includes lagged terms and therefore cannot be evaluated for the first time period within each panel. Instead of using just \(\mathbf{x}_{i, t-2}\) as an instrument, we could use all further lags of \(\mathbf{x}_{i t}\) as instruments as well.

\section*{Rational-expectations models}

Macroeconomic models typically assume that agents' expectations about the future are formed rationally. By rational expectations, we mean that agents use all information available when forming their forecasts, so the forecast error is uncorrelated with the information available when the forecast was made. Say that at time \(t\), people make a forecast, \(\widehat{y}_{t+1}\), of variable \(y\) in the next period. If \(\Omega_{t}\) denotes all available information at time \(t\), then rational expectations implies that \(E\left\{\left(\widehat{y}_{t+1}-y_{t+1}\right) \mid \Omega_{t}\right\}=0\). If \(\Omega_{t}\) denotes observable variables such as interest rates or prices, then this conditional expectation can serve as the basis of a moment condition for GMM estimation.

\section*{> Example 15: Fitting a Euler equation}

In a well-known article, Hansen and Singleton (1982) consider a model of portfolio decision making and discuss parameter estimation using GMM. We will consider a simple example with one asset in which the agent can invest. A consumer wants to maximize the present value of his or her lifetime utility derived from buying a good. On the one hand, the consumer is impatient, so he or she would rather buy today than wait until tomorrow. On the other hand, by buying less today, the consumer can invest more money, earning more interest that can be used to buy more of the good tomorrow. Thus, there is a tradeoff between having cake today or sacrificing a bit today to have more cake tomorrow.

If we assume a specific form for the agent's utility function, known as the constant relative-risk aversion utility function, we can show that the Euler equation is
\[
E\left[\mathbf{z}_{t}\left\{1-\beta\left(1+r_{t+1}\right)\left(c_{t+1} / c_{t}\right)^{-\gamma}\right\}\right]=\mathbf{0}
\]
where \(\beta\) and \(\gamma\) are the parameters to estimate, \(r_{t}\) is the return to the financial asset, and \(c_{t}\) is consumption in period \(t\). \(\beta\) measures the agent's discount factor. If \(\beta\) is near 1 , the agent is patient and is more willing to forgo consumption this period. If \(\beta\) is close to 0 , the agent is less patient and prefers to consume more now. The parameter \(\gamma\) characterizes the agent's utility function. If \(\gamma=0\), the utility function is linear. As \(\gamma\) tends toward 1 , the utility function tends toward \(u=\log (c)\).

We have data on three-month Treasury bills \(\left(r_{t}\right)\) and consumption expenditures \(\left(c_{t}\right)\). As instruments, we will use lagged rates of return and past growth rates of consumption. We will use the two-step estimator and a weight matrix that allows for heteroskedasticity and autocorrelation up to four lags with the Bartlett kernel. In Stata, we type
```

. use https://www.stata-press.com/data/r18/cr
. generate cgrowth = c / L.c
(1 missing value generated)
. gmm (1 - {b=1}*(1+F.r)*(F.c/c)^(-1*{gamma=1})),
> inst(L.r L2.r cgrowth L.cgrowth) wmat(hac nw 4) twostep
note: 1 missing value returned for equation 1 at initial values.
Step 1
Iteration 0: GMM criterion Q(b) = .00226482
Iteration 1: GMM criterion Q(b) = .00054369
Iteration 2: GMM criterion Q(b) = .00053904
Iteration 3: GMM criterion Q(b) = .00053904
Step 2
Iteration 0: GMM criterion Q(b) = .0600729
Iteration 1: GMM criterion Q(b) = .0596369
Iteration 2: GMM criterion Q(b) = . }059636
GMM estimation
Number of parameters = 2
Number of moments = 5
Initial weight matrix: Unadjusted Number of obs = 239
GMM weight matrix: HAC Bartlett 4

```
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline & Coefficient & \[
\begin{gathered}
\text { HAC } \\
\text { std. err. }
\end{gathered}
\] & z & \(P>|z|\) & \multicolumn{2}{|l|}{[95\% conf. interval]} \\
\hline /b & . 9204617 & . 0134646 & 68.36 & 0.000 & . 8940716 & . 9468518 \\
\hline /gamma & -4.222361 & 1.473895 & -2.86 & 0.004 & -7.111143 & -1.333579 \\
\hline
\end{tabular}

HAC standard errors based on Bartlett kernel with 4 lags.
Instruments for equation 1: L.r L2.r cgrowth L.cgrowth _cons
The note at the top of the output is given because the forward operator in our substitutable expression says that residuals cannot be computed for the last observation. In addition, two observations are omitted because the L2.r instrument has missing values in the first two time periods. Therefore, of the 242 observations in our dataset, 239 are used to fit the model. Our estimate of \(\beta\) is near 1 , in line with expectations and published results. However, our estimate of \(\gamma\) implies risk-loving behavior and therefore a poorly specified model.

Also notice our use of the forward operator to refer to the values of \(r\) and \(c\) one period ahead; time-series operators are allowed in substitutable expressions as long as you have previously tsset (see [TS] tsset) your data. See [U] 13.10 Time-series operators for more information on time-series operators.

\section*{System estimators}

In many economic models, two or more variables are determined jointly through a system of simultaneous equations. Indeed, some of the earliest work in econometrics, including that of the Cowles Commission, was centered around estimation of the parameters of simultaneous equations. The 2SLS and IV estimators we have already discussed are used in some circumstances to estimate such parameters. Here we focus on the joint estimation of all the parameters of systems of equations, and we begin with the well-known three-stage least-squares (3SLS) estimator.

Recall that the 2SLS estimator is based on the moment conditions \(E(\mathbf{z u})=\mathbf{0}\). The 2SLS estimator can be used to estimate the parameters of one equation of a system of structural equations. Moreover, with the 2SLS estimator, we do not even need to specify the structural relationship among all the endogenous variables; we need to specify only the equation on which interest focuses and simply assume reduced-form relationships among the endogenous regressors of the equation of interest and the exogenous variables of the model. If we are willing to specify the complete system of structural equations, then assuming our model is correctly specified, by estimating all the equations jointly, we can obtain estimates that are more efficient than equation-by-equation 2SLS.

In [R] reg3, we fit a simple two-equation macroeconomic model,
\[
\begin{align*}
\text { consump } & =\beta_{0}+\beta_{1} \text { wagepriv }+\beta_{2} \text { wagegovt }+\epsilon_{1}  \tag{11}\\
\text { wagepriv } & =\beta_{3}+\beta_{4} \text { consump }+\beta_{5} \text { govt }+\beta_{6} \text { capital1 }+\epsilon_{2} \tag{12}
\end{align*}
\]
where consump represents aggregate consumption; wagepriv and wagegovt are total wages paid by the private and government sectors, respectively; govt is government spending; and capitall is the previous period's capital stock. We are not willing to assume that \(\epsilon_{1}\) and \(\epsilon_{2}\) are independent, so we must treat both consump and wagepriv as endogenous. Suppose that a random shock makes \(\epsilon_{2}\) positive. Then by (12), wagepriv will be higher than it otherwise would. Moreover, \(\epsilon_{1}\) will be either higher or lower, depending on the correlation between it and \(\epsilon_{2}\). The shock to \(\epsilon_{2}\) has made both wagepriv and \(\epsilon_{1}\) move, which implies that in (11), wagepriv is an endogenous regressor. A similar argument shows that consump is an endogenous regressor in the second equation. In our model, wagegovt, govt, and capital1 are all exogenous variables.

Let \(\mathbf{z}_{1}\) and \(\mathbf{z}_{2}\) denote the instruments for the first and second equations, respectively; we will discuss what comprises them shortly. We have two sets of moment conditions:
\[
E\left\{\begin{array}{c}
\mathbf{z}_{1}\left(\text { consump }-\beta_{0}-\beta_{1} \text { wagepriv }-\beta_{2} \text { wagegovt }\right)  \tag{13}\\
\mathbf{z}_{2}\left(\text { wagepriv }-\beta_{3}-\beta_{4} \text { consump }-\beta_{5} \text { govt }-\beta_{6} \text { capital1 }\right)
\end{array}\right\}=\mathbf{0}
\]

One of the defining characteristics of 3SLS is that the errors are homoskedastic conditional on the instrumental variables. Using this assumption, we have
\[
E\left[\left\{\begin{array}{l}
\mathbf{z}_{1} \epsilon_{1}  \tag{14}\\
\mathbf{z}_{2} \epsilon_{2}
\end{array}\right\}\left\{\begin{array}{ll}
\mathbf{z}_{1}^{\prime} \epsilon_{1} & \left.\mathbf{z}_{2}^{\prime} \epsilon_{2}\right\}
\end{array}\right]=\left\{\begin{array}{ll}
\sigma_{11} E\left(\mathbf{z}_{1} \mathbf{z}_{1}^{\prime}\right) & \sigma_{12} E\left(\mathbf{z}_{1} \mathbf{z}_{2}^{\prime}\right) \\
\sigma_{21} E\left(\mathbf{z}_{2} \mathbf{z}_{1}^{\prime}\right) & \sigma_{22} E\left(\mathbf{z}_{2} \mathbf{z}_{2}^{\prime}\right)
\end{array}\right\}\right.
\]
where \(\sigma_{i j}=\operatorname{cov}\left(\epsilon_{i}, \epsilon_{j}\right)\). Let \(\boldsymbol{\Sigma}\) denote the \(2 \times 2\) matrix with typical element \(\sigma_{i j}\).
The second defining characteristic of the 3SLS estimator is that it uses all the exogenous variables as instruments for all equations; here \(\mathbf{z}_{1}=\mathbf{z}_{2}=\) (wagegovt, govt, capital1, 1 ), where the 1 indicates a constant term. From our discussion on the weight matrix and two-step estimation, we want to use the sample analogue of the matrix inverse of the right-hand side of (14) as our weight matrix.

To implement the 3SLS estimator, we apparently need to know \(\boldsymbol{\Sigma}\) or at least have a consistent estimator of it. The solution is to fit (11) and (12) by 2SLS, use the sample residuals \(\widehat{\epsilon_{1}}\) and \(\widehat{\epsilon_{2}}\) to estimate \(\boldsymbol{\Sigma}\), then estimate the parameters of (13) via GMM by using the weight matrix just discussed.

\section*{> Example 16: 3SLS estimation}

3SLS is easier to do using gmm than it sounds. The 3SLS estimator is a two-step GMM estimator. In the first step, we do the equivalent of 2SLS on each equation, and then we compute a weight matrix based on (14). Finally, we perform a second step of GMM with this weight matrix.

In Stata, we type
```

. use https://www.stata-press.com/data/r18/klein, clear
. gmm (eq1: consump - {xb: wagepriv wagegovt _cons})
> (eq2: wagepriv - {xc: consump govt capital1 _cons}),
> instruments(eq1: wagegovt govt capital1)
> instruments(eq2: wagegovt govt capital1)
> winitial(unadjusted, independent) wmatrix(unadjusted) twostep
Step 1
Iteration 0: GMM criterion Q(b) = 4195.4487
Iteration 1: GMM criterion Q(b) = . 22175631
Iteration 2: GMM criterion Q(b) = . 22175631 (backed up)
Step 2
Iteration 0: GMM criterion Q(b) = .09716589
Iteration 1: GMM criterion Q(b) = .07028208
Iteration 2: GMM criterion Q(b) = . 07028208
GMM estimation
Number of parameters = 7
Number of moments = 8
Initial weight matrix: Unadjusted Number of obs = 22
GMM weight matrix: Unadjusted

```
\begin{tabular}{lr|rlrlrr}
\hline & & Coefficient & Std. err. & \(z\) & P>|z| & [95\% conf. interval] \\
\hline xb & & & & & & & \\
& wagepriv & .8012754 & .1279329 & 6.26 & 0.000 & .5505314 & 1.052019 \\
& wagegovt & 1.029531 & .3048424 & 3.38 & 0.001 & .432051 & 1.627011 \\
& _cons & 19.3559 & 3.583772 & 5.40 & 0.000 & 12.33184 & 26.37996 \\
\hline xc & & & & & & & \\
& consump & .4026076 & .2567312 & 1.57 & 0.117 & -.1005764 & .9057916 \\
& govt & 1.177792 & .5421253 & 2.17 & 0.030 & .1152461 & 2.240338 \\
& capital1 & -.0281145 & .0572111 & -0.49 & 0.623 & -.1402462 & .0840173 \\
& _cons & 14.63026 & 10.26693 & 1.42 & 0.154 & -5.492552 & 34.75306 \\
\hline
\end{tabular}

Instruments for equation eq1: wagegovt govt capital1 _cons
Instruments for equation eq2: wagegovt govt capital1 _cons
The independent suboption of the winitial() option tells gmm to assume that the residuals are independent across moment conditions; this suboption sets \(\sigma_{21}=\sigma_{12}=0\) in (14). Assuming both homoskedasticity and cross-equation independence is equivalent to fitting the two equations of our model independently by 2 SLS. The wmatrix () option controls how the weight matrix is computed on the basis of the first-step parameter estimates before the second step of estimation; here we request a weight matrix that assumes conditional homoskedasticity but that does not impose the cross-equation independence like the initial weight matrix we used. In this example, we also illustrated how to name residual equations and how equation names can be used in the instruments () option. Our results are identical to those in \([R]\) reg3.

We could have specified our instruments with the syntax
```

instruments(wagegovt govt capital1)

```
because gmm uses the variables listed in the instruments() option for all equations unless you specify which equations the list of instruments is to be used with. However, we wanted to emphasize
that the same instruments are being used for both equations; in a moment, we will discuss an estimator that does not use the same instruments in all equations.

In the previous example, if we omit the twostep option, the resulting coefficients will be equivalent to equation-by-equation 2SLS, which Wooldridge (2010, 216) calls the "system 2SLS estimator". Eliminating the twostep option makes the wmatrix() option irrelevant, so that option can be eliminated as well.

So far, we have developed the traditional 3SLS estimator. Wooldridge (2010, chap. 8) discusses the "GMM 3SLS" estimator, which extends the traditional 3SLS estimator by allowing for heteroskedasticity and different instruments for different equations.

Generalizing (14) to an arbitrary number of equations, we have
\[
\begin{equation*}
E\left(\mathbf{Z}^{\prime} \boldsymbol{\epsilon} \boldsymbol{\epsilon}^{\prime} \mathbf{Z}\right)=E\left(\mathbf{Z}^{\prime} \boldsymbol{\Sigma} \mathbf{Z}\right) \tag{15}
\end{equation*}
\]
where
\[
\mathbf{Z}=\left[\begin{array}{cccc}
\mathbf{z}_{1} & \mathbf{0} & \cdots & \mathbf{0} \\
\mathbf{0} & \mathbf{z}_{2} & \cdots & \mathbf{0} \\
\vdots & \vdots & \ddots & \vdots \\
\mathbf{0} & \mathbf{0} & \cdots & \mathbf{z}_{m}
\end{array}\right]
\]
and \(\boldsymbol{\Sigma}\) is now \(m \times m\). Equation (15) is the multivariate analogue of a homoskedasticity assumption; for each equation, the error variance is constant for all observations, as is the covariance between any two equations' errors.

We can relax this homoskedasticity assumption by considering different weight matrices. For example, if we continue to assume that observations are independent but not necessarily identically distributed, then by specifying wmatrix (robust), we would obtain a weight matrix that allows for heteroskedasticity:
\[
\widehat{W}=\frac{1}{N} \sum_{i} \mathbf{Z}_{i}^{\prime} \widehat{\epsilon}_{i} \hat{\epsilon}_{i}^{\prime} \mathbf{Z}_{i}
\]

This is the weight matrix in Wooldridge's \((2010,218)\) Procedure 8.1 , "GMM with Optimal Weighting Matrix". By default, gmm would report standard errors based on his covariance matrix (8.27); specifying vce (unadjusted) would provide the optimal GMM standard errors. If you have multiple observations for each individual or firm in your dataset, you could specify wmatrix (cluster id), where id identifies individuals or firms. This would allow arbitrary within-individual correlation, though it does not account for an individual-specific fixed or random effect. In both cases, we would continue to use winitial (unadjusted, independent) so that the first-step estimates are the system 2SLS estimates.

Wooldridge (2010, sec. 9.6) discusses instances where it is necessary to use different instruments in different equations. The GMM 3SLS estimator with different instruments in different equations but with conditional homoskedasticity is what Hayashi \((2000,275)\) calls the "full-information instrumental variables efficient" (FIVE) estimator. Implementing the FIVE estimator is easy with gmm. For example, say we have a two-equation system where kids, age, income, and education are all valid instruments for the first equation but where education is not a valid instrument for the second equation. Then, our syntax would take the form
```

gmm (rexp ) (rexp 2), instruments(1:kids age income education)
instruments(2:kids age income)

```

The following syntax is equivalent:
```

gmm (rexp ) ( rexp ), instruments(kids age income)
instruments(1:education)

```

Because we did not specify a list of equations in the second example's first instruments () option, those variables are used as instruments in both equations. You can use whichever syntax you prefer. The first requires a bit more typing but is arguably more transparent.

If all the regressors in the model are exogenous, then the traditional 3SLS estimator is the seemingly unrelated regression (SUR) estimator. Here you would specify all the regressors as instruments.

\section*{Dynamic panel-data models}

Commands in Stata that work with panel data expect the data to be in the "long" format, meaning that each row of the dataset consists of one subobservation that is a member of a logical observation (represented by the panel identifier variable). See [D] reshape for a discussion of the long versus "wide" data forms. gmm is no exception in this respect when used with panel data. From a theoretical perspective, however, it is sometimes easier to view GMM estimators for panel data as system estimators in which we have \(N\) observations on a system of \(T\) equations, where \(N\) and \(T\) are the number of observations and panels, respectively, rather than a single-equation estimator with \(N T\) observations. Usually, each of the \(T\) equations will in fact be the same, though we will want to specify different instruments for each of these equations.

In a dynamic panel-data model, lagged values of the dependent variable are included as regressors. Here we consider a simple model with one lag of the dependent variable \(y\) as a regressor and a vector of strictly exogenous regressors, \(\mathbf{x}_{i t}\) :
\[
\begin{equation*}
y_{i t}=\rho y_{i, t-1}+\mathbf{x}_{i t}^{\prime} \boldsymbol{\beta}+u_{i}+\epsilon_{i t} \tag{16}
\end{equation*}
\]
\(u_{i}\) can be either a fixed- or a random-effect term in the sense that we do not require \(\mathbf{x}_{i t}\) to be independent of it. Even with the assumption that \(\epsilon_{i t}\) is i.i.d., the presence of both \(y_{i, t-1}\) and \(u_{i}\) in (16) renders both the standard fixed- and random-effects estimators to be inconsistent because of the well-known Nickell (1981) bias. OLS regression of \(y_{i t}\) on \(y_{i, t-1}\) and \(\mathbf{x}_{i t}\) also produces inconsistent estimates because \(y_{i, t-1}\) will be correlated with the error term.

\section*{- Technical note}

Stata has the xtabond, xtdpd, and xtdpdsys commands (see [XT] xtabond, [XT] xtdpd, and [XT] xtdpdsys) to fit equations like (16); for everyday use, those commands are preferred because they offer features such as Windmeijer (2005) bias-corrected standard errors to account for the bias of traditional two-step GMM standard errors seen in dynamic panel-data models and, being linear estimators, only require you to specify variable names instead of complete equations. However, using gmm has several pedagogical advantages, including the ability to tie those model-specific commands into a more general framework, a clear illustration of how certain types of instrument matrices for panel-data models are formed, and demonstrations of several advanced features of gmm.

First-differencing (16) removes the panel-specific \(u_{i}\) term:
\[
\begin{equation*}
y_{i t}-y_{i, t-1}=\rho\left(y_{i, t-1}-y_{i, t-2}\right)+\left(\mathbf{x}_{i t}-\mathbf{x}_{i, t-1}\right)^{\prime} \boldsymbol{\beta}+\left(\epsilon_{i t}-\epsilon_{i, t-1}\right) \tag{17}
\end{equation*}
\]

However, now \(\left(y_{i, t-1}-y_{i, t-2}\right)\) is correlated with \(\left(\epsilon_{i t}-\epsilon_{i, t-1}\right)\). Thus, we need an instrument that is correlated with the former but not the latter. The lagged variables in (17) mean that the equation is not estimable for \(t<3\), so consider when \(t=3\). We have
\[
\begin{equation*}
y_{i 3}-y_{i 2}=\rho\left(y_{i 2}-y_{i 1}\right)+\left(\mathbf{x}_{i 3}-\mathbf{x}_{i 2}\right)^{\prime} \boldsymbol{\beta}+\left(\epsilon_{i 3}-\epsilon_{i 2}\right) \tag{18}
\end{equation*}
\]

In the Arellano-Bond (1991) estimator, lagged levels of the dependent variable are used as instruments. With our assumption that the \(\epsilon_{i t}\) are i.i.d., (16) intimates that \(y_{i 1}\) can serve as an instrumental variable when we fit (18).

Next, consider (17) when \(t=4\). We have
\[
y_{i 4}-y_{i 3}=\rho\left(y_{i 3}-y_{i 2}\right)+\left(\mathbf{x}_{i 4}-\mathbf{x}_{i 3}\right)^{\prime} \boldsymbol{\beta}+\left(\epsilon_{i 4}-\epsilon_{i 3}\right)
\]

Now, (16) shows that both \(y_{i 1}\) and \(y_{i 2}\) are uncorrelated with the error term \(\left(\epsilon_{i 4}-\epsilon_{i 3}\right)\), so we have two instruments available. For \(t=5\), you can show that \(y_{i 1}, y_{i 2}\), and \(y_{i 3}\) can serve as instruments. As may now be apparent, one of the key features of these dynamic panel-data models is that the available instruments depend on the time period, \(t\), as was the case for some of the panel Poisson models we considered earlier. Because the \(\mathbf{x}_{i t}\) are strictly exogenous by assumption, they can serve as their own instruments.

The initial weight matrix that is appropriate for the GMM dynamic panel-data estimator is slightly more involved than the unadjusted matrix that we have used in most of our previous examples and that assumes the errors are i.i.d. First, rewrite (17) for panel \(i\) as
\[
\mathbf{y}_{i}-\mathbf{y}_{i}^{L}=\rho\left(\mathbf{y}_{i}^{L}-\mathbf{y}_{i}^{L L}\right)+\left(\mathbf{X}_{i}-\mathbf{X}_{i}^{L}\right) \boldsymbol{\beta}+\left(\boldsymbol{\epsilon}_{i}-\boldsymbol{\epsilon}_{i}^{L}\right)
\]
where \(\mathbf{y}_{i}=\left(y_{i 3}, \ldots, y_{i T}\right)\) and \(\mathbf{y}_{i}^{L}=\left(y_{i 2}, \ldots, y_{i, T-1}\right), \mathbf{y}_{i}^{L L}=\left(y_{i 1}, \ldots, y_{i, T-2}\right)\), and \(\mathbf{X}_{i}, \mathbf{X}_{i}^{L}, \boldsymbol{\epsilon}_{i}\), and \(\boldsymbol{\epsilon}_{i}^{L}\) are defined analogously. Let \(\mathbf{Z}\) denote the full matrix of instruments for panel \(i\), including the variables specified in both the instruments() and xtinstruments() options; the exact structure is detailed in Methods and formulas.

By assumption, \(\epsilon_{i t}\) is i.i.d., so the first difference \(\left(\epsilon_{i t}-\epsilon_{i, t-1}\right)\) is necessarily autocorrelated with correlation -0.5 . Therefore, we should not use a weight matrix that assumes the errors are independent. For dynamic panel-data models, we can show that the appropriate initial weight matrix is
\[
\widehat{\mathbf{W}}=\left(\frac{1}{N} \sum_{i} \mathbf{Z}_{i}^{\prime} \mathbf{H}_{D} \mathbf{Z}_{i}\right)^{-1}
\]
where
\[
\mathbf{H}_{D}=\left[\begin{array}{cccccc}
1 & -0.5 & 0 & \ldots & 0 & 0 \\
-0.5 & 1 & -0.5 & \ldots & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & 0 & \ldots & 1 & -0.5 \\
0 & 0 & 0 & \ldots & -0.5 & 1
\end{array}\right]
\]

We can obtain this initial weight matrix by specifying winitial (xt D). The letter D indicates that the equation we are estimating is specified in first differences.

\section*{> Example 17: Arellano-Bond estimator}

Say we want to fit the model
\[
\begin{equation*}
\mathrm{n}_{i t}=\rho \mathrm{n}_{i, t-1}+\beta_{1} \mathrm{w}_{i t}+\beta_{2} \mathrm{w}_{i, t-1}+\beta_{3} \mathbf{k}_{i t}+\beta_{4} \mathrm{k}_{i, t-1}+u_{i}+\epsilon_{i t} \tag{19}
\end{equation*}
\]
where we assume that \(\mathrm{w}_{i t}\) and \(\mathrm{k}_{i t}\) are strictly exogenous. First-differencing, our residual equation is
\[
\begin{align*}
\epsilon_{i t}^{*}=\left(\epsilon_{i t}-\epsilon_{i, t-1}\right)= & \mathrm{n}_{i t}-\mathrm{n}_{i, t-1}-\rho\left(\mathrm{n}_{i, t-1}-\mathrm{n}_{i, t-2}\right)-\beta_{1}\left(\mathrm{w}_{i t}-\mathrm{w}_{i, t-1}\right) \\
& -\beta_{2}\left(\mathrm{w}_{i, t-1}-\mathrm{w}_{i, t-2}\right)-\beta_{3}\left(\mathrm{k}_{i t}-\mathrm{k}_{i, t-1}\right)-\beta_{4}\left(\mathrm{k}_{i, t-1}-\mathrm{k}_{i, t-2}\right) \tag{20}
\end{align*}
\]

In Stata, we type
```

. use https://www.stata-press.com/data/r18/abdata
. gmm (D.n - {rho}*LD.n - {xb:D.w LD.w D.k LD.k}),
> xtinstruments(n, lags(2/.)) instruments(D.w LD.w D.k LD.k, noconstant)
> deriv(/rho = -1*LD.n) deriv(/xb = -1) winitial(xt D) onestep
Step 1
Iteration 0: GMM criterion Q(b) = .0011455
Iteration 1: GMM criterion Q(b) = .00009103
Iteration 2: GMM criterion Q(b) = .00009103
GMM estimation
Number of parameters = 5
Number of moments= = 32
Initial weight matrix: XT D Number of obs = 751

```
                            (Std. err. adjusted for 140 clusters in id)

Instruments for equation 1:
    XT-style: L(2/.).n
    Standard: D.w LD.w D.k LD.k

Because w and k are strictly exogenous, we specified their variants that appear in (20) in the instruments () option; because there is no constant term in the model, we specified noconstant to omit the constant from the instrument list.

We specified xtinstruments ( n , lags(2/.)) to tell gmm what instruments to use for the lagged dependent variable included as a regressor in (19). On the basis of our previous discussion, lags two and higher of \(\mathrm{n}_{i t}\) can serve as instruments. The lags (2/.) suboption tells gmm that the first available instrument for \(\mathrm{n}_{i t}\) is the lag-two value \(\mathrm{n}_{i, t-2}\). The "." tells gmm to use all further lags of \(\mathrm{n}_{i t}\) as instruments as well. The instrument matrices in dynamic panel-data models can become large if the dataset has many time periods per panel. In those cases, you could specify, for example, lags (2/4) to use just lags two through four instead of using all available lags.

Our results are identical to those we would obtain using xtabond with the syntax
xtabond n L(0/1).w L(0/1).k, lags(1) noconstant vce(robust)

If we had left off the vce (robust) option in our call to xtabond, we would have had to specify vce (unadjusted) in our call to gmm to obtain the same standard errors.

\section*{Technical note}
gmm automatically excludes observations for which there are no valid observations for the panelstyle instruments. However, it keeps in the estimation sample those observations for which fewer than the maximum number of instruments you requested are available. For example, if you specify the lags (2/4) suboption, you have requested three instruments, but gmm will keep observations even if only one or two instruments are available.

\section*{> Example 18: Two-step Arellano-Bond estimator}

Here we refit the model from example 17, using the two-step GMM estimator.
```

. gmm (D.n - {rho}*LD.n - {xb:D.w LD.w D.k LD.k}),
> xtinstruments(n, lags(2/.)) instruments(D.w LD.w D.k LD.k, noconstant)
> deriv(/rho = -1*LD.n) deriv(/xb = -1) winitial(xt D) wmatrix(robust)
> vce(unadjusted)
Step 1
Iteration 0: GMM criterion Q(b) = .0011455
Iteration 1: GMM criterion Q(b) = .00009103
Iteration 2: GMM criterion Q(b) = .00009103
Step 2
Iteration 0: GMM criterion Q(b) = .44107941
Iteration 1: GMM criterion Q(b) = .4236729
Iteration 2: GMM criterion Q(b) = . 4236729 (backed up)
GMM estimation
Number of parameters = 5
Number of moments = 32
Initial weight matrix: XT D Number of obs = 751
GMM weight matrix: Robust

```
\begin{tabular}{lr|rrrrrr}
\hline & & Coefficient & Std. err. & z & P>|z| & [95\% conf. interval] \\
\hline rho & & & & & & & \\
& _cons & .8044783 & .0534763 & 15.04 & 0.000 & .6996667 & .90929 \\
\hline xb & & & & & & & \\
& W & & & & & & \\
& D1. & -.5154978 & .0335506 & -15.36 & 0.000 & -.5812557 & -.4497399 \\
& LD. & .4059309 & .0637294 & 6.37 & 0.000 & .2810235 & .5308384 \\
& k & & & & & & \\
& D1. & .3556204 & .0390892 & 9.10 & 0.000 & .2790071 & .4322337 \\
& LD. & -.2204521 & .046439 & -4.75 & 0.000 & -.3114709 & -.1294332 \\
\hline
\end{tabular}
Instruments for equation 1:
    XT-style: L(2/.).n
    Standard: D.w LD.w D.k LD.k

Our results match those you would obtain with the command
```

xtabond n L(0/1).(w k), lags(1) noconstant twostep

```

\section*{- Technical note}

If we had specified vce(robust) in our call to gmm, we would have obtained the traditional sandwich-based robust covariance matrix, but our standard errors would not match those we would obtain by specifying vce (robust) with the xtabond command. The xtabond, xtdpd, and xtdpdsys commands implement a bias-corrected robust VCE for the two-step GMM dynamic panel-data estimator. Traditional VCEs computed after the two-step dynamic panel-data estimator have been shown to exhibit often-severe bias; see Windmeijer (2005).

Neither of the two dynamic panel-data examples (17 and 18) we have fit so far include a constant term. When a constant term is included, the dynamic panel-data estimator is in fact a two-equation system estimator. For notational simplicity, consider a simple model containing just a constant term and one lag of the dependent variable:
\[
y_{i t}=\alpha+\rho y_{i, t-1}+u_{i}+\epsilon_{i t}
\]

First-differencing to remove the \(u_{i}\) term, we have
\[
\begin{equation*}
y_{i t}-y_{i, t-1}=\rho\left(y_{i, t-1}-y_{i, t-2}\right)+\left(\epsilon_{i t}-\epsilon_{i, t-1}\right) \tag{21}
\end{equation*}
\]

This has also eliminated the constant term. If we assume \(E\left(u_{i}\right)=0\), which is reasonable if a constant term is included in the model, then we can recover \(\alpha\) by including the moment condition
\[
\begin{equation*}
y_{i t}=\alpha+\rho y_{i, t-1}+\epsilon_{i t}^{\prime} \tag{22}
\end{equation*}
\]
where \(\epsilon_{i t}^{\prime}=u_{i}+\epsilon_{i t}\). The parameter \(\rho\) continues to be identified by (21), so the only instrument we use with (22) is a constant term. As before, the error term \(\left(\epsilon_{i, t}-\epsilon_{i, t-1}\right)\) is necessarily autocorrelated with correlation coefficient -0.5 , though the error term \(\epsilon_{i t}^{\prime}\) is white noise. Therefore, our initial weight matrix should be
\[
\widehat{\mathbf{W}}=\left(\frac{1}{N} \sum_{i} \mathbf{Z}_{i}^{\prime} \mathbf{H} \mathbf{Z}_{i}\right)^{-1}
\]
where
\[
\mathbf{H}=\left[\begin{array}{cc}
\mathbf{H}_{D} & \mathbf{0} \\
\mathbf{0} & \mathbf{I}
\end{array}\right]
\]
and \(\mathbf{I}\) is a conformable identity matrix.
One complication arises concerning the relevant estimation sample. Looking at (21), we apparently lose the first two observations from each panel because of the presence of \(y_{i, t-2}\), but in (22), we need to sacrifice only one observation for \(y_{i, t-1}\). For most multiple-equation models, we need to use the same estimation sample for all equations. However, in dynamic panel-data models, we can use more observations to fit the equation in level form [(22) here] than the equation in first differences [equation (21)]. To request this behavior, we specify the nocommonesample option to gmm. That option tells gmm to use as many observations as possible for each equation, ignoring the loss of observations due to lagging or differencing.

\section*{> Example 19: Arellano-Bond estimator with constant term}

Here we fit the model
\[
\mathrm{n}_{i t}=\alpha+\rho \mathrm{n}_{i, t-1}+u_{i}+\epsilon_{i t}
\]

Without specifying derivatives, our command would be
```

. gmm (D.n - {rho}*LD.n) (n - {alpha} - {rho}*L.n),
> xtinstruments(1: n, lags(2/.)) instruments(1:, noconstant) onestep
> winitial(xt DL) vce(unadj) nocommonesample

```

We would specify winitial (xt DL) to obtain the required initial weight matrix. The notation DL indicates that our first residual equation is in first differences and that the second residual equation is in levels (not first-differenced). We exclude a constant in the instrument list for the first equation because first-differencing removed the constant term. Because we do not specify the instruments() option for the second residual equation, a constant is used by default.

This example also provides us the opportunity to illustrate how to specify derivatives for multipleequation GMM models. Within the derivative() option, instead of specifying just the parameter name, now you must specify the equation name or number, a slash, and the parameter name to which the derivative applies. In Stata, we type
```

. gmm (D.n - {rho}*LD.n) (n - {alpha} - {rho}*L.n),
> xtinstruments(1: n, lags(2/.)) instruments(1:, noconstant)
> derivative(1/rho = -1*LD.n) derivative(2/alpha = -1)
> derivative(2/rho = -1*L.n) winitial(xt DL) vce(unadj) nocommonesample onestep
Step 1
Iteration 0: GMM criterion Q(b) = .09894466
Iteration 1: GMM criterion Q(b) = .00023508
Iteration 2: GMM criterion Q(b) = .00023508
GMM estimation
Number of parameters = 2
Number of moments = 29
Initial weight matrix: XT DL Number of obs = *

```
\begin{tabular}{r|rrrrrr}
\hline & Coefficient & Std. err. & \(z\) & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline /rho & 1.023349 & .0608293 & 16.82 & 0.000 & .9041259 & 1.142572 \\
/alpha & -.0690864 & .0660343 & -1.05 & 0.295 & -.1985112 & .0603384 \\
\hline
\end{tabular}
* Number of observations for equation 1: 751

Number of observations for equation 2: 891
```

Instruments for equation 1:
XT-style: L(2/.).n
Instruments for equation 2:
Standard: _cons

```

These results are identical to those we would obtain by typing
xtabond \(n\), lags(1)
Because we specified nocommonesample, gmm did not report the number of observations used in the header of the output. In this dataset, there are in fact 1,031 observations on 140 panels. In the second equation, the presence of the lagged value of n reduces the sample size for that equation to \(1031-140=891\). In the first equation, we lose the first two observations per panel because of lagging and differencing, which leads to 751 usable observations. These tallies are listed after the coefficient table in the output.
- Technical note

Specifying
xtinstruments(x1 x2 x3, lags(1/3))
differs from
```

instruments(L(1/3).(x1 x2 x3))

```
in how observations are excluded from the estimation sample. When you use the latter syntax, gmm must exclude the first three observations from each panel when computing the residual equation: you requested that three lags of each regressor be used as instruments, so the first residual that could be interacted with those instruments is the one for \(t=4\). On the other hand, when you use xtinstruments (), you are telling gmm that you would like to use up to the first three lags of x 1 , x 2 , and x 3 as instruments but that using just one lag is acceptable. Because most panel datasets have a relatively modest number of observations per panel, dynamic instrument lists are typically used so that the number of usable observations is maximized. Dynamic instrument lists also accommodate the fact that there are more valid instruments for later time periods than earlier time periods.

Specifying panel-style instruments using the xtinstruments () option also affects how the standard instruments specified in the instruments() option are treated. To illustrate, we will suppose that we have a balanced panel dataset with \(T=5\) observations per panel and that we specify
```

. gmm ..., xtinstruments(w, lags(1/2)) instruments(x)

```

We will lose the first observation because we need at least one lag of w to serve as an instrument. Our instrument matrix for panel \(i\) will therefore be
\[
\mathbf{Z}_{i}=\left[\begin{array}{cccc}
w_{i 1} & 0 & 0 & 0  \tag{23}\\
0 & w_{i 1} & 0 & 0 \\
0 & w_{i 2} & 0 & 0 \\
0 & 0 & w_{i 2} & 0 \\
0 & 0 & w_{i 3} & 0 \\
0 & 0 & 0 & w_{i 3} \\
0 & 0 & 0 & w_{i 4} \\
x_{i 2} & x_{i 3} & x_{i 4} & x_{i 5} \\
1 & 1 & 1 & 1
\end{array}\right]
\]

The vector of ones in the final row represents the constant term implied by the instruments () option. Because we lost the first observation, the residual vector \(\mathbf{u}_{i}\) will be \(4 \times 1\). Thus, our moment conditions for the \(i\) th panel can be written in matrix notation as
\[
E\left\{\mathbf{Z}_{i} \mathbf{u}_{i}(\boldsymbol{\beta})\right\}=E\left\{\mathbf{Z}_{i}\left[\begin{array}{c}
u_{i 2}(\boldsymbol{\beta}) \\
u_{i 3}(\boldsymbol{\beta}) \\
u_{i 4}(\boldsymbol{\beta}) \\
u_{i 5}(\boldsymbol{\beta})
\end{array}\right]\right\}=\mathbf{0}
\]

The moment conditions corresponding to the final two rows of (23) say that
\[
E\left\{\sum_{t=2}^{T=4} x_{i t} u_{i t}(\boldsymbol{\beta})\right\}=0 \quad \text { and } \quad E\left\{\sum_{t=2}^{T=4} u_{i t}(\boldsymbol{\beta})\right\}=0
\]

Because we specified panel-style instruments with the xtinstruments() option, gmm no longer uses moment conditions for strictly exogenous variables of the form \(E\left\{x_{i t} u_{i t}(\beta)\right\}=0\) for each \(t\). Instead, the moment conditions now stipulate that the average (over \(t\) ) of \(x_{i t} u_{i t}(\boldsymbol{\beta})\) has expectation zero. This corresponds to the approach proposed by Arellano and Bond (1991, 280) and others.

When you request panel-style instruments with the xtinstruments() option, the number of instruments in the \(\mathbf{Z}_{i}\) matrix increases quadratically in the number of periods. The dynamic paneldata estimators we have discussed in this section are designed for datasets that contain a large number of panels and a modest number of time periods. When the number of time periods is large, estimators that use standard (non-panel-style) instruments are more appropriate.

We have focused on the Arellano-Bond dynamic panel-data estimator because of its relative simplicity. gmm can additionally fit any models that can be formulated using the xtdpd and xtdpdsys commands; see [XT] xtdpd and [XT] xtdpdsys. The key is to determine the appropriate instruments to use for the level and difference equations. You may find it useful to fit a version of your model with those commands to determine what instruments and XT-style instruments to use. We conclude this section with an example using the Arellano-Bover/Blundell-Bond estimator.
> Example 20: Arellano-Bover/Blundell-Bond estimator
We fit a small model that includes one lag of the dependent variable n as a regressor as well as the contemporaneous and first lag of w, which we assume are strictly exogenous. We could fit our model with xtdpdsys by using the syntax
xtdpdsys n L(0/1).w, lags(1) twostep

When we apply virtually all the syntax issues we have discussed so far, the equivalent gmm command is
```

. gmm (n - {rho}*L.n - {w}*w - {lagw}*L.w - {c})
> (D.n - {rho}*LD.n - {w}*D.w - {lagw}*LD.w),
> xtinst(1: D.n, lags(1/1)) xtinst(2: n, lags(2/.))
> inst(2: D.w LD.w, noconstant)
> deriv(1/rho = -1*L.n) deriv(1/w = -1*w)
> deriv(1/lagw = -1*L.w) deriv(1/c = -1)
> deriv(2/rho = -1*LD.n) deriv(2/w = -1*D.w)
> deriv(2/lagw = -1*LD.w)
> winit(xt LD) wmatrix(robust) vce(unadjusted) nocommonesample
Step 1
Iteration 0: GMM criterion Q(b) = . }1017033
Iteration 1: GMM criterion Q(b) = .00022772
Iteration 2: GMM criterion Q(b) = . 00022772
Step 2
Iteration 0: GMM criterion Q(b) = . 59965014
Iteration 1: GMM criterion Q(b) = . 56578186
Iteration 2: GMM criterion Q(b) = .56578186
GMM estimation
Number of parameters = 4
Number of moments = 39
Initial weight matrix: XT LD Number of obs = *
GMM weight matrix: Robust

```
\begin{tabular}{r|rrrrrr}
\hline & Coefficient & Std. err. & \multicolumn{1}{c}{z} & \(\mathrm{P}>|\mathrm{z}|\) & {\([95 \%\) conf. interval] } \\
\hline /rho & 1.122738 & .0206512 & 54.37 & 0.000 & 1.082263 & 1.163214 \\
/w & -.6719909 & .0246148 & -27.30 & 0.000 & -.7202351 & -.6237468 \\
/lagw & .571274 & .0403243 & 14.17 & 0.000 & .4922398 & .6503083 \\
/c & .154309 & .17241 & 0.90 & 0.371 & -.1836084 & .4922263 \\
\hline
\end{tabular}
* Number of observations for equation 1: 891

Number of observations for equation 2: 751
Instruments for equation 1:
XT-style: LD.n
Standard: _cons
Instruments for equation 2:
XT-style: L(2/.).n
Standard: D.w LD.w

\section*{Details of moment-evaluator programs}

In examples \(11,12,13\), and 14 , we used moment-evaluator programs to evaluate moment conditions that could not be specified using the interactive version of gmm. In example 13, we also showed how to pass additional information to an evaluator program. Here we discuss how to make moment-evaluator programs provide derivatives and accept weights.

The complete specification for a moment-evaluator program's syntax statement is
```

syntax varlist if [weight], at(name) options [derivatives(varlist)]

```

The macro 'varlist' contains the list of variables that we are to fill in with the values of our residual equations. The macro 'if' represents an if condition that restricts the estimation sample. The macro 'at' represents a vector containing the parameter values at which we are to evaluate our
residual equations. options represent other options that you specify in your call to gmm and want to have passed to your moment-evaluator programs. In example 13, we included the mylhs () option so that we could pass the name of the dependent variable to our evaluator program.

Two new elements of the syntax statement allow for weights and derivatives. weight specifies the types of weights your program allows. The interactive version of gmm allows for fweights, aweights, and pweights. However, unless you explicitly allow your moment evaluator program to accept weights, you cannot specify weights in your call to gmm with the moment-evaluator program version.

The derivatives() option is used to pass to your program a set of variables that you are to fill in with the derivatives of your residual equations with respect to the parameters.

To indicate that your program can calculate derivatives, you specify either the hasderivatives or the haslfderivatives option to gmm. The hasderivatives option indicates that your program calculates parameter-level derivatives; that method requires more work but can be applied to any GMM problem. The haslfderivatives option requires less work but can be used only when the model's residual equations satisfy certain restrictions and when you use the \{lcname: varlist\} syntax with the parameters() option.

We first consider how to write the derivative computation logic to work with the hasderivatives option and provide an example; then, we do the same for the haslfderivatives option.

Say that you specify \(k\) parameters in the nparameters() or parameters() option and \(q\) equations in the nequations() or equations() option and that you specify hasderivatives. Then, 'derivatives' will contain \(k \times q\) variables. The first \(k\) variables are for the derivatives of the first residual equation with respect to the \(k\) parameters, the second \(k\) variables are for the derivatives of the second residual equation, and so on.

\section*{> Example 21: Specifying derivatives with simple parameter names}

To focus on how to specify derivatives, we return to the simple moment-evaluator program we used in example 11, in which we had three regressors, and extend it to supply derivatives. The residual equation corresponding to moment condition (9) is
\[
u_{i t}(\boldsymbol{\beta})=y_{i t}-\mu_{i t} \frac{\bar{y}_{i}}{\bar{\mu}_{i}}
\]
where \(\mu_{i t}, \bar{\mu}_{i}\), and \(\bar{y}_{i}\) were defined previously. Now,
\[
\begin{equation*}
\frac{\partial}{\partial \beta_{j}} u_{i t}(\boldsymbol{\beta})=-\mu_{i t} \frac{\bar{y}_{i}}{\bar{\mu}_{i}^{2}}\left(x_{i t}^{(j)} \bar{\mu}_{i}-\frac{1}{T} \sum_{l=1}^{l=T} x_{i l}^{(j)} \mu_{i l}\right) \tag{24}
\end{equation*}
\]
where \(x_{i t}^{(j)}\) represents the \(j\) th element of \(\mathbf{x}_{i t}\).

Our moment-evaluator program is
```

program gmm_poideriv
version 18.0 // (or version 18.5 for StataNow)
syntax varlist if, at(name) [derivatives(varlist)]
quietly {
// Calculate residuals as before
tempvar mu mubar ybar
generate double 'mu' = exp(x1*'at'[1,1] + x2*'at'[1,2] ///
+ x3*'at'[1,3]) 'if'
egen double 'mubar' = mean('mu') 'if', by(id)
egen double 'ybar' = mean(y) 'if', by(id)
replace 'varlist' = y - 'mu'*'ybar'/'mubar' 'if'
// Did -gmm- request derivatives?
if "'derivatives'" == "" {
exit // no, so we are done
}
// Calculate derivatives
// We need the panel means of x1*mu, x2*mu, and x3*mu
tempvar work x1mubar x2mubar x3mubar
generate double 'work' = x1*'mu' 'if'
egen double 'x1mubar' = mean('work') 'if', by(id)
replace 'work' = x2*'mu' 'if'
egen double 'x2mubar' = mean('work') 'if', by(id)
replace 'work' = x3*'mu' 'if'
egen double 'x3mubar' = mean('work') 'if', by(id)
local d1: word 1 of 'derivatives'
local d2: word 2 of 'derivatives'
local d3: word 3 of 'derivatives'
replace 'd1' = -1*'mu'*'ybar'/'mubar'`2*(x1*'mubar' - 'x1mubar')     replace 'd2' = -1*'mu'*'ybar'/'mubar'`2*(x2*'mubar' - 'x2mubar')
replace 'd3' = -1*'mu'*'ybar'/'mubar'`2*(x3*'mubar' - 'x3mubar')
}
end

```

The derivatives() option is made optional in the syntax statement by placing it in square brackets. If gmm needs to evaluate your residual equations but does not need derivatives at that time, then the derivatives() option will be empty. In our program, we check to see whether that is the case and, if so, exit without calculating derivatives. As is often the case with \([\mathrm{R}] \mathrm{ml}\) as well, the portion of our program devoted to derivatives is longer than the code to compute the objective function.

The first part of our derivative code computes the term
\[
\begin{equation*}
\frac{1}{T} \sum_{l=1}^{l=T} x_{i l}^{(j)} \mu_{i l} \tag{25}
\end{equation*}
\]
for \(x_{i t}^{(j)}=\mathrm{x} 1, \mathrm{x} 2\), and, x 3 . The 'derivatives' macro contains three variable names corresponding to the three parameters of the 'at' matrix. We extract those names into local macros ' d 1 ', ' d 2 ', and ' d 3 ' and then fill in the variables those macros represent with the derivatives shown in (24).

With our program written, we fit our model by typing
```

. use https://www.stata-press.com/data/r18/poisson1, clear
. gmm gmm_poideriv, nequations(1) parameters(b1 b2 b3)
> instruments(x1 x2 x3, noconstant) vce(cluster id) onestep hasderivatives
Step 1
Iteration 0: GMM criterion Q(b) = 51.99142
Iteration 1: GMM criterion Q(b) = .04345191
Iteration 2: GMM criterion Q(b) = 8.720e-06
Iteration 3: GMM criterion Q(b) = 7.115e-13
Iteration 4: GMM criterion Q(b) = 5.130e-27
note: model is exactly identified.
GMM estimation

```
Number of parameters \(=3\)
Number of moments \(=3\)
Initial weight matrix: Unadjusted \(\quad\) Number of obs \(=409\)
(Std. err. adjusted for 45 clusters in id)
\begin{tabular}{r|rrrrrr}
\hline & \multicolumn{4}{|c}{\begin{tabular}{l} 
Robust \\
std. err.
\end{tabular}} & z & P>|z| \\
& Coefficient & [95\% conf. interval] \\
\hline /b1 & 1.94866 & .1000265 & 19.48 & 0.000 & 1.752612 & 2.144709 \\
/b2 & -2.966119 & .0923592 & -32.12 & 0.000 & -3.14714 & -2.785099 \\
\hline & 1.008634 & .1156561 & 8.72 & 0.000 & .781952 & 1.235315 \\
\hline
\end{tabular}

Instruments for equation 1: x1 x2 x3
Our results are identical to those in example 11. Another way to verify that our program calculates derivatives correctly would be to type
```

. gmm gmm_poideriv, nequations(1) parameters(b1 b2 b3)
> instruments(x1 x2 x3, noconstant) vce(cluster id) onestep

```

Without the hasderivatives or haslfderivatives option, gmm will not request derivatives from your program, even if it contains code to compute them. If you have trouble obtaining convergence with the hasderivatives or haslfderivatives option but do not have trouble without specifying one of them, then you need to recheck your derivatives.

After example 11, we remarked that the evaluator program would have to be changed to accommodate different regressors. We then showed how you can specify parameters using the syntax \{lcname: varlist\} and then use matrix score to compute linear combinations of variables. To specify derivatives when you specify parameters using this notation, ensure that your residual equations satisfy the "linear-form restriction" analogous to the restrictions of linear-form evaluators used by ml. See \([R] \mathbf{m l}\) and Pitblado, Poi, and Gould (2024) for more information about linear-form evaluators.

A GMM residual equation satisfies the linear-form restriction if the equation can be written in terms of a single observation in the dataset and if the equation for observation \(i\) does not depend on any observations \(j \neq i\). Cross-sectional models satisfy the linear-form restriction. Time-series models satisfy the linear-form restriction only when no lags or leads are used.

Panel-data models often do not satisfy the linear-form restriction. For example, recall moment condition (9) for a panel Poisson model. That residual equation included panel-level mean terms \(\bar{y}_{i}\) and \(\bar{\mu}_{i}\), so the residual equation for an individual observation depends on all the observations in the same panel.

When a residual equation does not satisfy the linear-form restriction, neither will its derivatives. To apply the chain rule, we need a way to multiply the lcname-level derivative by each of the variables
in the equation to obtain parameter-level derivatives. In (24), for example, there is no way to factor out each \(x_{i t}^{(j)}\) variable and obtain an lcname-level derivative that we then multiply by each of the \(x_{i t}^{(j)}\).

Suppose we do have a model with \(q=2\) moment conditions, both of which do satisfy the linear-form restriction, and we specify the parameters() option like this:
```

. gmm ..., parameters({eq1:x1 x2 _cons} {eq2:_cons} {eq3:x1 x2 _cons})

```

We have specified \(n=3\) lcnames in the parameters() option: eq1, eq2, and eq3. When we specify the haslfderivatives option, gmm will pass \(n \times q=3 \times 2=6\) variables in the derivatives() option. The first three variables are to be filled with
\[
\frac{\partial}{\partial \mathrm{eq} 1} u_{1 i}(\boldsymbol{\beta}) \quad \frac{\partial}{\partial \mathrm{eq} 2} u_{1 i}(\boldsymbol{\beta}) \quad \text { and } \quad \frac{\partial}{\partial \mathrm{eq} 3} u_{1 i}(\boldsymbol{\beta})
\]
where \(u_{1 i}(\boldsymbol{\beta})\) is the \(i\) th observation for the first moment equation. Then, the second three variables are to be filled with
\[
\frac{\partial}{\partial \mathrm{eq} 1} u_{2 i}(\boldsymbol{\beta}) \quad \frac{\partial}{\partial \mathrm{eq} 2} u_{2 i}(\boldsymbol{\beta}) \quad \text { and } \quad \frac{\partial}{\partial \mathrm{eq} 3} u_{2 i}(\boldsymbol{\beta})
\]
where \(u_{2 i}(\boldsymbol{\beta})\) is the second moment equation. In this example, we filled in a total of six variables with derivatives. If we instead used the hasderivatives option, we would have filled \(k \times q=7 \times 2=14\) variables; moreover, if we wanted to change the number of variables in our model, we would have modified our evaluator program.

\section*{\(>\) Example 22: Specifying derivatives with linear-form residual equations}

In examples 9 and 10, we showed how to specify derivatives with an exponential regression model when using the interactive version of gmm. Here we show how to write a moment-evaluator program for the exponential regression model, including derivatives.

The residual equation for observation \(i\) is
\[
u_{i}=y_{i}-\exp \left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\right)
\]
where \(\mathbf{x}_{i}\) may include a constant term. The derivative with respect to the linear combination \(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\) is
\[
\frac{\partial u_{i}}{\partial \mathbf{x}_{i}^{\prime} \boldsymbol{\beta}}=-\exp \left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\right)
\]

To verify that this residual equation satisfies the linear-form restriction, we see that for the \(j\) th element of \(\boldsymbol{\beta}\), we have
\[
\frac{\partial u_{i}}{\partial \beta_{j}}=-x_{i j} \exp \left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\right)=\frac{\partial u_{i}}{\partial \mathbf{x}_{i}^{\prime} \boldsymbol{\beta}} \times x_{i j}
\]
so that given \(\partial u_{i} / \partial \mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\), gmm can apply the chain rule to obtain the derivatives with respect to the individual parameters.

Our moment-evaluator program is
```

program gmm_poideriv2
version 18.0 // (or version 18.5 for StataNow)
syntax varlist if, at(name) [derivatives(varlist)]
quietly {
tempvar mu
matrix score double 'mu' = 'at' 'if', eq(\#1)
replace 'mu' = exp('mu')
local depvar : coleq 'at'
local depvar : word 1 of 'depvar'
replace 'varlist' = 'depvar' - 'mu' 'if'
// Did -gmm- request derivatives?
if "'derivatives'" == "" {
exit // no, so we are done
}
// Calculate derivatives
// The derivatives macro only has one variable
// for this model.
replace 'derivatives' = -1*'mu' 'if'
}
end

```

To fit our model of doctor visits treating income as an endogenous regressor, we type
```

. use https://www.stata-press.com/data/r18/docvisits
. gmm gmm_poideriv2, nequations(1)
> instruments(private chronic female age black hispanic)
> parameters({docvis:private chronic female income _cons}) haslfderivatives
Step 1
Iteration 0: GMM criterion Q(b) = 16.910173
Iteration 1: GMM criterion Q(b) = .82270871
Iteration 2: GMM criterion Q(b) = .21831995
Iteration 3: GMM criterion Q(b) = . }1268593
Iteration 4: GMM criterion Q(b) = . }1267236
Iteration 5: GMM criterion Q(b) = . }1267236
Step 2
Iteration 0: GMM criterion Q(b) = .00234641
Iteration 1: GMM criterion Q(b) = .00215957
Iteration 2: GMM criterion Q(b) = .00215911
Iteration 3: GMM criterion Q(b) = .00215911
GMM estimation
Number of parameters = 5
Number of moments = 7
Initial weight matrix: Unadjusted Number of obs = 4,412
GMM weight matrix: Robust

```
\begin{tabular}{r|rrrrrr}
\hline & \multicolumn{5}{|c}{\begin{tabular}{c} 
Robust \\
std. err.
\end{tabular}} & \multicolumn{1}{c}{\(z\)} \\
& Coefficient & P>|z| & [95\% conf. interval] \\
\hline private & .535335 & .159904 & 3.35 & 0.001 & .221929 & .848741 \\
chronic & 1.090126 & .0617659 & 17.65 & 0.000 & .9690668 & 1.211185 \\
female & .6636579 & .0959885 & 6.91 & 0.000 & .475524 & .8517918 \\
income & .0142855 & .0027162 & 5.26 & 0.000 & .0089618 & .0196092 \\
_cons & -.5983477 & .138433 & -4.32 & 0.000 & -.8696714 & -.327024 \\
\hline
\end{tabular}

Instruments for equation 1: private chronic female age black hispanic _cons
Our results match those shown in example 10.

We can change the variables in our model just by changing the parameters() and instruments () options; we do not need to make any changes to the moment-evaluator program, because we used linear-form derivatives.

Depending on your model, allowing your moment-evaluator program to accept weights may be as easy as modifying the syntax command to allow them, or it may require significantly more work. If your program uses only commands such as generate and replace, then just modifying the syntax command is all you need to do; gmm takes care of applying the weights to the observation-level residuals when computing the sample moments, derivatives, and weight matrices. On the other hand, if your moment-evaluator program computes residuals using statistics that depend on multiple observations, then you must apply the weights passed to your program when computing those statistics.

In our examples of panel Poisson with strictly exogenous regressors (11 and 21), we used the statistics \(\bar{\mu}_{i}\) and \(\bar{y}_{i}\) when computing the residuals. If we are to allow weights with our momentevaluator program, then we must incorporate those weights when computing \(\bar{\mu}_{i}\) and \(\bar{y}_{i}\). Moreover, looking at the derivative in (24), we see that the term highlighted in (25) is in fact a sample mean, so we must incorporate weights when computing it.

\section*{> Example 23: Panel Poisson with derivatives and weights}

Here we modify the program in example 21 to accept frequency weights. One complication arises: we had been using egen to compute \(\bar{\mu}_{i}\) and \(\bar{y}_{i}\). egen does not accept weights, so we must compute \(\bar{\mu}_{i}\) and \(\bar{y}_{i}\) ourselves, incorporating any weights the user may specify. Our program is
```

program gmm_poiderivfw
version 18.0 // (or version 18.5 for StataNow)
syntax varlist if [fweight/], at(name) [derivatives(varlist)]
quietly {
if "'exp'" == "" { // no weights
local exp 1 // weight each observation equally
}
// Calculate residuals as before
tempvar mu mubar ybar sumwt
generate double 'mu' = exp(x1*'at'[1,1] + x2*'at'[1,2] ///
+ x3*'at'[1,3]) 'if'
bysort id: generate double 'sumwt' = sum('exp')
by id: generate double 'mubar' = sum('mu'*'exp')
by id: generate double 'ybar' = sum(y*'exp')
by id: replace 'mubar' = 'mubar'[_N] / 'sumwt'[_N]
by id: replace 'ybar' = 'ybar'[_N] / 'sumwt' [_N]
replace 'varlist' = y - 'mu'*'ybar'/'mubar' 'if'
// Did -gmm- request derivatives?
if "'derivatives’" == "" {
exit // no, so we are done
}
// Calculate derivatives
// We need the panel means of x1*mu, x2*mu, and x3*mu
tempvar work x1mubar x2mubar x3mubar
generate double 'work' = x1*'mu' 'if'
by id: generate double 'x1mubar' = sum('work'*'exp')
by id: replace 'x1mubar' = 'x1mubar'[_N] / 'sumwt'[_N]
replace 'work' = x2*'mu' 'if'
by id: generate double 'x2mubar' = sum('work'*'exp')
by id: replace 'x2mubar' = 'x2mubar'[_N] / 'sumwt'[_N]
replace 'work' = x3*'mu' 'if'
by id: generate double 'x3mubar' = sum('work'*'exp')
by id: replace 'x3mubar' = 'x3mubar'[_N] / 'sumwt'[_N]

```
```

local d1: word 1 of 'derivatives'
local d2: word 2 of 'derivatives'
local d3: word 3 of 'derivatives'
replace 'd1' = -1*'mu'*'ybar'/'mubar'^2*(x1*'mubar' - 'x1mubar')
replace 'd2' = -1*'mu'*'ybar'/'mubar'^2*(x2*'mubar' - 'x2mubar')
replace 'd3' = -1*'mu'*'ybar'/'mubar'` 2*(x3*'mubar' - 'x3mubar')
}
end

```

Our syntax command now indicates that fweights are allowed. The first part of our code looks at the macro 'exp'. If it is empty, then the user did not specify weights in their call to gmm; and we set the macro equal to 1 so that we weight each observation equally. After we compute \(\mu_{i t}\), we calculate \(\bar{\mu}_{i}\) and \(\bar{y}_{i}\), accounting for weights. To compute frequency-weighted means for each panel, we just multiply each observation by its respective weight, sum over all observations in the panel, then divide by the sum of the weights for the panel. (See [U] 20.24 Weighted estimation for information on how to handle aweights and pweights.) We use the same procedure to compute the frequency-weighted variant of expression (25) in the derivative calculations. To use our program, we type
```

. use https://www.stata-press.com/data/r18/poissonwts
. gmm gmm_poiderivfw [fw=fwt], nequations(1) parameters(b1 b2 b3)
> instruments(x1 x2 x3, noconstant) vce(cluster id) onestep hasderivatives
Step 1
Iteration 0: GMM criterion Q(b) = 49.8292
Iteration 1: GMM criterion Q(b) = .11136736
Iteration 2: GMM criterion Q(b) = .00008519
Iteration 3: GMM criterion Q(b) = 7.110e-11
Iteration 4: GMM criterion Q(b) = 5.596e-23
note: model is exactly identified.
GMM estimation
Number of parameters = 3
Number of moments = 3
Initial weight matrix: Unadjusted Number of obs = 819

```
(Std. err. adjusted for 45 clusters in id)
\begin{tabular}{l|rrrrrr}
\hline & \multicolumn{4}{|c}{ Robust } & & \\
& Coefficient & std. err. & z & P>|z| & [95\% conf. interval] \\
\hline /b1 & 1.967766 & .111795 & 17.60 & 0.000 & 1.748652 & 2.186881 \\
/b2 & -3.060838 & .0935561 & -32.72 & 0.000 & -3.244205 & -2.877472 \\
/b3 & 1.037594 & .1184227 & 8.76 & 0.000 & .80549 & 1.269698 \\
\hline
\end{tabular}

Instruments for equation 1: x1 x2 x3
Testing whether our program works correctly with frequency weights is easy. A frequency-weighted dataset is just a compact form of a larger dataset in which identical observations are omitted and a frequency-weight variable is included to tell us how many times each observation in the smaller dataset appears in the larger dataset. Therefore, we can expand our smaller dataset by the frequency-weight variable and then refit our model without specifying frequency weights. If we obtain the same results, our program works correctly. When we type
```

. expand fw
. gmm gmm_poiderivfw, nequations(1) parameters(b1 b2 b3)
> instruments(x1 x2 x3, noconstant) vce(cluster id) onestep

```
we obtain the same results as before.

\section*{Stored results}
gmm stores the following in e():
Scalars
e(N)
e(k)
e(k_eq)
e(k_eq_model)
e(k_aux)
e(n_moments)
e(n_eq)
e(Q)
e(J)
e(J_df)
e(k_i)
e(has_xtinst)
e(N_clust)
e(type)
e(rank)
e(ic)
e(converged)
Macros
e(cmd)
e(cmdline)
e(title)
e(title_2)
e(clustvar)
e(inst_i)
e(eqnames)
e(winit)
e(winitname)
e(estimator)
e(rhs)
e(params_i)
e(wmatrix)
e(vce)
e(vcetype)
e(params)
e(sexp_i)
e(evalprog)
e(evalopts)
e(nocommonesample)
\(e\) (technique)
e(properties)
e(estat_cmd)
e(predict)
e(marginsok)
e(marginsnotok)
e(marginsprop)
e(asbalanced)
e(asobserved)
Matrices
e(b)
e(init)
e(Wuser)
e(W)
e(S)
e(G)
e(N_byequation)
e(V)
e(V_modelbased)
number of observations
number of parameters
number of equations in \(e(b)\)
number of equations in overall model test
number of auxiliary parameters
number of moments
number of equations in moment-evaluator program criterion function
Hansen \(J \chi^{2}\) statistic
\(J\) statistic degrees of freedom
number of parameters in equation \(i\)
1 if panel-style instruments specified, 0 otherwise
number of clusters
1 if interactive version, 2 if moment-evaluator program version rank of \(e(V)\)
number of iterations used by iterative GMM estimator
1 if converged, 0 otherwise

\section*{gmm}
command as typed
title specified in title()
title specified in title2()
name of cluster variable
equation \(i\) instruments
equation names
initial weight matrix used
name of user-supplied initial weight matrix
onestep, twostep, or igmm
variables specified in variables()
equation \(i\) parameters
wmtype specified in wmatrix()
vcetype specified in vce()
title used to label Std. err.
parameter names
substitutable expression for equation \(i\)
moment-evaluator program
options passed to moment-evaluator program
nocommonesample, if specified
optimization technique
b V
program used to implement estat
program used to implement predict
predictions allowed by margins
predictions disallowed by margins
signals to the margins command
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
initial values of the estimators
user-supplied initial weight matrix
weight matrix used for final round of estimation
moment covariance matrix used in robust VCE computations
averages of derivatives of moment conditions
number of observations per equation, if nocommonesample specified
variance-covariance matrix
model-based variance

Functions
```

    e(sample) marks estimation sample
    ```

In addition to the above, the following is stored in r() :

Matrices
\(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

\section*{Methods and formulas}

Let \(q\) denote the number of moment conditions. For observation \(i, i=1, \ldots, N\), write the \(j\) th moment equation as \(\mathbf{z}_{i j} u_{i j}\left(\boldsymbol{\beta}_{j}\right)\) for \(j=1, \ldots, q . \mathbf{z}_{i j}\) is a \(1 \times m_{j}\) vector, where \(m_{j}\) is the number of instruments specified for equation \(j\). Let \(m=m_{1}+\cdots+m_{q}\).

Our notation can incorporate moment conditions of the form \(h_{i j}\left(\mathbf{w}_{i j} ; \boldsymbol{\beta}_{j}\right)\) with instruments \(\mathbf{w}_{i j}\) by defining \(\mathbf{z}_{i j}=1\) and \(u_{i j}\left(\boldsymbol{\beta}_{j}\right)=h_{i j}\left(\mathbf{w}_{i j} ; \beta_{j}\right)\), so except when necessary, we do not distinguish between the two types of moment conditions. We could instead use notation so that all our moment conditions are of the form \(h_{i j}\left(\mathbf{w}_{i j} ; \beta_{j}\right)\), or we could adopt notation that explicitly combines both forms of moment equations. However, because moment conditions of the form \(\mathbf{z}_{i j}^{\prime} u_{i j}\left(\boldsymbol{\beta}_{j}\right)\) are arguably more common, we use that notation.

Let \(\boldsymbol{\beta}\) denote a \(k \times 1\) vector of parameters, consisting of all the unique parameters of \(\boldsymbol{\beta}_{1}, \ldots, \boldsymbol{\beta}_{q}\). Then, we can stack the moment conditions and write them more compactly as \(\mathbf{Z}_{i}^{\prime} \mathbf{u}_{i}(\boldsymbol{\beta})\), where
\[
\mathbf{Z}_{i}^{\prime}=\left[\begin{array}{cccc}
\mathbf{z}_{i 1} & \mathbf{0} & \cdots & \mathbf{0} \\
\mathbf{0} & \mathbf{z}_{i 2} & \cdots & \mathbf{0} \\
\vdots & \vdots & \ddots & \vdots \\
\mathbf{0} & \mathbf{0} & \cdots & \mathbf{z}_{i q}
\end{array}\right] \quad \text { and } \quad \mathbf{u}_{i}(\boldsymbol{\beta})=\left[\begin{array}{c}
u_{i 1}\left(\boldsymbol{\beta}_{1}\right) \\
u_{i 2}\left(\boldsymbol{\beta}_{2}\right) \\
\vdots \\
u_{i q}\left(\boldsymbol{\beta}_{q}\right)
\end{array}\right]
\]

The GMM estimator \(\widehat{\boldsymbol{\beta}}\) is the value of \(\beta\) that minimizes
\[
\begin{equation*}
Q(\boldsymbol{\beta})=\left\{N^{-1} \sum_{i=1}^{N} \mathbf{Z}_{i}^{\prime} \mathbf{u}_{i}(\boldsymbol{\beta})\right\}^{\prime} \mathbf{W}\left\{N^{-1} \sum_{i=1}^{N} \mathbf{Z}_{i}^{\prime} \mathbf{u}_{i}(\boldsymbol{\beta})\right\} \tag{A1}
\end{equation*}
\]
for \(q \times q\) weight matrix \(\mathbf{W}\).
By default, gmm minimizes (A1) using the Gauss-Newton method. See Hayashi \((2000,498)\) for a derivation. This technique is typically faster than quasi-Newton methods and does not require second-order derivatives.

Methods and formulas are presented under the following headings:

\footnotetext{
Initial weight matrix
Weight matrix
Variance-covariance matrix
Hansen's J statistic
Panel-style instruments
Marginal predictions with unconditional standard errors
}

\section*{Initial weight matrix}

If you specify winitial(unadjusted), then we create matrix \(\boldsymbol{\Lambda}\) with typical submatrix
\[
\boldsymbol{\Lambda}_{r s}=N^{-1} \sum_{i=1}^{N} \mathbf{z}_{i r}^{\prime} \mathbf{z}_{i s}
\]
for \(r=1, \ldots, q\) and \(s=1, \ldots, q\). If you include the independent suboption, then we set \(\boldsymbol{\Lambda}_{r s}=\mathbf{0}\) for \(r \neq s\). The weight matrix \(\mathbf{W}\) equals \(\boldsymbol{\Lambda}^{-1}\).

If you specify winitial (identity), then we set \(\mathbf{W}=\mathbf{I}_{q}\).
If you specify winitial (xt xtspec), then you must specify one or two items in xtspec, one for each equation. gmm allows you to specify at most two moment equations when you specify winitial (xt xtspec), one in first-differences, and one in levels. We create the block-diagonal matrix \(\mathbf{H}\) with typical block \(\mathbf{H}_{j}\). If the \(j\) th element of xtspec is " \(L\) ", then \(\mathbf{H}_{j}\) is the identity matrix of suitable dimension. If the \(j\) th element of \(x t s p e c\) is " \(D\) ", then
\[
\mathbf{H}_{j}=\left[\begin{array}{cccccc}
1 & -0.5 & 0 & \ldots & 0 & 0 \\
-0.5 & 1 & -0.5 & \ldots & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & 0 & \ldots & 1 & -0.5 \\
0 & 0 & 0 & \ldots & -0.5 & 1
\end{array}\right]
\]

Then,
\[
\boldsymbol{\Lambda}_{H}=N_{G}^{-1} \sum_{g=1}^{g=N_{G}} \mathbf{Z}_{g}^{\prime} \mathbf{H} \mathbf{Z}_{g}
\]
where \(g\) indexes panels in the dataset, \(N_{G}\) is the number of panels, \(\mathbf{Z}_{g}\) is the full instrument matrix for panel \(g\), and \(\mathbf{W}=\boldsymbol{\Lambda}_{H}^{-1}\). See Panel-style instruments below for a discussion of how \(\mathbf{Z}_{g}\) is formed.

If you specify winitial (matname), then we set \(\mathbf{W}\) equal to Stata matrix matname.

\section*{Weight matrix}

Specification of the weight matrix applies only to the two-step and iterative estimators. When you use the onestep option, the wmatrix() option is ignored.

We first evaluate (A1) using the initial weight matrix described above and then compute \(\mathbf{u}_{i}(\widehat{\boldsymbol{\beta}})\). In all cases, \(\mathbf{W}=\boldsymbol{\Lambda}^{-1}\). If you specify wmatrix (unadjusted), then we create \(\boldsymbol{\Lambda}\) to have typical submatrix
\[
\boldsymbol{\Lambda}_{r s}=\sigma_{r s} N^{-1} \sum_{i=1}^{N} \mathbf{z}_{i r}^{\prime} \mathbf{z}_{i s}
\]
where
\[
\sigma_{r s}=N^{-1} \sum_{i=1}^{N} u_{i r}(\widehat{\boldsymbol{\beta}}) u_{i s}(\widehat{\boldsymbol{\beta}})
\]
and \(r\) and \(s\) index moment equations. For all types of weight matrices, if the independent suboption is specified, then \(\boldsymbol{\Lambda}_{r s}=\mathbf{0}\) for \(r \neq s\), where \(\boldsymbol{\Lambda}_{r s}\) measures the covariance between moment conditions for equations \(r\) and \(s\).

If you specify wmatrix(robust), then
\[
\boldsymbol{\Lambda}=N^{-1} \sum_{i=1}^{N} \mathbf{Z}_{i} \mathbf{u}_{i}(\widehat{\boldsymbol{\beta}}) \mathbf{u}_{i}^{\prime}(\widehat{\boldsymbol{\beta}}) \mathbf{Z}_{i}^{\prime}
\]

If you specify wmatrix(cluster clustvar), then
\[
\boldsymbol{\Lambda}=N^{-1} \sum_{c=1}^{c=N_{C}} \mathbf{q}_{c} \mathbf{q}_{c}^{\prime}
\]
where \(c\) indexes clusters, \(N_{C}\) is the number of clusters, and
\[
\mathbf{q}_{c}=\sum_{i \in c_{j}} \mathbf{Z}_{i} \mathbf{u}_{i}(\widehat{\boldsymbol{\beta}})
\]

If you specify wmatrix (hac kernel [\#]), then
\[
\begin{aligned}
\boldsymbol{\Lambda}= & N^{-1} \sum_{i=1}^{N} \mathbf{Z}_{i} \mathbf{u}_{i}(\widehat{\boldsymbol{\beta}}) \mathbf{u}_{i}(\widehat{\boldsymbol{\beta}})^{\prime} \mathbf{Z}_{i}^{\prime}+ \\
& N^{-1} \sum_{l=1}^{l=N-1} \sum_{i=l+1}^{N} K(l, \lambda)\left\{\mathbf{Z}_{i} \mathbf{u}_{i}(\widehat{\boldsymbol{\beta}}) \mathbf{u}_{i-l}^{\prime}(\widehat{\boldsymbol{\beta}}) \mathbf{Z}_{i-l}^{\prime}+\mathbf{Z}_{i-l} \mathbf{u}_{i-l}(\widehat{\boldsymbol{\beta}}) \mathbf{u}_{i}^{\prime}(\widehat{\boldsymbol{\beta}}) \mathbf{Z}_{i}^{\prime}\right\}
\end{aligned}
\]
where \(\lambda=\) \# if \# is specified and \(\lambda=N-2\) otherwise. Define \(z=l /(\lambda+1)\). If kernel is bartlett or nwest, then
\[
K(l, \lambda)= \begin{cases}1-z & 0 \leq z \leq 1 \\ 0 & \text { otherwise }\end{cases}
\]

If kernel is parzen or gallant, then
\[
K(l, \lambda)= \begin{cases}1-6 z^{2}+6 z^{3} & 0 \leq z \leq 0.5 \\ 2(1-z)^{3} & 0.5<z \leq 1 \\ 0 & \text { otherwise }\end{cases}
\]

If kernel is quadraticspectral or andrews, then
\[
K(l, \lambda)= \begin{cases}1 & z=0 \\ 3\{\sin (\theta) / \theta-\cos (\theta)\} / \theta^{2} & \text { otherwise }\end{cases}
\]
where \(\theta=6 \pi z / 5\).
If wmatrix (hac kernel opt) is specified, then gmm uses Newey and West's (1994) automatic lag-selection algorithm, which proceeds as follows. Define \(\mathbf{h}\) to be an \(m \times 1\) vector of ones. Note that this definition of \(\mathbf{h}\) is slightly different from the one used by ivregress. There the element of \(\mathbf{h}\) corresponding to the constant term equals 0 , effectively ignoring the effect of the constant in determining the optimal lag length. Here we include the effect of the constant term. Now, define
\[
\begin{aligned}
f_{i} & =\left\{\mathbf{Z}_{i}^{\prime} \mathbf{u}_{i}(\boldsymbol{\beta})\right\}^{\prime} \mathbf{h} \\
\widehat{\sigma}_{j} & =N^{-1} \sum_{i=j+1}^{N} f_{i} f_{i-j} \quad j=0, \ldots, \lambda^{*} \\
\widehat{s}^{(q)} & =2 \sum_{j=1}^{j=\lambda^{*}} \widehat{\sigma}_{j} j^{q} \\
\widehat{s}^{(0)} & =\widehat{\sigma}_{0}+2 \sum_{j=1}^{j=\lambda^{*}} \widehat{\sigma}_{j} \\
\widehat{\gamma} & =c_{\gamma}\left\{\left(\frac{\widehat{s}^{(q)}}{\widehat{s}^{(0)}}\right)^{2}\right\}^{1 /(2 q+1)} \\
\lambda & =\widehat{\gamma} N^{1 /(2 q+1)}
\end{aligned}
\]
where \(q, \lambda^{*}\), and \(c_{\gamma}\) depend on the kernel specified:
\begin{tabular}{cccc} 
Kernel & \(q\) & \(\lambda^{*}\) & \(c_{\gamma}\) \\
\hline Bartlett/Newey-West & 1 & int \(\left\{20(T / 100)^{2 / 9}\right\}\) & 1.1447 \\
Parzen/Gallant & 2 & int \(\left\{20(T / 100)^{4 / 25}\right\}\) & 2.6614 \\
Quadratic spectral/Andrews & 2 & int \(\left\{20(T / 100)^{2 / 25}\right\}\) & 1.3221
\end{tabular}

Here \(\operatorname{int}(x)\) denotes the integer obtained by truncating \(x\) toward zero. For the Bartlett and Parzen kernels, the optimal lag is \(\min \left\{\operatorname{int}(\lambda), \lambda^{*}\right\}\). For the quadratic spectral kernel, the optimal lag is \(\min \left\{\lambda, \lambda^{*}\right\}\).

If wmatrix (hac kernel opt \#) is specified, then gmm uses \# instead of 20 in the definition of \(\lambda^{*}\) above to select the optimal lag.

\section*{Variance-covariance matrix}

If you specify vce(unadjusted), then the VCE matrix is computed as
\[
\begin{equation*}
\operatorname{Var}(\widehat{\boldsymbol{\beta}})=N^{-1}\left\{\overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})^{\prime} \mathbf{W} \overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})\right\}^{-1} \tag{A2}
\end{equation*}
\]
where
\[
\begin{equation*}
\overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})=\left.N^{-1} \sum_{i=1}^{N} \mathbf{Z}_{i}^{\prime} \frac{\partial \mathbf{u}_{i}(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}^{\prime}}\right|_{\boldsymbol{\beta}=\widehat{\boldsymbol{\beta}}} \tag{A3}
\end{equation*}
\]

For the two-step and iterated estimators, we use the weight matrix \(\mathbf{W}\) that was used to compute the final-round estimate \(\widehat{\boldsymbol{\beta}}\).

When you do not specify analytic derivatives, gmm must compute the Jacobian matrix (A3) numerically. By default, gmm computes each element of the matrix individually by using the Mata deriv() function; see [M-5] deriv( ). This procedure results in accurate derivatives but can be slow if your model has many instruments or parameters.

When you specify the quickderivatives option, gmm computes all derivatives corresponding to parameter \(\beta_{j}, j=1, \ldots, q\), at once, using two-sided derivatives with a step size of \(\left|\beta_{j}\right| \epsilon^{1 / 3}\), where \(\epsilon\) is the machine precision of a double precision number (approximately \(2.22045 \times 10^{-16}\) ). This method requires just two evaluations of the model's moments to compute an entire column of (A3) and therefore has the most impact when you specify many instruments or moment equations so that (A3) has many rows.

For the one-step estimator, how the unadjusted VCE is computed depends on the type of initial weight matrix requested and the form of the moment equations. If you specify two or more moment equations of the form \(h_{i j}\left(\mathbf{w}_{i j} ; \boldsymbol{\beta}_{j}\right)\), then gmm issues a note and computes a heteroskedasticity-robust VCE because here the matrix \(\mathbf{Z}^{\prime} \mathbf{Z}\) is necessarily singular; moreover, here you must use the identity matrix as the initial weight matrix. Otherwise, if you specify winitial (unadjusted) or winitial (identity), then gmm first computes an unadjusted weight matrix based on \(\widehat{\boldsymbol{\beta}}\) before evaluating (A2). If you specify winitial (matname), then (A2) is evaluated on the basis of matname; the user is responsible for verifying that the VCE and other statistics so produced are appropriate.

All types of robust VCEs computed by gmm take the form
\[
\operatorname{Var}(\widehat{\boldsymbol{\beta}})=N^{-1}\left\{\overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})^{\prime} \mathbf{W} \overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})\right\}^{-1} \overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})^{\prime} \mathbf{W S W} \overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})\left\{\overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})^{\prime} \mathbf{W} \overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})\right\}^{-1}
\]

For the one-step estimator, \(\mathbf{W}\) represents the initial weight matrix requested using the winitial() option, and \(\mathbf{S}\) is computed on the basis of the specification of the vce() option. The formulas for the \(\mathbf{S}\) matrix are identical to the ones that define the \(\boldsymbol{\Lambda}\) matrix in Weight matrix above, except that \(\mathbf{S}\) is computed after the moment equations are reevaluated using the final estimate of \(\widehat{\boldsymbol{\beta}}\). For the two-step and iterated GMM estimators, computation of \(\mathbf{W}\) is controlled by the wmatrix() option on the basis of the penultimate estimate of \(\widehat{\boldsymbol{\beta}}\).

For details on computation of the VCE matrix with dynamic panel-data models, see Panel-style instruments below.

\section*{Hansen's J statistic}

Hansen's (1982) \(J\) test of overidentifying restrictions is \(J=N \times Q(\widehat{\boldsymbol{\beta}}) . J \sim \chi^{2}(m-k)\). If \(m<k\), gmm issues an error message without estimating the parameters. If \(m=k\), the model is just-identified and \(J\) is saved as missing ("."). For the two-step and iterated GMM estimators, the \(J\) statistic is based on the last-computed weight matrix as determined by the wmatrix() option. For the one-step estimator, gmm recomputes a weight matrix as described in the second paragraph of Variance-covariance matrix above. To obtain Hansen's \(J\) statistic, you use estat overid; see [R] gmm postestimation.

\section*{Panel-style instruments}

Here we discuss several issues that arise only when you specify panel-style instruments by using the xtinstruments () option. When you specify the xtinstruments () option, we can no longer consider the instruments for one observation in isolation; instead, we must consider the instrument matrix for an entire panel at once. In the following discussion, we let \(T\) denote the number of time periods in a panel. To accommodate unbalanced datasets, conceptually we simply use zeros as instruments and residuals for time periods that are missing in a panel.

We consider the case where you specify both an equation in levels and an equation in differences, yielding two residual equations. Let \(u_{p t}^{L}(\boldsymbol{\beta})\) denote the residual for the level equation for panel \(p\) in period \(t\), and let \(u_{p t}^{D}(\boldsymbol{\beta})\) denote the residual for the corresponding difference equation. Now, define the \((2 T-1) \times 1\) vector \(\mathbf{u}_{p}(\boldsymbol{\beta})\) as
\[
\mathbf{u}_{p}(\boldsymbol{\beta})=\left[u_{p 1}^{L}(\boldsymbol{\beta}), u_{p 2}^{L}(\boldsymbol{\beta}), \ldots, u_{p T}^{L}(\boldsymbol{\beta}), u_{p 2}^{D}(\boldsymbol{\beta}), u_{p 3}^{D}(\boldsymbol{\beta}), \ldots, u_{p T}^{D}(\boldsymbol{\beta})\right]
\]

The \(T+1\) element of \(\mathbf{u}_{p}\) is \(u_{p 2}^{D}(\boldsymbol{\beta})\) because we lose the first observation of the difference equation because of differencing.

We write the moment conditions for the \(p\) th panel as \(\mathbf{Z}_{p} \mathbf{u}_{p}(\boldsymbol{\beta})\). To see how \(\mathbf{Z}_{p}\) is defined, we will let \(\mathbf{w}_{p t}^{L}\) and \(\mathbf{w}_{p t}^{D}\) denote the vectors of panel-style instruments for the level and difference equations, respectively, and let time be denoted by \(t\); we discuss their dimensions momentarily. Also let \(\mathbf{x}_{p t}^{L}\) and \(\mathbf{x}_{p t}^{D}\) denote the vectors of instruments specified in instruments () for the level and difference equations at time \(t\). Without loss of generality, for our discussion, we assume that you specify the level equation first. Then, \(\mathbf{Z}_{p}\) has the form
\[
\mathbf{Z}_{p}=\left[\begin{array}{cccccccc}
\mathbf{w}_{1}^{L} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0}  \tag{A4}\\
\mathbf{0} & \mathbf{w}_{2}^{L} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
\mathbf{0} & \mathbf{0} & \cdots & \mathbf{w}_{T}^{L} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\
\mathbf{x}_{1}^{L} & \mathbf{x}_{2}^{L} & \cdots & \mathbf{x}_{T}^{L} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{w}_{1}^{D} & \mathbf{0} & \cdots & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{w}_{2}^{D} & \cdots & \mathbf{0} \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\
\mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{w}_{T}^{D} \\
\mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{x}_{1}^{D} & \mathbf{x}_{2}^{D} & \cdots & \mathbf{x}_{T}^{D}
\end{array}\right]
\]

To see how the \(\mathbf{w}\) vectors are formed, we will suppose you specify xtinstruments(1: d, lags (a/b))

Then, \(\mathbf{w}_{t}^{L}\) will be a \((b-a+1) \times 1\) vector consisting of \(d_{t-a}, \ldots, d_{t-b}\). If \((t-a) \leq 0\), then instead, we set \(\mathbf{w}_{t}^{L}=0\). If \((t-a)>0\) but \((t-b) \leq 0\), then we create \(\mathbf{w}_{t}^{L}\) to consist of \(d_{t-a}, \ldots, d_{1}\). With this definition, \((b-a+1)\) defines the maximum number of lags of \(d\) used, but gmm will proceed with fewer lags if all \((b-a+1)\) lags are not available. If you specify two panel-style instruments, d and e, say, then \(\mathbf{w}_{t}^{L}\) will consist of \(d_{t-a}, \ldots, d_{t-b}, e_{t-a}, \ldots, e_{t-b} . \mathbf{w}_{t}^{D}\) is handled analogously.

The \(\mathbf{x}_{t}^{L}\) vectors are simply \(j \times 1\) vectors, where \(j\) is the number of regular instruments specified with the instruments() option; these vectors include a " 1 " unless you specify the noconstant suboption.

Looking carefully at (A4), you will notice that for dynamic panel-data models, moment conditions corresponding to the instruments \(\mathbf{x}_{p t}^{L}\) take the form
\[
E\left[\sum_{t=1}^{t=T} \mathbf{x}_{p t}^{L} u_{p t}^{L}(\boldsymbol{\beta})\right]=\mathbf{0}
\]
and likewise for \(\mathbf{x}_{p t}^{D}\). Instead of having separate moment conditions for each time period, there is one moment condition equal to the average of individual periods' moments. See Arellano and Bond (1991, 280). To include separate moment conditions for each time period, instead of specifying, say,
you could instead first generate a variable called one equal to unity for all observations and specify
```

xtinstruments(1: x one)

```
(Creating the variable one is necessary because a constant is not automatically included in variable lists specified in xtinstruments().)

Unbalanced panels are essentially handled by including zeros in rows and columns of \(\mathbf{Z}_{p}\) and \(\mathbf{u}_{p}(\boldsymbol{\beta})\) corresponding to missing time periods. However, the numbers of instruments and moment conditions reported by gmm do not reflect this trickery and instead reflect the numbers of instruments and moment conditions that are not manipulated in this way. Moreover, gmm includes code to work through these situations efficiently without actually having to fill in zeros.

When you specify winitial (xt ...), the one-step unadjusted VCE is computed as
\[
\operatorname{Var}(\widehat{\boldsymbol{\beta}})=\widehat{\sigma}_{1}^{2} \boldsymbol{\Lambda}_{H}
\]
where \(\boldsymbol{\Lambda}_{H}\) is as defined previously,
\[
\widehat{\sigma}_{1}^{2}=(N-k)^{-1} \sum_{p=1}^{p=P} \mathbf{u}_{p}^{D}(\widehat{\boldsymbol{\beta}})^{\prime} \mathbf{u}_{p}^{D}(\widehat{\boldsymbol{\beta}})
\]
and \(\mathbf{u}_{p}^{D}(\widehat{\boldsymbol{\beta}})=\left[u_{p 2}^{D}(\widehat{\boldsymbol{\beta}}), \ldots, u_{p T}^{D}(\widehat{\boldsymbol{\beta}})\right]\). Here we use \((N-k)^{-1}\) instead of \(N^{-1}\) to match xtdpd.

\section*{Marginal predictions with unconditional standard errors}

Here we describe how margins computes unconditional standard errors when used with the vce (unconditional) option after gmm. These standard errors account for the estimation of parameters in gmm before margins is used to make marginal predictions. They also account for variation in the covariates over the population.
margins with the vce (unconditional) option uses linearization to estimate the unconditional variance of \(\widehat{\boldsymbol{\beta}}\). Linearization uses the variance estimator for the total of a score variable for the marginal prediction \(p(\widehat{\boldsymbol{\beta}})\) as an approximate estimator for \(\operatorname{Var}\{p(\widehat{\boldsymbol{\beta}})\}\). See [SVY] Variance estimation for more details. Our derivation of the standard errors here is similar to the derivation of the standard errors for two-step \(M\) estimators. See Wooldridge (2010, sec. 12.4) for the latter.

Let \(\mathbf{x}_{i}\) be a vector of covariate values, which includes all variables used in calculating the moment conditions, and let \(f\left(\mathbf{x}_{i}, \boldsymbol{\beta}\right)\) be a scalar-valued function returning the value of the predictions of interest.

Let \(\delta_{i}\left(S_{p}\right)\) indicate whether observation \(i\) is in the subpopulation of interest, \(S_{p}\),
\[
\delta_{i}\left(S_{p}\right)= \begin{cases}1, & i \in S_{p} \\ 0, & i \notin S_{p}\end{cases}
\]
margins computes \(\widehat{\theta}=p(\widehat{\boldsymbol{\beta}})\) via
\[
\widehat{\theta}=\frac{1}{w} \sum_{i=1}^{N} \delta_{i}\left(S_{p}\right) w_{i} f\left(\mathbf{x}_{i}, \widehat{\boldsymbol{\beta}}\right)
\]
where
\[
w .=\sum_{i=1}^{N} \delta_{i}\left(S_{p}\right) w_{i}
\]
and \(w_{i}\) is the weight for the \(i\) th observation.
In minimizing (A1), we see that the GMM estimator \(\widehat{\boldsymbol{\beta}}\) is the value of \(\boldsymbol{\beta}\) that solves the score equations
\[
\mathbf{0}=\overline{\mathbf{G}}(\boldsymbol{\beta})^{\prime} \mathbf{W}\left\{N^{-1} \sum_{i=1}^{N} \mathbf{Z}_{i}^{\prime} \mathbf{u}_{i}(\boldsymbol{\beta})\right\}
\]
where \(\overline{\mathbf{G}}(\boldsymbol{\beta})\) was defined in (A3).
By the mean-value theorem, for some points \(\boldsymbol{\beta}_{1}, \ldots, \boldsymbol{\beta}_{q}\) between \(\boldsymbol{\beta}\) and \(\widehat{\boldsymbol{\beta}}\), we have
\[
\mathbf{0}=\overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})^{\prime} \mathbf{W}\left[\left\{N^{-1} \sum_{i=1}^{N} \mathbf{Z}_{i}^{\prime} \mathbf{u}_{i}(\boldsymbol{\beta})\right\}+\overline{\mathbf{G}}_{m}(\widehat{\boldsymbol{\beta}}-\boldsymbol{\beta})\right]
\]
where
\[
\overline{\mathbf{G}}_{m(j, l)}=N^{-1} \sum_{i=1}^{N}\left[\frac{\partial\left\{\mathbf{Z}_{i}^{\prime} \mathbf{u}_{i}\left(\boldsymbol{\beta}_{j}\right)\right\}_{l}}{\partial \boldsymbol{\beta}_{j}^{\prime}}\right]_{j, l}
\]

So we have
\[
\begin{equation*}
\sqrt{N}(\widehat{\boldsymbol{\beta}}-\boldsymbol{\beta})=-\left\{\overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})^{\prime} \mathbf{W} \overline{\mathbf{G}}_{m}\right\}^{-1} \overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})^{\prime} \mathbf{W}\left\{N^{-0.5} \sum_{i=1}^{N} \mathbf{Z}_{i}^{\prime} \mathbf{u}_{i}(\boldsymbol{\beta})\right\} \tag{A5}
\end{equation*}
\]

The margin \(\widehat{\theta}\) is the solution to the score equations
\[
\frac{1}{N} \sum_{i=1}^{N} s_{i}(\theta, \widehat{\boldsymbol{\beta}})=\mathbf{0}
\]
where
\[
s_{i}(\theta, \boldsymbol{\beta})=w_{i} \delta_{i}\left(S_{p}\right)\left\{f\left(\mathbf{x}_{i}, \boldsymbol{\beta}\right)-\theta\right\}
\]

When we do a mean-value expansion about point \(\theta_{1}\) between \(\theta\) and \(\widehat{\theta}\), we get
\[
\mathbf{0}=\frac{1}{N} \sum_{i=1}^{N} s_{i}(\theta, \widehat{\boldsymbol{\beta}})-w \cdot(\widehat{\theta}-\theta)
\]

So we have
\[
\sqrt{N}(\widehat{\theta}-\theta)=w .^{-1} N^{-0.5} \sum_{i=1}^{N} s_{i}(\theta, \widehat{\boldsymbol{\beta}})
\]

Using the mean-value theorem again, for point \(\boldsymbol{\beta}_{m}\) between \(\boldsymbol{\beta}\) and \(\widehat{\boldsymbol{\beta}}\), we have
\[
w .^{-1} N^{-0.5} \sum_{i=1}^{N} s_{i}(\theta, \widehat{\boldsymbol{\beta}})=w .^{-1}\left\{N^{-0.5} \sum_{i=1}^{N} s_{i}(\theta, \boldsymbol{\beta})+\sqrt{N} \overline{\mathbf{J}}\left(\boldsymbol{\beta}_{m}\right)(\widehat{\boldsymbol{\beta}}-\boldsymbol{\beta})\right\}
\]
where \(\overline{\mathbf{J}}\left(\boldsymbol{\beta}_{m}\right)\) is the Jacobian of the margin at \(\boldsymbol{\beta}_{m}\),
\[
\overline{\mathbf{J}}\left(\boldsymbol{\beta}_{m}\right)=\left\{N^{-1} \sum_{i=1}^{N} w_{i} \delta_{i}\left(S_{p}\right) \frac{\partial f\left(\mathbf{x}_{i}, \boldsymbol{\beta}_{m}\right)}{\partial \boldsymbol{\beta}_{m}^{\prime}}\right\}
\]

Using (A5), we get
\(\sqrt{N}(\widehat{\theta}-\theta)=w^{-1} N^{-0.5}\left[\sum_{i=1}^{N} s_{i}(\theta, \boldsymbol{\beta})-\overline{\mathbf{J}}\left(\boldsymbol{\beta}_{m}\right)\left\{\overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})^{\prime} \mathbf{W} \overline{\mathbf{G}}_{m}\right\}^{-1} \overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})^{\prime} \mathbf{W}\left\{\sum_{i=1}^{N} \mathbf{Z}_{i}^{\prime} \mathbf{u}_{i}(\boldsymbol{\beta})\right\}\right]\)
\(\widehat{\theta}\) is asymptotically normal, and a consistent estimator of its variance is given by
\[
\widehat{\operatorname{Var}}\{\sqrt{N}(\widehat{\theta}-\theta)\}=\sum_{i=1}^{N} w^{-2}\left[s_{i}(\widehat{\theta}, \widehat{\boldsymbol{\beta}})-\overline{\mathbf{J}}(\widehat{\boldsymbol{\beta}})\left\{\overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})^{\prime} \mathbf{W} \overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})\right\}^{-1} \overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})^{\prime} \mathbf{W} \mathbf{Z}_{i}^{\prime} \mathbf{u}_{i}(\widehat{\boldsymbol{\beta}})\right]^{2}
\]

See Wooldridge (2010, sec 12.4 and 12.5) for details.
gmm returns \(N^{-1}\left\{\overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})^{\prime} \mathbf{W} \overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})\right\}^{-1}\) as the model-based variance. The scores are
\(-\overline{\mathbf{G}}(\widehat{\boldsymbol{\beta}})^{\prime} \mathbf{W} \mathbf{Z}_{i}^{\prime} \mathbf{u}_{i}(\widehat{\boldsymbol{\beta}})\) and may be predicted in postestimation; see \([\mathrm{R}] \mathbf{g m m}\) postestimation. These scores correspond to derivatives of the criterion function \(Q(\beta)\), scaled by \(-1 / 2\). See Cameron and Trivedi (2005, sec. 6.3.2) for more details.
margins estimates the asymptotic standard error of \(\widehat{\theta}\) from the model-based variance, the scores, and its own predictions of \(s_{i}\) and the Jacobian \(\overline{\mathbf{J}}(\widehat{\boldsymbol{\beta}})\).

\section*{References}

Andrews, D. W. K., W. Kim, and X. Shi. 2017. Commands for testing conditional moment inequalities and equalities. Stata Journal 17: 56-72.
Arellano, M., and S. Bond. 1991. Some tests of specification for panel data: Monte Carlo evidence and an application to employment equations. Review of Economic Studies 58: 277-297. https://doi.org/10.2307/2297968.
Baum, C. F. 2006. An Introduction to Modern Econometrics Using Stata. College Station, TX: Stata Press.
Blundell, R. W., R. Griffith, and F. Windmeijer. 2002. Individual effects and dynamics in count data models. Journal of Econometrics 108: 113-131. https://doi.org/10.1016/S0304-4076(01)00108-7.
Cameron, A. C., and P. K. Trivedi. 2005. Microeconometrics: Methods and Applications. New York: Cambridge University Press.
——. 2013. Regression Analysis of Count Data. 2nd ed. New York: Cambridge University Press.
——. 2022. Microeconometrics Using Stata. 2nd ed. College Station, TX: Stata Press.
Cattaneo, M. D. 2010. Efficient semiparametric estimation of multi-valued treatment effects under ignorability. Journal of Econometrics 155: 138-154. https://doi.org/10.1016/j.jeconom.2009.09.023.
Chamberlain, G. 1992. Comment: Sequential moment restrictions in panel data. Journal of Business and Economic Statistics 10: 20-26. https://doi.org/10.2307/1391799.
Clarke, D., and B. Matta. 2018. Practical considerations for questionable IVs. Stata Journal 18: 663-691.
Davidson, R., and J. G. MacKinnon. 1993. Estimation and Inference in Econometrics. New York: Oxford University Press.
Doris, A., D. O'Neill, and O. Sweetman. 2011. GMM estimation of the covariance structure of longitudinal data on earnings. Stata Journal 11: 439-459.

Drukker, D. M. 2014. Using gmm to solve two-step estimation problems. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2014/12/08/using-gmm-to-solve-two-step-estimation-problems/.
——. 2015. Understanding the generalized method of moments (GMM): A simple example. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2015/12/03/understanding-the-generalized-method-of-moments-gmm-a-simple-example/.
_-. 2016. An ordered-probit inverse probability weighted (IPW) estimator. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2016/09/13/an-ordered-probit-inverse-probability-weighted-ipw-estimator/.

Erickson, T., R. Parham, and T. M. Whited. 2017. Fitting the errors-in-variables model using high-order cumulants and moments. Stata Journal 17: 116-129.
Flynn, Z. L., and L. M. Magnusson. 2013. Parametric inference using structural break tests. Stata Journal 13: 836-861.
Greene, W. H. 2018. Econometric Analysis. 8th ed. New York: Pearson.
Hall, A. R. 2005. Generalized Method of Moments. Oxford: Oxford University Press.
Hamilton, J. D. 1994. Time Series Analysis. Princeton, NJ: Princeton University Press.
Hansen, L. P. 1982. Large sample properties of generalized method of moments estimators. Econometrica 50 : 1029-1054. https://doi.org/10.2307/1912775.
Hansen, L. P., and K. J. Singleton. 1982. Generalized instrumental variables estimation of nonlinear rational expectations models. Econometrica 50: 1269-1286. https://doi.org/10.2307/1911873.

Hayashi, F. 2000. Econometrics. Princeton, NJ: Princeton University Press.
Lindsey, C. 2016a. Estimating covariate effects after gmm. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2016/10/04/estimating-covariate-effects-after-gmm/.
——. 2016b. Testing model specification and using the program version of gmm. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2016/02/11/testing-model-specification-and-using-the-program-version-of-gmm/.
Lindsey, C., and J. Luedicke. 2016. Solving missing data problems using inverse-probability-weighted estimators. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2016/10/11/solving-missing-data-problems-using-inverse-probability-weighted-estimators/.
Lindsey, C., and E. Pinzon. 2016. Multiple-equation models: Estimation and marginal effects using gmm. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2016/08/02/multiple-equation-models-estimation-and-marginal-effects-using-gmm/.

Manski, C. F. 1988. Analog Estimation Methods in Econometrics. New York: Chapman and Hall/CRC.
Mátyás, L. 1999. Generalized Method of Moments Estimation. Cambridge: Cambridge University Press.
Mullahy, J. 1997. Instrumental-variable estimation of count data models: Applications to models of cigarette smoking behavior. Review of Economics and Statistics 79: 586-593. https://doi.org/10.1162/003465397557169.
Newey, W. K., and K. D. West. 1994. Automatic lag selection in covariance matrix estimation. Review of Economic Studies 61: 631-653. https://doi.org/10.2307/2297912.
Nickell, S. J. 1981. Biases in dynamic models with fixed effects. Econometrica 49: 1417-1426. https://doi.org/10.2307/1911408.
Pinzon, E. 2016. Estimation under omitted confounders, endogeneity, omitted variable bias, and related problems. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2017/02/20/estimation-under-omitted-confounders-endogeneity-omitted-variable-bias-and-related-problems/.

Pitblado, J. S., B. P. Poi, and W. W. Gould. 2024. Maximum Likelihood Estimation with Stata. 5th ed. College Station, TX: Stata Press.

Rajbhandari, A. 2015. Estimating parameters by maximum likelihood and method of moments using mlexp and gmm. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2015/10/15/estimating-parameters-by-maximum-likelihood-and-method-of-moments-using-mlexp-and-gmm/.
__. 2016. Tests of forecast accuracy and forecast encompassing. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2016/06/01/tests-of-forecast-accuracy-and-forecast-encompassing/.

Rios-Avila, F., and G. Canavire-Bacarreza. 2018. Standard-error correction in two-stage optimization models: A quasi-maximum likelihood estimation approach. Stata Journal 18: 206-222.

Ruud, P. A. 2000. An Introduction to Classical Econometric Theory. New York: Oxford University Press.

Wilde, J. 2008. A note on GMM estimation of probit models with endogenous regressors. Statistical Papers 49: 471-484. https://doi.org/10.1007/s00362-006-0027-2.

Windmeijer, F. 2000. Moment conditions for fixed effects count data models with endogenous regressors. Economics Letters 68: 21-24. https://doi.org/10.1016/S0165-1765(00)00228-7.
——. 2005. A finite sample correction for the variance of linear efficient two-step GMM estimators. Journal of Econometrics 126: 25-51. https://doi.org/10.1016/j.jeconom.2004.02.005.

Windmeijer, F., and J. M. C. Santos Silva. 1997. Endogeneity in count data models: An application to demand for health care. Journal of Applied Econometrics 12: 281-294. https://doi.org/10.1002/(SICI)1099-1255(199705)12: 3〈281::AID-JAE436〉3.0.CO;2-1.
Wooldridge, J. M. 1997. Multiplicative panel data models without the strict exogeneity assumption. Econometric Theory 13: 667-678. https://doi.org/10.1017/S0266466600006125.
-_ 1999. Distribution-free estimation of some nonlinear panel data models. Journal of Econometrics 90: 77-97. https://doi.org/10.1016/S0304-4076(98)00033-5.
——. 2010. Econometric Analysis of Cross Section and Panel Data. 2nd ed. Cambridge, MA: MIT Press.
Ye, X., and Y. Sun. 2018. Heteroskedasticity- and autocorrelation-robust F and t tests in Stata. Stata Journal 18: 951-980.

\section*{Also see}
[R] gmm postestimation - Postestimation tools for gmm
\([\mathrm{R}]\) ivregress - Single-equation instrumental-variables regression
[R] ml — Maximum likelihood estimation
[R] mlexp - Maximum likelihood estimation of user-specified expressions
[R] nl - Nonlinear least-squares estimation
[R] nlsur - Estimation of nonlinear systems of equations
[XT] xtabond - Arellano-Bond linear dynamic panel-data estimation
[XT] xtdpd - Linear dynamic panel-data estimation
[XT] xtdpdsys - Arellano-Bover/Blundell-Bond linear dynamic panel-data estimation
[U] 20 Estimation and postestimation commands

\section*{Title}
gmm postestimation - Postestimation tools for gmm
\begin{tabular}{llll} 
Postestimation commands & predict & margins & estat \\
Remarks and examples & Stored results & References & Also see
\end{tabular}

\section*{Postestimation commands}

The following postestimation command is of special interest after gmm:
\begin{tabular}{ll} 
Command & Description \\
\hline estat overid & perform test of overidentifying restrictions \\
\hline
\end{tabular}

The following standard postestimation commands are also available:
\begin{tabular}{ll} 
Command & Description \\
\hline contrast & contrasts and ANOVA-style joint tests of estimates \\
estat vce & \begin{tabular}{l} 
variance-covariance matrix of the estimators (VCE) \\
estimates \\
cataloging estimation results \\
table of estimation results
\end{tabular} \\
hausman & \begin{tabular}{l} 
Hausman's specification test \\
point estimates, standard errors, testing, and inference for linear combinations of \\
coefficients
\end{tabular} \\
margins & \begin{tabular}{l} 
marginal means, predictive margins, marginal effects, and average marginal effects \\
marginsplot \\
graph the results from margins (profile plots, interaction plots, etc.) \\
nlcom
\end{tabular} \\
predint estimates, standard errors, testing, and inference for nonlinear combinations \\
of coefficients
\end{tabular}\(\quad\)\begin{tabular}{l} 
residuals \\
predictnl \\
pwcompare \\
point estimates, standard errors, testing, and inference for generalized predictions \\
pairwise comparisons of estimates
\end{tabular}\(\quad\)\begin{tabular}{l} 
Wald tests of simple and composite linear hypotheses \\
testnl
\end{tabular}\(\quad\)\begin{tabular}{l} 
Wald tests of nonlinear hypotheses
\end{tabular}

\section*{predict}

\section*{Description for predict}
predict creates a new variable containing predictions such as linear predictions, residuals, and scores.

\section*{Menu for predict}

Statistics \(>\) Postestimation

\section*{Syntax for predict}
```

predict [type] newvar [if][in] [, xb equation(\#eqno|eqname)]
predict [type] newvar [if] [in], residuals [equation(\#eqno|eqname)]
predict [type] {stub*|newvar
predict [type] stub* [if] [in], scores

```

Residuals are available both in and out of sample; type predict ... if e(sample)... if wanted only for the estimation sample.
Scores are available only for observations within the estimation sample.
You specify one new variable and (optionally) equation(), or you specify stub* or \(q\) or \(p\) new variables, where \(q\) is the number of moment equations and \(p\) is the number of parameters in the model.

\section*{Option for predict}
\(\qquad\) Main
xb , the default, calculates the linear prediction.
residuals calculates the residuals, the predicted values of the moment equations. This option requires that the length of the new variable list be equal to the number of moment equations, \(q\). Otherwise, use stub* to have predict generate enumerated variables with prefix stub. If equation() is not specified, the \(j\) th new variable will contain the residuals for the \(j\) th moment equation.
equation(\#eqno \(\mid\) eqname) with xb refers to a linear prediction in the model, whereas equation (\#eqno \(\mid\) eqname) with residuals refers to a moment equation.
For xb , specifying equation(\#1) indicates that the calculation is to be made for the first linear prediction within curly braces. Specifying equation(xb1) would indicate that the calculation is to be made for the linear prediction xb 1 , assuming there is a linear prediction named xb 1 in the model, for instance, \(\{\mathrm{xb} 1:\) varlist \(\}\).
For residuals, specifying equation(\#1) indicates that the calculation is to be made for the first moment equation. Specifying equation(demand) would indicate that the calculation is to be made for the moment equation named demand, assuming there is a moment equation named demand in the model.

If you specify one new variable name and omit equation(), results are the same as if you had specified equation(\#1).
For more information on using predict after multiple-equation estimation commands, see \([R]\) predict.
scores calculates the parameter-level score equations, the first derivatives of the GMM criterion function with respect to the parameters scaled by -0.5 . This option requires that the length of the new variable list be equal to the number of parameters, \(p\). Otherwise, use stub* to have predict generate enumerated variables with prefix stub. The \(j\) th new variable will contain the \(j\) th score of the model.

\section*{margins}

\section*{Description for margins}
margins estimates margins of response for linear predictions.

\section*{Menu for margins}

Statistics \(>\) Postestimation

\section*{Syntax for margins}
```

margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [自redict(statistic ...) ...] [options]

```
statistic Description
\begin{tabular}{ll}
xb & linear prediction; the default \\
\(\underline{\text { residuals }}\) & not allowed with margins
\end{tabular}
xb defaults to the first equation.
Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see \([R]\) margins.

\section*{estat}

\section*{Description for estat}
estat overid reports Hansen's \(J\) statistic, which is used to determine the validity of the overidentifying restrictions in a GMM model. If the model is correctly specified in the sense that \(E\left\{\mathbf{z}_{i} u_{i}(\boldsymbol{\beta})\right\}=\mathbf{0}\), then the sample analog to that condition should hold at the estimated value of \(\boldsymbol{\beta}\). Hansen's \(J\) statistic is valid only if the weight matrix is optimal, meaning that it equals the inverse of the covariance matrix of the moment conditions. Therefore, estat overid only reports Hansen's \(J\) statistic after two-step or iterated estimation, or if you specified winitial (matname) when calling gmm. In the latter case, it is your responsibility to determine the validity of the \(J\) statistic.

\section*{Menu for estat}

Statistics \(>\) Postestimation

\section*{Syntax for estat}
```

        estat overid
    ```
collect is allowed with estat overid; see [U] 11.1.10 Prefix commands.

\section*{Remarks and examples}

As we noted in Introduction of [R] gmm, underlying generalized method of moments (GMM) estimators is a set of \(l\) moment conditions, \(E\left\{\mathbf{z}_{i} u_{i}(\boldsymbol{\beta})\right\}=\mathbf{0}\). When \(l\) is greater than the number of parameters, \(k\), any size- \(k\) subset of the moment conditions would yield a consistent parameter estimate. We remarked that the parameter estimates we would obtain would in general depend on which \(k\) moment conditions we used. However, if all our moment conditions are indeed valid, then the parameter estimates should not differ too much regardless of which \(k\) moment conditions we used to estimate the parameters. The test of overidentifying restrictions is a model specification test based on this observation. The test of overidentifying restrictions requires that the number of moment conditions be greater than the number of parameters in the model.

Recall that the GMM criterion function is
\[
Q=\left\{\frac{1}{N} \sum_{i} \mathbf{z}_{i} u_{i}(\boldsymbol{\beta})\right\}^{\prime} \mathbf{W}\left\{\frac{1}{N} \sum_{i} \mathbf{z}_{i} u_{i}(\boldsymbol{\beta})\right\}
\]

The test of overidentifying restrictions is remarkably simple. If \(\mathbf{W}\) is an optimal weight matrix, under the null hypothesis \(H_{0}: E\left\{\mathbf{z}_{i} u_{i}(\boldsymbol{\beta})\right\}=\mathbf{0}\), the test statistic \(J=N \times Q \sim \chi^{2}(l-k)\). A large test statistic casts doubt on the null hypothesis.

For the test to be valid, \(\mathbf{W}\) must be optimal, meaning that \(\mathbf{W}\) must be the inverse of the covariance matrix of the moment conditions:
\[
\mathbf{W}^{-1}=E\left\{\mathbf{z}_{i} u_{i}(\boldsymbol{\beta}) u_{i}^{\prime}(\boldsymbol{\beta}) \mathbf{z}_{i}^{\prime}\right\}
\]

Therefore, estat overid works only after the two-step and iterated estimators, or if you supplied your own initial weight matrix by using the winitial (matname) option to gmm and used the one-step estimator.

Often the overidentifying restrictions test is interpreted as a test of the validity of the instruments z. However, other forms of model misspecification can sometimes lead to a significant test statistic. See Hall (2005, sec. 5.1) for a discussion of the overidentifying restrictions test and its behavior in correctly and misspecified models.

\section*{> Example 1}

In example 6 of \([R]\) gmm, we fit an exponential regression model of the number of doctor visits based on the person's gender, income, possession of private health insurance, and presence of a chronic disease. We argued that the variable income may be endogenous; we used the person's age and race as additional instrumental variables. Here we refit the model and test the specification of the model. We type
```

. use https://www.stata-press.com/data/r18/docvisits
. gmm (docvis - exp({xb:private chronic female income} + {bO})),
> instruments(private chronic female age black hispanic)
(output omitted)
estat overid
Test of overidentifying restriction:
Hansen's J chi2(2) = 9.52598 ( }\textrm{p}=0.0085\mathrm{ )

```

The \(J\) statistic is significant even at the \(1 \%\) significance level, so we conclude that our model is misspecified. One possibility is that age and race directly affect the number of doctor visits, so we are not justified in excluding them from the model.

A simple technique to explore whether any of the instruments is invalid is to examine the statistics
\[
r_{j}=\mathbf{W}_{j j}^{1 / 2}\left\{\frac{1}{N} \sum_{i=1}^{N} z_{i j} u_{i}(\widehat{\boldsymbol{\beta}})\right\}
\]
for \(j=1, \ldots, k\), where \(\mathbf{W}_{j j}\) denotes the \(j\) th diagonal element of \(\mathbf{W}, u_{i}(\widehat{\boldsymbol{\beta}})\) denotes the sample residuals, and \(k\) is the number of instruments. If all the instruments are valid, then the scaled sample moments should at least be on the same order of magnitude. If one (or more) instrument's \(r_{j}\) is large in absolute value relative to the others, then that could be an indication that instrument is not valid.

In Stata, we type
```

. predict double r if e(sample), residuals // obtain residual from the model
. matrix W = e(W) // retrieve weight matrix
. local i 1
. // loop over each instrument and compute r_j
. foreach var of varlist private chronic female age black hispanic {
2. generate double r'var' = r*'var'*sqrt(W['i', 'i'])
3. local '++i'
4.}
. summarize r*

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| $r$ | 4,412 | .0344373 | 8.26176 | -151.1847 | 113.059 |
| rprivate | 4,412 | .007988 | 3.824118 | -72.66254 | 54.33852 |
| rchronic | 4,412 | .0026947 | 2.0707 | -43.7311 | 32.703 |
| rfemale | 4,412 | .0028168 | 1.566397 | -12.7388 | 24.43621 |
| rage | 4,412 | .0360978 | 4.752986 | -89.74112 | 55.58143 |
| rblack | 4,412 | -.0379317 | 1.062027 | -24.39747 | 27.34512 |
| rhispanic | 4,412 | -.017435 | 1.08567 | -5.509386 | 31.53512 |

```

We notice that the \(r_{j}\) statistics for age, black, and hispanic are larger than those for the other instruments in our model, supporting our suspicion that age and race may have a direct impact on the number of doctor visits.

\section*{Stored results}
estat overid stores the following in r() :
Scalars
\[
\begin{array}{ll}
\mathrm{r}(\mathrm{~J}) & \text { Hansen's } J \text { statistic } \\
\mathrm{r}\left(\mathrm{~J} \_ \text {df }\right) & J \text { statistic degrees of freedom } \\
\mathrm{r}\left(\mathrm{~J} \_\mathrm{p}\right) & J \text { statistic } p \text {-value }
\end{array}
\]

\section*{References}

Hall, A. R. 2005. Generalized Method of Moments. Oxford: Oxford University Press.
Lindsey, C. 2016. Testing model specification and using the program version of gmm. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2016/02/11/testing-model-specification-and-using-the-program-version-of-gmm/.

\section*{Also see}
[R] gmm - Generalized method of moments estimation
[U] 20 Estimation and postestimation commands

\section*{Title}
grmeanby - Graph means and medians by categorical variables
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & References &
\end{tabular}

\section*{Description}
grmeanby graphs the (optionally weighted) means or medians of varname according to the values of the variables in varlist. The variables in varlist may be string or numeric and, if numeric, may be labeled.

\section*{Quick start}

Graph means of v1 for each level of categorical variables cvar1, cvar2, and cvar3 grmeanby cvar1 cvar2 cvar3, sum(v1)

Same as above, but graph medians
grmeanby cvar1 cvar2 cvar3, sum(v1) median
Same as above, but use + as the marker
grmeanby cvar1 cvar2 cvar3, sum(v1) median msymbol(+)

\section*{Menu}

Statistics \(>\) Summaries, tables, and tests \(>\) Summary and descriptive statistics \(>\) Graph means/medians by groups

\section*{Syntax}
grmeanby varlist \([\) if \(][\) in \(][\) weight \(]\), summarize(varname) [options]
options
Main
* summarize (varname)
median
Plot
cline_options
marker_options
marker_label_options
Y axis, X axis, Titles, Legend, Overall
twoway_options

Description
graph mean (or median) of varname
graph medians; default is to graph means
change look of the lines
change look of markers (color, size, etc.)
add marker labels; change look or position
any options other than by () documented in
[G-3] twoway_options
*summarize(varname) is required.
aweights and fweights are allowed; see [U] 11.1.6 weight.

\section*{Options}
\(\qquad\) Main
summarize (varname) is required; it specifies the name of the variable whose mean or median is to be graphed.
median specifies that the graph is to be of medians, not means.
\(\qquad\) Plot
cline_options affect the rendition of the lines through the markers, including their color, pattern, and width; see [G-3] cline_options.
marker_options affect the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G-3] marker_options.
marker_label_options specify if and how the markers are to be labeled; see [G-3] marker_label_options.
\(\qquad\) Y axis, X axis, Titles, Legend, Overall
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

\section*{Remarks and examples}

The idea of graphing means of categorical variables was shown in Chambers and Hastie (1992, 3). Because this was shown in the context of an \(S\) function for making such graphs, it doubtless has roots going back further than that. grmeanby is, in any case, another implementation of what we will assume is their idea.

\section*{> Example 1}

Using a variation of our auto dataset, we graph the mean of mpg by foreign, rep77, rep78, and make:
. use https://www.stata-press.com/data/r18/auto1
(Automobile models)
. grmeanby foreign rep77 rep78 make, sum(mpg)
Means of mpg, Mileage (mpg)


If we had wanted a graph of medians rather than means, we could have typed
- grmeanby foreign rep77 rep78 make, sum(mpg) median

\section*{References}

Chambers, J. M., and T. J. Hastie, ed. 1992. Statistical Models in S. Pacific Grove, CA: Wadsworth and Brooks/Cole. Cox, N. J. 2014. Speaking Stata: Design plots for graphical summary of a response given factors. Stata Journal 14: 975-990.

\section*{Title}
hausman - Hausman specification test
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
Acknowledgment & References & Also see &
\end{tabular}

\section*{Description}
hausman performs Hausman's (1978) specification test.

\section*{Quick start}

Hausman test for stored models consistent and efficient hausman consistent efficient

Same as above, but compare fixed-effects and random-effects linear regression models hausman fixed random, sigmamore

Endogeneity test after ivprobit and probit with estimates stored in iv and noiv hausman iv noiv, equations(1:1)

Test of independence of irrelevant alternatives for model with all alternatives all and model with omitted alternative omitted
hausman omitted all, alleqs constant

\section*{Menu}

Statistics \(>\) Postestimation

\section*{Syntax}
hausman name-consistent [name-efficient] [, options]
name-consistent and name-efficient are names under which estimation results were stored via estimates store; see [R] estimates store. A period (.) may be used to refer to the last estimation results, even if these were not already stored. Not specifying name-efficient is equivalent to specifying the last estimation results as ".".
options
Main
constant
alleqs
skipeqs (eqlist)
equations(matchlist)
force
df (\#)
sigmamore
sigmaless

Advanced
tconsistent (string)
tefficient (string)

Description
include estimated intercepts in comparison; default is to exclude use all equations to perform test; default is first equation only skip specified equations when performing test
associate/compare the specified (by number) pairs of equations force performance of test, even though assumptions are not met use \# degrees of freedom
base both (co)variance matrices on disturbance variance estimate from efficient estimator
base both (co)variance matrices on disturbance variance estimate from consistent estimator
consistent estimator column header
efficient estimator column header
collect is allowed; see [U] 11.1.10 Prefix commands.

\section*{Options}

\section*{Main}
constant specifies that the estimated intercept(s) be included in the model comparison; by default, they are excluded. The default behavior is appropriate for models in which the constant does not have a common interpretation across the two models.
alleqs specifies that all the equations in the models be used to perform the Hausman test; by default, only the first equation is used.
skipeqs (eqlist) specifies in eqlist the names of equations to be excluded from the test. Equation numbers are not allowed in this context, because the equation names, along with the variable names, are used to identify common coefficients.
equations (matchlist) specifies, by number, the pairs of equations that are to be compared.
The matchlist in equations() should follow the syntax
\[
\#_{c}: \#_{e}\left[, \#_{c}: \#_{e}[, \ldots]\right]
\]
where \(\#_{c}\left(\#_{e}\right)\) is an equation number of the always-consistent (efficient under \(H_{0}\) ) estimator. For instance, equations ( \(1: 1\) ), equations ( \(1: 1,2: 2\) ), or equations ( \(1: 2\) ).
If equations() is not specified, then equations are matched on equation names.
equations() handles the situation in which one estimator uses equation names and the other does not. For instance, equations ( \(1: 2\) ) means that equation 1 of the always-consistent estimator is to be tested against equation 2 of the efficient estimator. equations ( \(1: 1,2: 2\) ) means that equation 1 is to be tested against equation 1 and that equation 2 is to be tested against equation 2 . If equations() is specified, the alleqs and skipeqs options are ignored.
force specifies that the Hausman test be performed, even though the assumptions of the Hausman test seem not to be met, for example, because the estimators were pweighted or the data were clustered.
\(\mathrm{df}(\#)\) specifies the degrees of freedom for the Hausman test. The default is the matrix rank of the variance of the difference between the coefficients of the two estimators.
sigmamore and sigmaless specify that the two covariance matrices used in the test be based on a common estimate of disturbance variance \(\left(\sigma^{2}\right)\).
sigmamore specifies that the covariance matrices be based on the estimated disturbance variance from the efficient estimator. This option provides a proper estimate of the contrast variance for so-called tests of exogeneity and overidentification in instrumental-variables regression.
sigmaless specifies that the covariance matrices be based on the estimated disturbance variance from the consistent estimator.

These options can be specified only when both estimators store e(sigma) or e(rmse), or with the xtreg command. e(sigma_e) is stored after the xtreg command with the fe or mle option. \(e\) (rmse) is stored after the xtreg command with the re option.
sigmamore or sigmaless are recommended when comparing fixed-effects and random-effects linear regression because they are much less likely to produce a non-positive-definite-differenced covariance matrix (although the tests are asymptotically equivalent whether or not one of the options is specified).

\section*{Advanced}
tconsistent (string) and tefficient (string) are formatting options. They allow you to specify the headers of the columns of coefficients that default to the names of the models. These options will be of interest primarily to programmers.

\section*{Remarks and examples}
hausman is a general implementation of Hausman's (1978) specification test, which compares an estimator \(\widehat{\theta}_{1}\) that is known to be consistent with an estimator \(\widehat{\theta}_{2}\) that is efficient under the assumption being tested. The null hypothesis is that the estimator \(\widehat{\theta}_{2}\) is indeed an efficient (and consistent) estimator of the true parameters. If this is the case, there should be no systematic difference between the two estimators. If there exists a systematic difference in the estimates, you have reason to doubt the assumptions on which the efficient estimator is based.

The assumption of efficiency is violated if the estimator is pweighted or the data are clustered, so hausman cannot be used. The test can be forced by specifying the force option with hausman. For an alternative to using hausman in these cases, see [R] suest.

To use hausman, you
. (compute the always-consistent estimator)
. estimates store name-consistent
. (compute the estimator that is efficient under H_0)
. hausman name-consistent .

Alternatively, you can turn this around:
. (compute the estimator that is efficient under H_0)
- estimates store name-efficient
. (fit the less-efficient model)
. (compute the always-consistent estimator)
. hausman . name-efficient
You can, of course, also compute and store both the always-consistent and efficient-under- \(H_{0}\) estimators and perform the Hausman test with
. hausman name-consistent name-efficient

\section*{> Example 1}

We are studying the factors that affect the wages of young women in the United States between 1968 and 1988, and we have a panel-data sample of individual women over that time span.


Sorted by: idcode year
We believe that a random-effects specification is appropriate for individual-level effects in our model. We fit a fixed-effects model that will capture all temporally constant individual-level effects.


We assume that this model is consistent for the true parameters and store the results by using estimates store under a name, fixed:
```

. estimates store fixed

```

Now we fit a random-effects model as a fully efficient specification of the individual effects under the assumption that they are random and follow a normal distribution. We then compare these estimates with the previously stored results by using the hausman command.
\begin{tabular}{|c|c|c|c|c|c|c|c|}
\hline \multicolumn{4}{|l|}{\multirow[t]{2}{*}{Random-effects GLS regression Group variable: idcode}} & Number & \(f\) obs & = & 28,494 \\
\hline & & & & Number & \(f\) groups & = & 4,710 \\
\hline \multicolumn{4}{|l|}{R-squared:} & \multicolumn{4}{|l|}{Obs per group:} \\
\hline \multicolumn{4}{|l|}{Within \(=0.1373\)} & & min & & 1 \\
\hline \multicolumn{4}{|l|}{Between \(=0.2552\)} & & avg & = & 6.0 \\
\hline \multicolumn{4}{|l|}{Overall \(=0.1797\)} & & max & = & 15 \\
\hline & & & & Wald ch & 2(3) & = & 5100.33 \\
\hline \multicolumn{4}{|l|}{\(\operatorname{corr}\left(u_{\text {_ }} \mathrm{i}, \mathrm{X}\right)=0\) (assumed)} & Prob > & hi2 & = & 0.0000 \\
\hline ln_wage & Coefficient & Std. err. & z & \(P>|z|\) & \multicolumn{3}{|l|}{[95\% conf. interval]} \\
\hline age & -. 0069749 & . 0006882 & -10.13 & 0.000 & -. 0083238 & & -. 0056259 \\
\hline msp & . 0046594 & . 0051012 & 0.91 & 0.361 & -. 0053387 & & . 0146575 \\
\hline ttl_exp & . 0429635 & . 0010169 & 42.25 & 0.000 & . 0409704 & & . 0449567 \\
\hline _cons & 1.609916 & . 0159176 & 101.14 & 0.000 & 1.578718 & & 1.641114 \\
\hline sigma_u & . 32648519 & & & & & & \\
\hline sigma_e & . 29751014 & & & & & & \\
\hline rho & . 54633481 & \multicolumn{6}{|l|}{(fraction of variance due to u_i)} \\
\hline
\end{tabular}
```

. hausman fixed ., sigmamore
_ Coefficients -_

|  | $(b)$ <br> fixed | $(B)$ <br> age | -.005485 | -.0069749 |
| ---: | :---: | :---: | ---: | :---: |
| msp | .0033427 | .0046594 | $(b-B)$ <br> Difference | sqrt(diag(V_b-V_B)) <br> Std. err. |
| ttl_exp | .0383604 | .0429635 | -.0014899 | .0004803 |

                        b = Consistent under HO and Ha; obtained from xtreg.
                    B = Inconsistent under Ha, efficient under HO; obtained from xtreg.
    Test of HO: Difference in coefficients not systematic
chi2(3) = (b-B)'[(V_b-V_B)^(-1)](b-B)
= 260.40
Prob > chi2 = 0.0000

```

Under the current specification, our initial hypothesis that the individual-level effects are adequately modeled by a random-effects model is resoundingly rejected. This result is based on the rest of our model specification, and random effects might be appropriate for some alternate model of wages.

Jerry Allen Hausman (1946- ) is an American economist and econometrician. He was born in West Virginia and went on to study economics at Brown and Oxford. He joined the MIT faculty in 1972 and continues to teach there. He currently researches new goods and their effects on consumer welfare and its measurement in the Consumer Price Index along with regulation and competition in the telecommunications industry.

Hausman is best known for his many contributions to econometrics. In 1978, he published his now famous paper giving the Hausman specification test. The work remains one of the most widely cited econometrics papers. He has also done extensive work in applied microeconomics pertaining to governments role in the economy, including antitrust regulation, public finance, and taxation.

In 1980, Hausman received the Frisch Medal, a biennial award from the Econometric Society recognizing exceptional applied work, for his paper with David Wise on attrition bias. In 1985, he won the John Bates Clark Award from the American Economics Association, which is given for outstanding contributions to economics by an economist under 40 years of age. In 2012, the Advances in Econometrics book series devoted an entire volume to Hausman and his contributions to econometrics.

\section*{> Example 2}

A stringent assumption of multinomial and conditional logit models is that outcome categories for the model have the property of independence of irrelevant alternatives (IIA). Stated simply, this assumption requires that the inclusion or exclusion of categories does not affect the relative risks associated with the regressors in the remaining categories.

One classic example of a situation in which this assumption would be violated involves the choice of transportation mode; see McFadden (1974). For simplicity, postulate a transportation model with the four possible outcomes: rides a train to work, takes a bus to work, drives the Ford to work, and drives the Chevrolet to work. Clearly, "drives the Ford" is a closer substitute to "drives the Chevrolet" than it is to "rides a train" (at least for most people). This means that excluding "drives the Ford" from the model could be expected to affect the relative risks of the remaining options and that the model would not obey the IIA assumption.

Using the data presented in \([\mathrm{R}]\) mlogit, we will use a simplified model to test for IIA. The choice of insurance type among indemnity, prepaid, and uninsured is modeled as a function of age and gender. The indemnity category is allowed to be the base category, and the model including all three outcomes is fit. The results are then stored under the name allcats.
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline \multicolumn{7}{|l|}{\begin{tabular}{l}
. use https://www.stata-press.com/d \\
(Health insurance data) \\
. mlogit insure age male
\end{tabular}} \\
\hline \multicolumn{7}{|l|}{Iteration 0: Log likelihood \(=-555.85446\)} \\
\hline \multicolumn{7}{|l|}{Iteration 1: Log likelihood \(=-551.32973\)} \\
\hline \multicolumn{7}{|l|}{Iteration 2: Log likelihood \(=-551.32802\)} \\
\hline \multicolumn{7}{|l|}{Iteration 3: Log likelihood = -551.32802} \\
\hline \multicolumn{5}{|l|}{Multinomial logistic regression} & Number of obs & \(=615\) \\
\hline & & & & & LR chi2(4) & \(=9.05\) \\
\hline & & & & & Prob > chi2 & \(=0.0598\) \\
\hline \multicolumn{5}{|l|}{Log likelihood = -551.32802} & Pseudo R2 & \(=0.0081\) \\
\hline insure & Coefficient & Std. err. & z & \(P>|z|\) & [95\% conf. & interval] \\
\hline Indemnity & \multicolumn{6}{|l|}{(base outcome)} \\
\hline \multicolumn{7}{|l|}{Prepaid} \\
\hline age & -. 0100251 & . 0060181 & -1.67 & 0.096 & -. 0218204 & . 0017702 \\
\hline male & . 5095747 & . 1977893 & 2.58 & 0.010 & . 1219147 & . 8972346 \\
\hline _cons & . 2633838 & . 2787575 & 0.94 & 0.345 & -. 2829708 & . 8097383 \\
\hline \multicolumn{7}{|l|}{Uninsure} \\
\hline age & -. 0051925 & . 0113821 & -0.46 & 0.648 & -. 0275011 & . 0171161 \\
\hline male & . 4748547 & . 3618462 & 1.31 & 0.189 & -. 2343508 & 1.18406 \\
\hline _cons & -1.756843 & . 5309602 & -3.31 & 0.001 & -2.797506 & -. 7161803 \\
\hline
\end{tabular}

Under the IIA assumption, we would expect no systematic change in the coefficients if we excluded one of the outcomes from the model. (For an extensive discussion, see Hausman and McFadden [1984].) We reestimate the parameters, excluding the uninsured outcome, and perform a Hausman test against the fully efficient full model.
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline \multicolumn{7}{|l|}{Iteration 0: Log likelihood \(=-394.8693\)} \\
\hline \multicolumn{7}{|l|}{Iteration 1: Log likelihood \(=-390.4871\)} \\
\hline \multicolumn{7}{|l|}{Iteration 2: Log likelihood \(=-390.48643\)} \\
\hline \multicolumn{7}{|l|}{Iteration 3: Log likelihood \(=-390.48643\)} \\
\hline \multicolumn{5}{|l|}{\multirow[t]{2}{*}{Multinomial logistic regression}} & Number of obs & \(=570\) \\
\hline & & & & & LR chi2(2) & \(=8.77\) \\
\hline & & & & & Prob > chi2 & \(=0.0125\) \\
\hline \multicolumn{5}{|l|}{Log likelihood \(=-390.48643\)} & Pseudo R2 & \(=0.0111\) \\
\hline insure & Coefficient & Std. err & z & \(P>|z|\) & [95\% conf. & interval] \\
\hline Indemnity & (base outc & & & & & \\
\hline \multicolumn{7}{|l|}{Prepaid} \\
\hline age & -. 0101521 & . 0060049 & -1.69 & 0.091 & -. 0219214 & . 0016173 \\
\hline male & . 5144003 & . 1981735 & 2.60 & 0.009 & . 1259874 & . 9028133 \\
\hline _cons & . 2678043 & . 2775563 & 0.96 & 0.335 & -. 276196 & . 8118046 \\
\hline
\end{tabular}
hausman . allcats, alleqs constant
—— Coefficients —_
\begin{tabular}{r|cccc} 
& \((\mathrm{b})\) & \begin{tabular}{c}
\((B)\) \\
allcats
\end{tabular} & \begin{tabular}{c}
\((\mathrm{b}-\mathrm{B})\) \\
Difference
\end{tabular} & \begin{tabular}{c} 
sqrt(diag(V_b-V_B)) \\
Std. err.
\end{tabular} \\
\hline age & -.0101521 & -.0100251 & -.0001269 & \\
male & .5144003 & .5095747 & .0048256 & .0123338 \\
_cons & .2678043 & .2633838 & .0044205 &. \\
\hline
\end{tabular}
b = Consistent under HO and Ha ; obtained from mlogit.
B = Inconsistent under Ha, efficient under HO; obtained from mlogit.
Test of HO: Difference in coefficients not systematic
\(\operatorname{chi2}(3)=(b-B) \cdot\left[\left(V \_b-V \_B\right)^{\wedge}(-1)\right](b-B)\)
\(=0.08\)
Prob > chi2 = 0.9944
(V_b-V_B is not positive definite)
The syntax of the if condition on the mlogit command simply identified the "Uninsured" category with the insure value label; see [U] 12.6.3 Value labels. On examining the output from hausman, we see that there is no evidence that the IIA assumption has been violated.

Because the Hausman test is a standardized comparison of model coefficients, using it with mlogit requires that the base outcome be the same in both competing models. In particular, if the most-frequent category (the default base outcome) is being removed to test for IIA, you must use the baseoutcome () option in mlogit to manually set the base outcome to something else. Or you can use the equation() option of the hausman command to align the equations of the two models.

Having the missing values for the square root of the diagonal of the covariance matrix of the differences is not comforting, but it is also not surprising. This covariance matrix is guaranteed to be positive definite only asymptotically (it is a consequence of the assumption that one of the estimators is efficient), and assurances are not made about the diagonal elements. Negative values along the diagonal are possible, and the fourth column of the table is provided mainly for descriptive use.

We can also perform the Hausman IIA test against the remaining alternative in the model:
```

. mlogit insure age male if insure != "Prepaid":insure
Iteration 0: Log likelihood = -132.59913
Iteration 1: Log likelihood = -131.78009
Iteration 2: Log likelihood = -131.76808
Iteration 3: Log likelihood = -131.76807
Multinomial logistic regression Number of obs = 338

```
    LR chi2(2) \(=1.66\)
    Prob > chi2 \(=0.4356\)
    Pseudo R2 \(=0.0063\)
Log likelihood = -131.76807
\begin{tabular}{r|rlllll}
\hline \multicolumn{1}{c|}{ insure } & Coefficient & Std. err. & \(z\) & \(\mathrm{P}>|z|\) & [95\% conf. interval] \\
\hline Indemnity & (base outcome) & & & & \\
\hline \multicolumn{2}{c}{ Uninsure } & & & & & \\
age & -.0041055 & .0115807 & -0.35 & 0.723 & -.0268033 & .0185923 \\
male & .4591074 & .3595663 & 1.28 & 0.202 & -.2456296 & 1.163844 \\
_cons & -1.801774 & .5474476 & -3.29 & 0.001 & -2.874752 & -.7287968 \\
\hline
\end{tabular}
```

. hausman . allcats, alleqs constant
_ Coefficients -

|  | (b) | (B) <br> allcats | (b-B) <br> Difference | sqrt(diag(V_b-V_B)) <br> Std.err. |
| ---: | ---: | ---: | ---: | ---: |
| age | -.0041055 | -.0051925 | .001087 | .0021355 |
| male | .4591074 | .4748547 | -.0157473 | .1333421 |
| cons | -1.801774 | -1.756843 | -.0449311 | .1 |

                        b = Consistent under HO and Ha; obtained from mlogit.
        B = Inconsistent under Ha, efficient under HO; obtained from mlogit.
    Test of HO: Difference in coefficients not systematic
chi2(3) = (b-B)'[(V_b-V_B)^(-1)](b-B)
= -0.18
Warning: chi2 < O ==> model fitted on these data
fails to meet the asymptotic assumptions
of the Hausman test; see suest for a
generalized test.

```

Here the \(\chi^{2}\) statistic is actually negative. We might interpret this result as strong evidence that we cannot reject the null hypothesis. Such a result is not an unusual outcome for the Hausman test, particularly when the sample is relatively small-there are only 45 uninsured individuals in this dataset.

Are we surprised by the results of the Hausman test in this example? Not really. Judging from the \(z\) statistics on the original multinomial logit model, we were struggling to identify any structure in the data with the current specification. Even when we were willing to assume IIA and computed the efficient estimator under this assumption, few of the effects could be identified as statistically different from those on the base category. Trying to base a Hausman test on a contrast (difference) between two poor estimates is just asking too much of the existing data.

In example 2, we encountered a case in which the Hausman was not well defined. Unfortunately, in our experience this happens fairly often. Stata provides an alternative to the Hausman test that overcomes this problem through an alternative estimator of the variance of the difference between the two estimators. This other estimator is guaranteed to be positive semidefinite. This alternative estimator also allows a widening of the scope of problems to which Hausman-type tests can be applied by relaxing the assumption that one of the estimators is efficient. For instance, you can perform Hausman-type tests to clustered observations and survey estimators. See [R] suest for details.

\section*{Stored results}
hausman stores the following in \(r()\) :
Scalars
```

r(chi2) < \chi
r(df) degrees of freedom for the statistic
r(p) p-value for the }\mp@subsup{\chi}{}{2
r(rank) rank of (V_b-V_B)^(-1)

```

\section*{Methods and formulas}

The Hausman statistic is distributed as \(\chi^{2}\) and is computed as
\[
H=\left(\beta_{c}-\beta_{e}\right)^{\prime}\left(V_{c}-V_{e}\right)^{-1}\left(\beta_{c}-\beta_{e}\right)
\]
where
\(\beta_{c}\) is the coefficient vector from the consistent estimator
\(\beta_{e}\) is the coefficient vector from the efficient estimator
\(V_{c}\) is the covariance matrix of the consistent estimator
\(V_{e}\) is the covariance matrix of the efficient estimator
When the difference in the variance matrices is not positive definite, a Moore-Penrose generalized inverse is used. As noted in Gouriéroux and Monfort (1995, 125-128), the choice of generalized inverse is not important asymptotically.

The number of degrees of freedom for the statistic is the rank of the difference in the variance matrices. When the difference is positive definite, this is the number of common coefficients in the models being compared.

\section*{Acknowledgment}

Portions of hausman are based on an early implementation by Jeroen Weesie of the Department of Sociology at Utrecht University, The Netherlands.

\section*{References}

Baltagi, B. H. 2011. Econometrics. 5th ed. Berlin: Springer.
Gouriéroux, C. S., and A. Monfort. 1995. Statistics and Econometric Models, Vol. 2: Testing, Confidence Regions, Model Selection, and Asymptotic Theory. Trans. Q. Vuong. Cambridge: Cambridge University Press.

Hausman, J. A. 1978. Specification tests in econometrics. Econometrica 46: 1251-1271. https://doi.org/10.2307/1913827.
Hausman, J. A., and D. L. McFadden. 1984. Specification tests for the multinomial logit model. Econometrica 52: 1219-1240. https://doi.org/10.2307/1910997.

McFadden, D. L. 1974. Measurement of urban travel demand. Journal of Public Economics 3: 303-328. https://doi.org/10.1016/0047-2727(74)90003-6.

\section*{Also see}
[R] Irtest - Likelihood-ratio test after estimation
[R] suest - Seemingly unrelated estimation
[R] test - Test linear hypotheses after estimation
[XT] xtreg - Linear models for panel data \({ }^{+}\)

\section*{Title}
heckman - Heckman selection model

Description
Menu
Options for Heckman selection model (ML)
Remarks and examples
Methods and formulas
Also see

\section*{Quick start}

Syntax
Options for Heckman selection model (two-step)
Stored results
References

\section*{Description}
heckman fits regression models with selection by using either Heckman's two-step consistent estimator or full maximum likelihood.

\section*{Quick start}

Heckman model of y on x 1 with v 1 predicting selection when binary variable selected indicates selection status
heckman y x1, select (selected \(=\mathrm{v} 1 \mathrm{x} 1\) )
Same as above, and generate v2 containing the inverse of Mills's ratio
heckman y x1, select(selected = v1 x1) mills(v2)
Same as above
heckman y x1, select(selected = v1 x1) nshazard(v2)
Fit a Heckman model using the two-step estimation method
heckman y x1, select (selected \(=\mathrm{v} 1 \mathrm{x} 1\) ) twostep
Same as above, and display first-stage probit estimates
heckman y x1, select(selected = v1 x1) twostep first

\section*{Menu}

Statistics \(>\) Sample-selection models \(>\) Heckman selection model

\section*{Syntax}

Basic syntax
\[
\text { heckman depvar }[\text { indepvars }] \text {, select (varlist }) \text { [twostep }]
\]
or
heckman depvar \([\) indepvars \(]\), select (depvar \({ }_{s}=\) varlist \(\left._{s}\right)\) [twostep]

Full syntax for maximum likelihood estimates only
```

heckman depvar [indepvars] [if] [in] [weight],
select([depvars = ] varlists [, noconstant offset(\mp@subsup{\mathrm{ varname }}{o}{})])
[heckman_ml_options]

```

Full syntax for Heckman's two-step consistent estimates only heckman depvar [indepvars] [if] [in], twostep select([depvar \(=]\) varlist \(_{s}[\), noconstant \(\left.]\right)\) [heckman_ts_options]

Description

Model mle
* select ()
noconstant
offset (varname)
constraints (constraints)
SE/Robust
vce (vcetype)

Reporting
level (\#)
first
lrmodel
nshazard (newvar)
mills (newvar)
nocnsreport
display_options

Maximization
maximize_options
collinear
coeflegend
use maximum likelihood estimator; the default
specify selection equation: dependent and independent
variables; whether to have constant term and offset variable suppress constant term
include varname in model with coefficient constrained to 1
apply specified linear constraints
vcetype may be oim, robust, cluster clustvar, opg, bootstrap, or jackknife
set confidence level; default is level (95)
report first-step probit estimates
perform the likelihood-ratio model test instead of the default Wald test
generate nonselection hazard variable
synonym for nshazard ()
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
control the maximization process; seldom used
keep collinear variables
display legend instead of statistics
* select ( ) is required.

The full specification is select \(\left(\left[\right.\right.\) depvar \(\left._{s}=\right] \operatorname{varlist}_{S}\left[\right.\), noconstant offset \(\left(\right.\) varname \(\left.\left.\left._{O}\right)\right]\right)\).

\section*{Model}
*twostep
*select ()
noconstant
rhosigma
rhotrunc
rholimited
rhoforce
SE
vce (vcetype)
Reporting
level(\#)
first
nshazard (newvar)
mills (newvar)
display_options
coeflegend
produce two-step consistent estimate
specify selection equation: dependent and independent
variables; whether to have constant term
suppress constant term
truncate \(\rho\) to \([-1,1]\) with consistent \(\sigma\)
truncate \(\rho\) to \([-1,1]\)
truncate \(\rho\) in limited cases
do not truncate \(\rho\)
vcetype may be conventional, bootstrap, or jackknife
set confidence level; default is level (95)
report first-step probit estimates
generate nonselection hazard variable
synonym for nshazard ()
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
display legend instead of statistics
*twostep and select ( ) are required.
The full specification is select \(\left(\left[\right.\right.\) depvar \(\left._{s}=\right]\) varlist \(_{s}[\), noconstant \(\left.]\right)\).
indepvars and varlists may contain factor variables; see [U] 11.4.3 Factor variables.
depvar, indepvars, varlist \(t_{s}\), and depvar \({ }_{s}\) may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bayes, bootstrap, by, collect, fp, jackknife, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: heckman.
Weights are not allowed with the bootstrap prefix; see [R] bootstrap.
twostep, vce(), first, lrmodel, and weights are not allowed with the svy prefix; see [SVY] svy.
pweights, fweights, and iweights are allowed with maximum likelihood estimation; see [U] 11.1.6 weight. No weights are allowed if twostep is specified.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options for Heckman selection model (ML)}
\(\qquad\)
mle requests that the maximum likelihood estimator be used. This is the default.
select ( \(\left[\right.\) depvar \(\left._{s}=\right]\) varlist \(_{s}\left[\right.\), noconstant offset (varname \({ }_{o}\) ) \(]\) ) specifies the variables and options for the selection equation. It is an integral part of specifying a Heckman model and is required. The selection equation should contain at least one variable that is not in the outcome equation.
 and 1 indicating a selected observation. If depvar \({ }_{s}\) is not specified, observations for which depvar is not missing are assumed selected, and those for which depvar is missing are assumed not selected.
noconstant suppresses the selection constant term (intercept).
offset ( varname \(_{o}\) ) specifies that selection offset varname \({ }_{o}\) be included in the model with the coefficient constrained to be 1 .
noconstant, offset (varname), constraints (constraints); see [R] Estimation options.

\section*{SE/Robust}
vce(vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.

Reporting
level (\#); see [R] Estimation options.
first specifies that the first-step probit estimates of the selection equation be displayed before estimation.
lrmodel; see [R] Estimation options.
nshazard(newvar) and mills (newvar) are synonyms; either will create a new variable containing the nonselection hazard—what Heckman (1979) referred to as the inverse of the Mills ratio-from the selection equation. The nonselection hazard is computed from the estimated parameters of the selection equation.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap (\#), fvwrapon (style), cformat (\% fint), pformat (\%fmt), sformat (\% fmt), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate(\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from (init_specs); see [R] Maximize. These options are seldom used.

Setting the optimization type to technique (bhhh) resets the default vcetype to vce (opg).
The following options are available with heckman but are not shown in the dialog box: collinear, coeflegend; see [R] Estimation options.

\section*{Options for Heckman selection model (two-step)}

\footnotetext{
Model
twostep specifies that Heckman's (1979) two-step efficient estimates of the parameters, standard errors, and covariance matrix be produced.
}
select ([depvar \(=]\) varlist \(_{s}\) [, noconstant \(]\) ) specifies the variables and options for the selection equation. It is an integral part of specifying a Heckman model and is required. The selection equation should contain at least one variable that is not in the outcome equation.
If depvar \({ }_{s}\) is specified, it should be coded as 0 or 1 , with 0 indicating an observation not selected and 1 indicating a selected observation. If depvar \({ }_{s}\) is not specified, observations for which depvar is not missing are assumed selected, and those for which depvar is missing are assumed not selected.
noconstant suppresses the selection constant term (intercept).
noconstant; see [R] Estimation options.
rhosigma, rhotrunc, rholimited, and rhoforce are rarely used options to specify how the two-step estimator (option twostep) handles unusual cases in which the two-step estimate of \(\rho\) is outside the admissible range for a correlation, \([-1,1]\). When \(\operatorname{abs}(\rho)>1\), the two-step estimate of the coefficient variance-covariance matrix may not be positive definite and thus may be unusable for testing. The default is rhosigma.
rhosigma specifies that \(\rho\) be truncated, as with the rhotrunc option, and that the estimate of \(\sigma\) be made consistent with \(\widehat{\rho}\), the truncated estimate of \(\rho\). So, \(\widehat{\sigma}=\boldsymbol{\beta}_{m} \widehat{\rho}\), see Methods and formulas for the definition of \(\boldsymbol{\beta}_{m}\). Both the truncated \(\rho\) and the new estimate of \(\widehat{\sigma}\) are used in all computations to estimate the two-step covariance matrix.
rhotrunc specifies that \(\rho\) be truncated to lie in the range \([-1,1]\). If the two-step estimate is less than \(-1, \rho\) is set to -1 ; if the two-step estimate is greater than \(1, \rho\) is set to 1 . This truncated value of \(\rho\) is used in all computations to estimate the two-step covariance matrix.
rholimited specifies that \(\rho\) be truncated only in computing the diagonal matrix \(\mathbf{D}\) as it enters \(\mathbf{V}_{\text {twostep }}\) and \(\mathbf{Q}\); see Methods and formulas. In all other computations, the untruncated estimate of \(\rho\) is used.
rhoforce specifies that the two-step estimate of \(\rho\) be retained, even if it is outside the admissible range for a correlation. This option may, in rare cases, lead to a non-positive-definite covariance matrix.

These options have no effect when estimation is by maximum likelihood, the default. They also have no effect when the two-step estimate of \(\rho\) is in the range \([-1,1]\).

SE
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (conventional) and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
vce (conventional), the default, uses the two-step variance estimator derived by Heckman.

Reporting
level(\#); see [R] Estimation options.
first specifies that the first-step probit estimates of the selection equation be displayed before estimation.
nshazard (newvar) and mills (newvar) are synonyms; either will create a new variable containing the nonselection hazard - what Heckman (1979) referred to as the inverse of the Mills ratio-from the selection equation. The nonselection hazard is computed from the estimated parameters of the selection equation.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fint), pformat (\%fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.

The following option is available with heckman but is not shown in the dialog box:
coeflegend; see [R] Estimation options.

\section*{Remarks and examples}

The Heckman selection model (Gronau 1974; Lewis 1974; Heckman 1976) assumes that there exists an underlying regression relationship,
\[
y_{j}=\mathbf{x}_{j} \boldsymbol{\beta}+u_{1 j} \quad \text { regression equation }
\]

The dependent variable, however, is not always observed. Rather, the dependent variable for observation \(j\) is observed if
\[
\mathbf{z}_{j} \gamma+u_{2 j}>0 \quad \text { selection equation }
\]
where
\[
\begin{gathered}
u_{1} \sim N(0, \sigma) \\
u_{2} \sim N(0,1) \\
\operatorname{corr}\left(u_{1}, u_{2}\right)=\rho
\end{gathered}
\]

When \(\rho \neq 0\), standard regression techniques applied to the first equation yield biased results. heckman provides consistent, asymptotically efficient estimates for all the parameters in such models.

In one classic example, the first equation describes the wages of women. Women choose whether to work, and thus, from our point of view as researchers, whether we observe their wages in our data. If women made this decision randomly, we could ignore that not all wages are observed and use ordinary regression to fit a wage model. Such an assumption of random participation, however, is unlikely to be true; women who would have low wages may be unlikely to choose to work, and thus the sample of observed wages is biased upward. In the jargon of economics, women choose not to work when their personal reservation wage is greater than the wage offered by employers. Thus, women who choose not to work might have even higher offer wages than those who do work - they may have high offer wages, but they have even higher reservation wages. We could tell a story that competency is related to wages, but competency is rewarded more at home than in the labor force.

In any case, in this problem - which is the paradigm for most such problems - a solution can be found if there are some variables that strongly affect the chances for observation (the reservation wage) but not the outcome under study (the offer wage). Such a variable might be the number of children in the home. (Theoretically, we do not need such identifying variables, but without them, we depend on functional form to identify the model. It would be difficult for anyone to take such results seriously because the functional form assumptions have no firm basis in theory.)

\section*{> Example 1}

In the syntax for heckman, depvar and indepvars are the dependent variable and regressors for the underlying regression model to be fit \((\mathbf{y}=\mathbf{X} \boldsymbol{\beta})\), and varlist \(t_{s}\) are the variables \((\mathbf{Z})\) thought to determine whether depvar is observed or unobserved (selected or not selected). In our female wage example, the number of children at home would be included in the second list. By default, heckman assumes that missing values (see [U] 12.2.1 Missing values) of depvar imply that the dependent variable is
unobserved (not selected). With some datasets, it is more convenient to specify a binary variable (depvar \({ }_{s}\) ) that identifies the observations for which the dependent is observed/selected (depvar \({ }_{s} \neq 0\) ) or not observed (depvar \(=0\) ); heckman will accommodate either type of data. Here we have a (fictional) dataset on 2,000 women, 1,343 of whom work:
```

. use https://www.stata-press.com/data/r18/womenwk

```
. summarize age educ married children wage
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline age & 2,000 & 36.208 & 8.28656 & 20 & 59 \\
education & 2,000 & 13.084 & 3.045912 & 10 & 20 \\
married & 2,000 & .6705 & .4701492 & 0 & 1 \\
children & 2,000 & 1.6445 & 1.398963 & 0 & 5 \\
wage & 1,343 & 23.69217 & 6.305374 & 5.88497 & 45.80979
\end{tabular}

We will assume that the hourly wage is a function of education and age, whereas the likelihood of working (the likelihood of the wage being observed) is a function of marital status, the number of children at home, and (implicitly) the wage (via the inclusion of age and education, which we think determine the wage):
\begin{tabular}{|c|c|c|c|}
\hline \multicolumn{4}{|l|}{Iteration 0: Log likelihood \(=-5178.7009\)} \\
\hline \multicolumn{4}{|l|}{Iteration 1: Log likelihood \(=-5178.3049\)} \\
\hline \multicolumn{4}{|l|}{Iteration 2: Log likelihood \(=-5178.3045\)} \\
\hline \multirow[t]{3}{*}{\begin{tabular}{l}
Heckman selection model \\
(regression model with sample selection)
\end{tabular}} & Number of obs & = & 2,000 \\
\hline & Selected & = & 1,343 \\
\hline & Nonselected & = & 657 \\
\hline & Wald chi2(2) & = & 508.44 \\
\hline Log likelihood = -5178.304 & Prob > chi2 & = & 0.0000 \\
\hline
\end{tabular}
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline wage & Coefficient & Std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & \multicolumn{2}{|l|}{[95\% conf. interval]} \\
\hline \multicolumn{7}{|l|}{wage} \\
\hline education & . 9899537 & . 0532565 & 18.59 & 0.000 & . 8855729 & 1.094334 \\
\hline age & . 2131294 & . 0206031 & 10.34 & 0.000 & . 1727481 & . 2535108 \\
\hline _cons & . 4857752 & 1.077037 & 0.45 & 0.652 & -1.625179 & 2.59673 \\
\hline \multicolumn{7}{|l|}{select} \\
\hline married & . 4451721 & . 0673954 & 6.61 & 0.000 & . 3130794 & . 5772647 \\
\hline children & . 4387068 & . 0277828 & 15.79 & 0.000 & . 3842534 & . 4931601 \\
\hline education & . 0557318 & . 0107349 & 5.19 & 0.000 & . 0346917 & . 0767718 \\
\hline age & . 0365098 & . 0041533 & 8.79 & 0.000 & . 0283694 & . 0446502 \\
\hline _cons & -2.491015 & . 1893402 & -13.16 & 0.000 & -2.862115 & -2.119915 \\
\hline /athrho & . 8742086 & . 1014225 & 8.62 & 0.000 & . 6754241 & 1.072993 \\
\hline /lnsigma & 1.792559 & . 027598 & 64.95 & 0.000 & 1.738468 & 1.84665 \\
\hline rho & . 7035061 & . 0512264 & & & . 5885365 & . 7905862 \\
\hline sigma & 6.004797 & . 1657202 & & & 5.68862 & 6.338548 \\
\hline lambda & 4.224412 & . 3992265 & & & 3.441942 & 5.006881 \\
\hline
\end{tabular}
heckman assumes that wage is the dependent variable and that the first variable list (educ and age) are the determinants of wage. The variables specified in the select() option (married, children, educ, and age) are assumed to determine whether the dependent variable is observed (the selection equation). Thus, we fit the model
\[
\text { wage }=\beta_{0}+\beta_{1} \text { educ }+\beta_{2} \text { age }+u_{1}
\]
and we assumed that wage is observed if
\[
\gamma_{0}+\gamma_{1} \text { married }+\gamma_{2} \text { children }+\gamma_{3} \text { educ }+\gamma_{4} \text { age }+u_{2}>0
\]
where \(u_{1}\) and \(u_{2}\) have correlation \(\rho\).
The reported results for the wage equation are interpreted exactly as though we observed wage data for all women in the sample; the coefficients on age and education level represent the estimated marginal effects of the regressors in the underlying regression equation. The results for the two ancillary parameters require some explanation. heckman does not directly estimate \(\rho\); to constrain \(\rho\) within its valid limits, and for numerical stability during optimization, it estimates the inverse hyperbolic tangent of \(\rho\) :
\[
\operatorname{atanh} \rho=\frac{1}{2} \ln \left(\frac{1+\rho}{1-\rho}\right)
\]

This estimate is reported as /athrho. In the bottom panel of the output, heckman undoes this transformation for you: the estimated value of \(\rho\) is 0.7035061 . The standard error for \(\rho\) is computed using the delta method, and its confidence intervals are the transformed intervals of /athrho.

Similarly, \(\sigma\), the standard error of the residual in the wage equation, is not directly estimated; for numerical stability, heckman instead estimates \(\ln \sigma\). The untransformed sigma is reported at the end of the output: 6.004797.

Finally, some researchers-especially economists—are used to the selectivity effect summarized not by \(\rho\) but by \(\lambda=\rho \sigma\). heckman reports this, too, along with an estimate of the standard error and confidence interval.

\section*{- Technical note}

If each of the equations in the model had contained many regressors, the heckman command could have become long. An alternate way of specifying our wage model would be to use Stata's global macros. The following lines are an equivalent way of specifying our model:
```

. global wageeq "wage educ age"
. global seleq "married children educ age"
. heckman $wageeq, select($seleq)
(output omitted)

```

\section*{- Technical note}

The reported model \(\chi^{2}\) test is a Wald test that all coefficients in the regression model (except the constant) are 0 . heckman is an estimation command, so you can use test, testnl, or lrtest to perform tests against whatever nested alternate model you choose; see \([R]\) test, \([R]\) testnl, and [R] Irtest.

The estimation of \(\rho\) and \(\sigma\) in the forms atanh \(\rho\) and \(\ln \sigma\) extends the range of these parameters to infinity in both directions, thus avoiding boundary problems during the maximization. Tests of \(\rho\) must be made in the transformed units. However, because \(\operatorname{atanh}(0)=0\), the reported test for atanh \(\rho=0\) is equivalent to the test for \(\rho=0\).

The likelihood-ratio test reported at the bottom of the output is an equivalent test for \(\rho=0\) and is computationally the comparison of the joint likelihood of an independent probit model for the selection equation and a regression model on the observed wage data against the Heckman model likelihood. Because \(\chi^{2}=61.20\), this clearly justifies the Heckman selection equation with these data.

\section*{> Example 2}

This command supports the Huber/White/sandwich estimator of variance under the vce (robust) and vce(cluster clustvar) options or when pweights are used for population-weighted data; see [U] 20.22 Obtaining robust variance estimates. We can obtain robust standard errors for our wage model by specifying clustering on county of residence (the county variable).


Wald test of indep. eqns. (rho \(=0\) ) : chi2 \((1)=38.81 \quad\) Prob \(>\) chi2 \(=0.0000\)
The robust standard errors tend to be a bit larger, but we notice no systematic differences. This finding is not surprising because the data were not constructed to have any county-specific correlations or any other characteristics that would deviate from the assumptions of the Heckman model.

\section*{> Example 3}

Stata also produces Heckman's (1979) two-step efficient estimator of the model with the twostep option. Maximum likelihood estimation of the parameters can be time consuming with large datasets, and the two-step estimates may provide a good alternative in such cases. Continuing with the women's wage model, we can obtain the two-step estimates with Heckman's consistent covariance estimates by typing


\section*{Technical note}

The Heckman selection model depends strongly on the model being correct, much more so than ordinary regression. Running a separate probit or logit for sample inclusion followed by a regression, referred to in the literature as the two-part model (Manning, Duan, and Rogers 1987) — not to be confused with Heckman's two-step procedure-is an especially attractive alternative if the regression part of the model arose because of taking a logarithm of zero values. When the goal is to analyze an underlying regression model or to predict the value of the dependent variable that would be observed in the absence of selection, however, the Heckman model is more appropriate. When the goal is to predict an actual response, the two-part model is usually the better choice.

The Heckman selection model can be unstable when the model is not properly specified or if a specific dataset simply does not support the model's assumptions. For example, let's examine the solution to another simulated problem.
\begin{tabular}{|c|c|c|c|}
\hline \multicolumn{4}{|l|}{. heckman yt x1 x2 x3, select(z1 z2) nonrtol} \\
\hline \multicolumn{4}{|l|}{Iteration 0: Log likelihood = -111.94996} \\
\hline Iteration 1: Log likelihood \(=-110.82258\) & & & \\
\hline Iteration 2: Log likelihood = -110.17707 & & & \\
\hline Iteration 3: Log likelihood \(=-107.70663\) & (not concave) & & \\
\hline Iteration 4: Log likelihood \(=-107.07729\) (output omitted) & (not concave) & & \\
\hline Iteration 36: Log likelihood = -104.0825 & & & \\
\hline Heckman selection model & Number of obs & & 150 \\
\hline (regression model with sample selection) & Selected & & 63 \\
\hline & Nonselected & & 87 \\
\hline & Wald chi2(3) & & \(8.84 \mathrm{e}+08\) \\
\hline Log likelihood = -104.0825 & Prob > chi2 & & 0.0000 \\
\hline
\end{tabular}
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline yt & Coefficient & Std. err & z & \(\mathrm{P}>|\mathrm{z}|\) & \multicolumn{2}{|l|}{[95\% conf. interval]} \\
\hline \multicolumn{7}{|l|}{yt} \\
\hline x1 & . 8974192 & . 0002164 & 4146.52 & 0.000 & . 896995 & . 8978434 \\
\hline x2 & -2.525303 & . 0001244 & \(-2.0 e+04\) & 0.000 & -2.525546 & -2.525059 \\
\hline x3 & 2.855786 & . 0002695 & \(1.1 \mathrm{e}+04\) & 0.000 & 2.855258 & 2.856314 \\
\hline _cons & . 6975003 & . 0907873 & 7.68 & 0.000 & . 5195604 & . 8754402 \\
\hline \multicolumn{7}{|l|}{select} \\
\hline z1 & -. 6826482 & . 0889871 & -7.67 & 0.000 & -. 8570598 & -. 5082367 \\
\hline z2 & 1.003678 & . 1308344 & 7.67 & 0.000 & . 7472471 & 1.260108 \\
\hline _cons & -. 3605665 & . 1219011 & -2.96 & 0.003 & -. 5994883 & -. 1216447 \\
\hline /athrho & 16.11489 & 260.7581 & 0.06 & 0.951 & -494.9617 & 527.1914 \\
\hline /lnsigma & -. 5396877 & . 1303548 & -4.14 & 0.000 & -. 7951785 & -. 284197 \\
\hline rho & 1 & \(1.05 \mathrm{e}-11\) & & & -1 & 1 \\
\hline sigma & . 5829302 & . 0759878 & & & . 4515006 & . 7526184 \\
\hline lambda & . 5829302 & . 0759878 & & & . 433997 & . 7318635 \\
\hline
\end{tabular}

LR test of indep. eqns. (rho \(=0\) ) : chi2 (1) \(=25.67 \quad\) Prob \(>\) chi2 \(=0.0000\)
The model has converged to a value of \(\rho\) that is 1.0 - within machine-rounding tolerances. Given the form of the likelihood for the Heckman selection model, this implies a division by zero, and it is surprising that the model solution turns out as well as it does. Reparameterizing \(\rho\) has allowed the estimation to converge, but we clearly have problems with the estimates. Moreover, if this had occurred in a large dataset, waiting for convergence might take considerable time.

This dataset was not intentionally developed to cause problems. It is actually generated by a "Heckman process" and when generated starting from different random values can be easily estimated. The luck of the draw here merely led to data that, despite the source, did not support the assumptions of the Heckman model.

The two-step model is generally more stable when the data are problematic. It even tolerates estimates of \(\rho\) less than -1 and greater than 1 . For these reasons, the two-step model may be preferred when exploring a large dataset. Still, if the maximum likelihood estimates cannot converge, or converge to a value of \(\rho\) that is at the boundary of acceptable values, there is scant support for fitting a Heckman selection model on the data. Heckman (1979) discusses the implications of \(\rho\) being exactly 1 or 0 , together with the implications of other possible covariance relationships among the model's determinants.

\section*{Stored results}
heckman (maximum likelihood) stores the following in e() :
Scalars
e(N)
e(N_selected)
e(N_nonselected)
e(k)
\(e\left(k \_e q\right)\)
e(k_eq_model)
e(k_aux)
e(k_dv)
\(e\left(d f \_m\right)\)
e(11)
e(11_0)
e(N_clust)
e(lambda)
e(selambda)
e(sigma)
e(chi2)
e(chi2_c)
\(e(p)\)
e(p_c)
e(rho)
\(e\) (rank)
e(rankO)
e(ic)
e(rc)
\(e\) (converged)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(wtype)
e(wexp)
e(title)
e(title2)
e(clustvar)
e(offset1)
e(offset2)
e(mills)
e(chi2type)
e(chi2_ct)
e(vce)
e(vcetype)
e(opt)
e(which)
\(e\) (method)
e(ml_method)
e(user)
e(technique)
e(properties)
e(predict)
e(marginsok)
e(marginsnotok)
e(asbalanced)
e(asobserved)
number of observations
number of selected observations
number of nonselected observations
number of parameters
number of equations in \(e(b)\)
number of equations in overall model test
number of auxiliary parameters
number of dependent variables
model degrees of freedom
log likelihood
log likelihood, constant-only model
number of clusters
\(\lambda\)
standard error of \(\lambda\)
sigma
\(\chi^{2}\)
\(\chi^{2}\) for comparison test
\(p\)-value for model test
\(p\)-value for comparison test
\(\rho\)
rank of e(V)
rank of e(V) for constant-only model
number of iterations
return code
1 if converged, 0 otherwise
heckman
command as typed
names of dependent variables
weight type
weight expression
title in estimation output
secondary title in estimation output
name of cluster variable
offset for regression equation
offset for selection equation
variable containing nonselection hazard (inverse of Mills's ratio)
Wald or LR; type of model \(\chi^{2}\) test
Wald or LR; type of model \(\chi^{2}\) test corresponding to e(chi2_c)
vcetype specified in vce()
title used to label Std. err.
type of optimization
\(\max\) or min; whether optimizer is to perform maximization or minimization
ml
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement predict
predictions allowed by margins
predictions disallowed by margins
factor variables fvset as asbalanced
factor variables fvset as asobserved
```

Matrices
e(b) coefficient vector
e(Cns) constraints matrix
e(ilog) iteration log (up to 20 iterations)
e(gradient) gradient vector
e(V) variance-covariance matrix of the estimators
e(V_modelbased) model-based variance
Functions
e(sample) marks estimation sample

```

In addition to the above, the following is stored in r() :

Matrices
\[
r(\text { table })
\]
matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r\) () are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.
heckman (two-step) stores the following in e():
Scalars
e(N)
e(N_selected)
e(N_nonselected)
e(df_m)
e(lambda)
e(selambda)
e(sigma)
e(chi2)
e (p)
e(rho)
\(e\) (rank)
e(fconverged)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(title)
e(title2)
e(mills)
e(chi2type)
e(vce)
\(e\) (rhometh)
\(e\) (method)
e(properties)
e(predict)
e(marginsok)
e(marginsnotok)
e(asbalanced)
e(asobserved)
Matrices
e(b)
e(V)
Functions
e(sample) marks estimation sample

In addition to the above, the following is stored in r() :
Matrices
\(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r -class command is run after the estimation command.

James Joseph Heckman (1944- ) was born in Chicago in 1944 and studied mathematics at Colorado College and economics at Princeton. He has taught economics at Columbia and (since 1973) at the University of Chicago. He has worked on developing a scientific basis for economic policy evaluation, with emphasis on models of individuals or disaggregated groups and the problems and possibilities created by heterogeneity, diversity, and unobserved counterfactual states. In 2000, he shared the Nobel Prize in Economics with Daniel L. McFadden.

\section*{Methods and formulas}

Cameron and Trivedi \((2022,974-981)\) and Greene (2018, 950-957) provide good introductions to the Heckman selection model. Adkins and Hill (2011, sec. 16.8) describe the two-step estimator with an application using Stata. Jones (2007, 35-40) illustrates Heckman estimation with an application to health economics.

Regression estimates using the nonselection hazard (Heckman 1979) provide starting values for maximum likelihood estimation.

The regression equation is
\[
y_{j}=\mathbf{x}_{j} \boldsymbol{\beta}+u_{1 j}
\]

The selection equation is
\[
\mathbf{z}_{j} \gamma+u_{2 j}>0
\]
where
\[
\begin{gathered}
u_{1} \sim N(0, \sigma) \\
u_{2} \sim N(0,1) \\
\operatorname{corr}\left(u_{1}, u_{2}\right)=\rho
\end{gathered}
\]

The \(\log\) likelihood for observation \(j, \ln L_{j}=l_{j}\), is
\(l_{j}= \begin{cases}w_{j} \ln \Phi\left\{\frac{\mathbf{z}_{j} \gamma+\left(y_{j}-\mathbf{x}_{j} \boldsymbol{\beta}\right) \rho / \sigma}{\sqrt{1-\rho^{2}}}\right\}-\frac{w_{j}}{2}\left(\frac{y_{j}-\mathbf{x}_{j} \boldsymbol{\beta}}{\sigma}\right)^{2}-w_{j} \ln (\sqrt{2 \pi} \sigma) & y_{j} \text { observed } \\ w_{j} \ln \Phi\left(-\mathbf{z}_{j} \gamma\right) & y_{j} \text { not observed }\end{cases}\)
where \(\Phi(\cdot)\) is the standard cumulative normal and \(w_{j}\) is an optional weight for observation \(j\).
In the maximum likelihood estimation, \(\sigma\) and \(\rho\) are not directly estimated. Directly estimated are \(\ln \sigma\) and \(\operatorname{atanh} \rho\) :
\[
\operatorname{atanh} \rho=\frac{1}{2} \ln \left(\frac{1+\rho}{1-\rho}\right)
\]

The standard error of \(\lambda=\rho \sigma\) is approximated through the propagation of error (delta) method; that is,
\[
\operatorname{Var}(\lambda) \approx \mathbf{D} \operatorname{Var}\{(\operatorname{atanh} \rho \ln \sigma)\} \mathbf{D}^{\prime}
\]
where \(\mathbf{D}\) is the Jacobian of \(\lambda\) with respect to atanh \(\rho\) and \(\ln \sigma\).
With maximum likelihood estimation, this command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce(robust) and vce (cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.

The maximum likelihood version of heckman also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

The two-step estimates are computed using Heckman's (1979) procedure.
Probit estimates of the selection equation
\[
\operatorname{Pr}\left(y_{j} \text { observed } \mid \mathbf{z}_{j}\right)=\Phi\left(\mathbf{z}_{j} \boldsymbol{\gamma}\right)
\]
are obtained. From these estimates, the nonselection hazard—what Heckman (1979) referred to as the inverse of the Mills ratio, \(m_{j}\)-for each observation \(j\) is computed as
\[
m_{j}=\frac{\phi\left(\mathbf{z}_{j} \widehat{\gamma}\right)}{\Phi\left(\mathbf{z}_{j} \widehat{\gamma}\right)}
\]
where \(\phi\) is the normal density. We also define
\[
\delta_{j}=m_{j}\left(m_{j}+\widehat{\gamma} \mathbf{z}_{j}\right)
\]

Following Heckman, the two-step parameter estimates of \(\boldsymbol{\beta}\) are obtained by augmenting the regression equation with the nonselection hazard \(\mathbf{m}\). Thus, the regressors become \([\mathbf{X} \mathbf{m}]\), and we obtain the additional parameter estimate \(\boldsymbol{\beta}_{m}\) on the variable containing the nonselection hazard.

A consistent estimate of the regression disturbance variance is obtained using the residuals from the augmented regression and the parameter estimate on the nonselection hazard,
\[
\widehat{\sigma}^{2}=\frac{\mathbf{e}^{\prime} \mathbf{e}+\boldsymbol{\beta}_{m}^{2} \sum_{j=1}^{N} \delta_{j}}{N}
\]

The two-step estimate of \(\rho\) is then
\[
\widehat{\rho}=\frac{\boldsymbol{\beta}_{m}}{\widehat{\sigma}}
\]

Heckman derived consistent estimates of the coefficient covariance matrix on the basis of the augmented regression.

Let \(\mathbf{W}=[\mathbf{X} \mathbf{m}]\) and \(\mathbf{R}\) be a square, diagonal matrix of dimension \(N\), with \(\left(1-\widehat{\rho}^{2} \delta_{j}\right)\) as the diagonal elements. The conventional VCE is
\[
\mathbf{V}_{\text {twostep }}=\widehat{\sigma}^{2}\left(\mathbf{W}^{\prime} \mathbf{W}\right)^{-1}\left(\mathbf{W}^{\prime} \mathbf{R W}+\mathbf{Q}\right)\left(\mathbf{W}^{\prime} \mathbf{W}\right)^{-1}
\]
where
\[
\mathbf{Q}=\widehat{\rho}^{2}\left(\mathbf{W}^{\prime} \mathbf{D} \mathbf{Z}\right) \mathbf{V}_{\mathbf{p}}\left(\mathbf{Z}^{\prime} \mathbf{D} \mathbf{W}\right)
\]
where \(\mathbf{D}\) is the square, diagonal matrix of dimension \(N\) with \(\delta_{j}\) as the diagonal elements; \(\mathbf{Z}\) is the data matrix of selection equation covariates; and \(\mathbf{V}_{p}\) is the variance-covariance estimate from the probit estimation of the selection equation.

\section*{References}

Adkins, L. C., and R. C. Hill. 2011. Using Stata for Principles of Econometrics. 4th ed. Hoboken, NJ: Wiley.
Baum, C. F. 2006. An Introduction to Modern Econometrics Using Stata. College Station, TX: Stata Press.
Cameron, A. C., and P. K. Trivedi. 2022. Microeconometrics Using Stata. 2nd ed. College Station, TX: Stata Press.
Chiburis, R., and M. Lokshin. 2007. Maximum likelihood and two-step estimation of an ordered-probit selection model. Stata Journal 7: 167-182.
Cook, J. A., J.-S. Lee, and N. Newberger. 2021. On identification and estimation of Heckman models. Stata Journal 21: 972-998.

D'Haultfœuille, X., A. Maurel, X. Qiu, and Y. Zhang. 2020. Estimating selection models without an instrument with Stata. Stata Journal 20: 297-308.
Greene, W. H. 2018. Econometric Analysis. 8th ed. New York: Pearson.
Gronau, R. 1974. Wage comparisons-A selectivity bias. Journal of Political Economy 82: 1119-1143. https://doi.org/10.1086/260267.
Heckman, J. J. 1976. The common structure of statistical models of truncation, sample selection and limited dependent variables and a simple estimator for such models. Annals of Economic and Social Measurement 5: 475-492.
——. 1979. Sample selection bias as a specification error. Econometrica 47: 153-161. https://doi.org/10.2307/1912352.
Jones, A. M. 2007. Applied Econometrics for Health Economists: A Practical Guide. 2nd ed. Abingdon, UK: Radcliffe. Lewis, H. G. 1974. Comments on selectivity biases in wage comparisons. Journal of Political Economy 82: 1145-1155.

Manning, W. G., N. Duan, and W. H. Rogers. 1987. Monte Carlo evidence on the choice between sample selection and two-part models. Journal of Econometrics 35: 59-82. https://doi.org/10.1016/0304-4076(87)90081-9.
Tauchmann, H. 2014. Lee (2009) treatment-effect bounds for nonrandom sample selection. Stata Journal 14: 884-894.

\footnotetext{
Also see
[R] heckman postestimation - Postestimation tools for heckman
[R] heckoprobit - Ordered probit model with sample selection
[R] heckpoisson - Poisson regression with sample selection
[R] heckprobit - Probit model with sample selection
\([R]\) regress - Linear regression
[R] tobit - Tobit regression
[BAYES] bayes: heckman - Bayesian Heckman selection model
[CAUSAL] etregress - Linear regression with endogenous treatment effects
[ERM] eregress - Extended linear regression
[SVY] svy estimation - Estimation commands for survey data
[XT] xtheckman - Random-effects regression with sample selection
[U] 20 Estimation and postestimation commands
}

\section*{Title}
heckman postestimation - Postestimation tools for heckman
\begin{tabular}{ll} 
Postestimation commands & predict margins Remarks and examples \\
Reference & Also see
\end{tabular}

\section*{Postestimation commands}

The following postestimation commands are available after heckman:
\begin{tabular}{|c|c|}
\hline Command & Description \\
\hline contrast & contrasts and ANOVA-style joint tests of estimates \\
\hline *estat ic & Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) \\
\hline estat summarize & summary statistics for the estimation sample \\
\hline estat vce & variance-covariance matrix of the estimators (VCE) \\
\hline estat (svy) & postestimation statistics for survey data \\
\hline estimates & cataloging estimation results \\
\hline etable & table of estimation results \\
\hline \(\dagger\) hausman & Hausman's specification test \\
\hline lincom & point estimates, standard errors, testing, and inference for linear combinations of coefficients \\
\hline \(\dagger\) lrtest & likelihood-ratio test; not available with two-step estimator \\
\hline margins & marginal means, predictive margins, marginal effects, and average marginal effects \\
\hline marginsplot & graph the results from margins (profile plots, interaction plots, etc.) \\
\hline nlcom & point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients \\
\hline predict & linear predictions and their SEs, probabilities, etc. \\
\hline predictnl & point estimates, standard errors, testing, and inference for generalized predictions \\
\hline pwcompare & pairwise comparisons of estimates \\
\hline * suest & seemingly unrelated estimation \\
\hline test & Wald tests of simple and composite linear hypotheses \\
\hline testnl & Wald tests of nonlinear hypotheses \\
\hline
\end{tabular}

\footnotetext{
*estat ic and suest are not appropriate after heckman, twostep.
\(\dagger_{\text {hausman }}\) and lrtest are not appropriate with svy estimation results.
}

\section*{predict}

\section*{Description for predict}
predict creates a new variable containing predictions such as linear predictions, standard errors, probabilities, expected values, and nonselection hazards.

\section*{Menu for predict}

Statistics \(>\) Postestimation

\section*{Syntax for predict}

After ML or twostep
predict [type] newvar [if] [in] [, statistic nooffset]
After ML
predict [type] stub* [if] [in], scores
statistic Description

\section*{Main}
xb
stdp
stdf
xbsel
stdpsel
\(\operatorname{pr}(a, b)\)
\(\mathrm{e}(a, b)\)
ystar ( \(a, b\) )
ycond
yexpected
nshazard or mills psel
linear prediction; the default
standard error of the prediction
standard error of the forecast
linear prediction for selection equation
standard error of the linear prediction for selection equation
\(\operatorname{Pr}\left(y_{j} \mid a<y_{j}<b\right)\)
\(E\left(y_{j} \mid a<y_{j}<b\right)\)
\(E\left(y_{j}^{*}\right), y_{j}^{*}=\max \left\{a, \min \left(y_{j}, b\right)\right\}\)
\(E\left(y_{j} \mid y_{j}\right.\) observed \()\)
\(E\left(y_{j}^{*}\right), y_{j}\) taken to be 0 where unobserved
nonselection hazard (also called the inverse of Mills's ratio)
\(\operatorname{Pr}\left(y_{j}\right.\) observed \()\)

These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.
stdf is not allowed with svy estimation results.
where \(a\) and \(b\) may be numbers or variables; \(a\) missing ( \(a \geq\).) means \(-\infty\), and \(b\) missing ( \(b \geq\).) means \(+\infty\); see [U] 12.2.1 Missing values.

\section*{Options for predict}
\(\qquad\) Main
xb , the default, calculates the linear prediction \(\mathbf{x}_{j} \mathbf{b}\).
stdp calculates the standard error of the prediction, which can be thought of as the standard error of the predicted expected value or mean for the observation's covariate pattern. The standard error of the prediction is also referred to as the standard error of the fitted value.
stdf calculates the standard error of the forecast, which is the standard error of the point prediction for 1 observation. It is commonly referred to as the standard error of the future or forecast value. By construction, the standard errors produced by stdf are always larger than those produced by stdp; see Methods and formulas in [R] regress postestimation.
xbsel calculates the linear prediction for the selection equation.
stdpsel calculates the standard error of the linear prediction for the selection equation.
\(\operatorname{pr}(a, b)\) calculates \(\operatorname{Pr}\left(a<\mathbf{x}_{j} \mathbf{b}+u_{1}<b\right)\), the probability that \(y_{j} \mid \mathbf{x}_{j}\) would be observed in the interval \((a, b)\).
\(a\) and \(b\) may be specified as numbers or variable names; \(l b\) and \(u b\) are variable names;
\(\operatorname{pr}(20,30)\) calculates \(\operatorname{Pr}\left(20<\mathbf{x}_{j} \mathbf{b}+u_{1}<30\right) ; \operatorname{pr}(l b, u b)\) calculates \(\operatorname{Pr}\left(l b<\mathbf{x}_{j} \mathbf{b}+u_{1}<u b\right)\); and \(\operatorname{pr}(20, u b)\) calculates \(\operatorname{Pr}\left(20<\mathbf{x}_{j} \mathbf{b}+u_{1}<u b\right)\).
\(a\) missing \((a \geq\).\() means -\infty\); \(\operatorname{pr}(., 30)\) calculates \(\operatorname{Pr}\left(-\infty<\mathbf{x}_{j} \mathbf{b}+u_{j}<30\right)\);
\(\operatorname{pr}(l b, 30)\) calculates \(\operatorname{Pr}\left(-\infty<\mathbf{x}_{j} \mathbf{b}+u_{j}<30\right)\) in observations for which \(l b \geq\).
and calculates \(\operatorname{Pr}\left(l b<\mathbf{x}_{j} \mathbf{b}+u_{j}<30\right)\) elsewhere.
\(b\) missing ( \(b \geq\).) means \(+\infty\); \(\operatorname{pr}(20,\).\() calculates \operatorname{Pr}\left(+\infty>\mathbf{x}_{j} \mathbf{b}+u_{j}>20\right)\);
\(\operatorname{pr}(20, u b)\) calculates \(\operatorname{Pr}\left(+\infty>\mathbf{x}_{j} \mathbf{b}+u_{j}>20\right)\) in observations for which \(u b \geq\).
and calculates \(\operatorname{Pr}\left(20<\mathbf{x}_{j} \mathbf{b}+u_{j}<u b\right)\) elsewhere.
\(\mathrm{e}(a, b)\) calculates \(E\left(\mathbf{x}_{j} \mathbf{b}+u_{1} \mid a<\mathbf{x}_{j} \mathbf{b}+u_{1}<b\right)\), the expected value of \(y_{j} \mid \mathbf{x}_{j}\) conditional on \(y_{j} \mid \mathbf{x}_{j}\) being in the interval \((a, b)\), meaning that \(y_{j} \mid \mathbf{x}_{j}\) is truncated.
\(a\) and \(b\) are specified as they are for \(\operatorname{pr}()\).
ystar \((a, b)\) calculates \(E\left(y_{j}^{*}\right)\), where \(y_{j}^{*}=a\) if \(\mathbf{x}_{j} \mathbf{b}+u_{j} \leq a, y_{j}^{*}=b\) if \(\mathbf{x}_{j} \mathbf{b}+u_{j} \geq b\), and \(y_{j}^{*}=\mathbf{x}_{j} \mathbf{b}+u_{j}\) otherwise, meaning that \(y_{j}^{*}\) is censored. \(a\) and \(b\) are specified as they are for pr() .
ycond calculates the expected value of the dependent variable conditional on the dependent variable being observed, that is, selected; \(E\left(y_{j} \mid y_{j}\right.\) observed).
yexpected calculates the expected value of the dependent variable \(\left(y_{j}^{*}\right)\), where that value is taken to be 0 when it is expected to be unobserved; \(y_{j}^{*}=\operatorname{Pr}\left(y_{j}\right.\) observed \() E\left(y_{j} \mid y_{j}\right.\) observed \()\).
The assumption of 0 is valid for many cases where nonselection implies nonparticipation (for example, unobserved wage levels, insurance claims from those who are uninsured) but may be inappropriate for some problems (for example, unobserved disease incidence).
nshazard and mills are synonyms; both calculate the nonselection hazard—what Heckman (1979) referred to as the inverse of the Mills ratio-from the selection equation.
psel calculates the probability of selection (or being observed):
\(\operatorname{Pr}\left(y_{j}\right.\) observed \()=\operatorname{Pr}\left(\mathbf{z}_{j} \gamma+u_{2 j}>0\right)\).
scores, not available with twostep, calculates equation-level score variables.
The first new variable will contain \(\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)\).
The second new variable will contain \(\partial \ln L / \partial\left(\mathbf{z}_{j} \gamma\right)\).
The third new variable will contain \(\partial \ln L / \partial(\operatorname{atanh} \rho)\).
The fourth new variable will contain \(\partial \ln L / \partial(\ln \sigma)\).
nooffset is relevant when you specify offset (varname) for heckman. It modifies the calculations made by predict so that they ignore the offset variable; the linear prediction is treated as \(\mathbf{x}_{j} \mathbf{b}\) rather than as \(\mathbf{x}_{j} \mathbf{b}+\) offset \(_{j}\).

\section*{margins}

\section*{Description for margins}
margins estimates margins of response for linear predictions, probabilities, expected values, and nonselection hazards.

\section*{Menu for margins}

Statistics \(>\) Postestimation

\section*{Syntax for margins}
```

margins [marginlist] [, options]

```

statistic
xb
xbsel
\(\operatorname{pr}(a, b)\)
e \((a, b)\)
ystar ( \(a, b\) )
* ycond
* yexpected
nshazard or mills
psel
stdp
stdf
stdpsel

Description
linear prediction; the default
linear prediction for selection equation
\(\operatorname{Pr}\left(y_{j} \mid a<y_{j}<b\right)\)
\(E\left(y_{j} \mid a<y_{j}<b\right)\)
\(E\left(y_{j}^{*}\right), y_{j}^{*}=\max \left\{a, \min \left(y_{j}, b\right)\right\}\)
\(E\left(y_{j} \mid y_{j}\right.\) observed \()\)
\(E\left(y_{j}^{*}\right), y_{j}\) taken to be 0 where unobserved
nonselection hazard (also called the inverse of Mills's ratio)
\(\operatorname{Pr}\left(y_{j}\right.\) observed \()\)
not allowed with margins
not allowed with margins
not allowed with margins

\footnotetext{
*ycond and yexpected are not allowed with margins after heckman, twostep.
}

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

\section*{Remarks and examples}

\section*{> Example 1}

The default statistic produced by predict after heckman is the expected value of the dependent variable from the underlying distribution of the regression model. In the wage model of \([R]\) heckman, this is the expected wage rate among all women, regardless of whether they were observed to participate in the labor force:
```

. use https://www.stata-press.com/data/r18/womenwk
. heckman wage educ age, select(married children educ age) vce(cluster county)
(output omitted)
. predict heckwage
(option xb assumed; fitted values)

```

It is instructive to compare these predicted wage values from the Heckman model with an ordinary regression model - a model without the selection adjustment:
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline Source & SS & df & MS & Number of obs & = & 1,343 \\
\hline & & & & \(\mathrm{F}(2,1340)\) & = & 227.49 \\
\hline Model & 13524.0337 & 2 & 6762.01687 & Prob > F & = & 0.0000 \\
\hline Residual & 39830.8609 & 1,340 & 29.7245231 & R -squared & = & 0.2535 \\
\hline & & & & Adj R-squared & = & 0.2524 \\
\hline Total & 53354.8946 & 1,342 & 39.7577456 & Root MSE & = & 5.452 \\
\hline
\end{tabular}
\begin{tabular}{r|rcccrr}
\hline wage & Coefficient & Std. err. & \(t\) & P>|t| & [95\% conf. interval] \\
\hline education & .8965829 & .0498061 & 18.00 & 0.000 & .7988765 & .9942893 \\
age & .1465739 & .0187135 & 7.83 & 0.000 & .109863 & .1832848 \\
_cons & 6.084875 & .8896182 & 6.84 & 0.000 & 4.339679 & 7.830071 \\
\hline
\end{tabular}
. predict regwage
(option xb assumed; fitted values)
• summarize heckwage regwage
Variable

Because this dataset was concocted, we know the true coefficients of the wage regression equation to be \(1,0.2\), and 1 , respectively. We can compute the true mean wage for our sample.
. generate truewage \(=1+.2 *\) age \(+1 *\) educ
. summarize truewage
Variable

Whereas the mean of the predictions from heckman is within 18 cents of the true mean wage, ordinary regression yields predictions that are on average about \(\$ 1.80\) per hour too high because of the selection effect. The regression predictions also show somewhat less variation than the true wages.

The coefficients from heckman are so close to the true values that they are not worth testing. Conversely, the regression equation is significantly off but seems to give the right sense. Would we be led far astray if we relied on the OLS coefficients? The effect of age is off by more than 5 cents per year of age, and the coefficient on education level is off by about \(10 \%\). We can test the OLS coefficient on education level against the true value by using test.
```

. test educ = 1
( 1) education = 1
F( 1, 1340) = 4.31
Prob > F = 0.0380

```

The OLS coefficient on education is substantially lower than the true parameter; moreover, the difference from the true parameter is also statistically significant beyond the \(5 \%\) level. We can perform a similar test for the OLS age coefficient:
```

. test age = . 2
(1) age = . 2
F( 1, 1340) = 8.15
Prob > F = 0.0044

```

We find even stronger evidence that the OLS regression results are biased away from the true parameters.

\section*{> Example 2}

Several other interesting aspects of the Heckman model can be explored with predict. Continuing with our wage model, we can obtain the expected wages for women conditional on participating in the labor force with the ycond option. Let's get these predictions and compare them with actual wages for women participating in the labor force.
```

. use https://www.stata-press.com/data/r18/womenwk, clear
. heckman wage educ age, select(married children educ age)
(output omitted)
predict hondwage, ycond
. summarize wage hcndwage if wage != .

```
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline wage & 1,343 & 23.69217 & 6.305374 & 5.88497 & 45.80979 \\
hcndwage & 1,343 & 23.68239 & 3.335087 & 16.18337 & 33.7567
\end{tabular}

We see that the average predictions from heckman are close to the observed levels but do not have the same mean. These conditional wage predictions are available for all observations in the dataset but can be directly compared only with observed wages, where individuals are participating in the labor force.

What if we were interested in making predictions about mean wages for all women? Here the expected wage is 0 for those who are not expected to participate in the labor force, with expected participation determined by the selection equation. These values can be obtained with the yexpected option of predict. For comparison, a variable can be generated where the wage is set to 0 for nonparticipants.
```

. predict hexpwage, yexpected
. generate wage0 = wage
(657 missing values generated)
. replace wageO = O if wage == .
(657 real changes made)

```
\begin{tabular}{l|rrrrr}
. summarize hexpwage wage0 \\
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline hexpwage & 2,000 & 15.92511 & 5.979336 & 2.492469 & 32.45858 \\
wage0 & 2,000 & 15.90929 & 12.27081 & 0 & 45.80979
\end{tabular}

Again we note that the predictions from heckman are close to the observed mean hourly wage rate for all women. Why aren't the predictions using ycond and yexpected equal to their observed sample equivalents? For the Heckman model, unlike linear regression, the sample moments implied by the optimal solution to the model likelihood do not require that these predictions match observed data. Properly accounting for the additional variation from the selection equation requires that the model use more information than just the sample moments of the observed wages.

\section*{Reference}

Heckman, J. J. 1979. Sample selection bias as a specification error. Econometrica 47: 153-161. https://doi.org/10.2307/1912352.

\section*{Also see}
[R] heckman - Heckman selection model
[U] 20 Estimation and postestimation commands

\section*{Title}
heckoprobit - Ordered probit model with sample selection
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
heckoprobit fits maximum-likelihood ordered probit models with sample selection.

\section*{Quick start}

Ordered probit model of y on x with selection indicated by binary variable selected and predicted by v
heckoprobit \(y\) x, select (selected = v x)
Add indicator variables for each level of categorical variable a heckoprobit y x i.a, select(selected = v x i.a)

Account for complex sampling design using svyset data svy: heckoprobit y x, select(selected = v x)

\section*{Menu}

Statistics \(>\) Sample-selection models \(>\) Ordered probit model with selection

\section*{Syntax}
```

heckoprobit depvar indepvars [if] [in] [weight],
select( $\left[\right.$ depvar $\left._{s}=\right]$ varlist $_{s}\left[\right.$, noconstant offset $\left(\right.$ varname $\left.\left._{o}\right)\right]$ ) [options]

```
options
Description

Model
* select()
offset (varname)
constraints (constraints)
SE/Robust
vce (vcetype)

Reporting
level(\#)
first
noheader
nofootnote
nocnsreport
display_options
specify selection equation: dependent and independent variables; whether to have constant term and offset variable include varname in model with coefficient constrained to 1 apply specified linear constraints
vcetype may be oim, robust, cluster clustvar, opg, bootstrap, or jackknife
set confidence level; default is level(95)
report first-step probit estimates
do not display header above parameter table
do not display footnotes below parameter table
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
control the maximization process; seldom used
keep collinear variables
display legend instead of statistics
*select() is required.
The full specification is select \(\left(\left[\right.\right.\) depvar \(\left._{s}=\right] \operatorname{varlist}_{s}\left[\right.\), noconstant offset(varname \({ }_{o}\) ) \(]\) ). indepvars and varlist \({ }_{s}\) may contain factor variables; see [U] 11.4.3 Factor variables.
depvar, indepvars, depvar \({ }_{s}\), and varlist \(_{s}\) may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bayes, bootstrap, by, collect, jackknife, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: heckoprobit.
Weights are not allowed with the bootstrap prefix; see \([\mathrm{R}]\) bootstrap.
vce(), first, and weights are not allowed with the svy prefix; see [SVY] svy.
pweights, fweights, and iweights are allowed; see [U] 11.1.6 weight.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options}

required. The selection equation should contain at least one variable that is not in the outcome equation.
If depvar is specified, it should be coded as 0 or 1,0 indicating an observation not selected and \(1^{1}\) indicating a selected observation. If depvar \(r_{s}\) is not specified, observations for which depvar is not missing are assumed selected, and those for which depvar is missing are assumed not selected. noconstant suppresses the selection constant term (intercept).
offset ( varname \(_{o}\) ) specifies that selection offset varname \({ }_{o}\) be included in the model with the coefficient constrained to be 1 .
offset(varname), constraints (constraints); see [R] Estimation options.

SE/Robust
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.

Reporting
level(\#); see [R] Estimation options.
first specifies that the first-step probit estimates of the selection equation be displayed before estimation.
noheader suppresses the header above the parameter table.
nofootnote suppresses the footnotes displayed below the parameter table.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\%fint), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate (\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from(init_specs); see [R] Maximize. These options are seldom used.

Setting the optimization type to technique (bhhh) resets the default vcetype to vce (opg).
The following options are available with heckoprobit but are not shown in the dialog box: collinear, coeflegend; see [R] Estimation options.

\section*{Remarks and examples}
heckoprobit estimates the parameters of a regression model for an ordered categorical outcome from a nonrandom sample known as a selected sample. Selected samples suffer from "selection on unobservables" because the errors that determine whether a case is missing are correlated with the errors that determine the outcome.

For ordered categorical regression from samples that do not suffer from selection on unobservables, see \([R]\) oprobit or \([R]\) ologit. For regression of a continuous outcome variable from a selected sample, see \([R]\) heckman.

Even though we are interested in modeling a single ordinal outcome, there are two dependent variables in the ordered probit sample-selection model because we must also model the sampleselection process. First, there is the ordinal outcome \(y_{j}\). Second, there is a binary variable that indicates whether each case in the sample is observed or unobserved. To handle the sample-selection problem, we model both dependent variables jointly. Both variables are categorical. Their categorical values are determined by the values of linear combinations of covariates and normally distributed error terms relative to certain cutpoints that partition the real line. The error terms used in the determination of selection and the ordinal outcome value may be correlated.

The probability that the ordinal outcome \(y_{j}\) is equal to the value \(v_{h}\) is given by the probability that \(\mathbf{x}_{j} \boldsymbol{\beta}+u_{1 j}\) falls within the cutpoints \(\kappa_{h-1}\) and \(\kappa_{h}\),
\[
\operatorname{Pr}\left(y_{j}=v_{h}\right)=\operatorname{Pr}\left(\kappa_{h-1}<\mathbf{x}_{j} \boldsymbol{\beta}+u_{1 j} \leq \kappa_{h}\right)
\]
where \(\mathbf{x}_{j}\) is the outcome covariates, \(\boldsymbol{\beta}\) is the coefficients, and \(u_{1 j}\) is a random-error term. The observed outcome values \(v_{1}, \ldots, v_{H}\) are integers such that \(v_{i}<v_{m}\) for \(i<m . \kappa_{0}\) is taken as \(-\infty\) and \(\kappa_{H}\) is taken as \(+\infty\).

We model the selection process for the outcome by
\[
s_{j}=\mathbf{1}\left(\mathbf{z}_{j} \gamma+u_{2 j}>0\right)
\]
where \(s_{j}=1\) if we observed \(y_{j}\) and 0 otherwise, \(\mathbf{z}_{j}\) is the covariates used to model the selection process, \(\gamma\) is the coefficients for the selection process, \(\mathbf{1}(\cdot)\) denotes the indicator function, and \(u_{2 j}\) is a random-error term.
\(\left(u_{1 j}, u_{2 j}\right)\) have bivariate normal distribution with mean zero and variance matrix
\[
\left[\begin{array}{ll}
1 & \rho \\
\rho & 1
\end{array}\right]
\]

When \(\rho \neq 0\), standard ordered probit techniques applied to the outcome equation yield inconsistent results. heckoprobit provides consistent, asymptotically efficient estimates for all the parameters in such models.

De Luca and Perotti (2011) describe the maximum likelihood estimator used in heckoprobit.

\section*{> Example 1}

We have a simulated dataset containing a sample of 5,000 women, 3,480 of whom work. The outcome of interest is a woman's job satisfaction, and we suspect that unobservables that determine job satisfaction and the unobservables that increase the likelihood of employment are correlated. Women may make a decision to work based on how satisfying their job would be. We estimate the parameters of an ordered probit sample-selection model for the outcome of job satisfaction (satisfaction) with selection on employment (work). Age (age) and years of education (education) are used as outcome covariates, and we also expect that they affect selection. Additional covariates for selection are marital status (married) and the number of children at home (children).

Here we estimate the parameters of the model with heckoprobit. We use the factorial interaction of married and children in select(). This specifies that the number of children and marital status affect selection, and it allows the effect of the number of children to differ among married and nonmarried women. The factorial interaction is specified using factor-variable notation, which is described in [U] 11.4.3 Factor variables.
```

. use https://www.stata-press.com/data/r18/womensat
(Job satisfaction, female)
. heckoprobit satisfaction education age,
> select(work=education age i.married\#\#c.children)
Fitting oprobit model:
Iteration 0: Log likelihood = -3934.1474
Iteration 1: Log likelihood = -3571.886
Iteration 2: Log likelihood = -3570.2616
Iteration 3: Log likelihood = -3570.2616
Fitting selection model:
Iteration 0: Log likelihood = -3071.0775
Iteration 1: Log likelihood = -2565.5092
Iteration 2: Log likelihood = -2556.8369
Iteration 3: Log likelihood = -2556.8237
Iteration 4: Log likelihood = -2556.8237
Comparison: Log likelihood = -6127.0853
Fitting full model:
Iteration 0: Log likelihood = -6127.0853
Iteration 1: Log likelihood = -6093.8868
Iteration 2: Log likelihood = -6083.215
Iteration 3: Log likelihood = -6083.0376
Iteration 4: Log likelihood = -6083.0372

```
Ordered probit model with sample selection \(\quad\) Number of obs 5,000
    Selected \(=3,480\)
    Nonselected \(=1,520\)
Wald chi2(2) \(\quad=\quad 842.42\)
Log likelihood \(=-6083.037 \quad\) Prob \(>\) chi2 \(=0.0000\)
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline & Coefficient & Std. err. & z & \(P>|z|\) & [95\% conf & interval] \\
\hline satisfaction education age & \[
\begin{array}{r}
.1536381 \\
.0334463
\end{array}
\] & \[
\begin{aligned}
& .0068266 \\
& .0024049
\end{aligned}
\] & \[
\begin{aligned}
& 22.51 \\
& 13.91
\end{aligned}
\] & \[
\begin{aligned}
& 0.000 \\
& 0.000
\end{aligned}
\] & \[
\begin{aligned}
& .1402583 \\
& .0287329
\end{aligned}
\] & \[
\begin{array}{r}
.1670179 \\
.0381598
\end{array}
\] \\
\hline \multicolumn{7}{|l|}{work} \\
\hline education & . 0512494 & . 0068095 & 7.53 & 0.000 & . 037903 & . 0645958 \\
\hline age & . 0288084 & . 0026528 & 10.86 & 0.000 & . 023609 & . 0340078 \\
\hline 1.married & . 6120876 & . 0700055 & 8.74 & 0.000 & . 4748794 & . 7492958 \\
\hline children & . 5140995 & . 0288529 & 17.82 & 0.000 & . 4575489 & . 5706501 \\
\hline \[
\begin{gathered}
\text { married\# } \\
\text { c.children }
\end{gathered}
\] & & & & & & \\
\hline 1 & -. 1337573 & . 035126 & -3.81 & 0.000 & -. 202603 & -. 0649117 \\
\hline _cons & -2.203036 & . 125772 & -17.52 & 0.000 & -2.449545 & -1.956528 \\
\hline /cut1 & 1.728757 & . 1232063 & 14.03 & 0.000 & 1.487277 & 1.970237 \\
\hline /cut2 & 2.64357 & . 116586 & 22.67 & 0.000 & 2.415066 & 2.872075 \\
\hline /cut3 & 3.642911 & . 1178174 & 30.92 & 0.000 & 3.411993 & 3.873829 \\
\hline /athrho & . 7430919 & . 0780998 & 9.51 & 0.000 & . 5900191 & . 8961646 \\
\hline rho & . 6310096 & . 0470026 & & & . 5299093 & . 7144252 \\
\hline
\end{tabular}

The output shows several iteration logs. The first iteration log corresponds to running the ordered probit model for those observations in the sample where we have observed the outcome. The second iteration \(\log\) corresponds to running the selection probit model, which models whether we observe
our outcome of interest. If \(\rho=0\), the sum of the log likelihoods from these two models will equal the \(\log\) likelihood of the ordered probit sample-selection model; this sum is printed in the iteration \(\log\) as the comparison \(\log\) likelihood. The final iteration \(\log\) is for fitting the full ordered probit sample-selection model.

The Wald test in the header is highly significant, indicating a good model fit. All the covariates are statistically significant. The likelihood-ratio test in the footer indicates that we can reject the null hypothesis that the errors for outcome and selection are uncorrelated. This means that we should use the ordered probit sample-selection model instead of the simple ordered probit model.

The positive estimate of 0.63 for \(\rho\) indicates that unobservables that increase job satisfaction tend to occur with unobservables that increase the chance of having a job.

\section*{Stored results}
heckoprobit stores the following in e():
Scalars
\begin{tabular}{|c|c|}
\hline \(e(N)\) & number of observations \\
\hline e(N_selected) & number of selected observations \\
\hline e(N_nonselected) & number of nonselected observations \\
\hline e(N_cd) & number of completely determined observations \\
\hline e(k_cat) & number of categories \\
\hline e(k) & number of parameters \\
\hline e(k_eq) & number of equations in e(b) \\
\hline e(k_eq_model) & number of equations in overall model test \\
\hline \(e\left(k \_a u x\right)\) & number of auxiliary parameters \\
\hline e(k_dv) & number of dependent variables \\
\hline e(df_m) & model degrees of freedom \\
\hline e(ll) & log likelihood \\
\hline e(ll_c) & log likelihood, comparison model \\
\hline e(N_clust) & number of clusters \\
\hline e(chi2) & \(\chi^{2}\) \\
\hline e(chi2_c) & \(\chi^{2}\) for comparison test \\
\hline \(e(p)\) & \(p\)-value for model test \\
\hline \(e\left(p \_c\right)\) & \(p\)-value for comparison test \\
\hline e(rho) & \(\rho\) \\
\hline e(rank) & rank of e(V) \\
\hline e(ic) & number of iterations \\
\hline \(e(r c)\) & return code \\
\hline e(converged) & 1 if converged, 0 otherwise \\
\hline
\end{tabular}

Macros
e(cmd)
e(cmdline)
e(depvar)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(offset1)
e(offset2)
e(chi2type)
e(chi2_ct)
e(vce)
e(vcetype)
e(opt)
e(which)
\(e\left(m l \_m e t h o d\right)\)
e(user)
e(technique)
e(properties)
e(predict)
e(marginsok)
e(marginsnotok)
e(marginsdefault)
e(asbalanced)
e(asobserved)
Matrices
e(b)
e(Cns)
e(ilog)
e(gradient)
e(cat)
e(V)
e(V_modelbased)
Functions
e(sample)
heckoprobit
command as typed
names of dependent variables
weight type
weight expression
title in estimation output
name of cluster variable
offset for regression equation
offset for selection equation
Wald or LR; type of model \(\chi^{2}\) test
type of comparison \(\chi^{2}\) test
vcetype specified in vce()
title used to label Std. err.
type of optimization
\(\max\) or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement predict
predictions allowed by margins
predictions disallowed by margins
default predict() specification for margins
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
constraints matrix
iteration \(\log\) (up to 20 iterations)
gradient vector
category values
variance-covariance matrix of the estimators
model-based variance
marks estimation sample

In addition to the above, the following is stored in r() :

Matrices
r(table)
matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

\section*{Methods and formulas}

De Luca and Perotti (2011) provide an introduction to this model.
The ordinal outcome equation is
\[
y_{j}=\sum_{h=1}^{H} v_{h} 1\left(\kappa_{h-1}<\mathbf{x}_{j} \boldsymbol{\beta}+u_{1 j} \leq \kappa_{h}\right)
\]
where \(\mathbf{x}_{j}\) is the outcome covariates, \(\boldsymbol{\beta}\) is the coefficients, and \(u_{1 j}\) is a random-error term. The observed outcome values \(v_{1}, \ldots, v_{H}\) are integers such that \(v_{i}<v_{m}\) for \(i<m . \kappa_{1}, \ldots, \kappa_{H-1}\) are real numbers such that \(\kappa_{i}<\kappa_{m}\) for \(i<m\). \(\kappa_{0}\) is taken as \(-\infty\) and \(\kappa_{H}\) is taken as \(+\infty\).

The selection equation is
\[
s_{j}=\mathbf{1}\left(\mathbf{z}_{j} \gamma+u_{2 j}>0\right)
\]
where \(s_{j}=1\) if we observed \(y_{j}\) and 0 otherwise, \(\mathbf{z}_{j}\) is the covariates used to model the selection process, \(\gamma\) is the coefficients for the selection process, and \(u_{2 j}\) is a random-error term.
\(\left(u_{1 j}, u_{2 j}\right)\) have bivariate normal distribution with mean zero and variance matrix
\[
\left[\begin{array}{ll}
1 & \rho \\
\rho & 1
\end{array}\right]
\]

Let \(a_{j}=z_{j} \gamma+\operatorname{offset}_{j}^{\gamma}\) and \(b_{j}=x_{j} \boldsymbol{\beta}+\operatorname{offset}_{j}^{\beta}\). This yields the log likelihood
\[
\ln L=\sum_{j \notin S} w_{j} \ln \left\{\Phi\left(-a_{j}\right)\right\}+\sum_{h=1}^{H} \sum_{\substack{j \in S \\ y_{j}=v_{h}}} w_{j} \ln \left\{\Phi_{2}\left(a_{j}, \kappa_{h}-b_{j},-\rho\right)-\Phi_{2}\left(a_{j}, \kappa_{h-1}-b_{j},-\rho\right)\right\}
\]
where \(S\) is the set of observations for which \(y_{j}\) is observed, \(\Phi_{2}(\cdot)\) is the cumulative bivariate normal distribution function (with mean \(\left[\begin{array}{ll}0 & 0\end{array}\right]^{\prime}\) ), \(\Phi(\cdot)\) is the standard cumulative normal, and \(w_{j}\) is an optional weight for observation \(j\).

In the maximum likelihood estimation, \(\rho\) is not directly estimated. Directly estimated is atanh \(\rho\) :
\[
\operatorname{atanh} \rho=\frac{1}{2} \ln \left(\frac{1+\rho}{1-\rho}\right)
\]

From the form of the likelihood, it is clear that if \(\rho=0\), the log likelihood for the ordered probit sample-selection model is equal to the sum of the ordered probit model for the outcome \(y\) and the selection model. We can perform a likelihood-ratio test by comparing the log likelihood of the full model with the sum of the log likelihoods for the ordered probit and selection models.

\section*{References}

Baum, C. F. 2006. An Introduction to Modern Econometrics Using Stata. College Station, TX: Stata Press.
Cameron, A. C., and P. K. Trivedi. 2005. Microeconometrics: Methods and Applications. New York: Cambridge University Press.

Chiburis, R., and M. Lokshin. 2007. Maximum likelihood and two-step estimation of an ordered-probit selection model. Stata Journal 7: 167-182.
De Luca, G., and V. Perotti. 2011. Estimation of ordered response models with sample selection. Stata Journal 11: 213-239.

Heckman, J. J. 1979. Sample selection bias as a specification error. Econometrica 47: 153-161. https://doi.org/10.2307/1912352.
Miranda, A., and S. Rabe-Hesketh. 2006. Maximum likelihood estimation of endogenous switching and sample selection models for binary, ordinal, and count variables. Stata Journal 6: 285-308.

Muro, J., C. Suárez, and M. Zamora. 2010. Computing Murphy-Topel-corrected variances in a heckprobit model with endogeneity. Stata Journal 10: 252-258.
Van de Ven, W. P. M. M., and B. M. S. Van Pragg. 1981. The demand for deductibles in private health insurance: A probit model with sample selection. Journal of Econometrics 17: 229-252. https://doi.org/10.1016/0304-4076(81)90028-2.
Wooldridge, J. M. 2010. Econometric Analysis of Cross Section and Panel Data. 2nd ed. Cambridge, MA: MIT Press.

\section*{Also see}
\([R]\) heckoprobit postestimation - Postestimation tools for heckoprobit
[R] heckman - Heckman selection model
[R] heckpoisson - Poisson regression with sample selection
[R] heckprobit - Probit model with sample selection
[R] oprobit - Ordered probit regression
[R] probit - Probit regression
\([\mathrm{R}]\) regress - Linear regression
[R] tobit - Tobit regression
[BAYES] bayes: heckoprobit - Bayesian ordered probit model with sample selection
[ERM] eoprobit - Extended ordered probit regression
[SVY] svy estimation - Estimation commands for survey data
[U] 20 Estimation and postestimation commands

\section*{Title}
heckoprobit postestimation - Postestimation tools for heckoprobit
\begin{tabular}{lll} 
Postestimation commands & predict & margins \\
Remarks and examples & Methods and formulas & Also see
\end{tabular}

\section*{Postestimation commands}

The following postestimation commands are available after heckoprobit:
\begin{tabular}{|c|c|}
\hline Command & Description \\
\hline contrast & contrasts and ANOVA-style joint tests of estimates \\
\hline estat ic & Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) \\
\hline estat summarize & summary statistics for the estimation sample \\
\hline estat vce & variance-covariance matrix of the estimators (VCE) \\
\hline estat (svy) & postestimation statistics for survey data \\
\hline estimates & cataloging estimation results \\
\hline etable & table of estimation results \\
\hline * hausman & Hausman's specification test \\
\hline lincom & point estimates, standard errors, testing, and inference for linear combinations of coefficients \\
\hline * 1 rtest & likelihood-ratio test \\
\hline margins & marginal means, predictive margins, marginal effects, and average marginal effects \\
\hline marginsplot & graph the results from margins (profile plots, interaction plots, etc.) \\
\hline nlcom & point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients \\
\hline predict & probabilities, linear predictions and their SEs, etc. \\
\hline predictnl & point estimates, standard errors, testing, and inference for generalized predictions \\
\hline pwcompare & pairwise comparisons of estimates \\
\hline suest & seemingly unrelated estimation \\
\hline test & Wald tests of simple and composite linear hypotheses \\
\hline testnl & Wald tests of nonlinear hypotheses \\
\hline
\end{tabular}

\footnotetext{
*hausman and lrtest are not appropriate with svy estimation results.
}

\section*{predict}

\section*{Description for predict}
predict creates a new variable containing predictions such as probabilities, linear predictions, and standard errors.

\section*{Menu for predict}

Statistics \(>\) Postestimation

\section*{Syntax for predict}
```

predict [type] {stub*|newvar|newvarlist } [if] [in] [, statistic
outcome(outcome) nooffset]
predict [type] stub* [if] [in], scores

```
statistic Description

\section*{Main}
pmargin marginal probabilities; the default
p1 bivariate probabilities of levels with selection
p0 bivariate probabilities of levels with no selection
pcond1 probabilities of levels conditional on selection
pcond0 probabilities of levels conditional on no selection
psel selection probability
\(\overline{\mathrm{xb}} \quad\) linear prediction
stdp standard error of the linear prediction
xbsel linear prediction for selection equation
stdpsel standard error of the linear prediction for selection equation
If you do not specify outcome(), pmargin (with one new variable specified) assumes outcome(\#1).
You specify one or \(k\) new variables with pmargin, where \(k\) is the number of outcomes.
You specify one new variable with psel, xb, stdp, xbsel, and stdpsel.
These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

\section*{Options for predict}
\(\qquad\) Main
pmargin, the default, calculates the predicted marginal probabilities.
You specify one or \(k\) new variables, where \(k\) is the number of categories of the outcome variable \(y_{j}\). If you specify the outcome() option, you must specify one new variable. If you specify one new variable and do not specify outcome(), outcome(\#1) is assumed.

When outcome() is specified, the marginal probability that \(y_{j}\) is equal to the level outcome() is calculated. When outcome() is not specified, the marginal probabilities for each outcome level are calculated.
p1 calculates the predicted bivariate probabilities of outcome levels with selection.
You specify one or \(k\) new variables, where \(k\) is the number of categories of the outcome variable \(y_{j}\). If you specify the outcome() option, you must specify one new variable. If you specify one new variable and do not specify outcome(), outcome(\#1) is assumed.

When outcome() is specified, the bivariate probability that \(y_{j}\) is equal to the level outcome() and that \(y_{j}\) is observed is calculated. When outcome() is not specified, the bivariate probabilities for each outcome level and selection are calculated.
p0 calculates the predicted bivariate probabilities of outcome levels with no selection.
You specify one or \(k\) new variables, where \(k\) is the number of categories of the outcome variable \(y_{j}\). If you specify the outcome() option, you must specify one new variable. If you specify one new variable and do not specify outcome(), outcome(\#1) is assumed.

When outcome() is specified, the bivariate probability that \(y_{j}\) is equal to the level outcome() and that \(y_{j}\) is not observed is calculated. When outcome() is not specified, the bivariate probabilities for each outcome level and no selection are calculated.
pcond1 calculates the predicted probabilities of outcome levels conditional on selection.
You specify one or \(k\) new variables, where \(k\) is the number of categories of the outcome variable \(y_{j}\). If you specify the outcome() option, you must specify one new variable. If you specify one new variable and do not specify outcome(), outcome(\#1) is assumed.

When outcome() is specified, the probability that \(y_{j}\) is equal to the level outcome() given that \(y_{j}\) is observed is calculated. When outcome () is not specified, the probabilities for each outcome level conditional on selection are calculated.
pcond0 calculates the predicted probabilities of outcome levels conditional on no selection.
You specify one or \(k\) new variables, where \(k\) is the number of categories of the outcome variable \(y_{j}\). If you specify the outcome () option, you must specify one new variable. If you specify one new variable and do not specify outcome(), outcome(\#1) is assumed.

When outcome() is specified, the probability that \(y_{j}\) is equal to the level outcome() given that \(y_{j}\) is not observed is calculated. When outcome() is not specified, the probabilities for each outcome level conditional on no selection are calculated.
psel calculates the predicted univariate (marginal) probability of selection.
xb calculates the linear prediction for the outcome variable, which is \(\mathbf{x}_{j} \boldsymbol{\beta}\) if offset() was not specified and \(\mathbf{x}_{j} \boldsymbol{\beta}+\operatorname{offset}_{j}^{\beta}\) if offset () was specified.
stdp calculates the standard error of the prediction, which can be thought of as the standard error of the predicted expected value or mean for the observation's covariate pattern. The standard error of the prediction is also referred to as the standard error of the fitted value.
xbsel calculates the linear prediction for the selection equation, which is \(\mathbf{z}_{j} \gamma\) if offset () was not specified in select() and \(\mathbf{z}_{j} \gamma+\operatorname{offset}_{j}^{\gamma}\) if offset() was specified in select().
stdpsel calculates the standard error of the linear prediction for the selection equation.
outcome(outcome) specifies for which outcome the predicted probabilities are to be calculated. outcome() should contain either one value of the dependent variable or one of \#1, \#2, ..., with \#1 meaning the first category of the dependent variable, \#2 meaning the second category, etc.
nooffset is relevant only if you specified offset (varname) for heckoprobit. It modifies the calculations made by predict so that they ignore the offset variable; the linear prediction is treated as \(\mathbf{x}_{j} \mathbf{b}\) rather than as \(\mathbf{x}_{j} \mathbf{b}+\) offset \(_{j}\).
scores calculates equation-level score variables.
The first new variable will contain \(\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)\).
The second new variable will contain \(\partial \ln L / \partial\left(\mathbf{z}_{j} \gamma\right)\).
When the dependent variable takes \(k\) different values, new variables three through \(k+1\) will contain \(\partial \ln L / \partial\left(\kappa_{j-2}\right)\).
The last new variable will contain \(\partial \ln L / \partial(\operatorname{atanh} \rho)\).

\section*{margins}

\section*{Description for margins}
margins estimates margins of response for probabilities and linear predictions.

\section*{Menu for margins}

Statistics \(>\) Postestimation

\section*{Syntax for margins}
margins [marginlist] [, options]

statistic Description
\begin{tabular}{ll}
\hline default & \begin{tabular}{l} 
marginal probabilities for each outcome \\
pmargin
\end{tabular} \\
\hline p1 & \begin{tabular}{l} 
bivginal probabilities
\end{tabular} \\
p0 & bivariate probabilities of levels with selection \\
pcond1 & \begin{tabular}{l} 
probabilities of levels conditional on selection \\
probabilities of levels conditional on no selection
\end{tabular} \\
psel & \begin{tabular}{l} 
selection probability
\end{tabular} \\
\(\underline{\mathrm{xb}}\) & \begin{tabular}{l} 
linear prediction
\end{tabular} \\
\(\underline{\text { xbsel }}\) & linear prediction for selection equation \\
stdp & not allowed with margins \\
stdpsel & not allowed with margins
\end{tabular}
pmargin, p1, p0, pcond1, and pcond0 default to the first outcome.
Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

\section*{Remarks and examples}

\section*{> Example 1}

In example 1 of [R] heckoprobit, we examined a simulated dataset of 5,000 women, 3,480 of whom work and can thus report job satisfaction. Using job satisfaction (satisfaction) as the outcome variable and employment (work) as the selection variable, we estimated the parameters of an ordered probit sample-selection model. Covariates age (age), years of education (education), number of children (children), and marital status (married) are expected to affect selection. The outcome, job satisfaction, is affected by age (age) and education (education).

We first reestimate the parameters of the regression, but this time we request a robust variance estimator:
```

. use https://www.stata-press.com/data/r18/womensat
(Job satisfaction, female)
. heckoprobit satisfaction education age,
> select(work=education age i.married\#\#c.children) vce(robust)
(output omitted)

```

We then use margins (see [R] margins) to estimate the average marginal effect of education on the probability of having low job satisfaction.
```

. margins, dydx(education) vce(unconditional)
Average marginal effects Number of obs = 5,000
dy/dx wrt: education
1._predict: Pr(satisfaction=1), predict(pmargin outcome(1))
2._predict: Pr(satisfaction=2), predict(pmargin outcome(2))
3._predict: Pr(satisfaction=3), predict(pmargin outcome(3))
4._predict: Pr(satisfaction=4), predict(pmargin outcome(4))

```
\begin{tabular}{c|rrrrrr}
\hline & \multicolumn{5}{|c}{ Unconditional } \\
& dy/dx & std. err. & \(z\) & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline education & & & & & & \\
_predict & & & & & \\
1 & -.0234776 & .0019176 & -12.24 & 0.000 & -.027236 & -.0197192 \\
2 & -.0230858 & .0015143 & -15.24 & 0.000 & -.0260538 & -.0201178 \\
3 & -.0020603 & .001142 & -1.80 & 0.071 & -.0042987 & .000178 \\
4 & .0486238 & .0018593 & 26.15 & 0.000 & .0449795 & .052268 \\
\hline
\end{tabular}

The estimated average marginal effect of education on the probability of having low job satisfaction is approximately -0.023 .

\section*{Methods and formulas}

The ordinal outcome equation is
\[
y_{j}=\sum_{h=1}^{H} v_{h} 1\left(\kappa_{h-1}<\mathbf{x}_{j} \boldsymbol{\beta}+u_{1 j} \leq \kappa_{h}\right)
\]
where \(\mathbf{x}_{j}\) is the outcome covariates, \(\boldsymbol{\beta}\) is the coefficients, and \(u_{1 j}\) is a random-error term. The observed outcome values \(v_{1}, \ldots, v_{H}\) are integers such that \(v_{i}<v_{m}\) for \(i<m . \kappa_{1}, \ldots, \kappa_{H-1}\) are real numbers such that \(\kappa_{i}<\kappa_{m}\) for \(i<m . \kappa_{0}\) is taken as \(-\infty\) and \(\kappa_{H}\) is taken as \(+\infty\).

The selection equation is
\[
s_{j}=1\left(\mathbf{z}_{j} \gamma+u_{2 j}>0\right)
\]
where \(s_{j}=1\) if we observed \(y_{j}\) and 0 otherwise, \(\mathbf{z}_{j}\) is the covariates used to model the selection process, \(\gamma\) is the coefficients for the selection process, and \(u_{2 j}\) is a random-error term.
\(\left(u_{1 j}, u_{2 j}\right)\) have bivariate normal distribution with mean zero and variance matrix
\[
\left[\begin{array}{ll}
1 & \rho \\
\rho & 1
\end{array}\right]
\]

The probability of selection is
\[
\operatorname{Pr}\left(s_{j}=1\right)=\Phi\left(\mathbf{z}_{j} \gamma+\operatorname{offset}_{j}^{\gamma}\right)
\]
\(\Phi(\cdot)\) is the standard cumulative normal distribution function.
The probability of selection and the outcome \(y_{j}=v_{h}\) is
\[
\begin{aligned}
\operatorname{Pr}\left(y_{j}=v_{h}, s_{j}=1\right)= & \Phi_{2}\left(\mathbf{z}_{j} \gamma+\operatorname{offset}_{j}^{\gamma}, \kappa_{h}-x_{j} \boldsymbol{\beta}-\operatorname{offset}_{j}^{\beta},-\rho\right) \\
& -\Phi_{2}\left(\mathbf{z}_{j} \gamma+\operatorname{offset}_{j}^{\gamma}, \kappa_{h-1}-\mathbf{x}_{j} \boldsymbol{\beta}-\operatorname{offset}_{j}^{\beta},-\rho\right)
\end{aligned}
\]
\(\Phi_{2}(\cdot)\) is the cumulative bivariate normal distribution function (with mean \(\left[\begin{array}{ll}0 & 0\end{array}\right]^{\prime}\) ).
The probability of \(y_{j}\) not being selected and the outcome \(y_{j}=v_{h}\) is
\[
\begin{aligned}
\operatorname{Pr}\left(y_{j}=v_{h}, s_{j}=0\right)= & \Phi_{2}\left(-\mathbf{z}_{j} \boldsymbol{\gamma}-\operatorname{offset}_{j}^{\gamma}, \kappa_{h}-x_{j} \boldsymbol{\beta}-\operatorname{offset}_{j}^{\beta}, \rho\right) \\
& -\Phi_{2}\left(-\mathbf{z}_{j} \gamma-\operatorname{offset}_{j}^{\gamma}, \kappa_{h-1}-\mathbf{x}_{j} \boldsymbol{\beta}-\operatorname{offset}_{j}^{\beta}, \rho\right)
\end{aligned}
\]

The probability of outcome \(y_{j}=v_{h}\) given selection is
\[
\operatorname{Pr}\left(y_{j}=v_{h} \mid s_{j}=1\right)=\frac{\operatorname{Pr}\left(y_{j}=v_{h}, s_{j}=1\right)}{\operatorname{Pr}\left(s_{j}=1\right)}
\]

The probability of outcome \(y_{j}=v_{h}\) given \(y_{j}\) is not selected is
\[
\operatorname{Pr}\left(y_{j}=v_{h} \mid s_{j}=0\right)=\frac{\operatorname{Pr}\left(y_{j}=v_{h}, s_{j}=0\right)}{\operatorname{Pr}\left(s_{j}=0\right)}
\]

The marginal probabilities of the outcome \(y_{j}\) are
\[
\begin{aligned}
\operatorname{Pr}\left(y_{j}=v_{1}\right) & =\Phi\left(\kappa_{1}-x_{j} \boldsymbol{\beta}-\operatorname{offset}_{j}^{\beta}\right) \\
\operatorname{Pr}\left(y_{j}=v_{H}\right) & =1-\Phi\left(\kappa_{H-1}-x_{j} \boldsymbol{\beta}-\operatorname{offset}_{j}^{\beta}\right) \\
\operatorname{Pr}\left(y_{j}=v_{h}\right) & =\Phi\left(\kappa_{h}-x_{j} \boldsymbol{\beta}-\operatorname{offset}_{j}^{\beta}\right)-\Phi\left(\kappa_{h-1}-x_{j} \boldsymbol{\beta}-\operatorname{offset}_{j}^{\beta}\right)
\end{aligned}
\]

\section*{Also see}
[R] heckoprobit - Ordered probit model with sample selection
[U] 20 Estimation and postestimation commands

\section*{Title}
heckpoisson - Poisson regression with sample selection
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
heckpoisson fits a Poisson regression model with endogenous sample selection. This is sometimes called nonignorability of selection, missing not at random, or selection bias. Unlike the standard Poisson model, there is no assumption of equidispersion.

\section*{Quick start}

Poisson model of \(y\) on \(x 1\) with \(z 1\) predicting selection when binary variable selected indicates selection status
heckpoisson y x1, select (selected \(=\mathrm{z} 1\) )
Add categorical variable a using factor-variables syntax
heckpoisson y x1 i.a, select(selected = z1 i.a)
Report results as incidence-rate ratios
heckpoisson \(y\) x1 i.a, select(selected = z1 i.a) irr
Add robust standard errors
heckpoisson y x1 i.a, select(selected = z1 i.a) vce (robust)
Include exposure variable expose to account for different exposure levels
heckpoisson y x1 i.a, select(selected = z1 i.a) exposure(expose)

\section*{Menu}

Statistics \(>\) Sample-selection models \(>\) Poisson model with sample selection

\section*{Syntax}
```

heckpoisson depvar indepvars [if] [in] [weight],
$\underline{\text { select }}\left(\left[\right.\right.$ depvar $\left._{s}=\right]$ indepvars $_{s}\left[, \underline{\text { noconstant }} \underline{\text { offset }}\left(\right.\right.$ varname $\left.\left.\left._{o s}\right)\right]\right)$ [options $]$

```
options
Model
* select()
noconstant
exposure ( varname \(_{e}\) )
offset ( varname \(_{o}\) )
constraints (constraints)
SE/Robust
vce (vcetype)

Reporting
level(\#)
irr
nocnsreport
display_options

Integration
intpoints(\#)

Maximization
maximize_options
collinear
coeflegend

Description
specify selection equation: dependent and independent variables; whether to have constant term and offset variable suppress constant term
include \(\ln \left(\right.\) varname \(\left._{e}\right)\) in model with coefficient constrained to 1 include varname \(_{o}\) in model with coefficient constrained to 1 apply specified linear constraints
vcetype may be oim, robust, cluster clustvar, opg, bootstrap, or jackknife
set confidence level; default is level (95)
report incidence-rate ratios
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
set the number of integration (quadrature) points; default is intpoints(25)
control the maximization process; seldom used
keep collinear variables
display legend instead of statistics
*select () is required.
The full specification is select \(\left(\left[\right.\right.\) depvar \(\left._{s}=\right]\) indepvars \(_{s}\left[\right.\), noconstant offset (varname \({ }_{o s}\) ) \(]\) ). indepvars and indepvarss may contain factor variables; see [U] 11.4.3 Factor variables.
indepvars and indepvars \(s\) may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bootstrap, by, collect, jackknife, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands.
Weights are not allowed with the bootstrap prefix; see \([R]\) bootstrap.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options}
```

    Model
    select([\mp@subsup{depvar s = ] indepvars}{s}{[, noconstant offset(\mp@subsup{varname }{os}{})]}\mathrm{ ) specifies the variables} and options for the selection equation. It is an integral part of specifying a sample-selection model and is required.
    If depvar ${ }_{s}$ is specified, it should be coded as 0 or 1 , with 0 indicating an observation not selected and 1 indicating a selected observation. If depvar $r_{s}$ is not specified, then observations for which depvar is not missing are assumed selected and those for which depvar is missing are assumed not selected.
noconstant suppresses the selection constant term (intercept).
offset ( varname $_{o s}$ ) specifies that selection offset varname ${ }_{o s}$ be included in the model with the coefficient constrained to be 1 .
noconstant, exposure (varname ), offset ( varname $_{o}$ ), constraints (constraints); see [R] Estimation options.

```

SE/Robust
vce(vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.

\section*{Reporting}
level(\#); see [R] Estimation options.
irr reports estimated coefficients transformed to incidence-rate ratios, that is, \(e^{\beta_{i}}\) rather than \(\beta_{i}\). Standard errors and confidence intervals are similarly transformed. This option affects how results are displayed, not how they are estimated or stored. irr may be specified at estimation or when replaying previously estimated results.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% \(\% m t\) ), and nolstretch; see [R] Estimation options.

Integration
intpoints (\#) specifies the number of integration points to use for quadrature. The default is intpoints(25), which means that 25 quadrature points are used. The maximum number of allowed integration points is 128 .

The more integration points, the more accurate the approximation to the log likelihood. However, computation time increases with the number of quadrature points and is roughly proportional to the number of points used.

\section*{Maximization}
maximize_options: difficult, technique (algorithm_spec), iterate(\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from(init_specs); see [R] Maximize. These options are seldom used.

The following options are available with heckpoisson but are not shown in the dialog box: collinear, coeflegend; see [R] Estimation options.

\section*{Remarks and examples}

When analyzing observational data, we must consider the possibility that we cannot treat the observations for which we have data as if they were selected at random. Suppose we are interested in the number of after-school tutoring sessions a child attends. If unobservable variables that affect which students attend the sessions, for example, family stability, also affect the number of visits we observe, then a condition known as endogenous sample selection is present. This phenomenon is sometimes simply referred to as sample selection or called missing not at random, nonignorability of selection, or selection bias. When endogenous sample selection occurs, conventional estimation techniques are not appropriate. Cameron and Trivedi (2022, 974-981) and Greene (2018, 950-957) provide good introductions to the concept of endogenous sample selection.

The venerable Heckman estimator handles endogenous sample selection when the outcome of interest is modeled by linear regression; see [R] heckman. However, the Heckman estimator is not appropriate for count outcomes because its linear model for the outcome could produce negative predicted values and does not restrict the predicted values to integers.

There are different methods for estimating the parameters of a count-data model with endogenous sample selection. heckpoisson implements the maximum likelihood estimator derived in Terza (1998); see also Cameron and Trivedi (2013, chap. 10) for a discussion of this estimator.

The model consists of one equation for the count outcome, \(y\), and one equation for a binary selection indicator, \(s\). The indicator \(s\) is always observed and takes values of 0 or 1 . But the outcome \(y\) is observed only if \(s=1\), that is, we have complete information about the covariates of interest and selection status. However, the value of the primary outcome of interest, \(y\), is sometimes unknown.

More formally, the count outcome \(y\) is assumed to have a Poisson distribution, conditional on the covariates, with conditional mean
\[
E\left(y_{j} \mid \mathbf{x}_{j}, \epsilon_{1 j}\right)=\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{1 j}\right) \quad \text { Poisson regression equation }
\]

However, we only observe \(y\) for observation \(j\) if \(s_{j}=1\) :
\[
s_{j}=\left\{\begin{array}{ll}
1, & \text { if } \mathbf{w}_{j} \gamma+\epsilon_{2 j}>0 \\
0, & \text { otherwise }
\end{array} \quad\right. \text { selection equation }
\]
where
\[
\begin{gathered}
\epsilon_{1} \sim N(0, \sigma) \\
\epsilon_{2} \sim N(0,1) \\
\operatorname{corr}\left(\epsilon_{1}, \epsilon_{2}\right)=\rho
\end{gathered}
\]

When \(\rho \neq 0\), standard Poisson regression based on the observed \(y\) yields biased estimates. heckpoisson provides consistent, asymptotically efficient estimates for the parameters in such models.

Unlike the standard Poisson regression, the Poisson model with sample selection allows underdispersion and overdispersion.

\section*{> Example 1: Poisson model with sample selection}

Suppose we want to know the effect of research and development (R\&D) expenditures on the number of patents obtained by a firm in the last two years. The patent dataset contains fictional data on the number of patents (npatents) of 10,000 firms in different sectors. After reading in the data, we tabulate the frequencies of npatents against an indicator for whether a firm applied for patents (applied).
\begin{tabular}{|c|c|c|c|}
\hline Number of patents (last 2 yrs) & Applied f Not apply & \begin{tabular}{l}
atent \\
Apply
\end{tabular} & Total \\
\hline 0 & 0 & 1,127 & 1,127 \\
\hline 1 & 0 & 1,455 & 1,455 \\
\hline 2 & 0 & 1,131 & 1,131 \\
\hline 3 & 0 & 710 & 710 \\
\hline 4 & 0 & 479 & 479 \\
\hline 5 & 0 & 266 & 266 \\
\hline 6 & 0 & 126 & 126 \\
\hline 7 & 0 & 98 & 98 \\
\hline 8 & 0 & 66 & 66 \\
\hline 9 & 0 & 42 & 42 \\
\hline 10 & 0 & 19 & 19 \\
\hline 11 & 0 & 24 & 24 \\
\hline 12 & 0 & 5 & 5 \\
\hline 13 & 0 & 7 & 7 \\
\hline 14 & 0 & 5 & 5 \\
\hline 15 & 0 & 10 & 10 \\
\hline 17 & 0 & 1 & 1 \\
\hline 18 & 0 & 1 & 1 \\
\hline 19 & 0 & 2 & 2 \\
\hline 22 & 0 & 1 & 1 \\
\hline . & 4,425 & 0 & 4,425 \\
\hline Total & 4,425 & 5,575 & 10,000 \\
\hline
\end{tabular}

The output shows that npatents is missing for about half of the sample because some firms did not apply for any patents. Some firms prefer to keep their discoveries as trade secrets instead of applying for patents. The sample selection will be endogenous if the unobservable variables that affect which firms apply for patents also affect the number of patents obtained. Therefore, we do not want to use a standard Poisson model for these data.

We model npatents as a function of R\&D expenditures (expenditure) and a categorical variable indicating whether the firm is in the information technology (IT) sector (tech). We model the selection indicator applied as a function of expenditure, tech, and firm size (size), which is excluded from the outcome model.
```

. heckpoisson npatents expenditure i.tech,
> select(applied = expenditure size i.tech)
Initial: Log likelihood = -17442.266
Rescale: Log likelihood = -17442.266
Rescale eq: Log likelihood = -17442.266
(setting technique to bhhh)
Iteration 0: Log likelihood = -17442.266
Iteration 1: Log likelihood = -17441.444
Iteration 2: Log likelihood = -17440.72
Iteration 3: Log likelihood = -17440.438
Iteration 4: Log likelihood = -17440.438

```
Poisson regression with endogenous selection
(25 quadrature points)
Log likelihood \(=-17440.44 \quad\) Prob \(>\) chi2 \(=0.0000\)
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline npatents & Coefficient & Std. err. & z & \(P>|z|\) & [95\% conf & interval] \\
\hline npatents expenditure & . 497821 & . 0507866 & 9.80 & 0.000 & . 398281 & . 597361 \\
\hline tech IT sector _cons & \[
\begin{array}{r}
.5833501 \\
-1.855143
\end{array}
\] & \[
\begin{array}{r}
.0300366 \\
.208204
\end{array}
\] & \[
\begin{aligned}
& 19.42 \\
& -8.91
\end{aligned}
\] & \[
\begin{aligned}
& 0.000 \\
& 0.000
\end{aligned}
\] & \[
\begin{array}{r}
.5244795 \\
-2.263216
\end{array}
\] & \[
\begin{array}{r}
.6422207 \\
-1.447071
\end{array}
\] \\
\hline applied expenditure size & \[
\begin{aligned}
& .1369954 \\
& .2774201
\end{aligned}
\] & \[
\begin{aligned}
& .0447339 \\
& .0469132
\end{aligned}
\] & \[
\begin{aligned}
& 3.06 \\
& 5.91
\end{aligned}
\] & \[
\begin{aligned}
& 0.002 \\
& 0.000
\end{aligned}
\] & \[
\begin{aligned}
& .0493185 \\
& .1854718
\end{aligned}
\] & \[
\begin{aligned}
& .2246723 \\
& .3693683
\end{aligned}
\] \\
\hline IT sector _cons & \[
\begin{array}{r}
.2750208 \\
-1.660778
\end{array}
\] & \[
\begin{aligned}
& .0277032 \\
& .2631227
\end{aligned}
\] & \[
\begin{array}{r}
9.93 \\
-6.31
\end{array}
\] & \[
\begin{aligned}
& 0.000 \\
& 0.000
\end{aligned}
\] & \[
\begin{array}{r}
.2207236 \\
-2.176489
\end{array}
\] & \[
\begin{array}{r}
.329318 \\
-1.145066
\end{array}
\] \\
\hline /athrho /lnsigma & \[
\begin{array}{r}
1.161677 \\
-.3029685
\end{array}
\] & \[
\begin{aligned}
& .2847896 \\
& .0499674
\end{aligned}
\] & \[
\begin{array}{r}
4.08 \\
-6.06
\end{array}
\] & \[
\begin{aligned}
& 0.000 \\
& 0.000
\end{aligned}
\] & \[
\begin{array}{r}
.6034999 \\
-.4009028
\end{array}
\] & \[
\begin{array}{r}
1.719855 \\
-.2050342
\end{array}
\] \\
\hline \[
\begin{array}{r}
\text { rho } \\
\text { sigma }
\end{array}
\] & \[
\begin{aligned}
& .8215857 \\
& .7386224
\end{aligned}
\] & \[
\begin{array}{r}
.0925557 \\
.036907
\end{array}
\] & & & \[
\begin{aligned}
& .5395353 \\
& .6697151
\end{aligned}
\] & \[
\begin{aligned}
& .9378455 \\
& .8146195
\end{aligned}
\] \\
\hline
\end{tabular}

Wald test of indep. eqns. (rho \(=0): \operatorname{chi2}(1)=16.64\) Prob \(>\) chi2 \(=0.0000\)
The coefficient estimates reported by heckpoisson can be interpreted similarly to those reported by poisson. For example, the positive coefficient on expenditure tells us that increasing R\&D expenditures is associated with an increasing number of patents. However, the magnitude of the effect cannot be directly determined by the coefficients. The best way to obtain interpretable effects is by using margins. See example 1 in [R] heckpoisson postestimation for more information.

The estimated correlation between the selection errors and outcome errors is 0.8 , and the Wald test in the footer indicates that we can reject the null hypothesis of zero correlation. This positive and significant correlation estimate implies that unobservable factors that increase the number of patents a firm is awarded tend to occur with unobservable factors that also increase the chance of a firm being willing to apply for patents.

\section*{- Technical note}

In practice, we rely on the strength of the relationship between size and applied and the fact that size does not appear in the model for npatents to pin down the parameter estimates. Technically, we do not need this exclusion restriction, but identification from the functional form alone tends to be weak. For a discussion of this point, see Cameron and Trivedi (2022, 977-981).

\section*{> Example 2: Obtaining incidence-rate ratios}

In some cases, we may wish to view the parameters as incidence-rate ratios (IRRs). That is, we want to hold all the \(x\) 's in the model constant except one, say, the \(i\) th. The IRR for a one-unit change in \(x_{i}\) is
\[
\frac{e^{\ln (E)+\beta_{1} x_{1}+\cdots+\beta_{i}\left(x_{i}+1\right)+\cdots+\beta_{k} x_{k}+e_{1}}}{e^{\ln (E)+\beta_{1} x_{1}+\cdots+\beta_{i} x_{i}+\cdots+\beta_{k} x_{k}+e_{1}}}=e^{\beta_{i}}
\]

For instance, we may want to know the relative incidence rate of patents as the expenditure changes or the relative incidence rate of patents as sectors change from non-IT to IT.

We can use option irr to display the coefficient estimates transformed to IRRs. This option may be specified when we originally fit our model or on replay. Because we have already fit the model, we specify irr below using the replay syntax.
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline \multicolumn{4}{|l|}{Poisson regression with endogenous selection (25 quadrature points)} & \multicolumn{2}{|l|}{\multirow[t]{3}{*}{```
Number of obs
    Selected
    Nonselected
Wald chi2(2)
Prob > chi2
```}} & 10,000 5,575 4,425 \\
\hline & & & & & & 443.90 \\
\hline \multicolumn{4}{|l|}{Log likelihood = -17440.44} & & & 0.0000 \\
\hline npatents & IRR & Std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf & interval] \\
\hline \multirow[t]{3}{*}{\begin{tabular}{l}
npatents \\
expenditure \\
tech \\
IT sector _cons
\end{tabular}} & 1.645133 & . 0835508 & 9.80 & 0.000 & 1.489262 & 1.817316 \\
\hline & 1.792032 & . 0538265 & 19.42 & 0.000 & 1.689579 & 1.900697 \\
\hline & . 1564305 & . 0325695 & -8.91 & 0.000 & . 1040154 & . 2352583 \\
\hline \multirow[t]{5}{*}{\begin{tabular}{l}
applied \\
expenditure size \\
tech \\
IT sector \\
_cons
\end{tabular}} & & & & & & \\
\hline & . 1369954 & . 0447339 & 3.06 & 0.002 & . 0493185 & . 2246723 \\
\hline & . 2774201 & . 0469132 & 5.91 & 0.000 & . 1854718 & . 3693683 \\
\hline & . 2750208 & . 0277032 & 9.93 & 0.000 & . 2207236 & . 329318 \\
\hline & -1.660778 & . 2631227 & -6.31 & 0.000 & -2.176489 & -1.145066 \\
\hline \multirow[t]{2}{*}{\begin{tabular}{l}
/athrho \\
/lnsigma
\end{tabular}} & 1.161677 & . 2847896 & 4.08 & 0.000 & . 6034999 & 1.719855 \\
\hline & -. 3029685 & . 0499674 & -6.06 & 0.000 & -. 4009028 & -. 2050342 \\
\hline \multirow[t]{2}{*}{\[
\begin{array}{r}
\text { rho } \\
\text { sigma }
\end{array}
\]} & . 8215857 & . 0925557 & & & . 5395353 & . 9378455 \\
\hline & . 7386224 & . 036907 & & & . 6697151 & . 8146195 \\
\hline
\end{tabular}

\footnotetext{
Note: Estimates are transformed only in the first equation to incidence-rate ratios.
Note: _cons estimates baseline incidence rate.
Wald test of indep. eqns. (rho \(=0\) ) : chi2 \((1)=16.64\) Prob \(>\) chi2 \(=0.0000\)
}

The IRR for IT is about 1.8 , meaning that the expected number of patents in the IT sector is 1.8 times more than in the non-IT sector.

\section*{Stored results}
heckpoisson stores the following in e():
Scalars
```

        e(N)
    e(N_selected)
    e(N_nonselected)
    e(k)
    e(k_eq)
    e(k_eq_model)
    e(k_aux)
    e(k_dv)
    e(df_m)
    e(ll)
    e(N_clust)
    e(chi2)
    e(chi2_c)
    e(n_quad)
    e(p)
    e(p_c)
    e(rank)
    e(ic)
    e(rc)
    e(converged)
    ```
Macros
    e(cmd)
    e(cmdline)
    e(depvar)
    e(wtype)
    e(wexp)
    e(title)
    e(title2)
    e(clustvar)
    e(offset1)
    e(offset2)
    e(chi2type)
    e(chi2_ct)
    e(vce)
    e(vcetype)
    e(opt)
    e(which)
    e(ml_method)
    e(user)
    e(technique)
    e(properties)
    e(predict)
    e(marginsok)
    e(marginsnotok)
    e(asbalanced)
    e(asobserved)
        number of observations
        number of selected observations
        number of nonselected observations
        number of parameters
        number of equations in e(b)
        number of equations in overall model test
    number of auxiliary parameters
    number of dependent variables
    model degrees of freedom
    log likelihood
    number of clusters
    \(\chi^{2}\)
    \(\chi^{2}\) for comparison, \(\rho=0\) test
    number of quadrature points
    \(p\)-value for model test
    \(p\)-value for comparison test
    rank of \(e(V)\)
    number of iterations
    return code
    1 if converged, 0 otherwise
heckpoisson
command as typed
name of dependent variable
weight type
weight expression
title in estimation output
secondary title in estimation output
name of cluster variable
offset for regression equation
offset for selection equation
Wald; type of model \(\chi^{2}\) test
Wald; type of comparison \(\chi^{2}\) test
vcetype specified in vce()
title used to label Std. err.
type of optimization
max or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement predict
predictions allowed by margins
predictions disallowed by margins
factor variables fvset as asbalanced
factor variables fvset as asobserved
```

Matrices
e(b) coefficient vector
e(Cns) constraints matrix
e(ilog) iteration log (up to 20 iterations)
e(gradient) gradient vector
e(V) variance-covariance matrix of the estimators
e(V_modelbased) model-based variance
Functions
e(sample) marks estimation sample

```

In addition to the above, the following is stored in r() :
Matrices
\(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

\section*{Methods and formulas}
heckpoisson implements Terza's maximum likelihood estimator for the parameters of a count-data model with endogenous sample selection (Terza 1998).

Suppose that the count outcome \(y_{j}\) has covariates \(\mathbf{x}_{j}\) and that \(y_{j}\) has a Poisson distribution, conditional on \(\mathbf{x}_{j}\), with conditional mean
\[
E\left(y_{j} \mid \mathbf{x}_{j}, \epsilon_{1 j}\right)=\mu_{j}=\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{1 j}\right)
\]
and
\[
\operatorname{Pr}\left(Y=y_{j} \mid \mathbf{x}_{j}, \epsilon_{1 j}\right)=\frac{\mu_{j}^{y_{j}} e^{-\mu_{j}}}{y_{j}!}
\]

We only observe \(y_{j}\) when \(s_{j}\), the selection outcome, which is the binary outcome from a latent-variable model with covariates \(\mathbf{w}_{j}\), is equal to 1 .
\[
s_{j}= \begin{cases}1, & \text { if } \mathbf{w}_{j} \gamma+\epsilon_{2 j}>0 \\ 0, & \text { otherwise }\end{cases}
\]

The error terms \(\epsilon_{1}\) and \(\epsilon_{2}\) are assumed to have bivariate normal distribution with zero mean and covariance matrix
\[
\left[\begin{array}{cc}
\sigma^{2} & \sigma \rho \\
\sigma \rho & 1
\end{array}\right]
\]
where \(\sigma\) and \(\rho\) have their usual interpretation for the bivariate normal distribution. A nonzero \(\rho\) implies that the selected sample is not representative of the whole population and therefore that inference based on standard Poisson regression using the observed sample is incorrect.

In maximum likelihood estimation, \(\ln \sigma\) and atanh \(\rho\) are estimated rather than directly estimating \(\sigma\) and \(\rho\).
\[
\operatorname{atanh} \rho=\frac{1}{2} \ln \left(\frac{1+\rho}{1-\rho}\right)
\]

The joint log likelihood is given by
\[
\ln L(\boldsymbol{\theta})=\sum_{i=1}^{N}\left[s_{j} \times \ln \left\{\operatorname{Pr}\left(y_{j}, s_{j}=1\right) \mid \mathbf{x}_{j}, \mathbf{w}_{j}, \boldsymbol{\theta}\right\}+\left(1-s_{j}\right) \times \ln \left\{\operatorname{Pr}\left(s_{j}=0 \mid \mathbf{w}_{j}, \boldsymbol{\theta}\right)\right\}\right]
\]
where \(\boldsymbol{\theta}\) denotes \((\boldsymbol{\beta}, \gamma, \rho, \sigma)\) for notational simplicity.

The joint probability \(\operatorname{Pr}\left(y_{j}, s_{j}=1 \mid \mathbf{x}_{j}, \mathbf{w}_{j}, \boldsymbol{\theta}\right)\) can be obtained by integrating the conditional probability \(\operatorname{Pr}\left(y_{j}, s_{j}=1 \mid \mathbf{x}_{j}, \mathbf{w}_{j}, \boldsymbol{\theta}, \epsilon_{1}\right)\) over \(\epsilon_{1}\). More precisely,
\[
\begin{equation*}
\operatorname{Pr}\left(y_{j}, s_{j}=1 \mid \mathbf{x}_{j}, \mathbf{w}_{j}, \boldsymbol{\theta}\right)=\int_{-\infty}^{\infty} \operatorname{Pr}\left(y_{j} \mid \mathbf{x}_{j}, \epsilon_{1}\right) \Phi\left(\frac{\mathbf{w}_{j} \gamma+\rho / \sigma \epsilon_{1}}{\sqrt{1-\rho^{2}}}\right) \phi\left(\epsilon_{1} / \sigma\right) d \epsilon_{1} \tag{1}
\end{equation*}
\]
where \(\phi(\cdot)\) is the standard normal density function and \(\Phi(\cdot)\) is the standard normal cumulative density function. \(\operatorname{Pr}\left(s_{j}=0 \mid \mathbf{w}_{j}, \boldsymbol{\theta}\right)\) is similarly derived.
\[
\begin{equation*}
\operatorname{Pr}\left(s_{j}=0 \mid \mathbf{w}_{j}, \boldsymbol{\theta}\right)=\int_{-\infty}^{\infty} \Phi\left(-\frac{\mathbf{w}_{j} \gamma+\rho / \sigma \epsilon_{1}}{\sqrt{1-\rho^{2}}}\right) \phi\left(\epsilon_{1} / \sigma\right) d \epsilon_{1} \tag{2}
\end{equation*}
\]

The integrations in (1) and (2) have no closed form and must be approximated using Gauss-Hermite quadrature.

This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce (robust) and vce (cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.
heckpoisson also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

\section*{References}

Cameron, A. C., and P. K. Trivedi. 2013. Regression Analysis of Count Data. 2nd ed. New York: Cambridge University Press.
-. 2022. Microeconometrics Using Stata. 2nd ed. College Station, TX: Stata Press.
Greene, W. H. 2018. Econometric Analysis. 8th ed. New York: Pearson.
Terza, J. V. 1998. Estimating count data models with endogenous switching: Sample selection and endogenous treatment effects. Journal of Econometrics 84: 129-154. https://doi.org/10.1016/S0304-4076(97)00082-1.

\section*{Also see}
\([R]\) heckpoisson postestimation - Postestimation tools for heckpoisson
[R] heckman - Heckman selection model
\([R]\) heckoprobit - Ordered probit model with sample selection
[R] heckprobit - Probit model with sample selection
[R] poisson - Poisson regression
[CAUSAL] etpoisson - Poisson regression with endogenous treatment effects
[SVY] svy estimation - Estimation commands for survey data
[U] 20 Estimation and postestimation commands

\section*{Title}
heckpoisson postestimation - Postestimation tools for heckpoisson
\begin{tabular}{lll} 
Postestimation commands & predict & margins \\
Methods and formulas & Also see &
\end{tabular}

\section*{Postestimation commands}

The following postestimation commands are available after heckpoisson:
\begin{tabular}{|c|c|}
\hline Command & Description \\
\hline contrast & contrasts and ANOVA-style joint tests of estimates \\
\hline estat ic & Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) \\
\hline estat summarize & summary statistics for the estimation sample \\
\hline estat vce & variance-covariance matrix of the estimators (VCE) \\
\hline estat (svy) & postestimation statistics for survey data \\
\hline estimates & cataloging estimation results \\
\hline etable & table of estimation results \\
\hline * forecast & dynamic forecasts and simulations \\
\hline * hausman & Hausman's specification test \\
\hline lincom & point estimates, standard errors, testing, and inference for linear combinations of coefficients \\
\hline * 1 rtest & likelihood-ratio test \\
\hline margins & marginal means, predictive margins, marginal effects, and average marginal effects \\
\hline marginsplot & graph the results from margins (profile plots, interaction plots, etc.) \\
\hline nlcom & point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients \\
\hline predict & number of events, incidence rates, probabilities, etc. \\
\hline predictnl & point estimates, standard errors, testing, and inference for generalized predictions \\
\hline pwcompare & pairwise comparisons of estimates \\
\hline suest & seemingly unrelated estimation \\
\hline test & Wald tests of simple and composite linear hypotheses \\
\hline testnl & Wald tests of nonlinear hypotheses \\
\hline
\end{tabular}

\footnotetext{
*forecast, hausman, and lrtest are not appropriate with svy estimation results.
}

\section*{predict}

\section*{Description for predict}
predict creates new variables containing predictions such as number of events, incidence rates, conditional predicted number of events, probabilities, linear predictions, and equation-level scores.

\section*{Menu for predict}

\author{
Statistics \(>\) Postestimation
}

\section*{Syntax for predict}
```

predict [type] newvar [if] [in] [, statistic nooffset]
predict [type] stub* [if] [in], scores

```
statistic Description
Main
n
ir
ncond
\(\operatorname{pr}(n)\)
\(\operatorname{pr}(a, b)\)
psel
xb
xbsel
number of events; the default
incidence rate
predicted number of events conditional on \(y_{j}\) being observed
\(\operatorname{Pr}\left(y_{j}=n\right)\)
\(\operatorname{Pr}\left(a \leq y_{j} \leq b\right)\)
\(\operatorname{Pr}\left(y_{j}\right.\) observed \()\)
linear prediction
linear prediction for selection equation

These statistics are available both in and out of sample; type predict . . . if e(sample) ... if wanted only for the estimation sample.

\section*{Options for predict}
\(\qquad\) Main
\(\mathbf{n}\), the default, calculates the predicted number of events, which is \(\exp \left(\mathbf{x}_{j} \beta+\sigma^{2} / 2\right)\) if neither offset () nor exposure() was specified when the model was fit; is \(\exp \left(\mathbf{x}_{j} \beta+\sigma^{2} / 2+\right.\) offset \(\left._{j}\right)\) if offset() was specified; or is \(\exp \left(\mathbf{x}_{j} \beta+\sigma^{2} / 2\right) \times\) exposure \(_{i}\) if exposure() was specified.
ir calculates the incidence rate \(\exp \left(\mathbf{x}_{j} \beta+\sigma^{2} / 2\right)\), which is the predicted number of events when exposure is 1 . Specifying ir is equivalent to specifying \(n\) when neither offset() nor exposure() was specified when the model was fit.
ncond calculates the predicted number of events conditional on \(y_{j}\) being observed, which is \(\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}+\sigma^{2} / 2\right) \Phi\left(\mathbf{w}_{j} \gamma+\rho \sigma\right) / \Phi\left(\mathbf{w}_{j} \gamma\right)\).
\(\operatorname{pr}(n)\) calculates the probability \(\operatorname{Pr}\left(y_{j}=n\right)\), where \(n\) is a nonnegative integer that may be specified as a number or a variable.
\(\operatorname{pr}(a, b)\) calculates the probability \(\operatorname{Pr}\left(a \leq y_{j} \leq b\right)\), where \(a\) and \(b\) are nonnegative integers that may be specified as numbers or variables;
\(b\) missing ( \(b \geq\).) means \(+\infty\);
\(\operatorname{pr}(20,\).\() calculates \operatorname{Pr}\left(y_{j} \geq 20\right)\);
\(\operatorname{pr}(20, b)\) calculates \(\operatorname{Pr}\left(y_{j} \geq 20\right)\) in observations for which \(b \geq\). and calculates
\(\operatorname{Pr}\left(20 \leq y_{j} \leq b\right)\) elsewhere.
\(\operatorname{pr}(., b)\) produces a syntax error. A missing value in an observation of the variable \(a\) causes a missing value in that observation for \(\operatorname{pr}(a, b)\).
psel calculates the probability of selection (or being observed):
\(\operatorname{Pr}\left(y_{j}\right.\) observed \()=\operatorname{Pr}\left(\mathbf{w}_{j} \gamma+\epsilon_{2 j}>0\right)\)
xb calculates the linear prediction for the dependent count variable, which is \(\mathbf{x}_{j} \boldsymbol{\beta}\) if neither offset () nor exposure() was specified; \(\mathbf{x}_{j} \boldsymbol{\beta}+\operatorname{offset}_{j}^{\beta}\) if offset () was specified; or \(\mathbf{x}_{j} \boldsymbol{\beta}+\ln \left(\operatorname{exposure}_{j}\right)\) if exposure() was specified.
xbsel calculates the linear prediction for the selection equation, which is \(\mathbf{w}_{j} \gamma\) if offset () was not specified in select() and is \(\mathbf{w}_{j} \gamma+\operatorname{offset}_{j}^{\gamma}\) if offset() was specified in select().
nooffset is relevant only if you specified offset() or exposure() when you fit the model. It modifies the calculations made by predict so that they ignore the offset or exposure variable; the linear prediction is treated as \(\mathbf{x}_{j} \boldsymbol{\beta}\) rather than as \(\mathbf{x}_{j} \boldsymbol{\beta}+\) offset \(_{j}\) or \(\mathbf{x}_{j} \boldsymbol{\beta}+\ln \left(\right.\) exposure \(\left._{j}\right)\).
scores calculates equation-level score variables.
The first new variable will contain \(\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)\).
The second new variable will contain \(\partial \ln L / \partial\left(\mathbf{w}_{j} \gamma\right)\).
The third new variable will contain \(\partial \ln L / \partial \operatorname{atanh} \rho\).
The fourth new variable will contain \(\partial \ln L / \partial \ln \sigma\).

\section*{margins}

\section*{Description for margins}
margins estimates margins of response for number of events, incidence rates, conditional predicted number of events, probabilities, and linear predictions.

\section*{Menu for margins}

Statistics \(>\) Postestimation

\section*{Syntax for margins}
```

margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...) ...] [options]

```
statistic Description
\begin{tabular}{ll}
n & \begin{tabular}{l} 
number of events; the default \\
ir
\end{tabular} \\
\begin{tabular}{l} 
incidence rate
\end{tabular} \\
ncond & \(\operatorname{predicted~number~of~events~conditional~on~} y_{j}\) being observed \\
\(\mathrm{pr}(n, b)\) & \(\operatorname{Pr}\left(y_{j}=n\right)\) \\
psel & \(\operatorname{Pr}\left(a \leq y_{j} \leq b\right)\) \\
xb & \(\operatorname{Pr}\left(y_{j}\right.\) observed \()\) \\
xbsel & linear prediction \\
\hline
\end{tabular}

Statistics not allowed with margins are functions of stochastic quantities other than e(b). For the full syntax, see [R] margins.

\section*{Remarks and examples}
> Example 1: Obtaining margins for a count model with selection
In example 1 of \([R]\) heckpoisson, we fit a model for the number of patents. In that example, we are interested in the effect of \(R \& D\) expenditures on the number of patents received by a firm. We continue that example to determine the magnitude of the effect of \(R \& D\) expenditures on the number of patents and compare this effect for IT and non-IT sectors.

After reading in the data and fitting the model, we use margins to estimate the effect of an increase of a million dollars in \(R \& D\) expenditures (expenditure) on the number of patents (npatents) for firms in the IT and non-IT sectors (tech).

To do this, we use the at () option of margins. We use the observed values in our first scenario, so we tell margins to set expenditure equal to itself. For our second scenario, we tell margins to set expenditure equal to the observed value plus 1 because expenditures are measured in millions of dollars. We include the post option so that we can perform additional calculations later.


The output indicates that the expected number of patents for non-IT firms is about 1.28 compared with 2.29 for firms in the IT sector.

The second scenario shows the expected number of patents after our hypothetical increase in R\&D expenditures. In the non-IT sector, the expected number of patents received would be about 2.10 compared with 3.76 in the IT sector. It appears that increasing expenditures may have a larger effect for IT firms-the difference between the two scenarios is 1.47 for IT firms and only 0.82 for non-IT firms. We can test whether the effect of increasing expenditures is different for IT and non-IT firms. We use lincom to obtain an estimate of the difference in the differences between scenarios for the two sectors and a test of its significance. We ask for the differences by referring to the scenarios as 1. _at and 2. _at and by referring to the sector using the value that corresponds to the IT sector indicator, 1.tech for IT firms and 0.tech otherwise.
```

. lincom (_b[2._at\#1.tech] - _b[1._at\#1.tech]) -
> (_b[2._at\#0.tech] - _b[1._at\#0.tech])
( 1) 1bn._at\#Obn.tech - 1bn._at\#1.tech - 2._at\#Obn.tech + 2._at\#1.tech = 0

```
\begin{tabular}{r|rccccc}
\hline & Coefficient & Std. err. & \(z\) & \(\mathrm{P}>|\mathrm{z}|\) & {\([95 \%\) conf. interval] } \\
\hline\((1)\) & .6521006 & .0917299 & 7.11 & 0.000 & .4723134 & .8318878 \\
\hline
\end{tabular}

We find that the expected effect of increasing R\&D expenditures by one million dollars is 0.65 patents larger for IT firms than for non-IT firms, and this difference is significantly different from 0 .

\section*{Methods and formulas}

Suppose that the count outcome \(y_{j}\) has covariates \(\mathbf{x}_{j}\) and the selection outcome \(s_{j}\) has covariates \(\mathbf{w}_{j} . y_{j}\) is assumed to have a Poisson distribution, conditional on \(\mathbf{x}_{j}\), with conditional mean
\[
E\left(y_{j} \mid \mathbf{x}_{j}, \epsilon_{1 j}\right)=\mu_{j}=\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{1 j}\right)
\]
\(s_{j}\) is a binary outcome from a latent-variable model:
\[
s_{j}= \begin{cases}1, & \text { if } \mathbf{w}_{i} \gamma+\epsilon_{2 j}>0 \\ 0, & \text { otherwise }\end{cases}
\]

The expectation of \(y_{j}\) conditional on covariates \(\mathbf{x}_{j}\) for the whole population is
\[
E\left(y_{j} \mid \mathbf{x}_{j}\right)=\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}+\sigma^{2} / 2\right)
\]

Furthermore, if we want the expectation of \(y_{j}\) only if it was observed, then the formula is
\[
E\left(y_{j} \mid \mathbf{x}_{j}, \mathbf{w}_{j}, s_{j}=1\right)=\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}+\sigma^{2} / 2\right) \frac{\Phi\left(\mathbf{w}_{j} \gamma+\rho \sigma\right)}{\Phi\left(\mathbf{w}_{j} \gamma\right)}
\]

We note that if \(\rho=0\), this expectation is the same as its population version.
We can also predict the probability of \(y_{j}\) conditional on \(\mathbf{x}_{j}\). Note that although \(y_{j}\) is Poissondistributed conditional on \(\epsilon_{1}\) and \(\mathbf{x}_{j}\), the distribution of \(y_{j}\) is unknown unconditional on \(\epsilon_{1}\).
\[
\operatorname{Pr}\left(y_{j}=n \mid \mathbf{x}_{j}\right)=\int_{-\infty}^{\infty} \operatorname{Pr}\left(y_{j}=n \mid \mathbf{x}_{j}, \epsilon_{1}\right) \phi\left(\epsilon_{1} / \sigma\right) d \epsilon_{1}
\]

As in the implementation of log likelihood, we approximate this integral by Gauss-Hermite quadrature.

\section*{Also see}
[R] heckpoisson - Poisson regression with sample selection
[U] 20 Estimation and postestimation commands

\section*{Title}
heckprobit — Probit model with sample selection
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
heckprobit fits maximum-likelihood probit models with sample selection.

\section*{Quick start}

Probit model of \(y\) on \(x\) with sample selection indicated by binary variable selected and predicted by v
heckprobit \(y\) x, select (selected = v x)
Suppress iteration \(\log\)
heckprobit y x, select (selected \(=\mathrm{v}\) x) nolog
With cluster-robust standard errors for clustering by levels of cvar
heckprobit \(\mathrm{y} x\), select (selected \(=\mathrm{v} \mathrm{x}\) ) vce (cluster cvar)

\section*{Menu}

Statistics \(>\) Sample-selection models \(>\) Probit model with sample selection

\section*{Syntax}
```

heckprobit depvar indepvars [if] [in] [weight], select ([depvar $\left.{ }_{s}=\right]$ varlist $_{s}\left[\right.$, noconstant offset $\left(\right.$ varname $\left.\left._{o}\right)\right]$ ) [options]

```

\section*{options}

Description

\section*{Model}
* select ()

\section*{noconstant}
offset (varname)
constraints (constraints)
SE/Robust
vce (vcetype)

Reporting
level(\#)
first
lrmodel
nocnsreport
display_options

\section*{Maximization}
maximize_options
collinear
coeflegend or jackknife
specify selection equation: dependent and independent variables; whether to have constant term and offset variable suppress constant term
include varname in model with coefficient constrained to 1 apply specified linear constraints
vcetype may be oim, robust, cluster clustvar, opg, bootstrap,
set confidence level; default is level (95)
report first-step probit estimates
perform the likelihood-ratio model test instead of the default Wald test
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
control the maximization process; seldom used
keep collinear variables
display legend instead of statistics
*select () is required.
The full specification is select \(\left(\left[\right.\right.\) depvar \(\left._{s}=\right] \operatorname{varlist}_{s}\left[\right.\), noconstant offset \(\left(\right.\) varname \(\left.\left.\left._{o}\right)\right]\right)\). indepvars and varlist \({ }_{s}\) may contain factor variables; see [U] 11.4.3 Factor variables.
depvar, indepvars, depvar \({ }_{s}\), and varlist \(_{s}\) may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bayes, bootstrap, by, collect, fp, jackknife, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix
commands. For more details, see [BAYES] bayes: heckprobit.
Weights are not allowed with the bootstrap prefix; see \([\mathrm{R}]\) bootstrap.
vce(), first, lrmodel, and weights are not allowed with the svy prefix; see [SVY] svy.
pweights, fweights, and iweights are allowed; see [U] 11.1.6 weight.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options}

Model
select \(\left(\left[\right.\right.\) depvar \(\left._{s}=\right]\) varlist \(_{s}\left[\right.\), noconstant offset \(\left(\right.\) varname \(\left.\left.\left._{o}\right)\right]\right)\) specifies the variables and options for the selection equation. It is an integral part of specifying a selection model and is
required. The selection equation should contain at least one variable that is not in the outcome equation.
If depvar is specified, it should be coded as 0 or 1,0 indicating an observation not selected and 1 indicating a selected observation. If depvar \({ }_{s}\) is not specified, observations for which depvar is not missing are assumed selected, and those for which depvar is missing are assumed not selected. noconstant suppresses the selection constant term (intercept).
offset ( varname \(_{o}\) ) specifies that selection offset varname \({ }_{o}\) be included in the model with the coefficient constrained to be 1 .
noconstant, offset (varname), constraints (constraints); see [R] Estimation options.

\section*{SE/Robust}
vce(vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
level(\#); see [R] Estimation options.
first specifies that the first-step probit estimates of the selection equation be displayed before estimation.
lrmodel, nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap (\#), fvwrapon(style), cformat (\%fmt), pformat (\% fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate(\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from (init_specs); see [R] Maximize. These options are seldom used.

Setting the optimization type to technique (bhhh) resets the default vcetype to vce (opg).
The following options are available with heckprobit but are not shown in the dialog box:
collinear, coeflegend; see [R] Estimation options.

\section*{Remarks and examples}

The probit model with sample selection (Van de Ven and Van Pragg 1981) assumes that there exists an underlying relationship
\[
y_{j}^{*}=\mathbf{x}_{j} \boldsymbol{\beta}+u_{1 j} \quad \text { latent equation }
\]
such that we observe only the binary outcome
\[
y_{j}^{\text {probit }}=\left(y_{j}^{*}>0\right) \quad \text { probit equation }
\]

The dependent variable, however, is not always observed. Rather, the dependent variable for observation \(j\) is observed if
\[
y_{j}^{\text {select }}=\left(\mathbf{z}_{j} \gamma+u_{2 j}>0\right) \quad \text { selection equation }
\]
where
\[
\begin{gathered}
u_{1} \sim N(0,1) \\
u_{2} \sim N(0,1) \\
\operatorname{corr}\left(u_{1}, u_{2}\right)=\rho
\end{gathered}
\]

When \(\rho \neq 0\), standard probit techniques applied to the first equation yield biased results. heckprobit provides consistent, asymptotically efficient estimates for all the parameters in such models.

For the model to be well identified, the selection equation should have at least one variable that is not in the probit equation. Otherwise, the model is identified only by functional form, and the coefficients have no structural interpretation.

\section*{\(>\) Example 1}

We use the data from Pindyck and Rubinfeld (1998). In this dataset, the variables are whether children attend private school (private), number of years the family has been at the present residence (years), log of property tax (logptax), log of income (loginc), and whether one voted for an increase in property taxes (vote).

In this example, we alter the meaning of the data. Here we assume that we observe whether children attend private school only if the family votes for increasing the property taxes. This assumption is not true in the dataset, and we make it only to illustrate the use of this command.

We observe whether children attend private school only if the head of household voted for an increase in property taxes. We assume that the vote is affected by the number of years in residence, the current property taxes paid, and the household income. We wish to model whether children are sent to private school on the basis of the number of years spent in the current residence and the current property taxes paid.
. use https://www.stata-press.com/data/r18/school
. heckprobit private years logptax, select(vote=years loginc logptax)
Fitting probit model:
Iteration 0: Log likelihood \(=-17.122381\)
Iteration 1: Log likelihood \(=-16.243974\)
(output omitted )
Iteration 5: Log likelihood \(=-15.883655\)
Fitting selection model:
Iteration 0: Log likelihood \(=-63.036914\)
Iteration 1: Log likelihood \(=-58.534843\)
Iteration 2: Log likelihood \(=-58.497292\)
Iteration 3: Log likelihood \(=-58.497288\)
Comparison: Log likelihood = -74.380943
Fitting starting values:
Iteration 0: Log likelihood \(=-40.895684\)
Iteration 1: Log likelihood \(=-16.654497\) (output omitted)
Iteration 6: Log likelihood \(=-15.753765\)
Fitting full model:
\begin{tabular}{lclll} 
Iteration 0: & Log likelihood \(=-75.010619\) & (not concave) & \\
Iteration 1: & Log likelihood \(=-74.287758\) & & \\
Iteration 2: & Log likelihood \(=-74.250143\) & & \\
Iteration 3: & Log likelihood \(=-74.245088\) & & \\
Iteration 4: & Log likelihood \(=-74.244973\) & & \\
Iteration 5: & Log likelihood \(=-74.244973\) & & \\
Probit model with sample selection & Number of obs & \(=\) & 95 \\
& Selected & \(=\) & 59 \\
& Nonselected & \(=\) & 36 \\
& Wald chi2(2) & \(=\) & 1.04 \\
Log likelihood = -74.24497 & Prob \(>\) chi2 & \(=\) & 0.5935
\end{tabular}
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline & Coefficient & Std. err. & z & \(P>|z|\) & [95\% conf & interval] \\
\hline \multicolumn{7}{|l|}{private} \\
\hline years & -. 1142596 & . 1461715 & -0.78 & 0.434 & -. 4007505 & . 1722313 \\
\hline logptax & . 3516101 & 1.016483 & 0.35 & 0.729 & -1.64066 & 2.34388 \\
\hline _cons & -2.780667 & 6.905827 & -0.40 & 0.687 & -16.31584 & 10.75451 \\
\hline \multicolumn{7}{|l|}{vote} \\
\hline years & -. 0167511 & . 0147735 & -1.13 & 0.257 & -. 0457067 & . 0122045 \\
\hline loginc & . 9923023 & . 4430008 & 2.24 & 0.025 & . 1240368 & 1.860568 \\
\hline logptax & -1.278783 & . 5717545 & -2.24 & 0.025 & -2.399401 & -. 1581646 \\
\hline _cons & -. 5458205 & 4.070417 & -0.13 & 0.893 & -8.523692 & 7.432051 \\
\hline /athrho & -. 8663164 & 1.450017 & -0.60 & 0.550 & \(-3.708298\) & 1.975665 \\
\hline rho & -. 6994978 & . 7405281 & & & -. 9987983 & . 9622674 \\
\hline
\end{tabular}

The output shows several iteration logs. The first iteration log corresponds to running the probit model for those observations in the sample where we have observed the outcome. The second iteration log corresponds to running the selection probit model, which models whether we observe our outcome of interest. If \(\rho=0\), the sum of the log likelihoods from these two models will equal the log likelihood of the probit model with sample selection; this sum is printed in the iteration log as the comparison \(\log\) likelihood. The third iteration log shows starting values for the iterations.

The final iteration log is for fitting the full probit model with sample selection. A likelihood-ratio test of the log likelihood for this model and the comparison log likelihood is presented at the end of the output. If we had specified the vce (robust) option, this test would be presented as a Wald test instead of as a likelihood-ratio test.

\section*{> Example 2}

In example 1, we could have obtained robust standard errors by specifying the vce (robust) option. We do this here and also eliminate the iteration logs by using the nolog option:
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline \multicolumn{4}{|l|}{Probit model with sample selection} & \multicolumn{2}{|l|}{Number of obs Selected Nonselected} & 95
59
36 \\
\hline & & & & Wald ch & (2) & 2.55 \\
\hline \multicolumn{4}{|l|}{Log pseudolikelihood \(=-74.24497\)} & \multicolumn{2}{|l|}{Prob > chi2} & 0.2798 \\
\hline & Coefficient & Robust std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf & interval] \\
\hline \multicolumn{7}{|l|}{private} \\
\hline years & -. 1142596 & . 1113968 & -1.03 & 0.305 & -. 3325934 & . 1040741 \\
\hline logptax & . 3516101 & . 7358211 & 0.48 & 0.633 & -1.090573 & 1.793793 \\
\hline _cons & -2.780667 & 4.786652 & -0.58 & 0.561 & -12.16233 & 6.600998 \\
\hline \multicolumn{7}{|l|}{vote} \\
\hline years & -. 0167511 & . 0173344 & -0.97 & 0.334 & -. 0507259 & . 0172237 \\
\hline loginc & . 9923023 & . 4228042 & 2.35 & 0.019 & . 1636213 & 1.820983 \\
\hline logptax & -1.278783 & . 5095156 & -2.51 & 0.012 & -2.277415 & -. 2801506 \\
\hline _cons & -. 5458205 & 4.54389 & -0.12 & 0.904 & -9.451682 & 8.360041 \\
\hline /athrho & -. 8663164 & 1.63062 & -0.53 & 0.595 & -4.062272 & 2.329639 \\
\hline rho & -. 6994978 & . 8327621 & & & -. 9994078 & . 9812312 \\
\hline \multicolumn{7}{|l|}{Wald test of indep. eqns. (rho = 0) : chi2 (1) = 0.28 Prob > chi2 = 0.5952} \\
\hline
\end{tabular}

Regardless of whether we specify the vce (robust) option, the outcome is not significantly different from the outcome obtained by fitting the probit and selection models separately. This result is not surprising because the selection mechanism estimated was invented for the example rather than borne from any economic theory.

\section*{Stored results}
heckprobit stores the following in e():
Scalars
e(N)
e(N_selected)
e(N_nonselected)
e(k)
e(k_eq)
e(k_eq_model)
e(k_aux)
e(k_dv)
\(e\left(d f \_m\right)\)
e(11)
e(ll_0)
e(ll_c)
e(N_clust)
e(chi2)
e(chi2_c)
e (p)
e(p_c)
e(rho)
e(rank)
e(rankO)
e(ic)
e(rc)
e(converged)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(offset1)
e(offset2)
e(chi2type)
e(chi2_ct)
e(vce)
e(vcetype)
e(opt)
\(e\) (which)
e(ml_method)
e(user)
e(technique)
e(properties)
e(predict)
e(asbalanced)
e(asobserved)
Matrices
e(b)
e(Cns)
e(ilog)
e(gradient)
e(V)
e(V_modelbased)
Functions
e(sample)
number of observations
number of selected observations
number of nonselected observations
number of parameters
number of equations in \(e(b)\)
number of equations in overall model test
number of auxiliary parameters
number of dependent variables
model degrees of freedom
log likelihood
log likelihood, constant-only model
log likelihood, comparison model
number of clusters
\(\chi^{2}\)
\(\chi^{2}\) for comparison test
\(p\)-value for model test
\(p\)-value for comparison test
\(\rho\)
rank of \(e(V)\)
rank of \(e(V)\) for constant-only model
number of iterations
return code
1 if converged, 0 otherwise
heckprobit
command as typed
names of dependent variables
weight type
weight expression
title in estimation output
name of cluster variable
offset for regression equation
offset for selection equation
Wald or LR; type of model \(\chi^{2}\) test
type of comparison \(\chi^{2}\) test
vcetype specified in vce()
title used to label Std. err.
type of optimization
max or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement predict
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
constraints matrix
iteration \(\log\) (up to 20 iterations)
gradient vector
variance-covariance matrix of the estimators
model-based variance
marks estimation sample

In addition to the above, the following is stored in r() :
Matrices
\(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

\section*{Methods and formulas}

Van de Ven and Van Pragg (1981) provide an introduction and an explanation of this model.
The probit equation is
\[
y_{j}=\left(\mathbf{x}_{j} \boldsymbol{\beta}+u_{1 j}>0\right)
\]

The selection equation is
\[
\mathbf{z}_{j} \gamma+u_{2 j}>0
\]
where
\[
\begin{gathered}
u_{1} \sim N(0,1) \\
u_{2} \sim N(0,1) \\
\operatorname{corr}\left(u_{1}, u_{2}\right)=\rho
\end{gathered}
\]

The log likelihood is
\[
\begin{aligned}
\ln L=\sum_{\substack{j \in S \\
y_{j} \neq 0}} & w_{j} \ln \left\{\Phi_{2}\left(x_{j} \beta+\operatorname{offset}_{j}^{\beta}, z_{j} \gamma+\operatorname{offset}_{j}^{\gamma}, \rho\right)\right\} \\
& +\sum_{\substack{j \in S \\
y_{j}=0}} w_{j} \ln \left\{\Phi_{2}\left(-x_{j} \beta+\operatorname{offset}_{j}^{\beta}, z_{j} \gamma+\operatorname{offset}_{j}^{\gamma},-\rho\right)\right\} \\
& +\sum_{j \notin S} w_{j} \ln \left\{1-\Phi\left(z_{j} \gamma+\operatorname{offset}_{j}^{\gamma}\right)\right\}
\end{aligned}
\]
where \(S\) is the set of observations for which \(y_{j}\) is observed, \(\Phi_{2}(\cdot)\) is the cumulative bivariate normal distribution function (with mean \(\left[\begin{array}{ll}0 & 0\end{array}\right]^{\prime}\) ), \(\Phi(\cdot)\) is the standard cumulative normal, and \(w_{j}\) is an optional weight for observation \(j\).

In the maximum likelihood estimation, \(\rho\) is not directly estimated. Directly estimated is atanh \(\rho\) :
\[
\operatorname{atanh} \rho=\frac{1}{2} \ln \left(\frac{1+\rho}{1-\rho}\right)
\]

From the form of the likelihood, it is clear that if \(\rho=0\), the log likelihood for the probit model with sample selection is equal to the sum of the probit model for the outcome \(y\) and the selection model. We can perform a likelihood-ratio test by comparing the likelihood of the full model with the sum of the log likelihoods for the probit and selection models.

This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce(robust) and vce(cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.
heckprobit also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

\section*{References}

Baum, C. F. 2006. An Introduction to Modern Econometrics Using Stata. College Station, TX: Stata Press.
Chiburis, R., and M. Lokshin. 2007. Maximum likelihood and two-step estimation of an ordered-probit selection model. Stata Journal 7: 167-182.

De Luca, G. 2008. SNP and SML estimation of univariate and bivariate binary-choice models. Stata Journal 8: 190-220.

De Luca, G., and V. Perotti. 2011. Estimation of ordered response models with sample selection. Stata Journal 11: 213-239.

Heckman, J. J. 1979. Sample selection bias as a specification error. Econometrica 47: 153-161. https://doi.org/10.2307/1912352.
Lokshin, M., and Z. Sajaia. 2011. Impact of interventions on discrete outcomes: Maximum likelihood estimation of the binary choice models with binary endogenous regressors. Stata Journal 11: 368-385.

Miranda, A., and S. Rabe-Hesketh. 2006. Maximum likelihood estimation of endogenous switching and sample selection models for binary, ordinal, and count variables. Stata Journal 6: 285-308.
Muro, J., C. Suárez, and M. Zamora. 2010. Computing Murphy-Topel-corrected variances in a heckprobit model with endogeneity. Stata Journal 10: 252-258.

Pindyck, R. S., and D. L. Rubinfeld. 1998. Econometric Models and Economic Forecasts. 4th ed. New York: McGraw-Hill.

Van de Ven, W. P. M. M., and B. M. S. Van Pragg. 1981. The demand for deductibles in private health insurance: A probit model with sample selection. Journal of Econometrics 17: 229-252. https://doi.org/10.1016/0304-4076(81)90028-2.

\section*{Also see}
\([\mathrm{R}]\) heckprobit postestimation - Postestimation tools for heckprobit
[R] heckman - Heckman selection model
[R] heckoprobit - Ordered probit model with sample selection
[R] heckpoisson - Poisson regression with sample selection
[R] probit - Probit regression
[BAYES] bayes: heckprobit - Bayesian probit model with sample selection
[CAUSAL] etregress - Linear regression with endogenous treatment effects
[ERM] eprobit - Extended probit regression
[SVY] svy estimation - Estimation commands for survey data
[U] 20 Estimation and postestimation commands

\section*{Title}
heckprobit postestimation - Postestimation tools for heckprobit
Postestimation commands predict margins Remarks and examples

\section*{Postestimation commands}

The following postestimation commands are available after heckprobit:
\begin{tabular}{|c|c|}
\hline Command & Description \\
\hline contrast & contrasts and ANOVA-style joint tests of estimates \\
\hline estat ic & Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) \\
\hline estat summarize & summary statistics for the estimation sample \\
\hline estat vce & variance-covariance matrix of the estimators (VCE) \\
\hline estat (svy) & postestimation statistics for survey data \\
\hline estimates & cataloging estimation results \\
\hline etable & table of estimation results \\
\hline * hausman & Hausman's specification test \\
\hline lincom & point estimates, standard errors, testing, and inference for linear combinations of coefficients \\
\hline * lrtest & likelihood-ratio test \\
\hline margins & marginal means, predictive margins, marginal effects, and average marginal effects \\
\hline marginsplot & graph the results from margins (profile plots, interaction plots, etc.) \\
\hline nlcom & point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients \\
\hline predict & probabilities, linear predictions and their SEs, etc. \\
\hline predictnl & point estimates, standard errors, testing, and inference for generalized predictions \\
\hline pwcompare & pairwise comparisons of estimates \\
\hline suest & seemingly unrelated estimation \\
\hline test & Wald tests of simple and composite linear hypotheses \\
\hline testnl & Wald tests of nonlinear hypotheses \\
\hline
\end{tabular}

\footnotetext{
*hausman and lrtest are not appropriate with svy estimation results.
}

\section*{predict}

\section*{Description for predict}
predict creates a new variable containing predictions such as probabilities, linear predictions, and standard errors.

\section*{Menu for predict}

\author{
Statistics \(>\) Postestimation
}

\section*{Syntax for predict}
```

predict [type] newvar [if] [in] [, statistic nooffset]
predict [type] stub* [if] [in], scores

```
statistic Description

Main
\begin{tabular}{ll}
\(\underline{\text { pmargin }}\) & \(\Phi\left(\mathbf{x}_{j} \mathbf{b}\right)\), success probability; the default \\
p11 & \(\Phi_{2}\left(\mathbf{x}_{j} \mathbf{b}, \mathbf{z}_{j} \mathbf{g}, \rho\right)\), predicted probability \(\operatorname{Pr}\left(y_{j}^{\text {probit }}=1, y_{j}^{\text {select }}=1\right)\) \\
p10 & \(\Phi_{2}\left(\mathbf{x}_{j} \mathbf{b},-\mathbf{z}_{j} \mathbf{g},-\rho\right)\), predicted probability \(\operatorname{Pr}\left(y_{j}^{\text {probit }}=1, y_{j}^{\text {select }}=0\right)\) \\
p01 & \(\Phi_{2}\left(-\mathbf{x}_{j} \mathbf{b}, \mathbf{z}_{j} \mathbf{g},-\rho\right)\), predicted probability \(\operatorname{Pr}\left(y_{j}^{\text {probit }}=0, y_{j}^{\text {select }}=1\right)\) \\
p00 & \(\Phi_{2}\left(-\mathbf{x}_{j} \mathbf{b},-\mathbf{z}_{j} \mathbf{g}, \rho\right)\), predicted probability \(\operatorname{Pr}\left(y_{j}^{\text {probit }}=0, y_{j}^{\text {select }}=0\right)\) \\
psel & \(\Phi\left(\mathbf{z}_{j} \mathbf{g}\right)\), selection probability \\
\(\underline{\text { pcond }}\) & \(\Phi_{2}\left(\mathbf{x}_{j} \mathbf{b}, \mathbf{z}_{j} \mathbf{g}, \rho\right) / \Phi\left(\mathbf{z}_{j} \mathbf{g}\right)\), probability of success conditional on selection \\
xb & linear prediction \\
stdp & standard error of the linear prediction \\
\(\underline{\text { xbsel }}\) & \begin{tabular}{l} 
linear prediction for selection equation \\
stdpsel
\end{tabular} \\
\hline
\end{tabular}
\(\Phi(\cdot)\) is the standard normal distribution function, and \(\Phi_{2}(\cdot)\) is the bivariate normal distribution function.
These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

\section*{Options for predict}

Main pmargin, the default, calculates the univariate (marginal) predicted probability of success \(\operatorname{Pr}\left(y_{j}^{\text {probit }}=1\right)\).
p11 calculates the bivariate predicted probability \(\operatorname{Pr}\left(y_{j}^{\text {probit }}=1, y_{j}^{\text {select }}=1\right)\).
p10 calculates the bivariate predicted probability \(\operatorname{Pr}\left(y_{j}^{\text {probit }}=1, y_{j}^{\text {select }}=0\right)\). p01 calculates the bivariate predicted probability \(\operatorname{Pr}\left(y_{j}^{\text {probit }}=0, y_{j}^{\text {select }}=1\right)\).
p00 calculates the bivariate predicted probability \(\operatorname{Pr}\left(y_{j}^{\text {probit }}=0, y_{j}^{\text {select }}=0\right)\).
psel calculates the univariate (marginal) predicted probability of selection \(\operatorname{Pr}\left(y_{j}^{\text {select }}=1\right)\).
pcond calculates the conditional (on selection) predicted probability of success
\[
\operatorname{Pr}\left(y_{j}^{\text {probit }}=1, y_{j}^{\text {select }}=1\right) / \operatorname{Pr}\left(y_{j}^{\text {select }}=1\right) .
\]
xb calculates the probit linear prediction \(\mathbf{x}_{j} \mathbf{b}\).
stdp calculates the standard error of the prediction, which can be thought of as the standard error of the predicted expected value or mean for the observation's covariate pattern. The standard error of the prediction is also referred to as the standard error of the fitted value.
xbsel calculates the linear prediction for the selection equation.
stdpsel calculates the standard error of the linear prediction for the selection equation.
scores calculates equation-level score variables.
The first new variable will contain \(\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)\).
The second new variable will contain \(\partial \ln L / \partial\left(\mathbf{z}_{j} \gamma\right)\).
The third new variable will contain \(\partial \ln L / \partial(\operatorname{atanh} \rho)\).
nooffset is relevant only if you specified offset (varname) for heckprobit. It modifies the calculations made by predict so that they ignore the offset variable; the linear prediction is treated as \(\mathbf{x}_{j} \mathbf{b}\) rather than as \(\mathbf{x}_{j} \mathbf{b}+\) offset \(_{j}\).

\section*{margins}

\section*{Description for margins}
margins estimates margins of response for probabilities and linear predictions.

\section*{Menu for margins}

Statistics \(>\) Postestimation

\section*{Syntax for margins}
```

margins [marginlist] [, options]

```

statistic Description
\begin{tabular}{ll} 
pmargin & \(\Phi\left(\mathbf{x}_{j} \mathbf{b}\right)\), success probability; the default \\
p11 & \(\Phi_{2}\left(\mathbf{x}_{j} \mathbf{b}, \mathbf{z}_{j} \mathbf{g}, \rho\right)\), predicted probability \(\operatorname{Pr}\left(y_{j}^{\text {probit }}=1, y_{j}^{\text {select }}=1\right)\) \\
p10 & \(\Phi_{2}\left(\mathbf{x}_{j} \mathbf{b},-\mathbf{z}_{j} \mathbf{g},-\rho\right)\), predicted probability \(\operatorname{Pr}\left(y_{j}^{\text {probit }}=1, y_{j}^{\text {select }}=0\right)\) \\
p01 & \(\Phi_{2}\left(-\mathbf{x}_{j} \mathbf{b}, \mathbf{z}_{j} \mathbf{g},-\rho\right)\), predicted probability \(\operatorname{Pr}\left(y_{j}^{\text {probit }}=0, y_{j}^{\text {select }}=1\right)\) \\
p00 & \(\Phi_{2}\left(-\mathbf{x}_{j} \mathbf{b},-\mathbf{z}_{j} \mathbf{g}, \rho\right)\), predicted probability \(\operatorname{Pr}\left(y_{j}^{\text {probit }}=0, y_{j}^{\text {select }}=0\right)\) \\
psel & \(\Phi\left(\mathbf{z}_{j} \mathbf{g}\right)\), selection probability \\
\(\overline{\text { pcond }}\) & \(\Phi_{2}\left(\mathbf{x}_{j} \mathbf{b}, \mathbf{z}_{j} \mathbf{g}, \rho\right) / \Phi\left(\mathbf{z}_{j} \mathbf{g}\right)\), probability of success conditional on selection \\
xb & linear prediction \\
\(\underline{\text { xbsel }}\) & linear prediction for selection equation \\
stdp & not allowed with margins \\
stdpsel & not allowed with margins
\end{tabular}

Statistics not allowed with margins are functions of stochastic quantities other than e(b). For the full syntax, see \([R]\) margins.

\section*{Remarks and examples}
> Example 1
It is instructive to compare the marginal predicted probabilities with the predicted probabilities that we would obtain by ignoring the selection mechanism. To compare the two approaches, we will synthesize data so that we know the "true" predicted probabilities.

First, we need to generate correlated error terms, which we can do using a standard Cholesky decomposition approach. For our example, we will clear any data from memory and then generate errors that have a correlation of 0.5 by using the following commands. We set the seed so that interested readers can type in these same commands and obtain the same results.
```

. set seed 12309
. set obs 5000
Number of observations (_N) was 0, now 5,000.
. generate c1 = rnormal()
. generate c2 = rnormal()
. matrix P = (1,.5\.5,1)
. matrix A = cholesky(P)
. local fac1 = A[2,1]
. local fac2 = A[2,2]
. generate u1 = c1
. generate u2 = 'fac1'*c1 + 'fac2'*c2

```

We can check that the errors have the correct correlation by using the correlate command. We will also normalize the errors so that they have a standard deviation of one, so we can generate a bivariate probit model with known coefficients. We do that with the following commands:
```

. correlate u1 u2
(obs=5,000)

|  | u1 | u2 |
| ---: | ---: | ---: |
| u1 | 1.0000 |  |
| u2 | 0.5012 | 1.0000 |

. summarize u1
(output omitted)
. replace u1 = u1/r(sd)
(5,000 real changes made)
. summarize u2
(output omitted)
. replace u2 = u2/r(sd)
(5,000 real changes made)
. drop c1 c2
. generate x1 = runiform()-.5
. generate x2 = runiform()+1/3
. generate y1s = 0.5 + 4*x1 + u1
. generate y2s = 3-3*x2 + . 5*x1 + u2
. generate y1 = (y1s>0)
. generate y2 = (y2s>0)

```

We have now created two dependent variables, y1 and y2, which are defined by our specified coefficients. We also included error terms for each equation, and the error terms are correlated. We run heckprobit to verify that the data have been correctly generated according to the model
\[
\begin{aligned}
& y_{1}=.5+4 x_{1}+u_{1} \\
& y_{2}=3+.5 x_{1}-3 x_{2}+u_{2}
\end{aligned}
\]
where we assume that \(y_{1}\) is observed only if \(y_{2}=1\).
\begin{tabular}{lrlr}
. heckprobit y1 x1, sel(y2 = x1 x2) nolog & & \\
Probit model with sample selection & Number of obs & \(=\) & 5,000 \\
& Selected & \(=\) & 3,182 \\
& Nonselected & \(=\) & 1,818 \\
& Wald chi2(1) & \(=\) & 947.76 \\
Log likelihood \(=-3612.401\) & Prob \(>\) chi2 & \(=\) & 0.0000
\end{tabular}


Now that we have verified that we have generated data according to a known model, we can obtain and then compare predicted probabilities from the probit model with sample selection and a (usual) probit model.
```

. predict pmarg
(option pmargin assumed; Pr(y1=1))
. probit y1 x1 if y2==1
(output omitted)
. predict phat
(option pr assumed; Pr(y1))

```

Using the (marginal) predicted probabilities from the probit model with sample selection (pmarg) and the predicted probabilities from the (usual) probit model (phat), we can also generate the "true" predicted probabilities from the synthesized y1s variable and then compare the predicted probabilities:
. generate ptrue = normal (y1s)
. summarize pmarg ptrue phat
\begin{tabular}{r|rrrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline pmarg & 5,000 & .6089004 & .3249993 & .0632337 & .99354 \\
ptrue & 5,000 & .5967872 & .3534232 & \(2.78 \mathrm{e}-07\) & 1 \\
phat & 5,000 & .6588519 & .3113716 & .0910951 & .997021
\end{tabular}

Here we see that ignoring the selection mechanism (comparing the phat variable with the true ptrue variable) results in predicted probabilities that are much higher than the true values. Looking at the marginal predicted probabilities from the model with sample selection, however, results in more accurate predictions.

\section*{Also see}
[R] heckprobit - Probit model with sample selection
[U] 20 Estimation and postestimation commands

\section*{Title}
help - Display help in Stata
\begin{tabular}{llll} 
Description & Menu & Syntax & Options \\
Remarks and examples & Also see & &
\end{tabular}

\section*{Description}

The help command displays help information about the specified command or topic. help launches a new Viewer to display help for the specified command or topic or displays help on the console in Stata for Unix(console). If help is not followed by a command or a topic name, Stata displays advice for using the help system and documentation.

\section*{Menu}

Help > Stata command...

\section*{Syntax}
help [command_or_topic_name] [, nonew name(viewername) marker(markername)]

\section*{Options}
nonew specifies that a new Viewer window not be opened for the help topic if a Viewer window is already open. The default is for a new Viewer window to be opened each time help is typed so that multiple help files may be viewed at once. nonew causes the help file to be displayed in the topmost open Viewer.
name (viewername) specifies that help be displayed in a Viewer window named viewername. If the named window already exists, its contents will be replaced. If the named window does not exist, it will be created.
marker (markername) specifies that the help file be opened to the position of markername within the help file.

\section*{Remarks and examples}

To obtain help for any Stata command, type help command or select Help > Stata command... and fill in command.
help is best explained by examples.
\begin{tabular}{lll} 
To obtain help for ... & type & \\
regress & help regress & \\
postestimation tools for regress & help regress postestimation or \\
& help regress post \\
graph option xlabel() & help graph xlabel() \\
Stata function \(\operatorname{strpos}()\) & help strpos () \\
Mata function optimize() & help mata optimize()
\end{tabular}

Tips:
- help displays advice for using the help system and documentation.
- help guide displays a table of contents for basic Stata concepts.
- help estimation commands displays an alphabetical listing of all Stata estimation commands.
- help functions displays help on Stata functions by category.
- help mata functions displays a subject table of contents for Mata's functions.
- help ts glossary displays the glossary for the time-series manual, and similarly for the other Stata specialty manuals.

If you type help topic and help for topic is not found, Stata will automatically perform a search for topic.

For instance, try typing help forecasting. A forecasting help file is not found, so Stata executes search forecasting and displays the results in the Viewer.

See [U] 4 Stata's help and search facilities for a complete description of how to use help.

\section*{- Technical note}

When you type help topic, Stata first looks along the adopath for topic. sthlp; see [U] 17.5 Where does Stata look for ado-files?.

\section*{Video examples}

Quick help in Stata

\section*{Also see}
\([R]\) net search - Search the Internet for installable packages
[R] search - Search Stata documentation and other resources
[GSM] 4 Getting help
[GSW] 4 Getting help
[GSU] 4 Getting help
[U] 4 Stata's help and search facilities

\section*{Title}
hetoprobit - Heteroskedastic ordered probit regression
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
hetoprobit fits a heteroskedastic ordered probit model for an ordinal dependent variable. hetoprobit is a generalization of oprobit that allows the variance to be modeled as a function of independent variables and to differ between subjects or groups in the population.

\section*{Quick start}

Heteroskedastic ordinal probit model of y on x 1 , using x 2 to model the variance hetoprobit y x1, het(x2)

With robust standard errors
```

hetoprobit y x1, het(x2) vce(robust)

```

Perform a Wald test on the variance instead of a likelihood-ratio (LR) test hetoprobit \(y\) x1, het(x2) waldhet

\section*{Menu}

Statistics \(>\) Ordinal outcomes \(>\) Heteroskedastic ordered probit regression

\section*{Syntax}
```

hetoprobit depvar [indepvars] [if] [in] [weight],
$\operatorname{het}\left(\right.$ varlist $\left[\right.$, offset $\left(\right.$ varname $\left.\left._{o}\right)\right]$ ) [options $]$

```
options
Model
* \(\operatorname{het}(v a r l i s t[. .]\).
offset (varname)
constraints (constraints)
SE/Robust
vce (vcetype)

Reporting
level(\#)
waldhet
nocnsreport
display_options

Description
independent variables to model the variance and optional offset variable
include varname in model with coefficient constrained to 1
apply specified linear constraints
vcetype may be oim, robust, cluster clustvar, opg, bootstrap, or jackknife
set confidence level; default is level (95)
perform Wald test on variance instead of LR test
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

Maximization
maximize_options
noheader
notable
collinear
coeflegend
control the maximization process; seldom used
do not display header above coefficient table
do not display coefficient table
keep collinear variables
display legend instead of statistics
\({ }^{*} \operatorname{het}()\) is required. The full specification is het (varlist \(\left.\left.[\text {, offset (varname })_{o}\right)\right]\) ).
indepvars and varlist may contain factor variables; see [U] 11.4.3 Factor variables.
depvar, indepvars, and varlist may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bayes, bootstrap, by, collect, fp, jackknife, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: hetoprobit.
Weights are not allowed with the bootstrap prefix; see \([R]\) bootstrap.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
noheader, notable, collinear, and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options}
```

Model
het(varlist [, offset (varnameo) ]) specifies the independent variables and, optionally, the offset
variable in the variance function. het() is required.
offset (varname }\mp@subsup{)}{0}{}\mathrm{ ) specifies that offset varname }\mp@subsup{O}{0}{}\mathrm{ be included in the variance model with the
coefficient constrained to be 1.
offset(varname), constraints(constraints); see [R] Estimation options.

```
SE/Robust
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.

Reporting
level (\#); see [R] Estimation options.
waldhet specifies that a Wald test of whether lnsigma \(=0\) be performed instead of the LR test. nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap (\#), fvwrapon(style), cformat (\%fmt), pformat (\% fmt), sformat (\% \(\% m t\) ), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate (\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from(init_specs); see [R] Maximize. These options are seldom used.

The following options are available with hetoprobit but are not shown in the dialog box: noheader suppresses the header above the coefficient table. notable suppresses the display of the coefficient table. collinear, coeflegend; see [R] Estimation options.

\section*{Remarks and examples}
hetoprobit fits a maximum-likelihood heteroskedastic ordered probit model, which is a generalization of the ordered probit model (see [R] oprobit).

In ordinal regression models, the outcome is an ordinal variable-a variable that is categorical and ordered, for instance, "poor", "good", and "excellent". The specific values of the ordinal variable are irrelevant. It matters only that larger values are assumed to correspond to "higher" outcomes. To simplify the discussion in this entry, we assume without loss of generality that the dependent variable takes on the integer values \(0,1, \ldots, H\), for some value \(H>1\).

In ordered probit models, an underlying score is estimated as a linear function of the independent variables and a set of cutpoints. The probability of observing outcome \(y_{j}=h\), where \(h=0,1, \ldots, H\), corresponds to the probability that the value of the linear function, plus random error, is within the range of the cutpoints associated with the outcome
\[
\begin{aligned}
\operatorname{Pr}\left(y_{j}=h\right) & =\operatorname{Pr}\left(\kappa_{h}<\beta_{1} x_{1 j}+\beta_{2} x_{2 j}+\cdots+\beta_{k} x_{k j}+u_{j} \leq \kappa_{h+1}\right) \\
& =\Phi\left(\kappa_{h+1}-\mathbf{x}_{j} \boldsymbol{\beta}\right)-\Phi\left(\kappa_{h}-\mathbf{x}_{j} \boldsymbol{\beta}\right)
\end{aligned}
\]
where \(\mathbf{x}_{j}=\left(x_{1 j}, x_{2 j}, \ldots, x_{k j}\right)\) are the \(k\) independent variables that model the mean function; \(\boldsymbol{\beta}\) is a column vector of unknown parameters in the mean function; \(u_{j}\), where \(j=1, \ldots, N\), are normally distributed error terms; \(\kappa_{h}\), where \(h=1, \ldots, H\), are the unknown cutpoints that separate the different possible values of \(h\); and \(\Phi(\cdot)\) is the cumulative distribution function of the standard normal distribution. Also, by convention, to complete the intervals for the lowest and highest values of the outcome, \(\kappa_{0}=-\infty\) and \(\kappa_{H+1}=\infty\).

In conventional ordinal probit models, the error term is assumed i.i.d. normal with unit variance for all observations. hetoprobit generalizes the ordered probit model by representing the variance of the error term \(u_{j}\) as a multiplicative function of explanatory variables \(\mathbf{z}_{j}=\left(z_{1 j}, z_{2 j}, \ldots, z_{m j}\right)\). This approach was introduced by Harvey (1976), though we depart from Harvey slightly by modeling standard deviation rather than variance. More specifically, we model the natural logarithm of the standard deviation as a linear combination of the explanatory variables,
\[
\ln \sigma_{j}=\mathbf{z}_{j} \gamma
\]
where \(\gamma\) is a column vector of unknown parameters in the variance function.
With this generalization, the error variance may differ between subjects or between groups in the population, and
\[
\operatorname{Pr}\left(y_{j}=h\right)=\Phi\left\{\frac{\kappa_{h+1}-\mathbf{x}_{j} \boldsymbol{\beta}}{\exp \left(\mathbf{z}_{j} \boldsymbol{\gamma}\right)}\right\}-\Phi\left\{\frac{\kappa_{h}-\mathbf{x}_{j} \boldsymbol{\beta}}{\exp \left(\mathbf{z}_{j} \boldsymbol{\gamma}\right)}\right\}
\]

For the model to be identifiable, there can be no constant term in \(\mathbf{z}_{j} \gamma\). Also, as with \([\mathrm{R}]\) oprobit, there is no constant term in \(\mathbf{x}_{j} \boldsymbol{\beta}\). The role of the constant is subsumed by the cutpoints.

We estimate the coefficients \(\beta_{1}, \beta_{2}, \ldots, \beta_{k}\) and \(\gamma_{1}, \gamma_{2}, \ldots, \gamma_{m}\) together with the cutpoints \(\kappa_{1}\), \(\kappa_{2}, \ldots, \kappa_{H}\). If the model has no independent variables in \(\mathbf{x}_{j}\), only the cutpoints and the \(\gamma\) parameters are estimated.

Modeling of heteroskedastic variance has both constructive and defensive uses. It is known that differences in variance between subjects or between groups in the population can cause biased coefficient estimates and can complicate comparison of distinct groups. Thus, incorporating a model for variance can be necessary for proper inference, even if the variance function itself is not a topic of interest to the researcher. For discussion, see Williams (2010) and the references cited therein. There are also cases where modeling the differences between variances of different subjects or different groups in the population is one of the principal purposes of the study. We will discuss such a scenario in the examples below. See Reardon et al. (2017) and Alvarez and Brehm (1995) for additional examples.

\section*{> Example 1: Modeling heteroskedasticity of reported health status}

In this example, we will use a slightly modified subset of data from the 2015 Eating \& Health Module of the American Time Use Survey (atuS), conducted by the US Bureau of Labor Statistics. Our analysis will not account for the survey design. The atUS measures the amount of time people spend doing various activities, such as working, caring for children, volunteering, and socializing. Of interest to us is an ordinal response variable, health, which contains individuals' self-assessments of their overall health status on a five-point scale: 1 for "poor", 2 for "fair", 3 for "good", 4 for "very good", and 5 for "excellent".

We want to examine the role that age and other factors play in an individual's self-assessment of health. Age is a natural variable to include when modeling mean or typical health status. But we also suspect that the variation in health status is greater in an older population, as compared with a youthful population, which consists mainly of healthy individuals. If our suspicion is true, quantifying the relationship between variation in health status and age may have value, for example, in planning a healthcare strategy that is appropriately tailored both for the older Medicare population and for a younger cohort.

Thus, we will use hetoprobit to model heteroskedasticity induced by age. In modeling the variance term, in addition to age, we will include a factor variable, exercise, which indicates whether or not an individual exercised during the previous week. For purposes of illustration, imagine that we are not interested in exercise as a topic in its own right, but we are concerned that health variability among those who exercise may differ from the variability among those who do not. Therefore, we include exercise in the variance term to help insulate our estimation results against a possible hidden bias.

Our model will include three explanatory variables for the mean function: age, bmi (body mass index), and exercise.
```

. use https://www.stata-press.com/data/r18/eathealth15
(2015 ATUS Eating \& Health Module extract)
. hetoprobit health age bmi i.exercise, het(age i.exercise)
(output omitted)

```
Fitting ordered probit model:
Iteration 0: Log likelihood \(=-2905.7943\)
Iteration 1: Log likelihood \(=-2717.2752\)
Iteration 2: Log likelihood \(=-2716.9679\)
Iteration 3: Log likelihood \(=-2716.9679\)
Fitting full model:
Iteration 0: Log likelihood \(=-2716.9679\)
Iteration 1: Log likelihood \(=-2708.6752\)
Iteration 2: Log likelihood \(=-2708.5492\)
Iteration 3: Log likelihood = -2708.5491
Heteroskedastic ordered probit regression Number of obs \(=2,009\)
Log likelihood = -2708.5491
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline health & Coefficient & Std. err. & z & \(P>|z|\) & [95\% con & interval] \\
\hline \multicolumn{7}{|l|}{health} \\
\hline age & -. 0083348 & . 0015969 & -5.22 & 0.000 & -. 0114646 & -. 005205 \\
\hline bmi & -. 0564072 & . 0057392 & -9.83 & 0.000 & -. 0676558 & -. 0451586 \\
\hline exercise & & & & & & \\
\hline Yes & . 6493794 & . 0732137 & 8.87 & 0.000 & . 5058833 & . 7928755 \\
\hline \multicolumn{7}{|l|}{Insigma} \\
\hline age & . 0041401 & . 0011611 & 3.57 & 0.000 & . 0018643 & . 0064159 \\
\hline \multicolumn{7}{|l|}{exercise} \\
\hline Yes & -. 0773166 & . 0423038 & -1.83 & 0.068 & -. 1602305 & . 0055973 \\
\hline /cut1 & -3.903773 & . 2913163 & & & -4.474742 & -3.332803 \\
\hline /cut2 & -2.776111 & . 2262442 & & & -3.219541 & -2. 33268 \\
\hline /cut3 & -1.576396 & . 174352 & & & -1.918119 & -1. 234672 \\
\hline /cut4 & -. 4189882 & . 1524084 & & & -. 7177031 & -. 1202733 \\
\hline
\end{tabular}

LR test of lnsigma=0: chi2(2) \(=16.84\)
Prob > chi2 \(=0.0002\)
The LR test at the bottom of the output is a test of homogeneity of the variance function. The \(\chi^{2}(2)\) statistic of 16.84 is significant, indicating that heteroskedasticity is present. If you prefer the Wald test for heteroskedasticity, you can specify the waldhet option. The coefficients for the variance function are reported in the section of the table labeled insigma. Our results indicate that age is a significant contributor to the variance function but that exercise is not significant at a 0.05 level.

The LR test for the model that appears above the coefficient table is a joint test for inclusion of age, bmi, and exercise in the mean function. The null model for this test is the model consisting only of cutpoints and the heteroskedastic term. Coefficients for the mean function are reported in the section of the table labeled health. In this example, age, bmi, and exercise are significant components of the linear predictor of the mean.

The signs of the coefficients in the fitted model are directly interpretable. For example, the negative value for the coefficient of bmi implies that higher values of bmi predict lower values of health status. However, because of the probit link and the fact that we estimate variance with a log transformation, the numerical relationships between the coefficients of the model and the outcome variables are nonlinear. Postestimation commands recognize and account for these nonlinearities.

\section*{> Example 2: Predict the probability of a poor health rating}

Ordered probit models allow us to look at the probabilities of different outcomes of interest. Suppose we are interested in predictions of a reported health status of "poor" (health \(=1\) ) and how it differs across levels of bmi. First, we obtain the predicted probability of poor health.
```

. predict pr1, pr outcome(1)

```

We can now visualize how the predicted probability of poor health status differs across the range of bmi values in our sample.
```

. twoway scatter pr1 bmi, ytitle("Pr(poor health)")
> title("Effect of BMI on probability of poor health")

```


We see that predicted probabilities of poor health increase as body mass index increases.
> Example 3: Predictive margins and average marginal effect
The graph above plots the predicted probability of poor health for each individual in our dataset. We may also want to evaluate how the average predicted probability changes across levels of the covariates in the model. For instance, we can use the margins command to obtain the expected probability of having poor health across a range of ages.
```

. margins, at(age = (30(10)70)) predict(outcome(1))
Predictive margins Number of obs = 2,009
Model VCE: OIM
Expression: Pr(health==1), predict(outcome(1))
1._at: age = 30
2._at: age = 40
3._at: age = 50
4._at: age = 60
5._at: age = 70

```
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline & \multicolumn{3}{|c|}{Delta-method} & \(P>|z|\) & \multicolumn{2}{|l|}{[95\% conf. interval]} \\
\hline _at & & & & & & \\
\hline 1 & . 0225765 & . 0037522 & 6.02 & 0.000 & . 0152224 & . 0299306 \\
\hline 2 & . 0299079 & . 0038887 & 7.69 & 0.000 & . 0222862 & . 0375297 \\
\hline 3 & . 0388244 & . 0041936 & 9.26 & 0.000 & . 0306051 & . 0470438 \\
\hline 4 & . 0494246 & . 0049748 & 9.93 & 0.000 & . 0396742 & . 0591751 \\
\hline 5 & . 0617532 & . 0064443 & 9.58 & 0.000 & . 0491226 & . 0743839 \\
\hline
\end{tabular}

Based on our model, what would we expect if everyone was 30 years old but had the same distributions of bmi and exercise that we observed in our data? The first line in this table reports that the average predicted probability of poor health is 0.0226 in this case. The second line shows the average predicted probability of poor health if we set age \(=40\), and so on. We find that for age \(=70\), the average probability of reporting a poor health status has increased to 0.0618 . We can visualize this by typing marginsplot after margins.
```

. marginsplot
Variables that uniquely identify margins: age

```

Predictive margins with \(95 \%\) Cls


We have focused on the prediction of poor health. We could instead simultaneously obtain average predicted probabilities of poor, fair, good, very good, and excellent health status and plot them across our requested age range. In that case, we would type
```

. margins, at(age = (30(10)70))
. marginsplot

```

We might also be interested in characterizing the relationship between bmi and the probability of reporting poor health. The coefficients and cutpoints reported in hetoprobit are not easily interpreted. We can, however, use margins to estimate the average marginal effect of bmi on the probability of
reporting poor health. Because the average marginal effect depends on the value of bmi, we estimate it across a range of bmi values by typing
```

. margins, dydx(bmi) at(bmi = (20(5)35)) predict(outcome(1))
Average marginal effects Number of obs = 2,009
Model VCE: OIM
Expression: Pr(health==1), predict(outcome(1))
dy/dx wrt: bmi
1._at: bmi = 20
2._at: bmi = 25
3._at: bmi = 30
4._at: bmi = 35

```
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline & \multicolumn{2}{|l|}{\[
\begin{array}{cc} 
& \text { Delta-method } \\
\mathrm{dy} / \mathrm{dx} \quad \text { std. err. }
\end{array}
\]} & z & \(P>|z|\) & [95\% con & interval] \\
\hline \multicolumn{7}{|l|}{bmi} \\
\hline _at & & & & & & \\
\hline 1 & . 0015875 & . 0001721 & 9.23 & 0.000 & . 0012503 & . 0019248 \\
\hline 2 & . 0024512 & . 0002665 & 9.20 & 0.000 & . 0019288 & . 0029736 \\
\hline 3 & . 0036447 & . 0004377 & 8.33 & 0.000 & . 0027868 & . 0045027 \\
\hline 4 & . 0052124 & . 0007013 & 7.43 & 0.000 & . 0038379 & . 006587 \\
\hline
\end{tabular}

The average marginal effect of bmi on the probability of reporting poor health increases as bmi itself increases.

\section*{Example 4: Interpreting the variance function}

From the output of our hetoprobit command, we determined that variance of health status is affected by age. Let's consider to what extent. In this example, we assess the effect of age on the variance by using the margins command. We use the predict (sigma) option to obtain the average predicted standard deviation of the errors. We will look at ages 15 and 85 , which are the youngest and oldest ages, respectively, in our dataset.
```

. margins, predict(sigma) at(age = (15,85)) noatlegend
Predictive margins Number of obs = 2,009
Model VCE: OIM
Expression: Heteroskedastic standard deviation, predict(sigma)

```
\begin{tabular}{c|rrrrrr}
\hline & \multicolumn{4}{c}{\begin{tabular}{c} 
Delta-method \\
std. err.
\end{tabular}} & z & \(\mathrm{P}>|\mathrm{z}|\) \\
\hline Margin & [95\% conf. interval] \\
\hline 1 & 1.014732 & .0320905 & 31.62 & 0.000 & .9518354 & 1.077628 \\
2 & 1.355853 & .1398144 & 9.70 & 0.000 & 1.081822 & 1.629884 \\
\hline
\end{tabular}

Variation increases with age. The expected standard deviation of the error term changes from 1.015 at age 15 to 1.356 at age 85 .

\section*{Stored results}
hetoprobit stores the following in e() :
Scalars
e(N)
e(k_cat)
e(k)
e(k_eq)
e(k_eq_model)
e(k_aux)
e(k_dv)
e(df_m)
e(11)
e(11_0)
e(ll_c)
e(N_clust)
e(chi2)
e(chi2_c)
\(\mathrm{e}(\mathrm{p})\)
e(p_c)
e(df_m_c)
e(rank)
e(ic)
e(rc)
e(converged)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(offset1)
e(offset2)
e(chi2type)
e(chi2_ct)
e(vce)
e(vcetype)
e(opt)
\(e\) (which)
e(ml_method)
e(user)
e(technique)
\(e(\) properties)
e(predict)
e(marginsok)
e(marginsnotok)
\(e(\) marginsdefault)
e(asbalanced)
e(asobserved)
Matrices
e(b)
e(Cns)
e(ilog)
e(gradient)
e(cat)
e(V)
e(V_modelbased)
Functions
e(sample)
number of observations
number of categories
number of parameters
number of equations in \(e(b)\)
number of equations in overall model test
number of auxiliary parameters
number of dependent variables
model degrees of freedom ( \(x \beta\) term)
log likelihood
log likelihood, cutpoint-only (heteroskedastic) model
log likelihood, comparison (homoskedastic) model
number of clusters
\(\chi^{2}\)
\(\chi^{2}\) for heteroskedasticity test
\(p\)-value for model test
\(p\)-value for heteroskedasticity test
degrees of freedom for heteroskedasticity test
rank of \(e(V)\)
number of iterations
return code
1 if converged, 0 otherwise
hetoprobit
command as typed
name of dependent variable
weight type
weight expression
title in estimation output
name of cluster variable
offset for ordered probit equation
offset for variance equation
LR; type of model \(\chi^{2}\) test
LR or Wald; type of heteroskedasticity test corresponding to e(chi2_c)
vcetype specified in vce()
title used to label Std. err.
type of optimization
max or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement predict
predictions allowed by margins
predictions disallowed by margins
default predict() specification for margins
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
constraints matrix
iteration \(\log\) (up to 20 iterations)
gradient vector
category values
variance-covariance matrix of the estimators
model-based variance
marks estimation sample

In addition to the above, the following is stored in r() :
Matrices
\(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

\section*{Methods and formulas}
hetoprobit fits a cumulative probit model with heteroskedastic variance using maximum likelihood estimation. Namely, the model is that, for a subject with explanatory variables \(\mathbf{x}\) and \(\mathbf{z}\),
\[
\operatorname{Pr}(Y \leq h)=\Phi\left\{\frac{\kappa_{h+1}-\mathbf{x} \boldsymbol{\beta}}{\exp (\mathbf{z} \boldsymbol{\gamma})}\right\}
\]
where \(Y\) is an ordinal outcome taking on values \(h=0,1, \ldots, H\), and \(\Phi(\cdot)\) is the cdf of the standard normal distribution. The value \(\kappa_{h+1}\) is a cutpoint that separates the region corresponding to \(Y=h\) from regions for higher-valued categories. The effects \(\boldsymbol{\beta}\) and the effects \(\gamma\) are the same for each cumulative probability.

The log-likelihood function is
\[
\ln L=\sum_{j=1}^{N} w_{j} \sum_{h=0}^{H} I_{h}\left(y_{j}\right) \ln \left[\Phi\left\{\frac{\kappa_{h+1}-\mathbf{x}_{j} \boldsymbol{\beta}}{\exp \left(\mathbf{z}_{j} \boldsymbol{\gamma}\right)}\right\}-\Phi\left\{\frac{\kappa_{h}-\mathbf{x}_{j} \boldsymbol{\beta}}{\exp \left(\mathbf{z}_{j} \boldsymbol{\gamma}\right)}\right\}\right]
\]
where
\[
I_{h}\left(y_{j}\right)= \begin{cases}1 & \text { if } y_{j}=h \\ 0 & \text { otherwise }\end{cases}
\]
and \(y_{j}\), where \(j=1, \ldots, N\), is an observed value of \(Y ; w_{j}\) are optional weights; \(\kappa_{0}=-\infty\) and \(\kappa_{H+1}=\infty\); and all other terminology is defined in Remarks and examples above.

The log-likelihood function is maximized as described in [R] Maximize.
This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce(robust) and vce(cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.
hetoprobit also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

\section*{References}

Aitchison, J., and S. D. Silvey. 1957. The generalization of probit analysis to the case of multiple responses. Biometrika 44: 131-140. https://doi.org/10.2307/2333245.
Allison, P. D. 1999. Comparing logit and probit coefficients across groups. Sociological Methods and Research 28: 186-208. https://doi.org/10.1177/0049124199028002003.
Alvarez, R. M., and J. Brehm. 1995. American ambivalence towards abortion policy: Development of a heteroskedastic probit model of competing values. American Journal of Political Science 39: 1055-1082. https://doi.org/10.2307/2111669.

Harvey, A. C. 1976. Estimating regression models with multiplicative heteroscedasticity. Econometrica 44: 461-465. https://doi.org/10.2307/1913974.

Long, J. S., and J. Freese. 2014. Regression Models for Categorical Dependent Variables Using Stata. 3rd ed. College Station, TX: Stata Press.
McCullagh, P. 1980. Regression models for ordinal data (with discussion). Journal of the Royal Statistical Society, Series B 42: 109-142. https://doi.org/10.1111/j.2517-6161.1980.tb01109.x.

Reardon, S. F., B. R. Shear, K. E. Castellano, and A. D. Ho. 2017. Using heteroskedastic ordered probit models to recover moments of continuous test score distributions from coarsened data. Journal of Educational and Behavioral Statistics 42: 3-45. https://doi.org/10.3102/1076998616666279.
Williams, R. 2009. Using heterogeneous choice models to compare logit and probit coefficients across groups. Sociological Methods and Research 37: 531-559. https://doi.org/10.1177/0049124109335735.
-_. 2010. Fitting heterogeneous choice models with oglm. Stata Journal 10: 540-567.
Yatchew, A., and Z. Griliches. 1985. Specification error in probit models. Review of Economics and Statistics 67: 134-139. https://doi.org/10.2307/1928444.

\section*{Also see}
[R] hetoprobit postestimation - Postestimation tools for hetoprobit
\([R]\) hetprobit - Heteroskedastic probit model
[R] oprobit - Ordered probit regression
[BAYES] bayes: hetoprobit - Bayesian heteroskedastic ordered probit regression
[SVY] svy estimation - Estimation commands for survey data
[U] 20 Estimation and postestimation commands

\section*{Title}
hetoprobit postestimation - Postestimation tools for hetoprobit
\begin{tabular}{lll} 
Postestimation commands & predict & margins \\
Methods and formulas & Also see &
\end{tabular}

\section*{Postestimation commands}

The following postestimation commands are available after hetoprobit:
\begin{tabular}{|c|c|}
\hline Command & Description \\
\hline contrast & contrasts and ANOVA-style joint tests of estimates \\
\hline estat ic & Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) \\
\hline estat summarize & summary statistics for the estimation sample \\
\hline estat vce & variance-covariance matrix of the estimators (VCE) \\
\hline estat (svy) & postestimation statistics for survey data \\
\hline estimates & cataloging estimation results \\
\hline etable & table of estimation results \\
\hline * forecast & dynamic forecasts and simulations \\
\hline * hausman & Hausman's specification test \\
\hline lincom & point estimates, standard errors, testing, and inference for linear combinations of coefficients \\
\hline linktest & link test for model specification \\
\hline * Irtest & likelihood-ratio test \\
\hline margins & marginal means, predictive margins, marginal effects, and average marginal effects \\
\hline marginsplot & graph the results from margins (profile plots, interaction plots, etc.) \\
\hline nlcom & point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients \\
\hline predict & probabilities, linear predictions, etc. \\
\hline predictnl & point estimates, standard errors, testing, and inference for generalized predictions \\
\hline pwcompare & pairwise comparisons of estimates \\
\hline suest & seemingly unrelated estimation \\
\hline test & Wald tests of simple and composite linear hypotheses \\
\hline testnl & Wald tests of nonlinear hypotheses \\
\hline
\end{tabular}

\footnotetext{
*forecast, hausman, and lrtest are not appropriate with svy estimation results.
}

\section*{predict}

\section*{Description for predict}
predict creates a new variable containing predictions such as probabilities, linear predictions, and standard deviations.

\section*{Menu for predict}

Statistics \(>\) Postestimation

\section*{Syntax for predict}
```

    predict [type] {stub*|newvar|newvarlist } [if] [in] [, statistic
    outcome(outcome) nooffset]
    predict [type] stub* [if] [in], scores
    ```
    statistic Description
Main
    pr predicted probabilities; the default
    xb linear prediction
    stdp standard error of the linear prediction
    sigma standard deviation of the error term

You specify one or \(k\) new variables with pr , where \(k\) is the number of outcomes. If you specify one new variable and you do not specify outcome(), then outcome(\#1) is assumed.
You specify one new variable with xb , stdp, or sigma.
These statistics are available both in and out of sample; type predict . . . if e (sample) ... if wanted only for the estimation sample.

\section*{Options for predict}

\begin{abstract}
Main
pr , the default, computes the predicted probabilities for all outcomes or for a specific outcome. To compute probabilities for all outcomes, you specify \(k\) new variables, where \(k\) is the number of categories of the dependent variable. Alternatively, you can specify stub*; in which case, pr will store predicted probabilities in variables \(s t u b 1, s t u b 2, \ldots, s t u b k\). To compute the probability for a specific outcome, you specify one new variable and, optionally, the outcome value in option outcome(); if you omit outcome(), the first outcome value, outcome(\#1), is assumed.

Say that you fit a model by typing estimation_cmd y x1 x2, and y takes on four values. Then, you could type predict p1 p2 p3 p4 to obtain all four predicted probabilities; alternatively, you could type predict \(p *\) to generate the four predicted probabilities. To compute specific probabilities one at a time, you can type predict p1, outcome(\#1) (or simply predict p1), predict p2, outcome(\#2), and so on. See option outcome() for other ways to refer to outcome values.
\end{abstract}
xb calculates the linear prediction. The linear prediction is defined by ignoring the contribution of the estimated cutpoints.
stdp calculates the standard error of the linear prediction.
sigma calculates the standard deviation of the error term.
outcome (outcome) specifies for which outcome the predicted probabilities are to be calculated. outcome() should contain either one value of the dependent variable or one of \#1, \#2, ..., with \#1 meaning the first category of the dependent variable, \#2 meaning the second category, etc. outcome() is available only with the default pr option.
nooffset is relevant only if you specified offset (varname) for hetoprobit or within the het () option. nooffset modifies the calculations made by predict so that they ignore the offset variable: the linear prediction is treated as \(\mathbf{x}_{j} \mathbf{b}\) rather than as \(\mathbf{x}_{j} \mathbf{b}+\) offset \(_{j}^{b}\), and the prediction of \(\ln (\sigma)\) is treated as \(\mathbf{z}_{j} \mathbf{g}\) rather than as \(\mathbf{z}_{j} \mathbf{g}+\operatorname{offset}_{j}^{g}\). nooffset is not allowed with scores.
scores calculates equation-level score variables.
The first new variable will contain \(\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)\).
The next new variable will contain \(\partial \ln L / \partial\left(\mathbf{z}_{j} \gamma\right)\).
The next new variable will contain \(\partial \ln L / \partial \kappa_{1}\).
The next new variable (if any) will contain \(\partial \ln L / \partial \kappa_{2}\).

The last new variable will contain \(\partial \ln L / \partial \kappa_{H}\), where \(\kappa_{h}\) for \(h=1,2, \ldots, H\) refers to the \(h\) th cutpoint. If the linear predictor had no indepvars, the first new variable will contain \(\partial \ln L / \partial\left(\mathbf{z}_{j} \gamma\right)\).

\section*{margins}

\section*{Description for margins}
margins estimates margins of response for probabilities, linear predictions, and standard deviations.

\section*{Menu for margins}

Statistics \(>\) Postestimation

\section*{Syntax for margins}
```

margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...) ...] [options]]

```
statistic Description
\begin{tabular}{ll} 
Main & \\
default & probabilities for each outcome \\
pr & probability for a specified outcome \\
xb & linear prediction \\
stdp & not allowed with margins \\
sigma & standard deviation of the error term
\end{tabular}

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

\section*{Remarks and examples}

See [U] 20 Estimation and postestimation commands for an overview of postestimation commands, including information on obtaining the variance-covariance matrix of the estimators, predicted values, and hypothesis tests.

Once you have fit a model with hetoprobit, you may use the predict command to obtain the predicted probabilities for both the estimation sample and other samples. With the pr option, predict calculates the predicted probability of one or all ordinal value outcomes. With the xb option, predict calculates the linear prediction, \(\mathbf{x}_{j} \mathbf{b}\), where \(\mathbf{x}_{j}\) are the independent variables in the \(j\) th observation and \(\mathbf{b}\) is the estimated parameter vector. The linear prediction is defined ignoring the contribution of the cutpoints. With the sigma option, predict calculates the predicted standard deviations of the error term, \(\sigma_{j}=\exp \left(\mathbf{z}_{j} \mathbf{g}\right)\), where \(\mathbf{g}\) is the estimated coefficient vector for the variance model.

See example 2 in \([R]\) hetoprobit for an example of predict after hetoprobit. In example 3 and example 4, we demonstrate how to use margins to obtain marginal effects, to compute expected probabilities of outcome levels across values of covariates, and to characterize the variance as a function of covariates.

\section*{Methods and formulas}

For definitions of the terminology used in this section, see Remarks and examples in \([R]\) hetoprobit.
For outcome \(h\), the statistic pr is the estimated predicted probability
\[
\widehat{\operatorname{Pr}}\left(y_{j}=h\right)=\Phi\left\{\frac{\widehat{\kappa}_{h+1}-\mathbf{x}_{j} \mathbf{b}}{\exp \left(\mathbf{z}_{j} \mathbf{g}\right)}\right\}-\Phi\left\{\frac{\widehat{\kappa}_{h}-\mathbf{x}_{j} \mathbf{b}}{\exp \left(\mathbf{z}_{j} \mathbf{g}\right)}\right\}
\]
where \(\mathbf{x}_{j} \mathbf{b}\) and \(\mathbf{z}_{j} \mathbf{g}\) are the linear predictions of the regression and variance models, respectively, for the \(j\) th subject.

The statistic sigma is the estimated standard deviation of the modeled heteroskedastic error, namely,
\[
\widehat{\boldsymbol{\sigma}}_{j}=\exp \left(\mathbf{z}_{j} \mathbf{g}\right)
\]

If you specified offset (varname) with hetoprobit or within the het () option (and if you do not specify option nooffset with predict), then the specified offsets are applied by predict. Namely, the linear prediction is computed as \(\mathbf{x}_{j} \mathbf{b}+\operatorname{offset}_{j}^{b}\); the prediction of \(\ln (\sigma)\) is computed as \(\mathbf{z}_{j} \mathbf{g}+\operatorname{offset}_{j}^{g}\); and all other statistics are based on the resulting predictions. If you specify nooffset with predict, then the linear prediction is \(\mathbf{x}_{j} \mathbf{b}\) and the prediction of \(\ln (\sigma)\) is \(\mathbf{z}_{j} \mathbf{g}\), regardless of whether you specified the offset() option with hetoprobit or within het().

\section*{Also see}
[R] hetoprobit - Heteroskedastic ordered probit regression
[U] 20 Estimation and postestimation commands

\section*{Title}
hetprobit - Heteroskedastic probit model
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
hetprobit fits a maximum-likelihood heteroskedastic probit model.

\section*{Quick start}

Heteroskedastic probit model of y on x 1 , using x 2 to model the variance hetprobit \(y \mathrm{x} 1\), het ( x 2 )

With robust standard errors
```

hetprobit y x1, het(x2) vce(robust)

```

After fitting a model, reprint the table as a coefficient legend hetprobit, coeflegend

\section*{Menu}

Statistics \(>\) Binary outcomes \(>\) Heteroskedastic probit regression

\section*{Syntax}
```

hetprobit depvar [indepvars] [if] [in] [weight],
$\operatorname{het}\left(\operatorname{varlist}\left[\right.\right.$, offset $\left(\right.$ varname $\left.\left._{o}\right)\right]$ ) [options $]$

```
options
Description
Model
* het (varlist [...])
noconstant
offset (varname)
asis
constraints (constraints)
SE/Robust
vce (vcetype)

Reporting
level(\#)
lrmodel
waldhet
nocnsreport
display_options

\section*{Maximization}
maximize_options
collinear
coeflegend offset variable
independent variables to model the variance and optional
suppress constant term
include varname in model with coefficient constrained to 1
retain perfect predictor variables
apply specified linear constraints
vcetype may be oim, robust, cluster clustvar, opg, bootstrap, or jackknife
set confidence level; default is level (95)
perform the likelihood-ratio model test instead of the default Wald test perform Wald test on variance instead of LR test
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
\({ }^{*} \operatorname{het}()\) is required. The full specification is het(varlist \(\left[\right.\), offset(varname \({ }_{o}\) )]). indepvars and varlist may contain factor variables; see [U] 11.4.3 Factor variables.
depvar, indepvars, and varlist may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bayes, bootstrap, by, collect, fp, jackknife, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: hetprobit.
Weights are not allowed with the bootstrap prefix; see \([R]\) bootstrap.
vce(), lrmodel, and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options}

Model
het (varlist [, offset \(\left(\right.\) varname \(\left.\left._{o}\right)\right]\) ) specifies the independent variables and, optionally, the offset variable in the variance function. het () is required.
offset ( varname \(_{o}\) ) specifies that selection offset varname \({ }_{o}\) be included in the model with the coefficient constrained to be 1 .
noconstant, offset (varname); see [R] Estimation options.
asis forces the retention of perfect predictor variables and their associated perfectly predicted observations and may produce instabilities in maximization; see [R] probit.
constraints (constraints); see [R] Estimation options.

SE/Robust
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
\(\qquad\) Reporting
level(\#), lrmodel; see [R] Estimation options.
waldhet specifies that a Wald test of whether lnsigma \(=0\) be performed instead of the LR test. nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate (\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from(init_specs); see [R] Maximize. These options are seldom used.

Setting the optimization type to technique (bhhh) resets the default vcetype to vce (opg).
The following options are available with hetprobit but are not shown in the dialog box: collinear, coeflegend; see [R] Estimation options.

\section*{Remarks and examples}

Remarks are presented under the following headings:

\section*{Introduction}

Robust standard errors

\section*{Introduction}
hetprobit fits a maximum-likelihood heteroskedastic probit model, which is a generalization of the probit model. Let \(y_{j}, j=1, \ldots, N\), be a binary outcome variable taking on the value 0 (failure) or 1 (success). In the probit model, the probability that \(y_{j}\) takes on the value 1 is modeled as a nonlinear function of a linear combination of the \(k\) independent variables \(\mathbf{x}_{j}=\left(x_{1 j}, x_{2 j}, \ldots, x_{k j}\right)\),
\[
\operatorname{Pr}\left(y_{j}=1\right)=\Phi\left(\mathbf{x}_{j} \mathbf{b}\right)
\]
in which \(\Phi(\cdot)\) is the cumulative distribution function (CDF) of a standard normal random variable, that is, a normally distributed (Gaussian) random variable with mean 0 and variance 1 . The linear combination of the independent variables, \(\mathbf{x}_{j} \mathbf{b}\), is commonly called the index function, or index. Heteroskedastic probit generalizes the probit model by generalizing \(\Phi(\cdot)\) to a normal CDF with a variance that is no longer fixed at 1 but can vary as a function of the independent variables. hetprobit models the variance as a multiplicative function of these \(m\) variables \(\mathbf{z}_{j}=\left(z_{1 j}, z_{2 j}, \ldots, z_{m j}\right)\), following Harvey (1976):
\[
\sigma_{j}^{2}=\left\{\exp \left(\mathbf{z}_{j} \gamma\right)\right\}^{2}
\]

Thus, the probability of success as a function of all the independent variables is
\[
\operatorname{Pr}\left(y_{j}=1\right)=\Phi\left\{\mathbf{x}_{j} \mathbf{b} / \exp \left(\mathbf{z}_{j} \gamma\right)\right\}
\]

From this expression, it is clear that, unlike the index \(\mathbf{x}_{j} \mathbf{b}\), no constant term can be present in \(\mathbf{z}_{j} \gamma\) if the model is to be identifiable.

Suppose that the binary outcomes \(y_{j}\) are generated by thresholding an unobserved random variable, \(w\), which is normally distributed with mean \(\mathbf{x}_{j} \mathbf{b}\) and variance 1 such that
\[
y_{j}= \begin{cases}1 & \text { if } w_{j}>0 \\ 0 & \text { if } w_{j} \leq 0\end{cases}
\]

This process gives the probit model:
\[
\operatorname{Pr}\left(y_{j}=1\right)=\operatorname{Pr}\left(w_{j}>0\right)=\Phi\left(\mathbf{x}_{j} \mathbf{b}\right)
\]

Now, suppose that the unobserved \(w_{j}\) are heteroskedastic with variance
\[
\sigma_{j}^{2}=\left\{\exp \left(\mathbf{z}_{j} \boldsymbol{\gamma}\right)\right\}^{2}
\]

Relaxing the homoskedastic assumption of the probit model in this manner yields our multiplicative heteroskedastic probit model:
\[
\operatorname{Pr}\left(y_{j}=1\right)=\Phi\left\{\mathbf{x}_{j} \mathbf{b} / \exp \left(\mathbf{z}_{j} \gamma\right)\right\}
\]

\section*{\(>\) Example 1}

For this example, we generate simulated data for a simple heteroskedastic probit model and then estimate the coefficients with hetprobit:
```

. set obs }100
Number of observations (_N) was 0, now 1,000.
. set seed 1234567
. generate x = 1-2*runiform()
. generate xhet = runiform()
. generate sigma = exp(1.5*xhet)
. generate p = normal((0.3+2*x)/sigma)
. generate y = cond(runiform()<=p,1,0)
. hetprobit y x, het(xhet)
Fitting probit model:
Iteration 0: Log likelihood = -688.33746
Iteration 1: Log likelihood = -610.48362
Iteration 2: Log likelihood = -610.3626
Iteration 3: Log likelihood = -610.3626
Fitting full model:
Iteration 0: Log likelihood = -610.3626
Iteration 1: Log likelihood = -600.8767
Iteration 2: Log likelihood = -600.10154
Iteration 3: Log likelihood = -600.01544
Iteration 4: Log likelihood = -600.01521
Iteration 5: Log likelihood = -600.01521

```
\begin{tabular}{lllr} 
Heteroskedastic probit model & Number of obs & \(=\) & 1,000 \\
& Zero outcomes & \(=\) & 451 \\
& Nonzero outcomes & \(=\) & 549 \\
Log likelihood = -600.0152 & Wald chi2 (1) & \(=\) & 54.20 \\
& Prob \(>\) chi2 & \(=\) & 0.0000
\end{tabular}
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline y & Coefficient & Std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% con & interval] \\
\hline \multicolumn{7}{|l|}{y} \\
\hline x & 1.782479 & . 2421117 & 7.36 & 0.000 & 1.307949 & 2.257009 \\
\hline _cons & . 3140616 & . 0871121 & 3.61 & 0.000 & . 1433249 & . 4847982 \\
\hline \multicolumn{7}{|l|}{lnsigma} \\
\hline xhet & 1.31152 & . 3011689 & 4.35 & 0.000 & . 7212402 & 1.901801 \\
\hline
\end{tabular}

LR test of lnsigma=0: chi2(1) \(=20.69\)
Prob \(>\) chi2 \(=0.0000\)
Above, we created two variables, \(x\) and xhet, and then simulated the model
\[
\operatorname{Pr}(\mathrm{y}=1)=F\left\{\left(\beta_{0}+\beta_{1} \mathrm{x}\right) / \exp \left(\gamma_{1} \text { xhet }\right)\right\}
\]
for \(\beta_{0}=0.3, \beta_{1}=2\), and \(\gamma_{1}=1.5\). According to hetprobit's output, all coefficients are significant, and, as we would expect, the Wald test of the full model versus the constant-only model-for example, the index consisting of \(\beta_{0}+\beta_{1} \mathrm{x}\) versus that of just \(\beta_{0}\)-is significant with \(\chi^{2}(1)=54\). Likewise, the likelihood-ratio test of heteroskedasticity, which tests the full model with heteroskedasticity against the full model without, is significant with \(\chi^{2}(1)=21\). See [ R\(]\) Maximize for more explanation of the output. For this simple model, hetprobit took five iterations to converge. As stated elsewhere (Greene 2018, 764), this is a difficult model to fit, and it is not uncommon for it to require many iterations or for the optimizer to print out warnings and informative messages during the optimization. Slow convergence is especially common for models in which one or more of the independent variables appear in both the index and variance functions.

\section*{\(\square\) Technical note}

Stata interprets a value of 0 as a negative outcome (failure) and treats all other values (except missing) as positive outcomes (successes). Thus if your dependent variable takes on the values 0 and 1 , then 0 is interpreted as failure and 1 as success. If your dependent variable takes on the values 0 , 1 , and 2 , then 0 is still interpreted as failure, but both 1 and 2 are treated as successes.

\section*{Robust standard errors}

If you specify the vce(robust) option, hetprobit reports robust standard errors as described in [U] 20.22 Obtaining robust variance estimates. To illustrate the effect of this option, we will reestimate our coefficients by using the same model and data in our example, this time adding vce(robust) to our hetprobit command.
> Example 2


The vce(robust) standard errors for two of the three parameters are larger than the previously reported conventional standard errors. This is to be expected, even though (by construction) we have perfect model specification because this option trades off efficient estimation of the coefficient variance-covariance matrix for robustness against misspecification.

Specifying the vce(cluster clustvar) option relaxes the usual assumption of independence between observations to the weaker assumption of independence just between clusters; that is, hetprobit, vce(cluster clustvar) is robust with respect to within-cluster correlation. This option is less efficient than the xtgee population-averaged models because hetprobit inefficiently sums within cluster for the standard error calculation rather than attempting to exploit what might be assumed about the within-cluster correlation.

\section*{Stored results}
hetprobit stores the following in e() :
Scalars
e(N)
e(N_f)
e(N_s)
e(k)
e(k_eq)
e(k_eq_model)
e(k_dv)
e(df_m)
e(11)
e(ll_0)
e(ll_c)
e(N_clust)
e(chi2)
e(chi2_c)
e(p_c)
e(df_m_c)
e (p)
e(rank)
e(rankO)
e(ic)
e(rc)
e(converged)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(offset1)
e(offset2)
e(chi2type)
e(chi2_ct)
e(vce)
e(vcetype)
e(opt)
e(which)
\(e\) (method)
\(e(m l\) _method)
e(user)
e(technique)
e(properties)
e(predict)
e(asbalanced)
e(asobserved)
Matrices
e(b)
e(Cns)
e(ilog)
e(gradient)
e(V)
e(V_modelbased)
Functions
e(sample)
number of observations
number of zero outcomes
number of nonzero outcomes
number of parameters
number of equations in \(e(b)\)
number of equations in overall model test
number of dependent variables
model degrees of freedom
log likelihood
log likelihood, constant-only model
log likelihood, comparison model
number of clusters
\(\chi^{2}\)
\(\chi^{2}\) for heteroskedasticity test
\(p\)-value for heteroskedasticity test
degrees of freedom for heteroskedasticity test
\(p\)-value for model test
rank of \(e(V)\)
rank of \(e(V)\) for constant-only model
number of iterations
return code
1 if converged, 0 otherwise
hetprobit
command as typed
name of dependent variable
weight type
weight expression
title in estimation output
name of cluster variable
offset for probit equation
offset for variance equation
Wald or LR; type of model \(\chi^{2}\) test
Wald or LR; type of model \(\chi^{2}\) test corresponding to e(chi2_c)
vcetype specified in vce()
title used to label Std. err.
type of optimization
max or min; whether optimizer is to perform maximization or minimization
ml
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement predict
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
constraints matrix
iteration \(\log\) (up to 20 iterations)
gradient vector
variance-covariance matrix of the estimators
model-based variance
marks estimation sample

In addition to the above, the following is stored in r() :
Matrices
\(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

\section*{Methods and formulas}

The heteroskedastic probit model is a generalization of the probit model because it allows the scale of the inverse link function to vary from observation to observation as a function of the independent variables.

The log-likelihood function for the heteroskedastic probit model is
\[
\ln L=\sum_{j \in S} w_{j} \ln \Phi\left\{\mathbf{x}_{j} \boldsymbol{\beta} / \exp (\mathbf{z} \gamma)\right\}+\sum_{j \notin S} w_{j} \ln \left[1-\Phi\left\{\mathbf{x}_{j} \boldsymbol{\beta} / \exp (\mathbf{z} \boldsymbol{\gamma})\right\}\right]
\]
where \(S\) is the set of all observations \(j\) such that \(y_{j} \neq 0\) and \(w_{j}\) denotes the optional weights. \(\ln L\) is maximized as described in [ R\(]\) Maximize.

This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce (robust) and vce(cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.
hetprobit also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

\section*{References}

Blevins, J. R., and S. Khan. 2013. Distribution-free estimation of heteroskedastic binary response models in Stata. Stata Journal 13: 588-602.
Greene, W. H. 2018. Econometric Analysis. 8th ed. New York: Pearson.
Harvey, A. C. 1976. Estimating regression models with multiplicative heteroscedasticity. Econometrica 44: 461-465. https://doi.org/10.2307/1913974.

\section*{Also see}
[R] hetprobit postestimation - Postestimation tools for hetprobit
\([R]\) hetoprobit - Heteroskedastic ordered probit regression
[R] logistic - Logistic regression, reporting odds ratios
[R] probit — Probit regression
[BAYES] bayes: hetprobit - Bayesian heteroskedastic probit regression
[SVY] svy estimation - Estimation commands for survey data
[XT] xtprobit - Random-effects and population-averaged probit models
[U] 20 Estimation and postestimation commands

\section*{Title}
hetprobit postestimation - Postestimation tools for hetprobit
Postestimation commands predict margins Remarks and examples

\section*{Postestimation commands}

The following postestimation commands are available after hetprobit:
\begin{tabular}{|c|c|}
\hline Command & Description \\
\hline contrast & contrasts and ANOVA-style joint tests of estimates \\
\hline estat ic & Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) \\
\hline estat summarize & summary statistics for the estimation sample \\
\hline estat vce & variance-covariance matrix of the estimators (VCE) \\
\hline estat (svy) & postestimation statistics for survey data \\
\hline estimates & cataloging estimation results \\
\hline etable & table of estimation results \\
\hline * forecast & dynamic forecasts and simulations \\
\hline * hausman & Hausman's specification test \\
\hline lincom & point estimates, standard errors, testing, and inference for linear combinations of coefficients \\
\hline linktest & link test for model specification \\
\hline * Irtest & likelihood-ratio test \\
\hline margins & marginal means, predictive margins, marginal effects, and average marginal effects \\
\hline marginsplot & graph the results from margins (profile plots, interaction plots, etc.) \\
\hline nlcom & point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients \\
\hline predict & probabilities, linear predictions, etc. \\
\hline predictnl & point estimates, standard errors, testing, and inference for generalized predictions \\
\hline pwcompare & pairwise comparisons of estimates \\
\hline suest & seemingly unrelated estimation \\
\hline test & Wald tests of simple and composite linear hypotheses \\
\hline testnl & Wald tests of nonlinear hypotheses \\
\hline
\end{tabular}

\footnotetext{
*forecast, hausman, and lrtest are not appropriate with svy estimation results.
}

\section*{predict}

\section*{Description for predict}
predict creates a new variable containing predictions such as probabilities, linear predictions, and standard deviations.

\section*{Menu for predict}

Statistics \(>\) Postestimation

\section*{Syntax for predict}
```

predict [type] newvar [if] [in] [, statistic nooffset]
predict [type]stub* [if] [in], scores

```

\section*{statistic Description}

Main
pr probability of a positive outcome; the default
xb linear prediction
sigma standard deviation of the error term
These statistics are available both in and out of sample; type predict . . . if e (sample) ... if wanted only for the estimation sample.

\section*{Options for predict}

Main
pr , the default, calculates the probability of a positive outcome.
xb calculates the linear prediction.
sigma calculates the standard deviation of the error term.
nooffset is relevant only if you specified offset(varname) for hetprobit. It modifies the calculations made by predict so that they ignore the offset variable; the linear prediction is treated as \(\mathbf{x}_{j} \mathbf{b}\) rather than as \(\mathbf{x}_{j} \mathbf{b}+\) offset \(_{j}\).
scores calculates equation-level score variables.
The first new variable will contain \(\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)\).
The second new variable will contain \(\partial \ln L / \partial\left(\mathbf{z}_{j} \gamma\right)\).

\section*{margins}

\section*{Description for margins}
margins estimates margins of response for probabilities, linear predictions, and standard deviations.

\section*{Menu for margins}

Statistics \(>\) Postestimation

\section*{Syntax for margins}
margins [marginlist] [, options]
 statistic Description
\(\mathrm{pr} \quad\) probability of a positive outcome; the default
xb linear prediction
sigma standard deviation of the error term

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see \([\mathrm{R}]\) margins.

\section*{Remarks and examples}

Once you have fit a model, you can use the predict command to obtain the predicted probabilities for both the estimation sample and other samples; see [U] 20 Estimation and postestimation commands and [R] predict. predict without arguments calculates the predicted probability of a positive outcome. With the xb option, predict calculates the index function combination, \(\mathbf{x}_{j} \mathbf{b}\), where \(\mathbf{x}_{j}\) are the independent variables in the \(j\) th observation and \(\mathbf{b}\) is the estimated parameter vector. With the sigma option, predict calculates the predicted standard deviation, \(\sigma_{j}=\exp \left(\mathbf{z}_{j} \gamma\right)\).

\section*{> Example 1}

We use predict to compute the predicted probabilities and standard deviations based on the model in example 2 in \([\mathrm{R}]\) hetprobit to compare these with the actual values:
```

. predict phat
(option pr assumed; Pr(y))
. generate diff_p = phat - p
. summarize diff_p

```
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline diff_p & 1,000 & .0082805 & .0103027 & -.0169849 & .0396469
\end{tabular}
. predict sigmahat, sigma
. generate diff_s = sigmahat - sigma
. summarize diff_s
\begin{tabular}{r|rrrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline diff_s & 1,000 & -.2579493 & .2126614 & -.7661171 & -.000025
\end{tabular}

\section*{Also see}
[R] hetprobit - Heteroskedastic probit model
[U] 20 Estimation and postestimation commands

\section*{Title}
hetregress - Heteroskedastic linear regression

Description
Menu
Options for maximum likelihood estimation
Remarks and examples
Methods and formulas
Also see

Quick start
Syntax
Options for two-step GLS estimation
Stored results
References

\section*{Description}
hetregress fits a multiplicative heteroskedastic linear regression by modeling the variance as an exponential function of the specified variables using either maximum likelihood (ML; the default) or Harvey's two-step generalized least-squares (GLS) method.

\section*{Quick start}

Heteroskedastic regression model of y on x 1 , using x 2 to model the variance hetregress \(y\) x1, het (x2)

Using Harvey's two-step GLS estimator instead of the default ML
hetregress y x1, het(x2) twostep
With robust standard errors hetregress \(y\) x1, het(x2) vce(robust)

Perform a Wald test on the variance instead of a likelihood-ratio (LR) test hetregress \(y\) x1, het(x2) waldhet

\section*{Menu}

Statistics \(>\) Linear models and related \(>\) Heteroskedastic linear regression

\section*{Syntax}

Maximum likelihood estimation
hetregress depvar [indepvars] [if] [in] [weight] [, ml_options]

Two-step GLS estimation
hetregress depvar [indepvars] [if] [in], twostep het(varlist) [ts_options]
ml_options
Description
Model
mle use maximum likelihood estimator; the default
het (varlist)
noconstant
constraints (constraints)
apply specified linear constraints
SE/Robust
vce (vcetype)
vcetype may be oim, robust, cluster clustvar, opg, bootstrap, or jackknife

Reporting
level(\#)
set confidence level; default is level (95)
lrmodel
waldhet
perform the LR model test instead of the default Wald model test
nocnsreport
perform Wald test on variance instead of LR test
display_options
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

Maximization
maximize_options
control the maximization process; seldom used
collinear keep collinear variables
coeflegend
display legend instead of statistics

\section*{Model}
* twostep
* het (varlist) noconstant

SE
vce (vcetype)
Reporting
level (\#)
display_options
coeflegend
use two-step GLS estimator; default is maximum likelihood independent variables to model the variance suppress constant term
vcetype may be conventional, bootstrap, or jackknife
set confidence level; default is level (95)
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
display legend instead of statistics
*twostep and het() are required.
indepvars and varlist may contain factor variables; see [U] 11.4.3 Factor variables.
depvar, indepvars, and varlist may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bayes, bootstrap, by, collect, fp, jackknife, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: hetregress.
Weights are not allowed with the bootstrap prefix; see [R] bootstrap.
aweights are not allowed with the jackknife prefix; see [R] jackknife.
vce(), lrmodel, twostep, and weights are not allowed with the svy prefix; see [SVY] svy.
aweights, fweights, iweights, and pweights are allowed with maximum likelihood estimation; see [U] 11.1.6 weight. collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options for maximum likelihood estimation}
\(\qquad\) Model
mle requests that the maximum likelihood estimator be used. This is the default.
het (varlist) specifies the independent variables in the variance function. When the het () option is not specified, homoskedasticity is assumed and the waldhet option is not allowed.
noconstant, constraints (constraints); see [R] Estimation options.

SE/Robust
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
level(\#), lrmodel; see [R] Estimation options.
waldhet specifies that the Wald test of whether lnsigma2 \(=0\) be performed instead of the LR test.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fmt), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate (\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from(init_specs); see [R] Maximize. These options are seldom used.
Setting the optimization type to technique (bhhh) resets the default vcetype to vce (opg).
The following options are available with hetregress but are not shown in the dialog box: collinear, coeflegend; see [R] Estimation options.

\section*{Options for two-step GLS estimation}
\(\qquad\) Model
twostep specifies that the model be fit using Harvey's two-step GLS estimator. This option requires that the independent variables be specified in the het () option to model the variance. het (varlist) specifies the independent variables in the variance function. noconstant; see [R] Estimation options.

\section*{SE}
vce(vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (conventional) and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option. vce (conventional), the default, uses the two-step variance estimator derived by Heckman.
```

            Reporting
    level(\#); see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.

```

The following option is available with hetregress but is not shown in the dialog box:
coeflegend; see [R] Estimation options.

\section*{Remarks and examples}

Remarks are presented under the following headings:

\author{
Introduction \\ Maximum likelihood estimation \\ Two-step GLS estimation
}

\section*{Introduction}
hetregress fits a multiplicative heteroskedastic linear regression model using either ML or Harvey's two-step GLS method. Multiplicative heteroskedasticity occurs when the variances of the error terms are assumed to be a multiplicative function of one or more variables. When variables are not specified in the het () option, hetregress fits a homoskedastic linear regression model.

Heteroskedasticity arises in a regression when the variances of the error terms are not constant across observations. For example, wages may be heteroskedastic when predicted by age group. While there is little variability in wages among workers in their teens and early 20s, wages among workers in their 50s may vary greatly because of a variety of factors. Heteroskedasticity is often found in time-series data and cross-sectional measurements and is a common issue in econometrics, social science, and many other fields. For more detailed information on how to detect the presence of heteroskedasticity, see Tests for violation of assumptions in [R] regress postestimation.

We can use hetregress when the variance is assumed to have a form that is an exponential function of a linear combination of one or more variables. This is known as multiplicative heteroskedasticity and includes most of the useful formulations for variance as special cases. For example, in the special case of groupwise heteroskedasticity, the sample can be divided into groups where each group has a different variance.

A model with multiplicative heteroskedasticity can be written as
\[
\begin{equation*}
y_{i}=\mathbf{x}_{i} \boldsymbol{\beta}+\epsilon_{i} ; \quad \sigma_{i}^{2}=\exp \left(\mathbf{z}_{i} \boldsymbol{\alpha}\right) \tag{1}
\end{equation*}
\]
where \(y_{i}, i=1, \ldots, n\), is the dependent variable; \(\mathbf{x}_{i}=\left(x_{1 i}, x_{2 i}, \ldots, x_{k i}\right)\) are the \(k\) independent variables that model the mean function; and \(\mathbf{z}_{i}=\left(z_{1 i}, z_{2 i}, \ldots, z_{m i}\right)\) are the \(m\) variables that model the variance function. \(\boldsymbol{\beta}\) 's are unknown parameters in the mean function, and \(\boldsymbol{\alpha}\) 's are unknown parameters in the variance function. \(\epsilon_{i}\) 's are errors that are independent and identically distributed with mean 0 and variance \(\sigma_{i}^{2}\). Groupwise heteroskedasticity is modeled using (1) but where the \(\mathbf{z}_{i}\) 's are all indicator (dummy) variables for groups.

Harvey (1976) introduced two methods for dealing with multiplicative heteroskedasticity: ML estimation and two-step GLS estimation. By default, hetregress fits the multiplicative heteroskedastic regression model using ML. If the twostep option is specified, hetregress fits the model using the two-step GLS method. The ML estimates are more efficient than those obtained by the GLS estimator if the mean and variance function are correctly specified and the errors are normally distributed. By contrast, the two-step GLS estimates are more robust if the variance function is incorrect or the errors are nonnormal.

If the form of the variance is completely unknown, we may be better off using the oLS estimator instead of the ML and GLS estimators because it remains unbiased. However, we should then use the robust standard errors to correct for heteroskedasticity. Using robust standard errors for the OLS estimator allows us to make appropriate inferences without specifying any form for the variance. We discuss three modifications of the robust variance calculation in Robust standard errors of \([R]\) regress.

If the form of the variance is known and does not contain any unknown parameters, we can use the weighted least-squares estimator, also called the generalized least-squares estimator. For example, we can use weighted least squares to correct for heteroskedasticity if the variance is proportional to one of the regressors. See section 9.5.2 of Greene (2018) for details, and also see Weighted regression in [ R ] regress.

Greene (2018) and Hill, Griffiths, and Lim (2018) compare the ML estimator and the GLS estimator with the robust OLS estimator. If the form of the heteroskedasticity is specified correctly, the ML and GLS estimators are more efficient than the robust OLS estimator. However, if the form of the heteroskedasticity is misspecified, the robust OLS estimator may be more efficient than the ML and GLS estimators.

\section*{Maximum likelihood estimation}
> Example 1: Multiplicative heteroskedasticity
Consider the following dataset from a study of household expenditure on food described in Hill, Griffiths, and Lim (2018, chap. 8). We want to investigate the relationship between average household expenditure on food and household income by fitting a model of weekly food expenditure (food_exp) on weekly income (income) using OLS.
```

. use https://www.stata-press.com/data/r18/foodexp
(Household expenditure on food)
. regress food_exp income
(output omitted)

```

However, we suspect that the variance for low-income families may be lower than that for highincome families, because low-income families usually have less money to spend on food, while high-income families can choose to spend more or less on food. We plot the least-squares residuals against the value of income by using the rvpplot command after regress.


The graph confirms our suspicions about a relationship between income and the residuals. We believe that the variance is some power function of income. Therefore, we fit a multiplicative heteroskedastic regression model. The model for each observation is
\[
\text { food_exp }_{i}=\beta_{0}+\beta_{1} \times \text { income }_{i}+\epsilon_{i}
\]
and the variance function can be written as
\[
\sigma_{i}^{2}=\sigma^{2} \times \text { income }_{i}^{\gamma}
\]
where \(\gamma\) is an unknown parameter of the variance function. To ensure that we will get positive values for the variance \(\sigma_{i}^{2}\) for all possible values of the unknown parameter \(\gamma\), we rewrite this function so that \(\sigma_{i}^{2}\) is an exponential function of a linear combination of \(\ln \left(\right.\) income \(\left._{i}\right)\) and a constant term:
\[
\sigma_{i}^{2}=\exp \left\{\alpha_{0}+\alpha_{1} \times \ln \left(\text { income }_{i}\right)\right\}
\]
where \(\alpha_{0}=\ln \left(\sigma^{2}\right)\) and \(\alpha_{1}=\gamma\).
To fit this model using hetregress, we first create a variable that contains the logarithm of income (logincome) and use it in the het() option to model the variance function. The constant term in the variance function is always assumed.
```

. generate double logincome = ln(income)
. hetregress food_exp income, het(logincome)
Fitting full model:
Iteration 0: Log likelihood = -227.889
Iteration 1: Log likelihood = -226.61039
Iteration 2: Log likelihood = -225.72188
Iteration 3: Log likelihood = -225.71519
Iteration 4: Log likelihood = -225.71519

```
\begin{tabular}{llll} 
Heteroskedastic linear regression & Number of obs & \(=\) & 40 \\
ML estimation & & & \\
& Wald chi2(1) & \(=\) & 135.11 \\
Log likelihood \(=-225.7152\) & Prob \(>\) chi2 & \(=\) & 0.0000
\end{tabular}
\begin{tabular}{c|rlccrr}
\hline food_exp & Coefficient & Std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline food_exp & & & & & & \\
income & 10.63444 & .9148876 & 11.62 & 0.000 & 8.841295 & 12.42759 \\
_cons & 76.07294 & 7.369143 & 10.32 & 0.000 & 61.62969 & 90.5162 \\
\hline lnsigma2 & & & & & & \\
logincome & 2.769762 & .4481606 & 6.18 & 0.000 & 1.891383 & 3.648141 \\
_cons & .4684052 & 1.310337 & 0.36 & 0.721 & -2.099809 & 3.036619 \\
\hline LR test of lnsigma2=0 \(: \operatorname{chi2}(1)=19.59\) & & & Prob > chi2 \(=0.0000\)
\end{tabular}

The LR test at the bottom of the output is a test for the parameters of the variance function. The \(\chi^{2}(1)\) statistic of 19.59 is significant, indicating that heteroskedasticity is present. If we had preferred the Wald test for heteroskedasticity instead of the LR test, we would have specified the waldhet option.

In addition to the estimated parameters for the mean function (under food_exp), hetregress reports estimated parameters and test statistics for the variance function. The significant \(z\) statistic for logincome also suggests the presence of heteroskedasticity. Relating the output back to our model, \(\exp (0.47) \approx 1.60\) is our estimate of \(\sigma^{2}\). The coefficient for logincome is 2.77. This is our estimate of \(\gamma\), and it can be interpreted as the multiplicative factor of the variance associated with income.

We can obtain more formal results by using nlcom:
```

. nlcom (sigma2: exp(_b[lnsigma2:_cons]))

```
    sigma2: exp(_b[lnsigma2:_cons])
\begin{tabular}{r|rlccrr}
\hline food_exp & Coefficient & Std. err. & z & P>|z| & [95\% conf. interval] \\
\hline sigma2 & 1.597445 & 2.093191 & 0.76 & 0.445 & -2.505135 & 5.700024 \\
\hline
\end{tabular}

Here sigma2 refers to \(\sigma^{2}\) in the variance function given above.

\section*{Two-step GLS estimation}
\(>\) Example 2: Groupwise heteroskedasticity
Here we will use a dataset of 725 faculty members' salaries described in DeMaris (2004) to determine whether there is evidence of a difference in salaries between male faculty and female faculty. In addition to sex (female), other variables that might affect the salaries are prior experience (priorexp), years in rank (yrrank), years at the university (yrbg), and marketability of discipline (salfac). We will treat female as a factor variable and all other variables as continuous variables.

We could fit this model with regress by including main effects and the interaction terms between female and all other variables (by using factor-variable notation).
```

. use https://www.stata-press.com/data/r18/salary, clear
(DeMaris (2004) - Faculty salaries)
. regress salary i.female\#\#(c.priorexp c.yrrank c.yrbg c.salfac)

```
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline Source & SS & df & MS & Number of obs & = & 725 \\
\hline & & & & F (9, 715) & = & 135.23 \\
\hline Model & \(8.9287 e+10\) & 9 & \(9.9207 \mathrm{e}+09\) & Prob > F & = & 0.0000 \\
\hline Residual & \(5.2453 e+10\) & 715 & 73360978.8 & R -squared & = & 0.6299 \\
\hline & & & & Adj R-squared & = & 0.6253 \\
\hline Total & \(1.4174 \mathrm{e}+11\) & 724 & 195773163 & Root MSE & = & 8565.1 \\
\hline
\end{tabular}
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline salary & Coefficient & Std. err. & t & \(P>|t|\) & [95\% conf & interval] \\
\hline 1.female & 5735.113 & 4987.433 & 1.15 & 0.251 & -4056.65 & 15526.88 \\
\hline priorexp & 1042.845 & 82.62092 & 12.62 & 0.000 & 880.6366 & 1205.054 \\
\hline yrrank & -47.80904 & 99.85936 & -0.48 & 0.632 & -243.8617 & 148.2436 \\
\hline yrbg & 1009.12 & 75.5161 & 13.36 & 0.000 & 860.8607 & 1157.38 \\
\hline salfac & 33601.3 & 2531.61 & 13.27 & 0.000 & 28631.02 & 38571.58 \\
\hline female\# c.priorexp & & & & & & \\
\hline 1 & -662.7972 & 159.6171 & -4.15 & 0.000 & -976.1715 & -349.4228 \\
\hline \begin{tabular}{l}
female\# \\
c. yrrank
\end{tabular} & & & & & & \\
\hline 1 & -447.8621 & 218.8695 & -2.05 & 0.041 & -877.5659 & -18.15833 \\
\hline female\#c.yrbg 1 & 115.4784 & 157.9372 & 0.73 & 0.465 & -194.5976 & 425.5545 \\
\hline female\# c.salfac & & & & & & \\
\hline 1 & -7618.157 & 5142.204 & -1.48 & 0.139 & -17713.78 & 2477.467 \\
\hline _cons & 2137.718 & 2634.266 & 0.81 & 0.417 & -3034.103 & 7309.539 \\
\hline
\end{tabular}

However, we believe that the variances differ between female faculty and male faculty. In this case, we will use estat hettest to perform the Breusch and Pagan (1979) test for heteroskedasticity. See Tests for violation of assumptions in [R] regress postestimation for more detailed information.
```

. estat hettest i.female
Breusch-Pagan/Cook-Weisberg test for heteroskedasticity
Assumption: Normal error terms
Variable: i.female
HO: Constant variance
chi2(1) = 11.80
Prob > chi2 = 0.0006

```

The results above suggest the presence of heteroskedasticity with respect to sex. This is a case of groupwise heteroskedasticity and can be modeled using hetregress by treating the sex variable (female) as a factor variable (i.female) in the het() option.

We add the twostep option to obtain two-step GLS estimates instead of ML estimates.


The Wald test for heteroskedasticity is reported at the bottom of the coefficient table instead of the LR test because there is no likelihood computed for the two-step GLS estimation.

Compared with the OLS results obtained using regress, the estimated coefficients for the mean function are not affected by heteroskedasticity, but their standard errors are. Also, the estimated variance in salaries for female faculty is about \(\exp (-0.6) \approx 0.5\) times the estimated variance in salaries for male faculty.

The results above suggest that priorexp, yrbg, and salfac have significant effects on the salary of male faculty. We see also that the effects of priorexp and yrrank on salaries are significantly different between males and females. For example, each additional year of experience for male faculty increases their salary by \(\$ 1,042.85\), and the estimated difference in effect is \(\$ 662.80\) less for female faculty than for male faculty.

To obtain an estimate for female faculty of the effect of experience on salary, we can use lincom.
. lincom priorexp + 1.female\#c.priorexp
( 1) [salary]priorexp + [salary]1.female\#c.priorexp \(=0\)
\begin{tabular}{r|rlcccc}
\hline salary & Coefficient & Std. err. & \(z\) & \(P>|z|\) & [95\% conf. interval] \\
\hline\((1)\) & 380.0481 & 110.8702 & 3.43 & 0.001 & 162.7465 & 597.3497 \\
\hline
\end{tabular}

We see that each additional year of experience increases salary by only \(\$ 380.05\) and that this effect is significant.

We can estimate the effect of each of the other variables on the salaries of female faculty if we wish.
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline salary & Coefficient & Std. err. & z & \(P>|z|\) & [95\% conf. & interval] \\
\hline (1) & -495.6711 & 158.1108 & -3.13 & 0.002 & -805.5627 & -185.7796 \\
\hline
\end{tabular}

The effect of yrrank is associated with a decrease in salary. The effect is significant for female faculty but not for male faculty.

\section*{Stored results}
hetregress (ML) stores the following in e():
Scalars
e(N)
e(k)
e(k_eq)
e(k_eq_model)
e(k_dv)
e(df_m)
e(11)
e(11_0)
e(ll_c)
e(N_clust)
e(chi2)
e(chi2_c)
e(p_c)
e(df_m_c)
e(p)
e(rank)
e(rank0)
e(ic)
e(rc)
e(converged)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(wtype)
e(wexp)
e(title)
e(title2)
number of observations
number of parameters
number of equations in \(e(b)\)
number of equations in overall model test
number of dependent variables
model degrees of freedom
log likelihood, full model
log likelihood, constant-only model
log likelihood, comparison model
number of clusters
\(\chi^{2}\) for mean model test
\(\chi^{2}\) for heteroskedasticity test
\(p\)-value for heteroskedasticity test
degrees of freedom for heteroskedasticity test
\(p\)-value for the mean model test
rank of e(V)
rank of \(e(V)\) for constant-only model
number of iterations
return code
1 if converged, 0 otherwise

\section*{hetregress}
command as typed
name of dependent variable
weight type
weight expression
title in estimation output
secondary title in estimation output
```

e(clustvar)
e(chi2type)
e(chi2_ct)
e(vce)
e(vcetype)
e(opt)
e(which)
e(method)
e(ml_method)
e(user)
e(technique)
e(properties)
e(predict)
e(marginsok)
e(marginsnotok)
e(asbalanced)
e(asobserved)

```
Matrices
e(b)
e(Cns)
e(ilog)
e(gradient)
e(V)
e(V_modelbased)
Functions
e(sample) marks estimation sample

In addition to the above, the following is stored in r() :
```

```
Matrices
```

```
```

```
Matrices
```

```
    \(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values,
    and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.
hetregress (two-step GLS) stores the following in e():
Scalars
e(N)
e(k)
e(df_m)
e(chi2)
e(chi2_c)
e(p_c)
e(df_m_c)
\(e(p)\)
\(e\) (rank)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(title)
e(title2)
e(chi2type)
e(chi2_ct)
e(vce)
\(e\) (method)
e(properties)
e(predict)
e(marginsok)
name of cluster variable
Wald or LR; type of model \(\chi^{2}\) test
Wald or LR; type of heteroskedastic \(\chi^{2}\) test corresponding to e(chi2_c)
vcetype specified in vce()
title used to label Std. err.
type of optimization
max or min; whether optimizer is to perform maximization or minimization ml
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement predict
predictions allowed by margins
predictions disallowed by margins
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
constraints matrix
iteration \(\log\) (up to 20 iterations)
gradient vector
variance-covariance matrix of the estimators
model-based variance
marks estimation sample
```

        e(marginsnotok)
        e(asbalanced)
        e(asobserved)
    Matrices
Functions

```
    e(b) coefficient vector
    \(e(V) \quad\) variance-covariance matrix of the estimators
    e(sample) marks estimation sample

In addition to the above, the following is stored in r() :

Matrices
\(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

\section*{Methods and formulas}

Methods and formulas are presented under the following headings:
Maximum likelihood estimation
Two-step GLS estimation

\section*{Maximum likelihood estimation}

By default, hetregress fits a multiplicative heteroskedastic regression using ML estimation. The log-likelihood function is
\[
\ln L=\sum_{i=1}^{n} \frac{w_{i}}{2}\left\{\frac{\left(y_{i}-\mathbf{x}_{i} \boldsymbol{\beta}\right)^{2}}{\exp \left(\mathbf{z}_{i} \boldsymbol{\alpha}\right)}-\ln (2 \pi)-\mathbf{z}_{i} \boldsymbol{\alpha}\right\}
\]
where \(y_{i}, i=1, \ldots, n\), is the dependent variable; \(\mathbf{x}_{i}=\left(x_{1 i}, x_{2 i}, \ldots, x_{k i}\right)\) are the \(k\) independent variables that model the mean function; \(\mathbf{z}_{i}=\left(z_{1 i}, z_{2 i}, \ldots, z_{m i}\right)\) are the \(m\) variables that model the variance function; and \(w_{i}\) are the weights. \(\boldsymbol{\beta}\) is a column vector of unknown parameters in the mean function, and \(\alpha\) is a column vector of unknown parameters in the variance function. The GLS estimates \(\widehat{\boldsymbol{\beta}}_{\mathrm{GLS}}\) and \(\widehat{\boldsymbol{\alpha}}_{\mathrm{GLS}}\) (described below) are used as the initial values in ML estimation. The \(\ln L\) function is maximized as described in [R] Maximize.

This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce (robust) and vce (cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.
hetregress also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

\section*{Two-step GLS estimation}
hetregress uses two-step GLS estimation when the twostep option is specified. Harvey (1976) describes the procedure in detail, but here are the main steps.
1. Use OLS to estimate regression coefficients \(\boldsymbol{\beta}\) and compute residuals \(\mathbf{e}_{i}, i=1, \ldots, n\).
2. Use OLS to regress the \(\log\)-squared residuals, \(\ln \left(\mathbf{e}_{i}^{2}\right)\), on \(\mathbf{z}\) and estimate \(\alpha\).
3. Perform correction for the OLS estimates of \(\boldsymbol{\alpha}\) to obtain \(\widehat{\boldsymbol{\alpha}}_{c}\) and their covariance matrix based on Harvey (1976).
4. Compute \(\widehat{\sigma}_{i}^{2}=\exp \left(\mathbf{z}_{i} \widehat{\boldsymbol{\alpha}}_{c}\right), i=1, \ldots, n\).
5. Refit the original regression model using \(\widehat{\sigma}_{i}^{2}\) 's as weights to obtain the GLS estimates \(\widehat{\boldsymbol{\alpha}}_{\text {GLS }}\) and \(\widehat{\boldsymbol{\beta}}_{\mathrm{GLS}}\).

\section*{References}

Breusch, T. S., and A. R. Pagan. 1979. A simple test for heteroscedasticity and random coefficient variation. Econometrica 47: 1287-1294. https://doi.org/10.2307/1911963.
DeMaris, A. 2004. Regression with Social Data: Modeling Continuous and Limited Response Variables. Hoboken, NJ: Wiley.
Greene, W. H. 2018. Econometric Analysis. 8th ed. New York: Pearson.
Harvey, A. C. 1976. Estimating regression models with multiplicative heteroscedasticity. Econometrica 44: 461-465. https://doi.org/10.2307/1913974.
Hill, R. C., W. E. Griffiths, and G. C. Lim. 2018. Principles of Econometrics. 5th ed. Hoboken, NJ: Wiley.

\section*{Also see}
\([R]\) hetregress postestimation - Postestimation tools for hetregress
[ R ] regress - Linear regression
[BAYES] bayes: hetregress - Bayesian heteroskedastic linear regression
[SVY] svy estimation - Estimation commands for survey data
[U] 20 Estimation and postestimation commands

\section*{Title}
hetregress postestimation - Postestimation tools for hetregress
Postestimation commands predict margins Remarks and examples

\section*{Postestimation commands}

The following postestimation commands are available after hetregress:
\begin{tabular}{|c|c|}
\hline Command & Description \\
\hline contrast & contrasts and ANOVA-style joint tests of estimates \\
\hline *estat ic & Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) \\
\hline estat summarize & summary statistics for the estimation sample \\
\hline estat vce & variance-covariance matrix of the estimators (VCE) \\
\hline *estat (svy) & postestimation statistics for survey data \\
\hline estimates & cataloging estimation results \\
\hline etable & table of estimation results \\
\hline \(\dagger\) forecast & dynamic forecasts and simulations \\
\hline \(\dagger\) hausman & Hausman's specification test \\
\hline lincom & point estimates, standard errors, testing, and inference for linear combinations of coefficients \\
\hline linktest & link test for model specification \\
\hline * \(\dagger\) lrtest & likelihood-ratio test \\
\hline margins & marginal means, predictive margins, marginal effects, and average marginal effects \\
\hline marginsplot & graph the results from margins (profile plots, interaction plots, etc.) \\
\hline nlcom & point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients \\
\hline predict & linear predictions and their SEs, etc. \\
\hline predictnl & point estimates, standard errors, testing, and inference for generalized predictions \\
\hline pwcompare & pairwise comparisons of estimates \\
\hline * suest & seemingly unrelated estimation \\
\hline test & Wald tests of simple and composite linear hypotheses \\
\hline testnl & Wald tests of nonlinear hypotheses \\
\hline
\end{tabular}

\footnotetext{
*estat ic, estat (svy), lrtest, and suest are not appropriate after hetregress, twostep.
\(\dagger_{\text {forecast, hausman, and }}\) lrtest are not appropriate with svy estimation results.
}

\section*{predict}

\section*{Description for predict}
predict creates a new variable containing predictions such as linear predictions, standard errors, and standard deviations.

\section*{Menu for predict}

Statistics \(>\) Postestimation

\section*{Syntax for predict}

After ML or two-step
predict [type] newvar [if] [in] [, statistic]
After ML
predict [type] stub* [if] [in], scores
statistic Description
Main
\(\mathrm{xb} \quad\) linear prediction; the default
stdp standard error of the linear prediction
sigma standard deviation of the error term

These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

\section*{Options for predict}
\(\qquad\) Main
xb , the default, calculates the linear prediction.
stdp calculates the standard errors of the linear prediction.
sigma calculates the standard deviations of the error term.
scores calculates equation-level score variables.
The first new variable will contain the derivative of the log likelihood with respect to the regression equation, \(\partial \ln L / \partial\left(\mathbf{x}_{i} \boldsymbol{\beta}\right)\).

The second new variable will contain the derivative of the log likelihood with respect to the scale equation (lnsigma2), \(\partial \ln L / \partial\left(\mathbf{z}_{i} \boldsymbol{\alpha}\right)\).

\section*{margins}

\section*{Description for margins}
margins estimates margins of response for linear predictions and of standard deviations.

\section*{Menu for margins}

Statistics \(>\) Postestimation

\section*{Syntax for margins}
```

margins [marginlist] [, options]

```


\section*{statistic}

Description
xb
stdp
linear prediction; the default
not allowed with margins
sigma
standard deviation of the error term

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see \([R]\) margins.

\section*{Remarks and examples}

Once you have fit a heteroskedastic regression model, you can use the predict command to obtain the predicted values both for the estimation sample and for other samples; see [U] 20 Estimation and postestimation commands and \([\mathrm{R}]\) predict. predict without arguments calculates the linear prediction from the fitted model \(\mathbf{x}_{i} \mathbf{b}\), where \(\mathbf{x}_{i}\) are the independent variables in the \(j\) th observation and \(\mathbf{b}\) is the estimated parameter vector for the mean model. With the stdp option, predict calculates the standard error of the linear prediction. With the sigma option, predict calculates the predicted standard deviations of the error term, \(\widehat{\sigma}_{j}=\exp \left(0.5 \times \mathbf{z}_{i} \mathbf{a}\right)\), where \(\mathbf{g}\) is the estimated parameter vector for the variance model.

\section*{> Example 1: Predicting heteroskedastic standard deviation}

We can use predict to compute the predicted values of the standard deviations for female and male faculty based on the model from example 2 in [R] hetregress.
```

. use https://www.stata-press.com/data/r18/salary
(DeMaris (2004) - Faculty salaries)
. hetregress salary i.female\#\#(c.priorexp c.yrrank c.yrbg c.salfac),
> het(i.female) twostep
(output omitted)
. predict sigma, sigma
. tabulate female, summarize(sigma)

| $1=$ female; |
| ---: | ---: | ---: | ---: |
| $0=$ male |$\quad$| Summary ofHeteroskedastic <br> deviation <br> Std. dev. |
| ---: |
| 0 |

```

The predicted standard deviation for male faculty is approximately \(7742 / 5829 \approx 1.3\) times the size for female faculty. We could have obtained the same results using margins with the predict (sigma) option.

\section*{> Example 2: Marginal means}

We can use margins to compute the adjusted mean salary for male and female faculty when other factors are fixed at their means:
```

. margins female, atmeans
Adjusted predictions Number of obs = 725
Model VCE: Conventional
Expression: Linear prediction, predict()
At: O.female = . }7048276\mathrm{ (mean)
1.female = . }2951724\mathrm{ (mean)
priorexp = 2.89931 (mean)
yrrank = 7.397241 (mean)
yrbg = 12.52966 (mean)
salfac = . }9399862 (mean

```
\begin{tabular}{c|rrrrrr}
\hline & \multicolumn{3}{|c}{\begin{tabular}{c} 
Delta-method \\
std. err.
\end{tabular}} & Margin & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline female & & & & & & \\
0 & 49036.29 & 420.3711 & 116.65 & 0.000 & 48212.37 & 49860.2 \\
1 & 43822.74 & 569.9368 & 76.89 & 0.000 & 42705.69 & 44939.8 \\
\hline
\end{tabular}

If everyone in the population were male faculty while holding all other factors at their mean values, the average salary would be \(\$ 49,036\). If, instead, everyone were female faculty, the average salary would be \(\$ 43,823\).

\section*{Also see}
\([\mathrm{R}]\) hetregress - Heteroskedastic linear regression
[U] 20 Estimation and postestimation commands

\section*{Title}
histogram - Histograms for continuous and categorical variables
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & References & Also see
\end{tabular}

\section*{Description}
histogram draws histograms of varname, which is assumed to be the name of a continuous variable unless the discrete option is specified.
hist is a synonym for histogram.

\section*{Quick start}

Histogram of v1
histogram v1
Add a normal density curve to the graph histogram v1, normal

Add a kernal density estimate to the graph histogram v1, normal kdensity

Add "My Title" as the title of the graph histogram v1, normal kdensity title("My Title")

Specify the number of bins as 10
histogram v1, bins(10)
Specify the width of the bins as 2
histogram v1, width(2)
Specify that v2 should be treated as discrete
histogram v2, discrete
Same as above, but with narrower bars and space between the bars
histogram v2, discrete barwidth(.8)
Add labels to the bars on the \(x\) axis
histogram v2, discrete barwidth(.8) xlabel(1 "Category 1" ///
2 "Category 2" 3 "Category 3" 4 "Category 4")
Show frequencies on the \(y\) axis
histogram v1, frequency
Show percentages on the \(y\) axis
histogram v1, percent
Produce histograms for each value of categorical variable catvar
histogram v1, by (catvar)
Same as above, but with histograms arranged in a single column
histogram v1, by (cvar, cols(1))

\section*{Menu}

Graphics > Histogram

\section*{Syntax}
\[
\text { histogram varname }[\text { if }][\text { in }][\text { weight }][,[\text { continuous_opts } \mid \text { discrete_opts }] \text { options }]
\]
continuous_opts
Main
bins (\#) set number of bins to \#
width(\#) set width of bins to \#
start (\#) set lower limit of first bin to \#
discrete_opts
Description
Main
discrete
width(\#)
start(\#)
options
Description
Main
density
fraction
frequency
percent
bar_options
binrescale
addlabels
addlabopts (marker_label_options)
Density plots
normal
normopts (line_options)
kdensity
kdenopts(kdensity_options)
Add plots
addplot (plot)
Y axis, X axis, Titles, Legend, Overall, By
twoway_options
draw as density; the default
draw as fractions
draw as frequencies
draw as percentages
rendition of bars
recalculate bin sizes when by () is specified add height labels to bars
affect rendition of labels
add a normal density to the graph affect rendition of normal density add a kernel density estimate to the graph affect rendition of kernel density
add other plots to the histogram
any options documented in [G-3] twoway_options

\footnotetext{
fweights are allowed; see [U] 11.1.6 weight.
}

\section*{Options}

Options are presented under the following headings:
Options for use in the continuous case
Options for use in the discrete case
Options for use in the continuous and discrete cases

\section*{Options for use in the continuous case}

\section*{Main}
bins(\#) and width (\#) are alternatives. They specify how the data are to be aggregated into bins: bins() by specifying the number of bins (from which the width can be derived) and width() by specifying the bin width (from which the number of bins can be derived).
If neither option is specified, results are the same as if bins ( \(k\) ) had been specified, where
\[
k=\min \{\operatorname{sqrt}(N), 10 \ln (N) / \ln (10)\}
\]
and where \(N\) is the (weighted) number of observations.
start(\#) specifies the theoretical minimum of varname. The default is start( \(m\) ), where \(m\) is the observed minimum value of varname.

Specify start () when you are concerned about sparse data, for instance, if you know that varname can have a value of 0 , but you are concerned that 0 may not be observed.
start (\#), if specified, must be less than or equal to \(m\), or else an error will be issued.

\section*{Options for use in the discrete case}
\(\qquad\) Main
discrete specifies that varname is discrete and that you want each unique value of varname to have its own bin (bar of histogram).
width (\#) is rarely specified in the discrete case; it specifies the width of the bins. The default is width \((d)\), where \(d\) is the observed minimum difference between the unique values of varname.
Specify width() if you are concerned that your data are sparse. For example, in theory varname could take on the values, say, \(1,2,3, \ldots, 9\), but because of the sparseness, perhaps only the values \(2,4,7\), and 8 are observed. Here the default width calculation would produce width(2), and you would want to specify width(1).
start (\#) is also rarely specified in the discrete case; it specifies the theoretical minimum value of varname. The default is start \((m)\), where \(m\) is the observed minimum value.
As with width(), specify start (\#) if you are concerned that your data are sparse. In the previous example, you might also want to specify start(1). start() does nothing more than add white space to the left side of the graph.
The value of \# in start () must be less than or equal to \(m\), or an error will be issued.

\section*{Options for use in the continuous and discrete cases}

\section*{Main}
density, fraction, frequency, and percent specify whether you want the histogram scaled to density units, fractional units, frequencies, or percentages. density is the default.
density scales the height of the bars so that the sum of their areas equals 1 .
fraction scales the height of the bars so that the sum of their heights equals 1 .
frequency scales the height of the bars so that each bar's height is equal to the number of observations in the category. Thus the sum of the heights is equal to the total number of observations.
percent scales the height of the bars so that the sum of their heights equals 100 .
bar_options are any of the options allowed by graph twoway bar; see [G-2] graph twoway bar.
One of the most useful bar_options is barwidth(\#), which specifies the width of the bars in varname units. By default, histogram draws the bars so that adjacent bars just touch. If you want gaps between the bars, do not specify histogram's width() option-which would change how the histogram is calculated-but specify the bar_option barwidth() or the histogram option gap, both of which affect only how the bar is rendered.

The bar_option horizontal cannot be used with the addlabels option.
binrescale specifies that bin size and plot range be recalculated for each group when by () is specified. If normal is specified, the mean and standard deviation of each overlaid normal density plot are recalculated in each group. Similarly, if kdensity is specified, the scaling of the overlaid kernel density plot is recalculated in each group.
addlabels specifies that the top of each bar be labeled with the density, fraction, or frequency, as determined by the density, fraction, and frequency options.
addlabopts(marker_label_options) specifies how to render the labels atop the bars. See [G-3] marker_label_options. Do not specify the marker_label_option mlabel(varname), which specifies the variable to be used; this is specified for you by histogram.
addlabopts() will accept more options than those documented in [G-3] marker_label_options. All options allowed by twoway scatter are also allowed by addlabopts(); see [G-2] graph twoway scatter. One particularly useful option is yvarformat (); see [G-3] advanced_options.

\section*{\(\sqrt{\text { Density plots }}\)}
normal specifies that the histogram be overlaid with an appropriately scaled normal density. The normal will have the same mean and standard deviation as the data.
normopts(line_options) specifies details about the rendition of the normal curve, such as the color and style of line used. See [G-2] graph twoway line.
kdensity specifies that the histogram be overlaid with an appropriately scaled kernel density estimate of the density. By default, the estimate will be produced using the Epanechnikov kernel with an "optimal" half-width. This default corresponds to the default of kdensity; see [R] kdensity. How the estimate is produced can be controlled using the kdenopts () option described below.
kdenopts(kdensity_options) specifies details about how the kernel density estimate is to be produced along with details about the rendition of the resulting curve, such as the color and style of line used; see [G-2] graph twoway kdensity. The kernel density estimate is described in [R] kdensity. As an example, if you wanted to produce kernel density estimates by using the Gaussian kernel with
optimal half-width, you would specify kdenopts (gauss) and if you also wanted a half-width of 5 , you would specify kdenopts (gauss width(5)).

Add plots
addplot (plot) allows adding more graph twoway plots to the graph; see [G-3] addplot_option.

Y axis, X axis, Titles, Legend, Overall, By
twoway_options are any of the options documented in [G-3] twoway_options. This includes, most importantly, options for titling the graph (see [G-3] title_options), options for saving the graph to disk (see [G-3] saving_option), and the by () option, which will allow you to simultaneously graph histograms for different subsets of the data (see [G-3] by_option).

\section*{Remarks and examples}

Remarks are presented under the following headings:
Histograms of continuous variables
Overlaying normal and kernel density estimates
Histograms of discrete variables
Use with by()
Video example
For an example of editing a histogram with the Graph Editor, see Pollock (2011, 29-31).

\section*{Histograms of continuous variables}
histogram assumes that the variable is continuous, so you need type only histogram followed by the variable name:


The small values reported for density on the \(y\) axis are correct; if you added up the area of the bars, you would get 1 . Nevertheless, many people are used to seeing histograms scaled so that the bar heights sum to 1 ,
```

. histogram volume, fraction
(bin=15, start=4103, width=1280.3533)

```

and others are used to seeing histograms so that the bar height reflects the number of observations,
. histogram volume, frequency
(bin=15, start=4103, width=1280.3533)


Regardless of the scale you prefer, you can specify other options to make the graph look more impressive:
. summarize volume
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline volume & 248 & 12320.68 & 2585.929 & 4103 & 23308.3
\end{tabular}
histogram volume, freq
\(>\quad\) xaxis(1 2)
\(>\quad\) ylabel(0(10)60)
\(>\) xlabel(12321 "mean"
\(>\quad 9735\) "-1 s.d."
\(>\quad 14907\) "+1 s.d."
\(>\quad 7149\) "-2 s.d."
\(>\quad 17493\) "+2 s.d."
\(>\quad 20078\) "+3 s.d."
\(>\quad 22664\) "+4 s.d."
\(>\), axis(2) grid)
\(>\quad x l a b e l(\), axis(1) nogrid)
> xtitle("", axis(2))
> subtitle("S\&P 500, January 2001 to December 2001")
> note("Source: Yahoo! Finance and Commodity Systems, Inc.")
(bin=15, start=4103, width=1280.3533)


Source: Yahoo! Finance and Commodity Systems, Inc.
For an explanation of the xaxis() option-it created the upper and lower \(x\) axissee [G-3] axis_choice_options. For an explanation of the ylabel() and xlabel() options, see [G-3] axis_label_options. For an explanation of the subtitle() and note() options, see [G-3] title_options.

\section*{Overlaying normal and kernel density estimates}

Specifying normal will overlay a normal density over the histogram. It would be enough to type . histogram volume, normal
but we will add the option to our more impressive rendition:
. summarize volume
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline volume & 248 & 12320.68 & 2585.929 & 4103 & 23308.3
\end{tabular}
```

histogram volume, freq normal
xaxis(1 2)
ylabel(0(10)60)
xlabel(12321 "mean"
9735 "-1 s.d."
14907 "+1 s.d."
7149 "-2 s.d."
17493 "+2 s.d."
20078 "+3 s.d."
22664 "+4 s.d."
xlabel(, axis(1) nogrid)
xtitle("", axis(2))
subtitle("S\&P 500, January 2001 to December 2001")
note("Source: Yahoo! Finance and Commodity Systems, Inc.")
(bin=15, start=4103, width=1280.3533)

```


If we instead wanted to overlay a kernel density estimate, we could specify kdensity in place of normal.

\section*{Histograms of discrete variables}

Specify histogram's discrete option when you wish to treat the data as discrete-when you wish each unique value of the variable to be assigned its own bin. For instance, in the automobile data, mpg is a continuous variable, but the mileage ratings have been measured to integer precision. If we were to type
```

. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. histogram mpg
(bin=8, start=12, width=3.625)

```
mpg would be treated as continuous and categorized into eight bins by the default number-of-bins calculation, which is based on the number of observations, 74.

Adding the discrete option makes a histogram with a bin for each of the 21 unique values.
```

. histogram mpg, discrete
(start=12, width=1)

```


Just as in the continuous case, the \(y\) axis was reported in density, and we could specify the fraction or frequency options if we wanted it to be reported differently. Below, we specify frequency, we specify addlabels to add a report of frequencies printed above the bars, and we specify xlabel (12(2)42) to label the values \(12,14, \ldots, 42\) on the \(x\) axis:
. histogram mpg, discrete freq addlabels xlabel(12(2)42)
(start=12, width=1)


\section*{Use with by()}
histogram may be used with graph twoway's by () ; for example,
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. histogram mpg, discrete by (foreign)


Here results would be easier to compare if the graphs were presented in one column:
. histogram mpg, discrete by(foreign, col(1))

\(\operatorname{col}(1)\) is a by () suboption-see [G-3] by_option-and there are other useful suboptions, such as total, which will add an overall total histogram. total is a suboption of by (), not an option of histogram, so you would type
. histogram mpg, discrete by(foreign, total)
and not histogram mpg, discrete by (foreign) total.

As another example, Lipset (1993) reprinted data from the New York Times (November 5, 1992) collected by the Voter Research and Surveys based on questionnaires completed by 15,490 US presidential voters from 300 polling places on election day in 1992.
```

. use https://www.stata-press.com/data/r18/voter
(1992 US presidential voters)
. histogram candi [fweight=pop], discrete fraction by(inc, total)
> gap(40) xlabel(2 3 4, valuelabel)

```


Graphs by Family income
We specified gap(40) to reduce the width of the bars by \(40 \%\). We also used xlabel()'s valuelabel suboption, which caused our bars to be labeled "Clinton", "Bush", and "Perot", rather than 2, 3, and 4; see [G-3] axis_label_options.

\section*{Video example}

Histograms in Stata

\section*{References}

Cox, N. J. 2004. Speaking Stata: Graphing distributions. Stata Journal 4: 66-88.
——. 2005. Speaking Stata: Density probability plots. Stata Journal 5: 259-273.
Harrison, D. A. 2005. Stata tip 20: Generating histogram bin variables. Stata Journal 5: 280-281.
Lipset, S. M. 1993. The significance of the 1992 election. PS: Political Science and Politics 26: 7-16. https://doi.org/10.2307/419496.
Pinna, M. 2022. Binned scatterplots with marginal histograms: binscatterhist. Stata Journal 22: 430-445.
Pollock, P. H., III. 2011. A Stata Companion to Political Analysis. 2nd ed. Washington, DC: CQ Press.

\section*{Also see}
[R] kdensity - Univariate kernel density estimation
[R] spikeplot - Spike plots and rootograms
[G-2] graph twoway histogram - Histogram plots

\section*{Title}

IC note - Calculating and interpreting information criteria

Description Remarks and examples Methods and formulas References
Also see

\section*{Description}

This entry discusses a statistical issue that arises when using the Bayesian (BIC), consistent Akaike's (CAIC), and corrected Akaike's (AICc) information criteria to compare models.

Stata calculates BIC, CAIC, and AICc using \(N=e(N)\), unless \(e\left(N_{-} i c\right)\) has been set; in that instance, it uses \(N=e\left(N \_i c\right)\). For example, choice-model cm commands set e(N_ic) to the number of cases because these commands use a data arrangement in which multiple Stata observations represent a single statistical observation, which is called a case.

Sometimes, it would be better if a different \(N\) than e(N) were used. Commands that calculate BIC, CAIC, and AICc have an n () option, allowing you to specify the \(N\) to be used.

In summary,
1. if you are comparing results estimated by the same estimation command, using the default BIC, CAIC, or AICc calculation is probably fine. There is an issue, but most researchers would ignore it.
2. if you are comparing results estimated by different estimation commands, you need to be on your guard.
a. If the different estimation commands share the same definitions of observations, independence, and the like, you are back to case 1.
b. If they differ in these regards, you need to think about the value of \(N\) that should be used. For example, logit and xtlogit differ in that the former assumes independent observations and the latter, independent panels.
c. If estimation commands differ in the events being used over which the likelihood function is calculated, the information criteria may not be comparable at all. We say information criteria because this would apply equally to the Akaike information criterion (AIC) and its possible extensions AICc and CAIC, as well as to the BIC. For instance, streg and stcox produce such incomparable results. The events used by streg are the actual survival times, whereas the events used by stcox are failures within risk pools, conditional on the times at which failures occurred.

\section*{Remarks and examples}

Remarks are presented under the following headings:

\footnotetext{
Background
The problem of determining \(N\)
The problem of conformable likelihoods
The first problem does not arise with AIC; the second problem does
Calculating BIC, AICc, and CAIC correctly
}

\section*{Background}

The AIC and the BIC are two popular measures for comparing maximum likelihood models. AIC and BIC are defined as
\[
\begin{aligned}
& \mathrm{AIC}=-2 \ln L+2 k \\
& \mathrm{BIC}=-2 \ln L+k \ln N
\end{aligned}
\]
where
\[
\begin{aligned}
\ln L & =\text { maximized log-likelihood } \\
k & =\text { number of parameters estimated } \\
N & =\text { number of observations }
\end{aligned}
\]

However, when sample size is small, AIC is biased, and Burnham and Anderson (2002) suggest to use AICc,
\[
\mathrm{AICc}=\mathrm{AIC}+\frac{2 k(k+1)}{N-k-1}
\]

CAIC is a consistent version of AIC and was proposed in Bozdogan (1987),
\[
\mathrm{CAIC}=-2 \ln L+k(\ln N+1)
\]

All four information criteria-AIC, BIC, CAIC, and AICc-can be viewed as measures that combine fit and complexity. Fit is measured negatively by \(-2 \ln L\); the larger the value, the worse the fit. Complexity is measured positively, for example, by \(2 k\) (AIC) or \(k \ln N\) (BIC).

Given two models fit on the same data, the model with the smaller value of the information criterion is considered to be better.

There is substantial literature on these measures: see Akaike (1974); Raftery (1995); Sakamoto, Ishiguro, and Kitagawa (1986); Schwarz (1978); Burnham and Anderson (2002); and Hurvich and Tsai (1989).

When Stata calculates the above measures, it uses the rank of \(\mathrm{e}(\mathrm{V})\) for \(k\), and it uses \(\mathrm{e}(\mathrm{N})\) for \(N . e(V)\) and \(e(N)\) are Stata notation for results stored by the estimation command.e(V) is the variance-covariance matrix of the estimated parameters, and \(e(N)\) is the number of observations in the dataset used in calculating the result.

\section*{The problem of determining \(\mathbf{N}\)}

The difference between AIC and the other three information criteria is that AIC uses the constant 2 to weight \(k\), whereas the complexity term for BIC, CAIC, and AICc depends on \(N\).

Determining what value of \(N\) should be used is problematic. Despite appearances, the definition " \(N\) is the number of observations" is not easy to make operational. \(N\) does not appear in the likelihood function itself, \(N\) is not the output of a standard statistical formula, and what is an observation is often subjective.
> Example 1
Often what is meant by \(N\) is obvious. Consider a simple logit model. What is meant by \(N\) is the number of observations that is statistically independent and that corresponds to \(M\), the number of observations in the dataset used in the calculation. We will write \(N=M\).

But now assume that the same dataset has a grouping variable and the data are thought to be clustered within group. To keep the problem simple, let's pretend that there are \(G\) groups and \(m\) observations within group, so that \(M=G \times m\). Because you are worried about intragroup correlation, you fit your model with xtlogit, grouping on the grouping variable. Now, you wish to calculate BIC. What is the \(N\) that should be used? \(N=M\) or \(N=G\) ?

That is a deep question. If the observations really are independent, then you should use \(N=M\). If the observations within group are not just correlated but are duplicates of one another, and they had to be so, then you should use \(N=G\) (Kass and Raftery 1995). Between those two extremes, you should probably use a number between \(N\) and \(G\), but determining what that number should be from measured correlations is difficult. Using \(N=M\) is conservative in that, if anything, it overweights complexity. Conservativeness, however, is subjective, too: using \(N=G\) could be considered more conservative in that fewer constraints are being placed on the data.

When the estimated correlation is high, our reaction would be that using \(N=G\) is probably more reasonable. Our first reaction, however, would be that using BIC to compare models is probably a misuse of the measure.

Stata uses \(N=M\). An informal survey of web-based literature suggests that \(N=M\) is the popular choice.

There is another reason, not so good, to choose \(N=M\). It makes across-model comparisons more likely to be valid when performed without thinking about the issue. Say that you wish to compare the logit and xtlogit results. Thus, you need to calculate
\[
\begin{aligned}
\mathrm{BIC}_{p} & =-2 \ln L_{p}+k \ln N_{p} \\
\mathrm{BIC}_{x} & =-2 \ln L_{x}+k \ln N_{x}
\end{aligned}
\]

Whatever \(N\) you use, you must use the same \(N\) in both formulas. Stata's choice of \(N=M\) at least meets that test.

\section*{> Example 2}

In the above example, using \(N=M\) is reasonable. Now, let's look at when using \(N=M\) is wrong, even if popular.

Consider a model fit by stcox. Using \(N=M\) is certainly wrong if for no other reason than \(M\) is not even a well-defined number. The same data can be represented by different datasets with different numbers of observations. For example, in one dataset, there might be one observation per subject. In another, the same subjects could have two records each, the first recording the first half of the time at risk and the second recording the remaining part. All statistics calculated by Stata on either dataset would be the same, but \(M\) would be different.

Deciding on the right definition, however, is difficult. Viewed one way, \(N\) in the Cox regression case should be the number of risk pools, \(R\), because the Cox regression calculation is made on the basis of the independent risk pools. Viewed another way, \(N\) should be the number of subjects, \(N_{\text {subj }}\), because, even though the likelihood function is based on risk pools, the parameters estimated are at the subject level.

You can decide which argument you prefer.
For parametric survival models, in single-record data, \(N=M\) is unambiguously correct. For multirecord data, there is an argument for \(N=M\) and for \(N=N_{\text {subj }}\).

\section*{The problem of conformable likelihoods}

The problem of conformable likelihoods does not concern \(N\). Researchers sometimes use information criteria such as BIC and AIC to make comparisons across models. For that to be valid, the likelihoods must be conformable; that is, the likelihoods must all measure the same thing.

It is common to think of the likelihood function as the \(\operatorname{Pr}\) (data| parameters), but in fact, the likelihood is
\[
\operatorname{Pr}(\text { particular events in the data } \mid \text { parameters })
\]

You must ensure that the events are the same.
For instance, they are not the same in the semiparametric Cox regression and the various parametric survival models. In Cox regression, the events are, at each failure time, that the subjects observed to fail in fact failed, given that failures occurred at those times. In the parametric models, the events are that each subject failed exactly when the subject was observed to fail.

The formula for AIC, AICc, CAIC, and BIC can be written as
\[
\text { measure }=-2 \ln L+\text { complexity }
\]

When you are comparing models, if the likelihoods are measuring different events, even if the models obtain estimates of the same parameters, differences in the information measures are irrelevant.

\section*{The first problem does not arise with AIC; the second problem does}

Regardless of model, the problem of defining \(N\) never arises with AIC because \(N\) is not used in the AIC calculation. AIC uses a constant 2 to weight complexity as measured by \(k\), rather than \(\ln N\).

However, for all four information criteria-AIC, AICc, CAIC, and BIC-the likelihood functions must be conformable; that is, they must be measuring the same event.

\section*{Calculating BIC, AICc, and CAIC correctly}

When using BIC, AICc, or CAIC to compare results, and especially when using them to compare results from different models, you should think carefully about how \(N\) should be defined. Then, specify that number by using the n() option:
. estimates stats full sub, all n(74)
Information criteria
\begin{tabular}{r|rrrr}
\hline Model & N & ll(null) & ll(model) & df \\
\hline full & 74 & -45.03321 & -20.59083 & 4 \\
sub & 74 & -45.03321 & -27.17516 & 3 \\
\hline
\end{tabular}
\begin{tabular}{r|rrrr}
\hline Model & AIC & BIC & AICc & CAIC \\
\hline full & 49.18167 & 58.39793 & 49.76138 & 62.39793 \\
sub & 60.35031 & 67.26251 & 60.69317 & 70.26251 \\
\hline
\end{tabular}

Legend: AIC is Akaike's information criterion.
BIC is Bayesian information criterion.
AICc is corrected Akaike's information criterion.
CAIC is consistent Akaike's information criterion.
Both estimates stats and estat ic allow the \(n()\) option; see \([R]\) estimates stats and \([R]\) estat ic.

\section*{Methods and formulas}

AIC, BIC, CAIC, and AICc are defined as
\[
\begin{aligned}
\text { AIC } & =-2 \ln L+2 k \\
\mathrm{BIC} & =-2 \ln L+k \ln N \\
\mathrm{CAIC} & =-2 \ln L+k(\ln N+1) \\
\mathrm{AICc} & =\mathrm{AIC}+\frac{2 k(k+1)}{N-k-1}
\end{aligned}
\]
where \(\ln L\) is the maximized log-likelihood of the model; \(k\) is the model degrees of freedom calculated as the rank of variance-covariance matrix of the parameters \(e(V)\), unless the df() option is specified; and \(N\) is the number of observations used in estimation or, more precisely, the number of independent terms in the likelihood. Operationally, \(N\) is defined as e(N), unless the estimation command returns \(e\left(N \_i c\right)\) or the \(n()\) option is specified with estimates stats or estat ic.

\section*{References}

Akaike, H. 1974. A new look at the statistical model identification. IEEE Transactions on Automatic Control 19: 716-723. https://doi.org/10.1109/TAC.1974.1100705.

Bozdogan, H. 1987. Model selection and Akaike's information criterion (AIC): The general theory and its analytical extensions. Psychometrika 52: 345-370. https://doi.org/10.1007/BF02294361.
Burnham, K. P., and D. R. Anderson. 2002. Model Selection and Multimodel Inference: A Practical InformationTheoretic Approach. 2nd ed. New York: Springer.

Hurvich, C. M., and C.-L. Tsai. 1989. Regression and time series model selection in small samples. Biometrika 76: 297-307. https://doi.org/10.1093/biomet/76.2.297.
Kass, R. E., and A. E. Raftery. 1995. Bayes factors. Journal of the American Statistical Association 90: 773-795. https://doi.org/10.1080/01621459.1995.10476572.

Raftery, A. E. 1995. Bayesian model selection in social research. In Vol. 25 of Sociological Methodology, ed. P. V. Marsden, 111-163. Oxford: Blackwell.

Sakamoto, Y., M. Ishiguro, and G. Kitagawa. 1986. Akaike Information Criterion Statistics. Dordrecht, The Netherlands: Reidel.

Schwarz, G. 1978. Estimating the dimension of a model. Annals of Statistics 6: 461-464. https://doi.org/10.1214/aos/1176344136.

\section*{Also see}
[R] estat ic - Display information criteria
[R] estimates stats - Model-selection statistics

\section*{Title}
icc - Intraclass correlation coefficients

Description
Menu
Options for one-way RE model
Remarks and examples
Methods and formulas
Also see

Quick start
Syntax
Options for two-way RE and ME models
Stored results
References

\section*{Description}
icc estimates intraclass correlations for one-way random-effects models, two-way random-effects models, or two-way mixed-effects models for both individual and average measurements. Intraclass correlations measuring consistency of agreement or absolute agreement of the measurements may be estimated.

\section*{Quick start}

Individual and average absolute-agreement intraclass correlation coefficients (ICCs) for ratings y of targets identified by tid in a one-way random-effects model
icc y tid
Same as above, but test that the individual and average ICCs are equal to 0.5
icc y tid, testvalue(.5)
Absolute-agreement ICCs for targets identified by tid and raters identified by rid in a two-way random-effects model
icc y tid rid
Same as above, but estimate consistency-of-agreement ICCs
icc y tid rid, consistency
Consistency-of-agreement ICCs when estimating random effects for targets and fixed effects for raters in a mixed-effects model
icc y tid rid, mixed
Same as above, but estimate absolute-agreement ICCs
icc y tid rid, mixed absolute
Same as above, but report \(90 \%\) confidence intervals and test that ICCs are equal to 0.3
icc y tid rid, mixed absolute level(90) testvalue(.3)

\section*{Menu}

Statistics \(>\) Summaries, tables, and tests \(>\) Summary and descriptive statistics \(>\) Intraclass correlations

\section*{Syntax}

Calculate intraclass correlations for one-way random-effects model icc depvar target \([\) if \(][\mathrm{in}]\) [, oneway_options]

Calculate intraclass correlations for two-way random-effects model icc depvar target rater [if] [in] [, twoway_re_options]

Calculate intraclass correlations for two-way mixed-effects model icc depvar target rater \([\) if \(][\) in], mixed [twoway_me_options]
oneway_options Description

\section*{Main}
absolute estimate absolute agreement; the default
testvalue(\#) test whether intraclass correlations equal \#; default is testvalue(0)

Reporting
level(\#)
format (\% \(\%\) fmt)
set confidence level; default is level (95)
display format for statistics and confidence intervals; default is format (\%9.0g)
twoway_re_options
Main
absolute
consistency
testvalue(\#)

Reporting
level(\#)
format (\% fmt)

Description
estimate absolute agreement; the default estimate consistency of agreement
test whether intraclass correlations equal \#; default is testvalue(0)
set confidence level; default is level (95)
display format for statistics and confidence intervals; default is format (\%9.0g)

\section*{twoway_me_options \\ Description}

\section*{Main}
* mixed
consistency
absolute
testvalue (\#)
estimate intraclass correlations for a mixed-effects model
estimate consistency of agreement; the default
estimate absolute agreement
test whether intraclass correlations equal \#;
default is testvalue (0)

Reporting
level (\#)
format (\% fmt)
set confidence level; default is level (95)
display format for statistics and confidence intervals; default is format ( \(\% 9.0 \mathrm{~g}\) )
* mixed is required.
bootstrap, by, collect, jackknife, and statsby are allowed; see [U] 11.1.10 Prefix commands.

\section*{Options for one-way RE model}
\(\qquad\) Main
absolute specifies that intraclass correlations measuring absolute agreement of the measurements be estimated. This is the default for random-effects models.
testvalue(\#) tests whether intraclass correlations equal \#. The default is testvalue(0).
\(\xrightarrow{\text { Reporting }}\)
level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [R] level.
format (\% fint) specifies how the intraclass correlation estimates and confidence intervals are to be formatted. The default is format \((\% 9.0 \mathrm{~g})\).

\section*{Options for two-way RE and ME models}
\(\qquad\) Main
mixed is required to calculate two-way mixed-effects models. mixed specifies that intraclass correlations for a mixed-effects model be estimated.
absolute specifies that intraclass correlations measuring absolute agreement of the measurements be estimated. This is the default for random-effects models. Only one of absolute or consistency may be specified.
consistency specifies that intraclass correlations measuring consistency of agreement of the measurements be estimated. This is the default for mixed-effects models. Only one of absolute or consistency may be specified.
testvalue(\#) tests whether intraclass correlations equal \#. The default is testvalue(0).
level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [R] level.
format ( \(\%\) fimt) specifies how the intraclass correlation estimates and confidence intervals are to be formatted. The default is format \((\% 9.0 \mathrm{~g})\).

\section*{Remarks and examples}

Remarks are presented under the following headings:
Introduction
One-way random effects
Two-way random effects
Two-way mixed effects
Adoption study
Relationship between ICCs
Tests against nonzero values

\section*{Introduction}

In some disciplines, such as psychology and sociology, data are often measured with error that can seriously affect statistical interpretation of the results. Thus, it is important to assess the amount of measurement error by evaluating the consistency or reliability of measurements. The intraclass correlation coefficient (ICC) is often used to measure the consistency or homogeneity of measurements.

Several versions of ICCs are introduced in the literature depending on the experimental design and goals of the study (see, for example, Shrout and Fleiss [1979] and McGraw and Wong [1996a]). Following Shrout and Fleiss (1979), we describe various forms of ICCs in the context of a reliability study of ratings of different targets (or objects of measurements) by several raters.

Consider \(n\) targets (for example, students, patients, athletes) that are randomly sampled from a population of interest. Each target is rated independently by a set of \(k\) raters (for example, teachers, doctors, judges). One rating per target and rater is obtained. It is of interest to determine the extent of the agreement of the ratings.

As noted by Shrout and Fleiss (1979) and McGraw and Wong (1996a), you need to answer several questions to decide what version of ICC is appropriate to measure the agreement in your study:
1. Is a one-way or two-way analysis-of-variance model appropriate for your study?
2. Are differences between raters' mean ratings relevant to the reliability of interest?
3. Is the unit of analysis an individual rating or the mean rating over several raters?
4. Is the consistency of agreement or the absolute agreement of ratings of interest?

Three types of analysis-of-variance models are considered for the reliability study: one-way random effects, two-way random effects, and two-way mixed effects. Mixed models contain both fixed effects and random effects. In the one-way random-effects model, each target is rated by a different set of \(k\) independent raters, who are randomly drawn from the population of raters. The target is the only random effect in this model; the effects due to raters and possibly due to rater-and-target interaction cannot be separated from random error. In the two-way random-effects model, each target is rated by the same set of \(k\) independent raters, who are randomly drawn from the population of raters. The random effects in this model are target and rater and possibly their interaction, although in the absence of repeated measurements for each rater on each target, the effect of an interaction cannot be separated from random error. In the two-way mixed-effects model, each target is rated by the
same set of \(k\) independent raters. Because they are the only raters of interest, rater is a fixed effect. The random effects are target and possibly target-and-rater interaction, but again the interaction effect cannot be separated from random error without repeated measurements for each rater and target. The definition of ICC depends on the chosen random-effects model; see Methods and formulas for details.

In summary, use a one-way model if there are no systematic differences in measurements due to raters and use a two-way model otherwise. If you want to generalize your results to a population of raters from which the observed raters are sampled, use a two-way random-effects model, treating raters as random. If you are interested only in the effects of the observed \(k\) raters, use a two-way mixed-effects model, treating raters as fixed. For example, suppose you compare judges' ratings of targets from different groups. If you use the combined data from \(k\) judges to compare the groups, the random-effects model is appropriate. If you compare groups separately for each judge and then pool the differences, the mixed-effects model is appropriate.

The definition of ICC also depends on the unit of analysis in a study-whether the agreement is measured between individual ratings (individual ICC) or between the averages of ratings over several raters (average ICC). The data on individual ratings are more common. The data on average ratings are typically used when individual ratings are deemed unreliable. The average ICC can also be used when teams of raters are used to rate a target. For example, the ratings of teams of physicians may be evaluated in this manner. When the unit of analysis is an average rating, you should remember that the interpretation of ICC pertains to average ratings and not individual ratings.

Finally, depending on whether consistency of agreement or absolute agreement is of interest, two types of ICC are used: consistency-of-agreement ICC (CA-ICC) and absolute-agreement ICC (AA-ICC). Under consistency of agreement, the scores are considered consistent if the scores from any two raters differ by the same constant value for all targets. This implies that raters give the same ranking to all targets. Under absolute agreement, the scores are considered in absolute agreement if the scores from all raters match exactly.

For example, suppose we observe three targets and two raters. The ratings are \((2,4),(4,6)\), and \((6,8)\), with rater 1 giving the scores \((2,4,6)\) and rater 2 giving the scores \((4,6,8)\), two points higher than rater 1 . The CA-ICC between individual ratings is 1 because the scores from rater 1 and rater 2 differ by a constant value (two points) for all targets. That rater 1 gives lower scores than rater 2 is deemed irrelevant under the consistency measure of agreement. The raters have the same difference of opinion on every target, and the variation between raters that is caused by this difference is not relevant. On the other hand, the AA-ICC between individual ratings is \(8 / 12=0.67\), where 8 is the estimated between-target variance and 12 is the estimated total variance of ratings.

Either CA-ICC or AA-ICC can serve as a useful measure of agreement depending on whether rater variability is relevant for determining the degree of agreement. As McGraw and Wong (1996a) point out, CA-ICC is useful when comparative judgments are made about objects of measurement. The CA-ICC represents correlation when the rater is fixed; the AA-ICC represents correlation when the rater is random.

See Shrout and Fleiss (1979) and McGraw and Wong (1996a) for more detailed guidelines about the choice of appropriate ICC.

Shrout and Fleiss (1979) and McGraw and Wong (1996a) describe 10 versions of ICCs based on the concepts above: individual and average AA-ICCs for a one-way model (consistency of agreement is not defined for this model); individual and average AA-ICCs and CA-ICCs for a two-way random-effects model; and individual and average AA-ICCs and CA-ICCs for a two-way mixed-effects model. Although each of these ICCs has its own definition and interpretation, the estimators for some are identical, leading to the same estimates of those ICCs; see Relationship between ICCs and Methods and formulas for details.

The icc command calculates ICCs for each of the three analysis-of-variance models. You can use option absolute to compute AA-ICCs or option consistency to compute CA-ICCs. By default, icc computes ICCs corresponding to the correlation between ratings and between average ratings made on the same target: AA-ICC for a random-effects model and CA-ICC for a mixed-effects model. As pointed out by Shrout and Fleiss (1979), although the data on average ratings might be needed for reliability, the generalization of interest might be individuals. For this reason, icc reports ICCs for both units, individual and average, for each model.

In addition to estimates of ICCs, icc provides confidence intervals and one-sided \(F\) tests. The \(F\) test of \(H_{o}: \rho=0\) versus \(H_{a}: \rho>0\) is the same for the individual and average ICCs, so icc reports one test. This is not true, however, for nonzero null hypotheses (see Tests against nonzero values for details), so icc reports a separate test in this case.

The icc command requires data in long form; see [D] reshape for how to convert data in wide form to long form. The data must also be balanced and contain one observation per target and rater. For unbalanced data, icc omits all targets with fewer than \(k\) ratings from computation. Under one-way models, \(k\) is determined as the largest number of observed ratings for a target. Under two-way models, \(k\) is the number of unique raters. If multiple observations per target and rater are detected, icc issues an error.

We demonstrate the use of icc using datasets from Shrout and Fleiss (1979) and McGraw and Wong (1996a). In the next three sections, we use an example from table 2 of Shrout and Fleiss (1979) with six targets and four judges. For instructional purposes, we analyze these data under each of the three different models: one-way random effects, two-way random effects, and two-way mixed effects.

\section*{One-way random effects}

In the one-way random-effects model, we assume that the \(n\) targets being rated are randomly selected from the population of potential targets. Each is rated by a different set of \(k\) raters randomly drawn from the population of potential raters. McGraw and Wong (1996a) describe an example of this setting, where behavioral genetics data are used to assess familial resemblance. Family units can be viewed as "targets", and children can be viewed as "raters". By taking a measurement on a child of the family unit, we obtain the "rating" of the family unit by the "child-rater". In this case, we can use ICC to assess similarity between children within a family or, in other words, assess if there is a family effect in these data.

As we mentioned in the introduction, only AA-ICC is defined for a one-way model. The consistency of agreement is not defined in this case, as each target is evaluated by a different set of raters. Thus, there is no between-rater variability in this model.

In a one-way model, the AA-ICC corresponds to the correlation coefficient between ratings within a target. It is also a ratio of the between-target variance of ratings to the total variance of ratings, the sum of the between-target and error variances.
\(>\) Example 1: One-way random-effects ICCs
Consider data from table 2 of Shrout and Fleiss (1979) stored in judges.dta. The data contain 24 ratings of \(n=6\) targets by \(k=4\) judges. We list the first eight observations:
```

. use https://www.stata-press.com/data/r18/judges
(Ratings of targets by judges)
. list in 1/8, sepby(target)

```
\begin{tabular}{l|rrr|}
\cline { 2 - 3 } & rating & target & judge \\
\cline { 2 - 4 } 1. & 9 & 1 & 1 \\
2. & 2 & 1 & 2 \\
3. & 5 & 1 & 3 \\
4. & 8 & 1 & 4 \\
\cline { 2 - 4 } 5. & 6 & 2 & 1 \\
6. & 1 & 2 & 2 \\
7. & 3 & 2 & 3 \\
8. & 2 & 2 & 4 \\
\hline
\end{tabular}

For a moment, let's ignore that targets are rated by the same set of judges. Instead, we assume that a different set of four judges is used to rate each target. In this case, the only systematic variation in the data is due to targets, so the one-way random-effects model is appropriate.

We use icc to estimate the intraclass correlations for these data. To compute ICCs for a one-way model, we specify the dependent variable rating followed by the target variable target:
\begin{tabular}{|c|c|c|c|}
\hline \multicolumn{4}{|l|}{Intraclass correlations One-way random-effects model Absolute agreement} \\
\hline \multirow[t]{2}{*}{Random effects: target} & & of targets & 6 \\
\hline & Num & of raters & 4 \\
\hline rating & ICC & [95\% conf & interval] \\
\hline \multirow[t]{2}{*}{Individual Average} & . 1657418 & -. 1329323 & .7225601 \\
\hline & . 4427971 & -. 8844422 & . 9124154 \\
\hline \multicolumn{4}{|l|}{F test that} \\
\hline \multicolumn{4}{|l|}{ICC=0.00: F \(5.0,18.0)=1.79 \quad\) Prob \(>\mathrm{F}=0.165\)} \\
\hline \multicolumn{4}{|l|}{Note: ICCs estimate correlations between individual measurement and between average measurements made on the same target.} \\
\hline
\end{tabular}
icc reports the AA-ICCs for both individual and average ratings. The individual AA-ICC corresponds to ICC(1) in McGraw and Wong (1996a) or ICC(1,1) in Shrout and Fleiss (1979). The average AA-ICC corresponds to \(\operatorname{ICC}(k)\) in McGraw and Wong (1996a) or \(\operatorname{ICC}(1, k)\) in Shrout and Fleiss (1979).

The estimated correlation between individual ratings is 0.17 , indicating little similarity between ratings within a target, low reliability of individual target ratings, or no target effect. The estimated intraclass correlation between ratings averaged over \(k=4\) judges is higher, 0.44 . (The average ICC will typically be higher than the individual ICC.) The estimated intraclass correlation measures the similarity or reliability of mean ratings from groups of four judges. We do not have statistical evidence that either ICC is different from zero based on reported confidence intervals and the one-sided \(F\) test.

Note that although the estimates of ICCs cannot be negative in this setting, the lower bound of the computed confidence interval may be negative. A common ad-hoc way of handling this is to truncate the lower bound at zero.

The estimates of both the individual and the average AA-ICC are also computed by the loneway command (see [R] loneway), which performs a one-way analysis of variance.

\section*{- Technical note}

Mean rating is commonly used when individual rating is unreliable because the reliability of a mean rating is always higher than the reliability of the individual rating when the individual reliability is positive.

In the previous example, we estimated low reliability of the individual ratings of a target, 0.17. The reliability increased to 0.44 for the ratings averaged over four judges. What if we had more judges?

We can use the Spearman-Brown formula (Spearman 1910; Brown 1910) to compute the \(m\)-average ICC based on the individual ICC:
\[
\operatorname{ICC}(m)=\frac{m \operatorname{ICC}(1)}{1+(m-1) \operatorname{ICC}(1)}
\]

Using this formula for the previous example, we find that the mean reliability over, say, 10 judges is \(10 \times 0.17 /(1+9 \times 0.17)=0.67\).

Alternatively, we can invert the Spearman-Brown formula to determine the number of judges (or the number of ratings of a target) we need to achieve the desired reliability. Suppose we would like an average reliability of 0.9 , then
\[
m=\frac{\operatorname{ICC}(m)\{(1-\operatorname{ICC}(1))\}}{\operatorname{ICC}(1)\{1-\operatorname{ICC}(m)\}}=\frac{0.9(1-0.17)}{0.17(1-0.9)}=44
\]

See, for example, Bliese (2000) for other examples.

\section*{Two-way random effects}

As before, we assume that the targets being rated are randomly selected from the population of potential targets. We now also assume that each target is evaluated by the same set of \(k\) raters, who have been randomly sampled from the population of raters. In this scenario, we want to generalize our findings to the population of raters from which the observed \(k\) raters were sampled. For example, suppose we want to estimate the reliability of doctors' evaluations of patients with a certain condition. Unless the reliability at a specific hospital is of interest, the doctors may be interchanged with others in the relevant population of doctors.

As for a one-way model, the AA-ICC corresponds to the correlation between measurements on the same target and is also a ratio of the between-target variance to the total variance of measurements in a two-way random-effects model. The total variance is now the sum of the between-target, between-rater, and error variances. Unlike a one-way model, the CA-ICC can be computed for a two-way randomeffects model when the consistency of agreement is of interest rather than the absolute agreement. The CA-ICC is also the ratio of the between-target variance to the total variance, but the total variance does not include the between-rater variance because the between-rater variability is irrelevant for the consistency of agreement.

Again, the two versions, individual and average, are available for each ICC.

\section*{> Example 2: Two-way random-effects ICCs}

Continuing with example 1 , recall that we previously ignored that each target is rated by the same set of four judges and instead assumed different sets of judges. We return to the original data setting. We want to evaluate the agreement between judges' ratings of targets in a population represented by the observed set of four judges.

In a two-way model, we must specify both the target and the rater variables. In icc, we now additionally specify the rater variable judge following the target variable target; the random-effects model is assumed by default.
. icc rating target judge
\begin{tabular}{l} 
Intraclass correlations \\
Two-way random-effects model \\
Absolute agreement
\end{tabular}
\begin{tabular}{r} 
Random effects: target \\
Random effects: judge
\end{tabular}
\begin{tabular}{rrrr} 
rating & Number of targets \(=\) & 6 \\
\hline Number of raters \(=\) & 4 \\
\hline Average & ICC & [95\% conf. interval] \\
\hline
\end{tabular}
```

F test that
ICC=0.00: F(5.0, 15.0) = 11.03 Prob > F = 0.000
Note: ICCs estimate correlations between individual measurements
and between average measurements made on the same target.

```

As for a one-way random-effects model, icc by default reports AA-ICCs that correspond to the correlation between ratings on a target. Notice that both individual and average ICCs are larger in the two-way random-effects model than in the previous one-way model- 0.29 versus 0.17 and 0.62 versus 0.44 , respectively. We also have statistical evidence to reject the null hypothesis that neither ICC is zero based on confidence intervals and the \(F\) test. If a one-way model is used when a two-way model is appropriate, the true ICC will generally be underestimated.

The individual AA-ICC corresponds to \(\operatorname{ICC}(A, 1)\) in McGraw and Wong \((1996\) a) or ICC \((2,1)\) in Shrout and Fleiss (1979). The average AA-ICC corresponds to ICC( \(A, k\) ) in McGraw and Wong (1996a) or \(\operatorname{ICC}(2, k)\) in Shrout and Fleiss (1979).

Instead of the absolute agreement, we can also assess the consistency of agreement. The individual and average CA-ICCs are considered in McGraw and Wong (1996a) and denoted as ICC(C,1) and ICC(C, \(k\) ), respectively. These ICCs are not considered in Shrout and Fleiss (1979) because they are not correlations in the strict sense. Although CA-ICCs do not estimate correlation, they can provide useful information about the reliability of the raters. McGraw and Wong (1996a) note that the practical value of the individual and average CA-ICCs in the two-way random-effects model setting is well documented in measurement theory, citing Hartmann (1982) and Suen (1988).

To estimate the individual and average CA-ICCs, we specify the consistency option:
\begin{tabular}{|c|c|c|c|}
\hline \multicolumn{4}{|l|}{Intraclass correlations Two-way random-effects model Consistency of agreement} \\
\hline \multirow[t]{3}{*}{\begin{tabular}{l}
Random effects: target \\
Random effects: judge
\(\qquad\)
\end{tabular}} & \multicolumn{3}{|r|}{\multirow[t]{2}{*}{\begin{tabular}{l}
Number of targets \(=\) \\
Number of raters =
\end{tabular}}} \\
\hline & & & \\
\hline & ICC & [95\% conf & interval] \\
\hline Individual & . 7148407 & . 3424648 & . 9458583 \\
\hline Average & . 9093155 & . 6756747 & . 9858917 \\
\hline \multicolumn{4}{|l|}{\(F\) test that} \\
\hline
\end{tabular}

We estimate that the consistency of agreement of ratings in the considered population of raters is high, 0.71, based on the individual CA-ICC. On the other hand, the absolute agreement of ratings is low, 0.29 , based on the individual AA-ICC from the previous output.

The measure of consistency of agreement among means, the average CA-ICC, is equivalent to Cronbach's alpha (Cronbach 1951); see [MV] alpha. The individual CA-ICC can also be equivalent to the Pearson's correlation coefficient between raters when \(k=2\); see McGraw and Wong (1996a) for details.

In the next example, we will see that the actual estimates of the individual and average AA-ICCs and CA-ICCs are the same whether we examine a random-effects model or a mixed-effects model. The differences between these ICCs are in their definitions and interpretations.

\section*{Two-way mixed effects}

As in a two-way random-effects model, we assume that the targets are randomly selected from the population of potential targets and that each is evaluated by the same set of \(k\) raters. In a mixed-effects model, however, we assume that these raters are the only raters of interest. So as before, the targets are random, but now the raters are fixed.

In the two-way mixed-effects model, the fixed effect of the rater does not contribute to the betweenrater random variance component to the total variance. As such, the definitions and interpretations of ICCs are different in a mixed-effects model than in a random-effects model. However, the estimates of ICCs as well as test statistics and confidence intervals are the same. The only exceptions are average AA-ICCs and CA-ICCs. These are not estimable in a two-way mixed-effects model including an interaction term between target and rater; see Relationship between ICCs and Methods and formulas for details.

In a two-way mixed-effects model, the CA-ICC corresponds to the correlation between measurements on the same target. As pointed out by Shrout and Fleiss (1979), when the rater variance is ignored, the correlation coefficient is interpreted in terms of rater consistency rather than rater absolute agreement. Formally, the CA-ICC is the ratio of the covariance between measurements on the target to the total variance of the measurements. The AA-ICC corresponds to the same ratio, but includes a variance of the fixed factor, rater, in its denominator.

\section*{> Example 3: Two-way mixed-effects ICCs}

Continuing with example 2, suppose that we are now interested in assessing the agreement of ratings from only the observed four judges. The judges are now fixed effects, and the appropriate model is a two-way mixed-effects model.

To estimate ICCs for a two-way mixed-effects model, we specify the mixed option with icc:
```

. icc rating target judge, mixed
Intraclass correlations
Two-way mixed-effects model
Consistency of agreement
Random effects: target Number of targets = 6
Fixed effects: judge Number of raters = 4

| rating | ICC | [95\% conf. interval] |  |
| ---: | ---: | :--- | ---: |
| Individual | .7148407 | .3424648 | .9458583 |
| Average | .9093155 | .6756747 | .9858917 |

F test that
ICC=0.00: F(5.0, 15.0) = 11.03 Prob > F = 0.000

```
Note: ICCs estimate correlations between individual measurements
        and between average measurements made on the same target.

As we described in the introduction, icc by default reports ICCs corresponding to the correlations. So, for a mixed-effects model, icc reports CA-ICCs by default. The individual and average CA-ICCs are denoted as \(\operatorname{ICC}(3,1)\) and \(\operatorname{ICC}(3, k)\) in Shrout and Fleiss (1979) and \(\operatorname{ICC}(C, 1)\) and \(\operatorname{ICC}(C, k)\) in McGraw and Wong (1996a).

Our estimates of the individual and average CA-ICCs are identical to the CA-ICC estimates obtained under the two-way random-effects model in example 2, but our interpretation of the results is different. Under a mixed-effects model, 0.71 and 0.91 are the estimates, respectively, of the correlation between individual measurements and the correlation between average measurements made on the same target.

We can also estimate the AA-ICCs in this setting by specifying the absolute option:
\begin{tabular}{|c|c|c|c|}
\hline \multicolumn{4}{|l|}{Intraclass correlations Two-way mixed-effects model Absolute agreement} \\
\hline \multirow[t]{2}{*}{\begin{tabular}{r}
\begin{tabular}{r} 
Random effects: target \\
Fixed effects: judge
\end{tabular} \\
\hline rating
\end{tabular}} & \multicolumn{3}{|r|}{```
Number of targets = 6
Number of raters = 4
```} \\
\hline & ICC & [95\% conf & interval] \\
\hline Individual & . 2897638 & . 0187865 & . 7610844 \\
\hline Average & . 6200505 & . 0711368 & . 927232 \\
\hline
\end{tabular}

The intraclass correlation estimates match the individual and average AA-ICCs obtained under the two-way random-effects model in example 2; but in a mixed-effects model, they do not represent correlations. We demonstrate the use of an individual AA-ICC in a mixed-effects setting in the next example.

The AA-ICCs under a mixed-effects model are not considered by Shrout and Fleiss (1979). They are denoted as \(\operatorname{ICC}(A, 1)\) and \(\operatorname{ICC}(A, k)\) in McGraw and Wong (1996a).

\section*{Adoption study}

In this section, we consider the adoption study described in McGraw and Wong (1996a). Adoption studies commonly include two effects of interest. One is the mean difference between the adopted child and its biological parents. It is used to determine if characteristics of adopted children differ on average from those of their biological parents. Another effect of interest is the correlation between genetically paired individuals and genetically unrelated individuals who live together. This effect is used to evaluate the impact of genetic differences on individual differences.

As discussed in McGraw and Wong (1996a), a consistent finding from adoption research using IQ as a trait characteristic is that while adopted children typically have higher IQs than their biological parents, their IQs correlate better with those of their biological parents than with those of their adoptive parents. Both effects are important, and there is additional need to reconcile the two findings. McGraw and Wong (1996a) propose to use the individual AA-ICC for this purpose.

\section*{> Example 4: Absolute-agreement ICC in a mixed-effects model}

The adoption.dta dataset contains the data from table 6 of McGraw and Wong (1996a) on IQ scores:
\begin{tabular}{|c|c|c|c|c|}
\hline \multicolumn{5}{|l|}{\begin{tabular}{l}
- use https://www.stata-press.com/data/r18/adoption (Biological mother and adopted child IQ scores) \\
. describe
\end{tabular}} \\
\hline \multicolumn{5}{|l|}{Contains data from https://www.stata-press.com/data/r18/adoption.dta} \\
\hline \multicolumn{2}{|l|}{Observations:} & \multicolumn{2}{|l|}{20} & Biological mother and adopted child IQ scores \\
\hline \multicolumn{2}{|l|}{\multirow[t]{2}{*}{Variables:}} & \multirow[t]{2}{*}{5} & & 15 May 2022 13:50 \\
\hline & & & & (_dta has notes) \\
\hline Variable & Storage & Display & Value & Variable label \\
\hline family & byte & \%9.0g & \multirow{4}{*}{mcvalues} & Adoptive family ID \\
\hline mc & byte & \%9.0g & & Whether mother or child \\
\hline \multirow[t]{2}{*}{iq3} & \multirow[t]{2}{*}{int} & \multirow[t]{2}{*}{\(\% 9.0 \mathrm{~g}\)} & & IQ scores, mother-child \\
\hline & & & & difference of 3 pts \\
\hline \multirow[t]{2}{*}{iq9} & \multirow[t]{2}{*}{int} & \multirow[t]{2}{*}{\%9.0g} & & IQ scores, mother-child \\
\hline & & & & difference of 9 pts \\
\hline \multirow[t]{2}{*}{iq15} & \multirow[t]{2}{*}{int} & \multirow[t]{2}{*}{\%9.0g} & & IQ scores, mother-child \\
\hline & & & & difference of 15 pts \\
\hline
\end{tabular}

Sorted by:
The family variable contains adoptive family identifiers, the mc variable records a mother or a child, and the iq3, iq9, and iq15 variables record IQ scores with differences between mother and child mean IQ scores of 3,9 , and 15 points, respectively.
. by mc, sort: summarize iq*
\begin{tabular}{r|ccccc}
-> mc = Mother \\
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline iq3 & 10 & 97 & 15.0037 & 62 & 116 \\
iq9 & 10 & 91 & 15.0037 & 56 & 110 \\
iq15 & 10 & 85 & 15.0037 & 50 & 104 \\
\hline mc = Child & & & & & \\
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline iq3 & 10 & 100 & 15.0037 & 65 & 119 \\
iq9 & 10 & 100 & 15.0037 & 65 & 119 \\
iq15 & 10 & 100 & 15.0037 & 65 & 119
\end{tabular}

The variances of the mother and child IQ scores are the same.
Children are fixed effects, so the mixed-effects model is appropriate for these data. We want to compare individual CA-ICC with individual AA-ICC for each of the three IQ variables. We could issue a separate icc command for each of the three IQ variables to obtain the intraclass correlations. Instead, we use reshape to convert our data to long form with one iq variable and the new diff variable recording mean differences:
```

. reshape long iq, i(family mc) j(diff)
(j = 3 9 15)

| Data | Wide | -> | Long |
| :--- | ---: | :--- | :--- |
| Number of observations | 20 | $\rightarrow>$ | 60 |
| Number of variables | 5 | $\rightarrow$ | 4 |
| j variable (3 values) |  | -> | diff |

xij variables:
iq3 iq9 iq15 -> iq

```

We can now use the by prefix with icc to estimate intraclass correlations for the three groups of interest:
. by diff, sort: icc iq family mc, mixed
```

-> diff = 3

```

Intraclass correlations
Two-way mixed-effects model
Consistency of agreement
Random effects: family Number of targets \(=10\)
Fixed effects: mc Number of raters \(=\) 2
\begin{tabular}{r|rrr}
\hline iq & ICC & [95\% conf. interval] \\
\hline Individual & .7142152 & .1967504 & .920474 \\
Average & .8332853 & .3288078 & .9585904 \\
\hline
\end{tabular}

F test that
ICC=0.00: F (9.0, 9.0) \(=6.00 \quad\) Prob \(>\mathrm{F}=0.007\)
Note: ICCs estimate correlations between individual measurements and between average measurements made on the same target.
```

-> diff = 9
Intraclass correlations
Two-way mixed-effects model
Consistency of agreement
Random effects: family Number of targets = 10
Fixed effects: mc Number of raters = 2

| iq | ICC | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: |
| Individual | .7142152 | .1967504 | .920474 |
| Average | .8332853 | .3288078 | .9585904 |

F test that
ICC=0.00: F(9.0, 9.0) = 6.00 Prob > F = 0.007

```

Note: ICCs estimate correlations between individual measurements and between average measurements made on the same target.
```

-> diff = 15

```
(output omitted)
The estimated CA-ICCs are the same in all three groups and are equal to the corresponding estimates of the Pearson's correlation coefficients because mothers' and childrens' IQ scores have the same variability. The scores differ only in means, and mean differences are irrelevant when measuring the consistency of agreement.

The AA-ICCs, however, differ across the three groups:
. by diff, sort: icc iq family mc, mixed absolute
\begin{tabular}{|c|c|c|c|}
\hline \multicolumn{4}{|l|}{Intraclass correlations Two-way mixed-effects model Absolute agreement} \\
\hline ```
Random effects: family
    Fixed effects: mc
``` & Num & \begin{tabular}{l}
targets \\
raters
\end{tabular} & 10 \\
\hline iq & ICC & [95\% conf & interval] \\
\hline Individual Average & \[
\begin{aligned}
& .7204023 \\
& .8374812
\end{aligned}
\] & \[
\begin{aligned}
& .2275148 \\
& .3706917
\end{aligned}
\] & \[
\begin{aligned}
& .9217029 \\
& .9592564
\end{aligned}
\] \\
\hline \multicolumn{4}{|l|}{\(F\) test that
\[
I C C=0.00: F(9.0,9.0)=6.00 \quad \text { Prob }>F=0.007
\]} \\
\hline \multicolumn{4}{|l|}{} \\
\hline \multicolumn{4}{|l|}{Intraclass correlations Two-way mixed-effects model Absolute agreement} \\
\hline ```
Random effects: family
    Fixed effects: mc
``` & \multicolumn{2}{|r|}{\begin{tabular}{l}
Number of targets \(=\) \\
Number of raters
\end{tabular}} & 10
2 \\
\hline iq & ICC & [95\% conf & interval] \\
\hline Individual & . 6203378 & . 0293932 & . 8905025 \\
\hline Average & . 7656895 & . 0571077 & . 9420802 \\
\hline
\end{tabular}
```

F test that
ICC=0.00: F(9.0, 9.0) = 6.00 Prob > F = 0.007

```
-> diff = 15
Intraclass correlations
Two-way mixed-effects model
Absolute agreement
Random effects: family Number of targets \(=10\)
    Fixed effects: mc Number of raters \(=\quad 2\)
\begin{tabular}{r|rcr}
\hline iq & ICC & [95\% conf. interval] \\
\hline Individual & .4854727 & -.1194157 & .8466905 \\
Average & .6536272 & -.2712191 & .9169815 \\
\hline
\end{tabular}
\(F\) test that
    ICC=0.00: F \((9.0,9.0)=6.00 \quad\) Prob \(>F=0.007\)

As the mean differences increase, the AA-ICCs decrease. Their attenuation reflects the difference in means between biological mother and child IQs while still measuring their agreement. Notice that for small mean differences, the estimates of AA-ICCs and CA-ICCs are very similar.

Note that our estimates match those given in McGraw and Wong (1996b), who correct the original table 6 of McGraw and Wong (1996a).

\section*{Relationship between ICCs}

In examples 2 and 3, we saw that the estimates of AA-ICCs and CA-ICCs are the same for twoway random-effects and two-way mixed-effects models. In this section, we consider the relationship between various forms of ICCs in more detail; also see Methods and formulas.

There are 10 different versions of ICCs, but only 6 different estimators are needed to compute them. These estimators include the two estimators for the individual and average AA-ICCs in a one-way model, the two estimators for the individual and average AA-ICCs in two-way models, and the two estimators for the individual and average CA-ICCs in two-way models.

Only individual and average AA-ICCs are defined for the one-way model. The estimates of AA-ICCs based on the one-way model will typically be smaller than individual and average estimates of AA-ICCs and CA-ICCs based on two-way models. The estimates of individual and average CA-ICCs will typically be larger than the estimates of individual and average AA-ICCs.

Although AA-ICCs and CA-ICCs have the same respective estimators in two-way random-effects and mixed-effects models, their definitions and interpretations are different. The AA-ICCs based on a random-effects model contain the between-rater variance component in the denominator of the variance ratio. The AA-ICCs based on a mixed-effects model contain the variance of the fixed-factor rater instead of the random between-rater variability. The AA-ICCs in a random-effects model represent correlations between any two measurements made on a target. The AA-ICCs in a mixed-effects model measure absolute agreement of measurements treating raters as fixed. The CA-ICCs for random-effects and mixed-effects models have the same definition but different interpretations. The CA-ICCs represent correlations between any two measurements made on a target in a mixed-effects model but estimate the degree of consistency among measurements treating raters as random in a random-effects model. The difference in the definitions of AA-ICCs and CA-ICCs is that CA-ICCs do not contain the between-rater variance in the denominator of the variance ratio.

For two-way models, the definitions and interpretations (but not the estimators) of ICCs also depend on whether the model contains an interaction between target and rater. For two-way models with interaction, ICCs include an additional variance component for the target-rater interaction in the denominator of the variance ratio. This component cannot be separated from random error because there is only one observation per target and rater.

Also, under a two-way mixed-effects model including interaction, the interaction components are not mutually independent, as they are in a two-way random-effects model. The considered version of the mixed-effects model places a constraint on the interaction effects-the sum of the interaction effects over levels of the fixed factor is zero; see, for example, chapter 7 in Kuehl (2000) for an introductory discussion of mixed models. In this version of the model, there is a correlation between the interaction effects. Specifically, the two interaction effects for the same target and two different raters are negatively correlated. As a result, the estimated intraclass correlation can be negative under a two-way mixed-effects model with interaction. Also, average AA-ICC and average CA-ICC cannot be estimated in a two-way mixed-effects model including interaction; see Methods and formulas and McGraw and Wong (1996a) for details.

\section*{Tests against nonzero values}

It may be of interest to test whether the intraclass correlation is equal to a value other than zero. icc supports testing against positive values through the use of the testvalue() option. Specifying testvalue(\#) provides a one-sided hypothesis test of \(H_{o}: \rho=\#\) versus \(H_{a}: \rho>\#\). The test is provided separately for both individual and average ICCs.

\section*{> Example 5: Testing ICC against a nonzero value}

We return to the two-way random-effects model for the judge and target data from Shrout and Fleiss (1979). Suppose we want to test whether the individual and average AA-ICCs are each equal to 0.2 . We specify the testvalue (0.2) option with icc:
```

. use https://www.stata-press.com/data/r18/judges, clear
(Ratings of targets by judges)
. icc rating target judge, testvalue(0.2)
Intraclass correlations
Two-way random-effects model
Absolute agreement

| Random effects: target | Number of targets $=$ | 6 |
| :--- | :--- | :--- |
| Random effects: judge | Number of raters $=$ | 4 |


| rating | ICC | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: |
| Individual | .2897638 | .0187865 | .7610844 |
| Average | .6200505 | .0711368 | .927232 |

F test that

```
Note: ICCs estimate correlations between individual measurements
    and between average measurements made on the same target.

We reject the null hypothesis that the average AA-ICC, labeled as \(\operatorname{ICC}(k)\) in the output, is equal to 0.2 , but we do not have statistical evidence to reject the null hypothesis that the individual AA-ICC, labeled as \(\operatorname{ICC}(1)\), is equal to 0.2 .

\section*{Stored results}
icc stores the following in \(r()\) :
Scalars
r(N_target)
r(N_rater)
r(icc_i)
r(icc_i_F)
r(icc_i_df1)
r(icc_i_df2)
r(icc_i_p)
r(icc_i_lb)
r(icc_i_ub)
r(icc_avg)
r(icc_avg_F)
r(icc_avg_df1)
r(icc_avg_df2)
r(icc_avg_p)
r(icc_avg_lb)
r(icc_avg_ub)
\(r\) (testvalue)
r(level)
Macros
\(r\) (model)
r(depvar)
\(r\) (target)
r(rater)
r(type)
number of targets
number of raters
intraclass correlation for individual measurements
\(F\) test statistic for individual ICC
numerator degrees of freedom for \(r\) (icc_i_F)
denominator degrees of freedom for \(\mathbf{r}\) (icc_i_F)
\(p\)-value for F test of individual ICC
lower endpoint for confidence intervals of individual ICC
upper endpoint for confidence intervals of individual ICC
intraclass correlation for average measurements
\(F\) test statistic for average ICC
numerator degrees of freedom for r (icc_avg_F)
denominator degrees of freedom for r (icc_avg_F) \(p\)-value for \(F\) test of average ICC
lower endpoint for confidence intervals of average ICC upper endpoint for confidence intervals of average ICC null hypothesis value confidence level
analysis-of-variance model
name of dependent variable
target variable
rater variable
type of ICC estimated (absolute or consistency)

\section*{Methods and formulas}

We observe \(y_{i j}\), where \(i=1, \ldots, n\) and \(j=1, \ldots, k . y_{i j}\) is the \(j\) th rating on the \(i\) th target. Let \(\alpha=1-l / 100\), where \(l\) is the significance level specified by the user.

Methods and formulas are presented under the following headings:

\author{
Mean squares \\ One-way random effects \\ Two-way random effects \\ Two-way mixed effects
}

\section*{Mean squares}

The mean squares within targets are
\[
\mathrm{WMS}=\sum_{i} \sum_{j} \frac{\left(y_{i j}-\bar{y}_{i .}\right)^{2}}{n(k-1)}
\]
where \(\bar{y}_{i}\). \(=\sum_{j} y_{i j} / k\).
The mean squares between targets are
\[
\mathrm{BMS}=\sum_{i} \frac{\left(\bar{y}_{i .}-\bar{y}_{. .}\right)^{2}}{n-1}
\]
where \(\bar{y}_{. .}=\sum_{i} \bar{y}_{i .} / n\).
These are the only mean squares needed to estimate ICC in the one-way random-effects model. For the two-way models, we need two additional mean squares.

The mean squares between raters are
\[
\mathrm{JMS}=\sum_{j} \frac{\left(\bar{y}_{\cdot j}-\bar{y}_{. .}\right)^{2}}{k-1}
\]
where \(\bar{y}_{\cdot j}=\sum_{i} y_{i j} / n\) and \(\bar{y}_{. .}=\sum_{j} \bar{y}_{\cdot j} / k\).
The residual or error mean square is
\[
\mathrm{EMS}=\frac{\sum_{i} \sum_{j}\left(y_{i j}-\bar{y}\right)^{2}-(k-1) \mathrm{JMS}-(n-1) \mathrm{BMS}}{(n-1)(k-1)}
\]

\section*{One-way random effects}

Under the one-way random-effects model, we observe
\[
\begin{equation*}
y_{i j}=\mu+r_{i}+\epsilon_{i j} \tag{M1}
\end{equation*}
\]
where \(\mu\) is the mean rating, \(r_{i}\) is the target random effect, and \(\epsilon_{i j}\) is random error. The \(r_{i} \mathrm{~s}\) are i.i.d. \(N\left(0, \sigma_{r}^{2}\right) ; \epsilon_{i j} \mathrm{~s}\) are i.i.d. \(N\left(0, \sigma_{\epsilon}^{2}\right)\) and are independent of \(r_{i}\) s. There is no rater effect separate from the residual error because each target is evaluated by a different set of raters.

The individual AA-ICC is the correlation between individual measurements on the same target:
\[
\rho_{1}=\operatorname{ICC}(1)=\operatorname{Corr}\left(y_{i j}, y_{i j^{\prime}}\right)=\frac{\sigma_{r}^{2}}{\sigma_{r}^{2}+\sigma_{\epsilon}^{2}}
\]

The average AA-ICC is the correlation between average measurements of size \(k\) made on the same target:
\[
\rho_{k}=\operatorname{ICC}(k)=\operatorname{Corr}\left(\bar{y}_{i .}, \bar{y}_{i .}^{\prime}\right)=\frac{\sigma_{r}^{2}}{\sigma_{r}^{2}+\sigma_{\epsilon}^{2} / k}
\]

They are estimated by
\[
\begin{aligned}
& \widehat{\rho}_{1}=\widehat{\mathrm{ICC}(1)}=\frac{\mathrm{BMS}-\mathrm{WMS}}{\mathrm{BMS}+(k-1) \mathrm{WMS}} \\
& \widehat{\rho}_{k}=\widehat{\mathrm{ICC}(k)}=\frac{\mathrm{BMS}-\mathrm{WMS}}{\mathrm{BMS}}
\end{aligned}
\]

Confidence intervals. Let \(F_{\text {obs }}=\) BMS \(/ \mathrm{WMS}\), let \(F_{l}\) be the \((1-\alpha / 2) \times 100\) th percentile of the \(F_{n-1, n(k-1)}\) distribution, and let \(F_{u}\) be the \((1-\alpha / 2) \times 100\) th percentile of the \(F_{n(k-1), n-1}\) distribution. Let \(F_{L}=F_{\text {obs }} / F_{l}\) and \(F_{U}=F_{\text {obs }} F_{u}\).

A \((1-\alpha) \times 100 \%\) confidence interval for \(\rho_{1}\) is
\[
\begin{equation*}
\left(\frac{F_{L}-1}{F_{l}+k-1}, \frac{F_{U}-1}{F_{U}+k-1}\right) \tag{1}
\end{equation*}
\]

A \((1-\alpha) \times 100 \%\) confidence interval for \(\rho_{k}\) is
\[
\begin{equation*}
\left(1-\frac{1}{F_{L}}, 1-\frac{1}{F_{U}}\right) \tag{2}
\end{equation*}
\]

Hypothesis tests. Consider a one-sided hypothesis test of \(H_{o}\) : ICC \(=\rho_{0}\) versus \(H_{a}\) : ICC \(>\rho_{0}\).
The test statistic for \(\rho_{1}\) is
\[
\begin{equation*}
F_{\rho_{1}}=\frac{\operatorname{BMS}}{\operatorname{WMS}} \frac{1-\rho_{0}}{1+(k-1) \rho_{0}} \tag{3}
\end{equation*}
\]

The test statistic for \(\rho_{k}\) is
\[
\begin{equation*}
F_{\rho_{k}}=\frac{\mathrm{BMS}}{\mathrm{WMS}}\left(1-\rho_{0}\right) \tag{4}
\end{equation*}
\]

Under the null hypothesis, both \(F_{\rho_{1}}\) and \(F_{\rho_{k}}\) have the \(F_{n-1, n(k-1)}\) distribution. When \(\rho_{0}=0\), the two test statistics coincide.

\section*{Two-way random effects}

In this setting, the target is evaluated by the same set of raters, who are randomly drawn from the population of raters. The underlying models with and without interaction are
\[
\begin{gather*}
y_{i j}=\mu+r_{i}+c_{j}+(r c)_{i j}+\epsilon_{i j}  \tag{M2}\\
y_{i j}=\mu+r_{i}+c_{j}+\epsilon_{i j} \tag{M2A}
\end{gather*}
\]
where \(y_{i j}\) is the rating of the \(i\) th target by the \(j\) th rater, \(\mu\) is the mean rating, \(r_{i}\) is the target random effect, \(c_{j}\) is the rater random effect, \((r c)_{i j}\) is the target-rater random effect, and \(\epsilon_{i j}\) is random error. The \(r_{i} \mathrm{~s}\) are i.i.d. \(N\left(0, \sigma_{r}^{2}\right), c_{j} \mathrm{~s}\) are i.i.d. \(N\left(0, \sigma_{c}^{2}\right),(r c)_{i j} \mathrm{~s}\) are i.i.d. \(N\left(0, \sigma_{r c}^{2}\right)\), and \(\epsilon_{i j} \mathrm{~s}\) are i.i.d. \(N\left(0, \sigma_{\epsilon}^{2}\right)\). Each effect is mutually independent of the others.

Below, we provide formulas for ICCs for model (M2). The corresponding ICCs for model (M2A) can be obtained by setting \(\sigma_{r c}^{2}=0\).

The individual AA-ICC is the correlation between individual measurements on the same target:
\[
\rho_{A, 1}=\operatorname{ICC}(A, 1)=\operatorname{Corr}\left(y_{i j}, y_{i j^{\prime}}\right)=\frac{\sigma_{r}^{2}}{\sigma_{r}^{2}+\sigma_{c}^{2}+\left(\sigma_{r c}^{2}+\sigma_{\epsilon}^{2}\right)}
\]

The average AA-ICC is the correlation between average measurements of size \(k\) made on the same target:
\[
\rho_{A, k}=\operatorname{ICC}(A, k)=\operatorname{Corr}\left(\bar{y}_{i .}, \bar{y}_{i .}^{\prime}\right)=\frac{\sigma_{r}^{2}}{\sigma_{r}^{2}+\left(\sigma_{c}^{2}+\sigma_{r c}^{2}+\sigma_{\epsilon}^{2}\right) / k}
\]

The consistency-of-agreement intraclass correlation for individual measurements, individual CA-ICC, is
\[
\rho_{C, 1}=\operatorname{ICC}(C, 1)=\frac{\sigma_{r}^{2}}{\sigma_{r}^{2}+\left(\sigma_{r c}^{2}+\sigma_{\epsilon}^{2}\right)}
\]

The consistency-of-agreement intraclass correlation for average measurements of size \(k\), average CA-ICC, is
\[
\rho_{C, k}=\operatorname{ICC}(C, k)=\frac{\sigma_{r}^{2}}{\sigma_{r}^{2}+\left(\sigma_{r c}^{2}+\sigma_{\epsilon}^{2}\right) / k}
\]

With one observation per target and rater, \(\sigma_{r c}^{2}\) and \(\sigma_{\epsilon}^{2}\) cannot be estimated separately.

The estimators of intraclass correlations, confidence intervals, and test statistics are the same for models (M2) and (M2A). The estimators of ICCs are
\[
\begin{aligned}
& \left.\widehat{\rho}_{A, 1}=\operatorname{ICC} \widehat{C A}, 1\right)=\frac{\mathrm{BMS}-\mathrm{EMS}}{\mathrm{BMS}+(k-1) \mathrm{EMS}+\frac{k}{n}(\mathrm{JMS}-\mathrm{EMS})} \\
& \widehat{\rho}_{A, k}=\mathrm{ICC} \widehat{(A, k)}=\frac{\mathrm{BMS}-\mathrm{EMS}}{\mathrm{BMS}+\frac{1}{n}(\mathrm{JMS}-\mathrm{EMS})} \\
& \widehat{\rho}_{C, 1}=\operatorname{ICC}(C, 1)=\frac{\mathrm{BMS}-\mathrm{EMS}}{\mathrm{BMS}+(k-1) \mathrm{EMS}} \\
& \widehat{\rho}_{C, k}=\operatorname{ICC(C,\mathrm {k})}=\frac{\mathrm{BMS}-\mathrm{EMS}}{\mathrm{BMS}}
\end{aligned}
\]

Confidence intervals. Let \(a=k \widehat{\rho}_{A, 1} /\left\{n\left(1-\widehat{\rho}_{A, 1}\right)\right\}, b=1+k \widehat{\rho}_{A, 1}(n-1) /\left\{n\left(1-\widehat{\rho}_{A, 1}\right)\right\}\), and
\[
\begin{equation*}
v=\frac{(a \mathrm{JMS}+b \mathrm{EMS})^{2}}{\frac{a^{2} \mathrm{JMS}}{}{ }^{2}-1}+\frac{b^{2} \mathrm{EMS}^{2}}{(n-1)(k-1)} \tag{5}
\end{equation*}
\]

Let \(F_{l}\) be the \((1-\alpha / 2) \times 100\) th percentile of the \(F_{n-1, v}\) distribution and \(F_{u}\) be the \((1-\alpha / 2) \times 100\) th percentile of the \(F_{v, n-1}\) distribution.

A \((1-\alpha) \times 100 \%\) confidence interval for \(\rho_{A, 1}\) is given by \((L, U)\), where
\[
\begin{align*}
L & =\frac{n\left(\mathrm{BMS}-F_{l} \mathrm{EMS}\right)}{F_{l}\{k \mathrm{JMS}+(k n-k-n) \mathrm{EMS}\}+n \mathrm{BMS}} \\
U & =\frac{n\left(F_{u} \mathrm{BMS}-\mathrm{EMS}\right)}{k \mathrm{JMS}+(k n-k-n) \mathrm{EMS}+n F_{u} \mathrm{BMS}} \tag{6}
\end{align*}
\]

A \((1-\alpha) \times 100 \%\) confidence intervals for \(\rho_{A, k}\) is a special case of (6) with \(k=1\), where \(a=\widehat{\rho}_{A, k} /\left\{n\left(1-\widehat{\rho}_{A, k}\right)\right\}, b=1+\widehat{\rho}_{A, k}(n-1) /\left\{n\left(1-\widehat{\rho}_{A, k}\right)\right\}\), and \(v\) is defined in (5).

To define confidence intervals for \(\rho_{C, 1}\) and \(\rho_{C, k}\), let \(F_{\text {obs }}=\) BMS \(/\) EMS, \(F_{l}\) be the \((1-\alpha / 2) \times 100\) th percentile of the \(F_{n-1,(n-1)(k-1)}\) distribution, and \(F_{u}\) be the \((1-\alpha / 2) \times 100\) th percentile of the \(F_{(n-1)(k-1), n-1}\) distribution. Let \(F_{L}=F_{\text {obs }} / F_{l}\) and \(F_{L}=F_{\text {obs }} F_{u}\).

A \((1-\alpha) \times 100 \%\) confidence intervals for \(\rho_{C, 1}\) and \(\rho_{C, k}\) are then as given by (1) and (2) for model (M1).
Hypothesis tests. Consider a one-sided hypothesis test of \(H_{o}\) : ICC \(=\rho_{0}\) versus \(H_{a}\) : ICC \(>\rho_{0}\). Let \(a=k \rho_{0} /\left\{n\left(1-\rho_{0}\right)\right\}\) and \(b=1+k \rho_{0}(n-1) /\left\{n\left(1-\rho_{0}\right)\right\}\).

The test statistic for \(\rho_{A, 1}\) is
\[
F_{\rho_{A, 1}}=\frac{\mathrm{BMS}}{a \mathrm{JMS}+b \mathrm{EMS}}
\]

Under the null hypothesis, \(F_{\rho_{A, 1}}\) has the \(F_{n-1, v}\) distribution, where \(v\) is defined in (5).
The test statistic for \(\rho_{A, k}\) is defined similarly, except \(a=\rho_{0} /\left\{n\left(1-\rho_{0}\right)\right\}\) and \(b=1+\) \(\rho_{0}(n-1) /\left\{n\left(1-\rho_{0}\right)\right\}\). Under the null hypothesis, \(F_{\rho_{A, k}}\) has the \(F_{n-1, v}\) distribution, where \(v\) is defined in (5). When \(\rho_{0}=0\), then \(a=0, b=1\), and the two test statistics coincide.

The test statistics for \(\rho_{C, 1}\) and \(\rho_{C, k}\) are defined by (3) and (4), respectively, with wMS replaced by EMS. Under the null hypothesis, both \(F_{\rho_{C, 1}}\) and \(F_{\rho_{C, k}}\) have the \(F_{n-1,(n-1)(k-1)}\) distribution. They also both have the same value when \(\rho_{0}=0\).

\section*{Two-way mixed effects}

In this setting, every target is evaluated by the same set of judges, who are the only judges of interest. The underlying models with and without interaction are
\[
\begin{gather*}
y_{i j}=\mu+r_{i}+c_{j}+(r c)_{i j}+\epsilon_{i j}  \tag{M3}\\
y_{i j}=\mu+r_{i}+c_{j}+\epsilon_{i j} \tag{M3A}
\end{gather*}
\]
where \(y_{i j}\) is the rating of the \(i\) th target by the \(j\) th rater, \(\mu\) is the mean rating, \(r_{i}\) is the target random effect, \(c_{j}\) is the rater random effect, \((r c)_{i j}\) is an interaction effect between target and rater, and \(\epsilon_{i j}\) is random error. The \(r_{i}\) s are i.i.d. \(N\left(0, \sigma_{r}^{2}\right),(r c)_{i j}\) s are \(N\left(0, \sigma_{r c}^{2}\right)\), and \(\epsilon_{i j}\) s are i.i.d. \(N\left(0, \sigma_{\epsilon}^{2}\right)\). Each random effect is mutually independent of the others. The \(c_{j} \mathrm{~s}\) are fixed such that \(\sum_{j} c_{j}=0\). The variance of \(c_{j} \mathrm{~s}\) is \(\theta_{c}^{2}=\sum c_{j}^{2} /(k-1)\).

In the presence of an interaction, two versions of a mixed-effects model may be considered. One assumes that \((r c)_{i j} \mathrm{~s}\) are i.i.d. \(N\left(0, \sigma_{r c}^{2}\right)\). Another assumes that \((r c)_{i j} \mathrm{~s}\) are \(N\left(0, \sigma_{r c}^{2}\right)\) with an additional constraint that \(\sum_{j}(r c)_{i j}=0\) (for example, Kuehl [2000]), so only interaction terms involving different targets are independent. The latter model is considered here.

We now define the intraclass correlations for individual measurements for model (M3).
The individual CA-ICC, the correlation between individual measurements on the same target, is
\[
\rho_{C, 1}=\operatorname{ICC}(C, 1)=\operatorname{Corr}\left(y_{i j}, y_{i j^{\prime}}\right)=\frac{\sigma_{r}^{2}-\sigma_{r c}^{2} /(k-1)}{\sigma_{r}^{2}+\left(\sigma_{r c}^{2}+\sigma_{\epsilon}^{2}\right)}
\]

The absolute-agreement intraclass correlation for individual measurements, individual AA-ICC, is
\[
\rho_{A, 1}=\operatorname{ICC}(A, 1)=\frac{\sigma_{r}^{2}-\sigma_{r c}^{2} /(k-1)}{\sigma_{r}^{2}+\theta_{c}^{2}+\left(\sigma_{r c}^{2}+\sigma_{\epsilon}^{2}\right)}
\]

Shrout and Fleiss (1979) show that the individual ICC could be negative in this case-a phenomenon first pointed out by Sitgreaves (1960). This can happen when the interaction term has a high variance relative to the targets and there are not many raters.

The individual intraclass correlations for model (M3A) have similar definitions with \(\sigma_{r c}^{2}=0\). The individual CA-ICC is the correlation between individual measurements on the same target, \(\operatorname{Corr}\left(y_{i j}, y_{i j^{\prime}}\right)\).

We now discuss the intraclass correlations that correspond to average measurements. Neither average AA-ICC, \(\rho_{A, k}\), nor average CA-ICC, \(\rho_{C, k}\), can be estimated under model (M3) (Shrout and Fleiss 1979; McGraw and Wong 1996a). The problem is that in this model, \(\sigma_{r}^{2}\), which is the covariance between two means based on \(k\) raters, cannot be estimated.

Specifically, the parameter \(\sigma_{r}^{2}\) appears only in the expectation of the between-target mean squares BMS. Under the restriction \(\sum_{j}(r c)_{i j}=0\),
\[
E(\mathrm{BMS})=k \sigma_{r}^{2}+\sigma_{\epsilon}^{2}
\]

Note that \(\sigma_{r c}^{2}\) does not appear in the expectation of between-target mean squares. With one observation per target and rater, \(\sigma_{r c}^{2}\) and \(\sigma_{\epsilon}^{2}\) cannot be estimated separately (only their sum \(\sigma_{r c}^{2}+\sigma_{\epsilon}^{2}\) can be estimated), so BMS alone cannot be used to estimate \(\sigma_{r}^{2}\).

Under model (M3A), however, there is no interaction (and thus no interaction variance component \(\sigma_{r c}^{2}\) ), so \(\rho_{A, k}\) or \(\rho_{C, k}\) can be estimated.

The average AA-ICC, the absolute-agreement intraclass correlation for average measurements of size \(k\), is
\[
\rho_{A, k}=\operatorname{ICC}(A, k)=\frac{\sigma_{r}^{2}}{\sigma_{r}^{2}+\left(\theta_{c}^{2}+\sigma_{\epsilon}^{2}\right) / k}
\]

The average CA-ICC, the correlation between average measurements of size \(k\) made on the same target, is
\[
\rho_{C, k}=\operatorname{ICC}(C, k)=\operatorname{Corr}\left(\bar{y}_{i .}, \bar{y}_{i .}^{\prime}\right)=\frac{\sigma_{r}^{2}}{\sigma_{r}^{2}+\sigma_{\epsilon}^{2} / k}
\]

The estimators of ICCs, their confidence intervals, and hypothesis tests are as described for two-way random-effects models, except \(\rho_{A, k}\) and \(\rho_{C, k}\) are not defined under model (M3).

\section*{References}

Bliese, P. D. 2000. Within-group agreement, non-independence, and reliability: Implications for data aggregation and analysis. In Multilevel Theory, Research, and Methods in Organizations: Foundations, Extensions, and New Directions, ed. K. J. Klein and S. W. J. Kozlowski, 349-381. San Francisco: Jossey-Bass.

Brown, W. 1910. Some experimental results in the correlation of mental abilities. British Journal of Psychology 3: 296-322. https://doi.org/10.1111/j.2044-8295.1910.tb00207.x.
Cronbach, L. J. 1951. Coefficient alpha and the internal structure of tests. Psychometrika 16: 297-334. https://doi.org/10.1007/BF02310555.

Hartmann, D. P. 1982. Assessing the dependability of observational data. In Using Observers to Study Behavior, 51-65. San Francisco: Jossey-Bass.
Kueh1, R. O. 2000. Design of Experiments: Statistical Principles of Research Design and Analysis. 2nd ed. Belmont, CA: Duxbury.

McGraw, K. O., and S. P. Wong. 1996a. Forming inferences about some intraclass correlation coefficients. Psychological Methods 1: 30-46. https://doi.org/10.1037/1082-989X.1.1.30.
——. 1996b. Forming inferences about some intraclass correlation coefficients: Correction. Psychological Methods 1: 390. https://doi.org/10.1037/1082-989X.1.4.390.

Shrout, P. E., and J. L. Fleiss. 1979. Intraclass correlations: Uses in assessing rater reliability. Psychological Bulletin 86: 420-428. https://doi.org/10.1037/0033-2909.86.2.420.
Sitgreaves, R. 1960. Book reviews: Intraclass Correlation and the Analysis of Variance, Ernest A. Haggard. Journal of the American Statistical Association 55: 384-385. https://doi.org/10.1177/001316445901900113.

Spearman, C. E. 1910. Correlation calculated from faulty data. British Journal of Psychology 3: 271-295. https://doi.org/10.1111/j.2044-8295.1910.tb00206.x.
Suen, H. K. 1988. Agreement, reliability, accuracy, and validity: Toward a clarification. Behavioral Assessment 10: 343-366.

\section*{Also see \\ [R] anova - Analysis of variance and covariance \\ [R] correlate - Correlations of variables \\ [R] loneway - Large one-way ANOVA, random effects, and reliability \\ [MV] alpha - Compute interitem correlations (covariances) and Cronbach's alpha}

Inequality — Inequality, poverty, and other distributional summaries

\section*{Description Remarks and examples References}

\section*{Description}

There are different approaches for measuring inequality and poverty, and different methods for summarizing the distribution of earnings, income, and wealth. Stata has some official features for these types of analyses. For example, the roctab command has an option to report Gini and Pietra indices, which are measures of income inequality; see \([R]\) roctab. With the cumul command, you can obtain the cumulative distribution function of, for example, household expenditures; see [R] cumul. Additionally, you may want to look at the shape of income distributions; see [R] kdensity and [R] histogram.

However, you might also be interested in the generalized Lorenz curve or a welfare index, which cannot be obtained from any official Stata commands. Fortunately, Stata users have developed an excellent suite of commands that create these graphs and report these types of indices, along with many other related measures. In this entry, we demonstrate how to find and install community-contributed commands related to inequality, poverty, and distributional summaries of earnings, income, and wealth.

\section*{Remarks and examples}

Suppose you are in search of a command to report inequality indices; you could type the following:
```

. search inequality

```

First, you will see official resources related to this keyword, including frequently asked questions (FAQs) and Stata Journal articles. Then, you will see community-contributed additions to Stata that are available from the Internet, which typically include additions published in the Stata Journal and those posted on the Statistical Software Components (SSC) Archive, provided by http://repec.org. For example, in the output from search, we find ineqdec0, which calculates inequality indices with decomposition by subgroup.
```

Search of web resources from Stata and other users
(output omitted)
ineqdec0 from http://fmwww.bc.edu/RePEc/bocode/i
'INEQDECO': module to calculate inequality indices with decomposition by
subgroup / ineqdec0 is a stripped-down version of ineqdeco (q.v.). /
Unlike the latter, it will provide estimates using samples /containing
zero or negative values for the variable of / interest. But as a result
ineqdecgini from http://fmwww.bc.edu/RePEc/bocode/i
'INEQDECGINI': module to estimate Gini coefficient with optional
decomposition by subgroups / ineqdecgini estimates the Gini coefficient
(a.k.a. Gini index) / of inequality plus, optionally, a decomposition by
population / subgroup into components representing inequality within /
(output omitted)
(end of search)

```

Let's take a closer look at this package:
1. Click on ineqdec0 from http://fmwww.bc.edu/RePEc/bocode/i.
2. Click on ineqdec0.hlp.

This leads us to the help file, where we could learn more about which indices are computed with this package. This particular command supports fweights and aweights, but not all communitycontributed commands do; it is important to look at the help file and confirm whether the command is a good fit for the analysis you want to perform. If it is, you can install it:
3. Click on the back button.
4. Click on click here to install.

Once you see a message indicating that the installation is complete, you may now use this command.
Another thing to keep in mind when installing packages is whether they are designed to work with complex survey data. This particular command is not. We can click on the back button once to return to the description of ineqdec0 and then once more to return to the search results. Scrolling up, we come across
```

Search of official help files, FAQs, Examples, and Stata Journals
(output omitted)
SJ-16-2 st0427_1 . . . . . . . . . . . . . . . . Software update for conindex
. . . . . . . . . O. O'Donnell, S. O'Neill, T. Van Ourti, and B. Walsh
(help conindex if installed)
Q2/16 SJ 16(2):521--522
program can now be used on data from complex survey designs
(output omitted)
(end of search)

```

This is an update for a package called conindex; this updated version of the command can be used with data from complex survey designs. The original version of the package was introduced in Stata Journal Volume 16 Number 1 (March 2016), and this update was published the following quarter. If we scroll up a bit more in the search results, we will find an even more recent update:
```

Search of official help files, FAQs, Examples, and Stata Journals
(output omitted)
SJ-18-3 st0427_2 . . . . . . . . . . . . . . . . Software update for conindex
. . . . . . . . . O. O'Donnell, S. O'Neill, T. Van Ourti, and B. Walsh
(help conindex if installed)
Q3/18 SJ 18(3):758--759
uses the community-contributed lorenz command and fix a small bug
in the graph option
(output omitted)
(end of search)

```

It is best to install the most recent version, which we can do as follows:
1. Click on st0427_2.
2. Click on click here to install.

The examples above demonstrate how to install community-contributed commands related to measures of inequality, but you would follow the same steps to find and install commands related to poverty measures, indices of social welfare, and the like.

For example, type the following:
. search poverty
In the list of results, you will find glcurve, which can be used to draw Lorenz curves, concentration curves, and other related curves. We can install this command as follows:
1. Click on glcurve from http://fmwww.bc.edu/RePEc/bocode/g.
2. Click on click here to install.

These are just a few examples of the many community-contributed commands that are available. You can use search to find more community-contributed commands used to obtain other measures of inequality and poverty, and distributional summaries of earnings, income, and wealth.

Max Otto Lorenz (1876-1959) was born in Burlington, Iowa. He did his undergraduate studies at the University of Iowa and received his PhD from the University of Wisconsin-Madison in 1906. In 1905, he published his only article, "Methods of measuring the concentration of wealth", in a scientific journal. In the article, he introduces what we now call the Lorenz curve, a term first introduced in a statistics textbook in 1912.

Lorenz worked his whole life in governmental statistical institutions. He was the Deputy Commissioner of Labor and Industrial Statistics for Wisconsin, worked for the US Bureau of the Census and the Bureau of Railway Economics, and was the Director of the Bureau of Statistics and the Bureau of Transport and Economic Statistics.

His hobbies included calendar reform and Interlingua, a proposed international language.

\section*{References}

Chávez Juárez, F. W., and I. Soloaga. 2014. iop: Estimating ex-ante inequality of opportunity. Stata Journal 14: 830-846.

Cox, N. J. 1999. gr35: Diagnostic plots for assessing Singh-Maddala and Dagum distributions fitted by MLE. Stata Technical Bulletin 48: 2-4. Reprinted in Stata Technical Bulletin Reprints, vol. 8, pp. 72-74. College Station, TX: Stata Press.

Foster, J., S. Seth, M. Lokshin, and Z. Sajaia. 2013. A Unified Approach to Measuring Poverty and Inequality: Theory and Practice. Washington, DC: World Bank.

Goldstein, R. 1995. sg31: Measures of diversity: Absolute and relative. Stata Technical Bulletin 23: 23-26. Reprinted in Stata Technical Bulletin Reprints, vol. 4, pp. 150-154. College Station, TX: Stata Press.
Guinea-Martin, D., and R. Mora. 2022. Computing decomposable multigroup indices of segregation. Stata Journal 22: 521-556.

Jann, B. 2016a. Assessing inequality using percentile shares. Stata Journal 16: 264-300.
-_. 2016b. Estimating Lorenz and concentration curves. Stata Journal 16: 837-866.
Jenkins, S. P. 1999a. sg104: Analysis of income distributions. Stata Technical Bulletin 48: 4-18. Reprinted in Stata Technical Bulletin Reprints, vol. 8, pp. 243-260. College Station, TX: Stata Press.
__. 1999b. sg106: Fitting Singh-Maddala and Dagum distributions by maximum likelihood. Stata Technical Bulletin 48: 19-25. Reprinted in Stata Technical Bulletin Reprints, vol. 8, pp. 261-268. College Station, TX: Stata Press.
-_. 2020. Comparing distributions of ordinal data. Stata Journal 20: 505-531.
Jenkins, S. P., and P. Van Kerm. 1999a. sg107: Generalized Lorenz curves and related graphs. Stata Technical Bulletin 48: 25-29. Reprinted in Stata Technical Bulletin Reprints, vol. 8, pp. 269-274. College Station, TX: Stata Press.
-_. 1999b. sg 107.1: Generalized Lorenz curves and related graphs. Stata Technical Bulletin 49: 23. Reprinted in Stata Technical Bulletin Reprints, vol. 9, p. 171. College Station, TX: Stata Press.
——. 2001. Generalized Lorenz curves and related graphs: An update for Stata 7. Stata Journal 1: 107-112.
__. 2004. gr0001_1: Software Updates: Generalized Lorenz curves and related graphs. Stata Journal 4: 490.
__. 2006. gr0001_2: Software Updates: Generalized Lorenz curves and related graphs. Stata Journal 6: 597.
——. 2007. gr0001_3: Software Updates: Generalized Lorenz curves and related graphs. Stata Journal 7: 280.
Jolliffe, D., and B. Krushelnytskyy. 1999. sg115: Bootstrap standard errors for indices of inequality. Stata Technical Bulletin 51: 28-32. Reprinted in Stata Technical Bulletin Reprints, vol. 9, pp. 191-196. College Station, TX: Stata Press.
Jolliffe, D., and A. Semykina. 1999. sg117: Robust standard errors for the Foster-Greer-Thorbecke class of poverty indices. Stata Technical Bulletin 51: 34-36. Reprinted in Stata Technical Bulletin Reprints, vol. 9, pp. 200-203. College Station, TX: Stata Press.

Kleiber, C., and S. Kotz. 2003. Statistical Size Distributions in Economics and Actuarial Sciences. Hoboken, NJ: Wiley.

López-Feldman, A. 2006. Decomposing inequality and obtaining marginal effects. Stata Journal 6: 106-111.
__. 2008. Software Updates: Decomposing inequality and obtaining marginal effects. Stata Journal 8: 594.
Lorenz, M. O. 1905. Methods of measuring the concentration of wealth. American Statistical Association 9: 209-219. https://doi.org/10.2307/2276207.

O'Donnell, O., S. O’Neill, T. Van Ourti, and B. Walsh. 2016a. conindex: Estimation of concentration indices. Stata Journal 16: 112-138.
__. 2016b. st0427_1: Software Updates: conindex: Estimation of concentration indices. Stata Journal 16: 521-522.
——. 2018. st0427_2: Software Updates: conindex: Estimation of concentration indices. Stata Journal 18: 758-759.
Savegnago, M. 2016. igmobil: A command for intergenerational mobility analysis in Stata. Stata Journal 16: 386-401.
Suppa, N. 2023. mpitb: A toolbox for multidimensional poverty indices. Stata Journal 23: 625-657.
Van Kerm, P. 1999. sg108: Computing poverty indices. Stata Technical Bulletin 48: 29-33. Reprinted in Stata Technical Bulletin Reprints, vol. 8, pp. 274-278. College Station, TX: Stata Press.

\section*{Title}
intreg - Interval regression
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
intreg fits a linear model with an outcome measured as point data, interval data, left-censored data, or right-censored data. As such, it is a generalization of the model fit by tobit.

\section*{Quick start}

Regression on x 1 and x 2 of an interval-measured dependent variable with lower endpoint y _lower and upper endpoint \(y\) _upper
intreg y_lower y_upper x1 x2
With robust standard errors
intreg y_lower y_upper x1 x2, vce(robust)
Model heteroskedasticity in the conditional variance as a function of \(x 3\)
intreg y_lower y_upper x1 x2, het(x3)
Adjust for complex survey design using svyset data
svy: intreg y_lower y_upper x1 x2

\section*{Menu}

Statistics \(>\) Linear models and related \(>\) Censored regression \(>\) Interval regression

\section*{Syntax}
intreg depvar \({ }_{1}\) depvar \({ }_{2}\) [indepvars] [if] [in] [weight] [, options]
depvar \(_{1}\) and depvar \({ }_{2}\) should have the following form:
\begin{tabular}{lrcc} 
Type of data & & depvar \(_{1}\) & depvar \(_{2}\) \\
\hline point data & \(a=[a, a]\) & \(a\) & \(a\) \\
interval data & {\([a, b]\)} & \(a\) & \(b\) \\
left-censored data & \((-\infty, b]\) & \(\cdot\) & \(b\) \\
right-censored data & {\([a,+\infty)\)} & \(a\) & \(\cdot\) \\
missing & &. & \(\cdot\) \\
\hline
\end{tabular}
options
Model
noconstant
het(varlist[, noconstant])
offset (varname)
constraints (constraints)
SE/Robust
vce (vcetype)

Reporting
level(\#)
nocnsreport
display_options
suppress constant term
independent variables to model the variance; use noconstant to suppress constant term
include varname in model with coefficient constrained to 1 apply specified linear constraints
vcetype may be oim, robust, cluster clustvar, opg, bootstrap, or jackknife
set confidence level; default is level (95)
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

\section*{Maximization}
maximize_options
collinear
coeflegend
control the maximization process; seldom used
keep collinear variables
display legend instead of statistics
indepvars and varlist may contain factor variables; see [U] 11.4.3 Factor variables.
depvar \({ }_{1}\), depvar \({ }_{2}\), indepvars, and varlist may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bayes, bootstrap, by, collect, fmm, fp, jackknife, mfp, nestreg, rolling, statsby, stepwise, and svy are allowed; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: intreg and [FMM] fmm: intreg.
Weights are not allowed with the bootstrap prefix; see [R] bootstrap.
aweights are not allowed with the jackknife prefix; see \([R]\) jackknife.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
aweights, fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options}
nodel
noconstant; see \([R]\) Estimation options.
he linear combination of varlist. The constant is included unless noconstant is specified.
offset (varname), constraints (constraints); see [R] Estimation options.

SE/Robust
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.

\footnotetext{
Reporting
level(\#), nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fmt), and nolstretch; see [R] Estimation options.
}

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate(\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from(init_specs); see [R] Maximize. These options are seldom used.

Setting the optimization type to technique (bhhh) resets the default vcetype to vce (opg).

The following options are available with intreg but are not shown in the dialog box: collinear, coeflegend; see [R] Estimation options.

\section*{Remarks and examples}
intreg fits a linear model to an outcome that may be either observed exactly or unobserved but known to fall within some interval. The values of the outcome variable may be observed (point data), unobserved but known to fall within an interval with fixed endpoints (interval-censored data), unobserved but known to fall within an interval that has a fixed upper endpoint (left-censored data), or unobserved but known to fall within an interval that has a fixed lower endpoint (right-censored data). Such censored data arise naturally in many contexts, such as wage data. Often, you know only that, for example, a person's salary is between \(\$ 30,000\) and \(\$ 40,000\).

The interval regression model fit by intreg is a generalization of the models fit by tobit because it extends censoring beyond left-censored data or right-censored data; see Cameron and Trivedi (2022, 965-967) for additional discussion of these data types. See Wooldridge (2020, sec. 17.4) for an introduction to censored and truncated regression models.

Regardless of the type of censoring, intreg requires the outcome to be stored in the dataset as interval data. That is, two dependent variables, depvar \({ }_{1}\) and depvar \(_{2}\), are used to hold the endpoints of the interval. If the data are left-censored, the lower endpoint is \(-\infty\) and is represented by a missing value in depvar. If the data are right-censored, the upper endpoint is \(+\infty\) and is represented by a missing value in depvar \({ }_{2}\). Point data are represented by the two endpoints being equal. Truly missing values of the dependent variable must be represented by missing values in both depvar \(r_{1}\) and depvar \({ }_{2}\).

\section*{> Example 1: Interval regression}
womenwage 2.dta contains the yearly wages of working women in interval form. Women were asked to indicate a category for their yearly income from employment. The categories were \(\$ 5,000\) or less, \(\$ 5,001-\$ 10,000, \ldots, \$ 25,001-\$ 30,000, \$ 30,001-\$ 40,000, \$ 40,001-\$ 50,000\), and more than \(\$ 50,000\). The lower and upper endpoints of the wage categories (in \(\$ 1,000 \mathrm{~s}\) ) are recorded in variables wage1 and wage2. Below, we list the first 10 observations in wage1 and wage2.
```

. use https://www.stata-press.com/data/r18/womenwage2
(Wages of women, fictional data)
. list wage1 wage2 in 1/10

```
\begin{tabular}{|rr|}
\hline wage1 & wage2 \\
\hline. & 5 \\
5 & 10 \\
5 & 10 \\
10 & 15 \\
15 & 20 \\
\hline 20 & 25 \\
25 & 30 \\
30 & 40 \\
40 & 50 \\
50 & \(\cdot\) \\
\hline
\end{tabular}

We see, for example, that the first respondent made \(\$ 5,000\) or less in a year, that the second respondent made between \(\$ 5,001\) and \(\$ 10,000\) in a year, and so on. The tenth respondent made at least \(\$ 50,000\) a year.

We now fit an interval regression model of women's wages using social and demographic characteristics as explanatory variables. The variables include the subject's age (age), years of schooling (school), job tenure (tenure), a dummy for living in a rural area (rural), and a dummy for never being married (nev_mar).
```

. intreg wage1 wage2 age c.age\#c.age i.nev_mar i.rural school tenure
Fitting constant-only model:
Iteration 0: Log likelihood = -967.24956
Iteration 1: Log likelihood = -967.1368
Iteration 2: Log likelihood = -967.1368
Fitting full model:
Iteration 0: Log likelihood = -856.65324
Iteration 1: Log likelihood = -856.33294
Iteration 2: Log likelihood = -856.33293
Interval regression Number of obs = 488
Uncensored = 0
Left-censored = 14
Right-censored = 6
Interval-cens. = 468
LR chi2(6) = 221.61
Log likelihood = -856.33293
Prob > chi2 = 0.0000

```
\begin{tabular}{r|rrrrrr}
\hline & Coefficient & Std. err. & \(z\) & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline age & .7914438 & .4433604 & 1.79 & 0.074 & -.0775265 & 1.660414 \\
c.age\#c.age & -.0132624 & .0073028 & -1.82 & 0.069 & -.0275757 & .0010509 \\
1.nev_mar & -.2075022 & .8119581 & -0.26 & 0.798 & -1.798911 & 1.383906 \\
1.rural & -3.043044 & .7757324 & -3.92 & 0.000 & -4.563452 & -1.522637 \\
school & 1.334721 & .1357873 & 9.83 & 0.000 & 1.068583 & 1.600859 \\
tenure & .8000664 & .1045077 & 7.66 & 0.000 & .5952351 & 1.004898 \\
_cons & -12.70238 & 6.367117 & -1.99 & 0.046 & -25.1817 & -.2230583 \\
\hline /lnsigma & 1.987823 & .0346543 & 57.36 & 0.000 & 1.919902 & 2.055744 \\
\hline sigma & 7.299626 & .2529634 & & & 6.82029 & 7.81265 \\
\hline
\end{tabular}

Because the conditional mean modeled by interval regression is linear, the coefficients are interpreted the same way they are in ordinary least-squares regression; see \([R]\) regress. For example, residing in a rural area lowers the expected income by \(\$ 3,043\) and each additional year of schooling raises the expected income by \(\$ 1,335\).
- Technical note

Instead of using intervals to record wages, we could treat the outcome as categorical with a higher category corresponding to a higher wage. Here we fit an ordered probit model for wagecat, created based on groups defined by the intervals, by using oprobit (see [R] oprobit) with the same covariates:
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline \multicolumn{7}{|l|}{Iteration 0: Log likelihood = -881.1491} \\
\hline \multicolumn{7}{|l|}{Iteration 1: Log likelihood \(=-764.31729\)} \\
\hline \multicolumn{7}{|l|}{Iteration 2: Log likelihood = -763.31191} \\
\hline \multicolumn{7}{|l|}{Iteration 3: Log likelihood \(=-763.31049\)} \\
\hline \multicolumn{7}{|l|}{Iteration 4: Log likelihood = -763.31049} \\
\hline \multicolumn{5}{|l|}{Ordered probit regression} & Number of obs & \(=488\) \\
\hline & & & & & LR chi2(6) & \(=235.68\) \\
\hline & & & & & Prob > chi2 & \(=0.0000\) \\
\hline \multicolumn{5}{|l|}{Log likelihood \(=-763.31049\)} & Pseudo R2 & \(=0.1337\) \\
\hline wagecat & Coefficient & Std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. & interval] \\
\hline age & . 1674519 & . 0620333 & 2.70 & 0.007 & . 0458689 & . 289035 \\
\hline c.age\#c.age & -. 0027983 & . 0010214 & -2.74 & 0.006 & -. 0048001 & -. 0007964 \\
\hline 1.nev_mar & -. 0046417 & . 1126737 & -0.04 & 0.967 & -. 225478 & . 2161946 \\
\hline \(1 . r\) rural & -. 5270036 & . 1100449 & -4.79 & 0.000 & -. 7426875 & -. 3113196 \\
\hline school & . 2010587 & . 0201189 & 9.99 & 0.000 & . 1616263 & . 2404911 \\
\hline tenure & . 0989916 & . 0147887 & 6.69 & 0.000 & . 0700063 & . 127977 \\
\hline /cut1 & 2.650637 & . 8957245 & & & . 8950495 & 4.406225 \\
\hline /cut2 & 3.941018 & . 8979167 & & & 2.181134 & 5.700903 \\
\hline /cut3 & 5.085205 & . 9056582 & & & 3.310148 & 6.860263 \\
\hline /cut4 & 5.875534 & . 9120933 & & & 4.087864 & 7.663204 \\
\hline /cut5 & 6.468723 & . 918117 & & & 4.669247 & 8.268199 \\
\hline /cut6 & 6.922726 & . 9215455 & & & 5.11653 & 8.728922 \\
\hline /cut7 & 7.34471 & . 9237628 & & & 5.534168 & 9.155252 \\
\hline /cut8 & 7.963441 & . 9338881 & & & 6.133054 & 9.793828 \\
\hline
\end{tabular}

We can directly compare the log likelihoods for the intreg and oprobit models because both likelihoods are discrete. If we had point data in our intreg estimation, the likelihood would be a mixture of discrete and continuous terms, and we could not compare it directly with the oprobit likelihood.

Here the oprobit \(\log\) likelihood is significantly larger (that is, less negative), so it fits better than the intreg model. The intreg model assumes normality, but the distribution of wages is skewed and definitely nonnormal. Normality is more closely approximated if we model the log of wages.
```

. generate logwage1 = log(wage1)
(14 missing values generated)
. generate logwage2 = log(wage2)
(6 missing values generated)
. intreg logwage1 logwage2 age c.age\#c.age i.nev_mar i.rural school tenure
Fitting constant-only model:
Iteration 0: Log likelihood = -889.23647
Iteration 1: Log likelihood = -889.06346
Iteration 2: Log likelihood = -889.06346
Fitting full model:
Iteration 0: Log likelihood = -773.81968
Iteration 1: Log likelihood = -773.36566
Iteration 2: Log likelihood = -773.36563
Interval regression Number of obs = 488
Uncensored = 0
Left-censored = 14
Right-censored = 6
Interval-cens. = 468
LR chi2(6) = 231.40
Prob > chi2 = 0.0000

```
\begin{tabular}{r|rrrrrrr}
\hline & Coefficient & Std. err. & \(\mathbf{z}\) & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline age & .0645589 & .0249954 & 2.58 & 0.010 & .0155689 & .1135489 \\
c.age\#c.age & -.0010812 & .0004115 & -2.63 & 0.009 & -.0018878 & -.0002746 \\
1.nev_mar & -.0058151 & .0454867 & -0.13 & 0.898 & -.0949674 & .0833371 \\
1.rural & -.2098361 & .0439454 & -4.77 & 0.000 & -.2959675 & -.1237047 \\
school & .0804832 & .0076783 & 10.48 & 0.000 & .0654341 & .0955323 \\
tenure & .0397144 & .0058001 & 6.85 & 0.000 & .0283464 & .0510825 \\
_cons & .7084023 & .3593193 & 1.97 & 0.049 & .0041495 & 1.412655 \\
\hline /lnsigma & -.906989 & .0356265 & -25.46 & 0.000 & -.9768157 & -.8371623 \\
\hline sigma & .4037381 & .0143838 & & & .3765081 & .4329373 \\
\hline
\end{tabular}

The log likelihood of this intreg model is close to the oprobit \(\log\) likelihood, and the \(z\) statistics for both models are similar.

\section*{Stored results}
intreg stores the following in e() :
Scalars
e(N)
e(N_unc)
e(N_lc)
e(N_rc)
e(N_int)
e(k)
e(k_aux)
\(e\left(k \_\right.\)eq)
e(k_eq_model)
e(k_dv)
e(df_m)
e(11)
e(11_0)
e(N_clust)
e(chi2)
\(e(p)\)
e(sigma)
e(se_sigma)
\(e\) (rank)
e(rankO)
\(e(i c)\)
\(e(r c)\)
e(converged)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(offset)
e(chi2type)
e(vce)
e(vcetype)
\(e\) (het)
e(ml_score)
e(opt)
\(e\) (which)
e(ml_method)
e(user)
e(technique)
e(properties)
e(predict)
e(marginsok)
e(asbalanced)
e(asobserved)
Matrices
e(b)
\(e\) (Cns)
e(ilog)
e(gradient)
e(V)
e(V_modelbased)
Functions
e(sample)
number of observations
number of uncensored observations
number of left-censored observations
number of right-censored observations
number of interval observations
number of parameters
number of auxiliary parameters
number of equations in \(e(b)\)
number of equations in overall model test
number of dependent variables
model degrees of freedom
log likelihood
log likelihood, constant-only model
number of clusters
\(\chi^{2}\)
\(p\)-value for model \(\chi^{2}\) test
sigma
standard error of sigma
rank of e(v)
rank of \(e(V)\) for constant-only model
number of iterations
return code
1 if converged, 0 otherwise

\section*{intreg}
command as typed
names of dependent variables
weight type
weight expression
title in estimation output
name of cluster variable
linear offset variable
Wald or LR; type of model \(\chi^{2}\) test
vcetype specified in vce()
title used to label Std. err.
heteroskedasticity, if het() specified
program used to implement scores
type of optimization
max or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement predict
predictions allowed by margins
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
constraints matrix
iteration \(\log\) (up to 20 iterations)
gradient vector
variance-covariance matrix of the estimators
model-based variance
marks estimation sample

In addition to the above, the following is stored in r() :
Matrices
\(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

\section*{Methods and formulas}

The regression equation of interest is
\[
y_{j}=\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j}
\]
where \(y_{j}\) is a continuous outcome for the \(j\) th observation-either observed or unobserved-with covariates \(\mathbf{x}_{j}\) and corresponding coefficients \(\boldsymbol{\beta}\). The model assumes that the error term is normally distributed; \(\epsilon \sim N\left(0, \sigma^{2}\right)\).

For observations \(j \in \mathcal{C}\), we observe \(y_{j}\), that is, point data. Observations \(j \in \mathcal{I}\) are intervals; we know only that the unobserved \(y_{j}\) is in the interval \(\left[y_{1 j}, y_{2 j}\right.\) ]. For these observations, the likelihood contribution is \(\operatorname{Pr}\left(y_{1 j} \leq Y_{j} \leq y_{2 j}\right)\), where \(Y_{j}\) denotes the random variable representing the dependent variable in the model. Observations \(j \in \mathcal{L}\) are left-censored; we know only that the unobserved \(y_{j}\) is less than or equal to \(y_{\mathcal{L} j}\), a censoring value that we do know. Similarly, observations \(j \in \mathcal{R}\) are right-censored; we know only that the unobserved \(y_{j}\) is greater than or equal to \(y_{\mathcal{R} j}\). The likelihoods for these censored observations contain terms of the form \(\operatorname{Pr}\left(Y_{j} \leq y_{\mathcal{L}} j\right)\) for left-censored data and \(\operatorname{Pr}\left(Y_{j} \geq y_{\mathcal{R} j}\right)\) for right-censored data.

The log likelihood is
\[
\begin{aligned}
\ln L=- & \frac{1}{2} \sum_{j \in \mathcal{C}} w_{j}\left\{\left(\frac{y_{j}-\mathbf{x}_{j} \boldsymbol{\beta}}{\sigma}\right)^{2}+\log 2 \pi \sigma^{2}\right\} \\
& +\sum_{j \in \mathcal{L}} w_{j} \log \Phi\left(\frac{y_{\mathcal{L} j}-\mathbf{x}_{j} \boldsymbol{\beta}}{\sigma}\right) \\
& +\sum_{j \in \mathcal{R}} w_{j} \log \left\{1-\Phi\left(\frac{y_{\mathcal{R} j}-\mathbf{x}_{j} \boldsymbol{\beta}}{\sigma}\right)\right\} \\
& +\sum_{j \in \mathcal{I}} w_{j} \log \left\{\Phi\left(\frac{y_{2 j}-\mathbf{x}_{j} \boldsymbol{\beta}}{\sigma}\right)-\Phi\left(\frac{y_{1 j}-\mathbf{x}_{j} \boldsymbol{\beta}}{\sigma}\right)\right\}
\end{aligned}
\]
where \(\Phi()\) is the cumulative standard normal distribution and \(w_{j}\) is the weight for the \(j\) th observation. If no weights are specified, \(w_{j}=1\). If aweights are specified, \(w_{j}=1\), and \(\sigma\) is replaced by \(\sigma / \sqrt{a_{j}}\) in the above, where \(a_{j}\) are the aweights normalized to sum to \(N\).

When the het () option is specified, \(\sigma\) is modeled as \(\ln (\sigma)=z_{j}^{\prime} \gamma\), where \(z\) represents the variables in het () and \(\gamma\) is a vector of the estimated parameters to model the variance.

Note that the likelihood for intreg subsumes that of the tobit models; see [R] tobit.
Maximization is as described in [R] Maximize. intreg stores the estimated \(\sigma\) in e(b) in the log metric; therefore, if you want to provide an initial value for \(\sigma\) or to specify a constraint on it, ensure you do so on the \(\log\) scale.

This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce (robust) and vce (cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.
intreg also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

\section*{References}

Bertanha, M., A. H. McCallum, A. Payne, and N. Seegert. 2022. Bunching estimation of elasticities using Stata. Stata Journal 22: 597-624.

Cameron, A. C., and P. K. Trivedi. 2022. Microeconometrics Using Stata. 2nd ed. College Station, TX: Stata Press.
Canette, I. 2016. Understanding truncation and censoring. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2016/12/13/understanding-truncation-and-censoring/.

Chernozhukov, V., I. Fernández-Val, S. Han, and A. Kowalski. 2019. Censored quantile instrumental-variable estimation with Stata. Stata Journal 19: 768-781.
Conroy, R. M. 2005. Stings in the tails: Detecting and dealing with censored data. Stata Journal 5: 395-404.
Davidson, R., and J. G. MacKinnon. 1993. Estimation and Inference in Econometrics. New York: Oxford University Press.
Goldberger, A. S. 1983. Abnormal selection bias. In Studies in Econometrics, Time Series, and Multivariate Statistics, ed. S. Karlin, T. Amemiya, and L. A. Goodman, 67-84. New York: Academic Press.

Hurd, M. 1979. Estimation in truncated samples when there is heteroscedasticity. Journal of Econometrics 11: 247-258. https://doi.org/10.1016/0304-4076(79)90039-3.
Long, J. S. 1997. Regression Models for Categorical and Limited Dependent Variables. Thousand Oaks, CA: Sage.
Pudney, S. 2019. intcount: A command for fitting count-data models from interval data. Stata Journal 19: 645-666.
Sánchez-Peñalver, A. 2019. Estimation methods in the presence of corner solutions. Stata Journal 19: 87-111.
Stewart, M. B. 1983. On least squares estimation when the dependent variable is grouped. Review of Economic Studies 50: 737-753. https://doi.org/10.2307/2297773.
Wooldridge, J. M. 2020. Introductory Econometrics: A Modern Approach. 7th ed. Boston: Cengage.

\section*{Also see}
[R] intreg postestimation - Postestimation tools for intreg
\([R]\) regress - Linear regression
[R] tobit - Tobit regression
[BAYES] bayes: intreg - Bayesian interval regression
[ERM] eintreg - Extended interval regression
[FMM] fmm: intreg - Finite mixtures of interval regression models
[ME] meintreg - Multilevel mixed-effects interval regression
[ST] stintreg - Parametric models for interval-censored survival-time data
[SVY] svy estimation - Estimation commands for survey data
[XT] xtintreg - Random-effects interval-data regression models
[XT] xttobit - Random-effects tobit models
[U] 20 Estimation and postestimation commands

\section*{Title}
intreg postestimation - Postestimation tools for intreg
Postestimation commands predict margins Remarks and examples

\section*{Postestimation commands}

The following postestimation commands are available after intreg:
\begin{tabular}{ll} 
Command & Description \\
\hline contrast & \begin{tabular}{l} 
contrasts and ANOVA-style joint tests of estimates \\
estat ic
\end{tabular} \\
\begin{tabular}{l} 
Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian in- \\
formation criteria (AIC, CAIC, AICc, and BIC)
\end{tabular} \\
estat summarize & summary statistics for the estimation sample \\
estat vce & variance-covariance matrix of the estimators (VCE) \\
estat (svy) & \begin{tabular}{l} 
postestimation statistics for survey data \\
cataloging estimation results
\end{tabular} \\
estimates & table of estimation results \\
etable \(h a u s m a n\) & Hausman's specification test \\
lincom & \begin{tabular}{l} 
point estimates, standard errors, testing, and inference for linear combinations of \\
*lrtest
\end{tabular} \\
likelihood-ratio test
\end{tabular}

\footnotetext{
*hausman and lrtest are not appropriate with svy estimation results.
}

\section*{predict}

\section*{Description for predict}
predict creates a new variable containing predictions such as linear predictions, standard errors, probabilities, and expected values.

\section*{Menu for predict}

Statistics \(>\) Postestimation

\section*{Syntax for predict}
```

predict [type] newvar [if][in] [, statistic nooffset]
predict [type] stub* [if] [in], scores

```
statistic Description

Main
\(\mathrm{xb} \quad\) linear prediction; the default
stdp standard error of the prediction
stdf standard error of the forecast
\(\operatorname{pr}(a, b) \quad \operatorname{Pr}\left(a<y_{j}<b\right)\)
\(\mathrm{e}(a, b) \quad E\left(y_{j} \mid a<y_{j}<b\right)\)
ystar \((a, b) \quad E\left(y_{j}^{*}\right), y_{j}^{*}=\max \left\{a, \min \left(y_{j}, b\right)\right\}\)
These statistics are available both in and out of sample; type predict ... if e(sample)... if wanted only for the estimation sample.
stdf is not allowed with svy estimation results.
where \(a\) and \(b\) may be numbers or variables; \(a\) missing ( \(a \geq\).) means \(-\infty\), and \(b\) missing ( \(b \geq\).) means \(+\infty\); see [U] 12.2.1 Missing values.

\section*{Options for predict}
\(\qquad\) Main
xb , the default, calculates the linear prediction.
stdp calculates the standard error of the prediction, which can be thought of as the standard error of the predicted expected value or mean for the observation's covariate pattern. The standard error of the prediction is also referred to as the standard error of the fitted value.
stdf calculates the standard error of the forecast, which is the standard error of the point prediction for 1 observation. It is commonly referred to as the standard error of the future or forecast value. By construction, the standard errors produced by stdf are always larger than those produced by stdp; see Methods and formulas in [R] regress postestimation.
\(\operatorname{pr}(a, b)\) calculates \(\operatorname{Pr}\left(a<\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j}<b\right)\), the probability that \(y_{j} \mid \mathbf{x}_{j}\) would be observed in the interval \((a, b)\).
\(a\) and \(b\) may be specified as numbers or variable names; \(l b\) and \(u b\) are variable names;
\(\operatorname{pr}(20,30)\) calculates \(\operatorname{Pr}\left(20<\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j}<30\right)\);
\(\operatorname{pr}(l b, u b)\) calculates \(\operatorname{Pr}\left(l b<\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j}<u b\right)\); and
\(\operatorname{pr}(20, u b)\) calculates \(\operatorname{Pr}\left(20<\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j}<u b\right)\).
\(a\) missing \((a \geq\).\() means -\infty ; \operatorname{pr}(., 30)\) calculates \(\operatorname{Pr}\left(-\infty<\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j}<30\right)\); \(\operatorname{pr}(l b, 30)\) calculates \(\operatorname{Pr}\left(-\infty<\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j}<30\right)\) in observations for which \(l b \geq\). and calculates \(\operatorname{Pr}\left(l b<\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j}<30\right)\) elsewhere.
\(b\) missing ( \(b \geq\). ) means \(+\infty\); \(\operatorname{pr}(20,\).\() calculates \operatorname{Pr}\left(+\infty>\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j}>20\right)\); \(\operatorname{pr}(20, u b)\) calculates \(\operatorname{Pr}\left(+\infty>\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j}>20\right)\) in observations for which \(u b \geq\). and calculates \(\operatorname{Pr}\left(20<\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j}<u b\right)\) elsewhere.
\(\mathrm{e}(a, b)\) calculates \(E\left(\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j} \mid a<\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j}<b\right)\), the expected value of \(y_{j} \mid \mathbf{x}_{j}\) conditional on \(y_{j} \mid \mathbf{x}_{j}\) being in the interval \((a, b)\), meaning that \(y_{j} \mid \mathbf{x}_{j}\) is truncated.
\(a\) and \(b\) are specified as they are for \(\operatorname{pr}()\).
ystar \((a, b)\) calculates \(E\left(y_{j}^{*}\right)\), where \(y_{j}^{*}=a\) if \(\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j} \leq a, y_{j}^{*}=b\) if \(\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j} \geq b\), and \(y_{j}^{*}=\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j}\) otherwise, meaning that \(y_{j}^{*}\) is censored. \(a\) and \(b\) are specified as they are for pr() .
nooffset is relevant only if you specified offset (varname). It modifies the calculations made by predict so that they ignore the offset variable; the linear prediction is treated as \(\mathbf{x}_{j} \boldsymbol{\beta}\) rather than as \(\mathbf{x}_{j} \boldsymbol{\beta}+\) offset \(_{j}\).
scores calculates equation-level score variables.
The first new variable will contain \(\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)\).
The second new variable will contain \(\partial \ln L / \partial \ln \sigma\).

\section*{margins}

\section*{Description for margins}
margins estimates margins of response for linear predictions, probabilities, and expected values.

\section*{Menu for margins}

Statistics \(>\) Postestimation

\section*{Syntax for margins}
```

margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...)...] [options]

```
statistic Description
xb
linear prediction; the default
\(\operatorname{pr}(a, b) \quad \operatorname{Pr}\left(a<y_{j}<b\right)\)
\(\mathrm{e}(a, b) \quad E\left(y_{j} \mid a<y_{j}<b\right)\)
ystar \((a, b) \quad E\left(y_{j}^{*}\right), y_{j}^{*}=\max \left\{a, \min \left(y_{j}, b\right)\right\}\)
stdp not allowed with margins
stdf not allowed with margins

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

\section*{Remarks and examples}

\section*{> Example 1: Marginal predictions}

Continuing with example 1 of \([\mathrm{R}]\) intreg, we compute women's expected wages conditional on a woman's wage being higher than \(\$ 5,000\). To do this, we can use the \(\mathrm{e}(a, b)\) option.
```

. use https://www.stata-press.com/data/r18/womenwage2
(Wages of women, fictional data)
. intreg wage1 wage2 age c.age\#c.age i.nev_mar i.rural school tenure
(output omitted)
. predict w1, e(5,.)
. summarize w1

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| w1 | 488 | 18.02362 | 4.583738 | 8.717687 | 35.31161 |

```

The predicted wages range from \(\$ 8,718\) to \(\$ 35,312\).
We can also examine whether the probability of earning more than \(\$ 5,000\) varies with age. We can use margins to compute the marginal means of the predicted probabilities at different ages.
```

. margins, predict(pr(5,.)) at(age=(20(5)50))
Predictive margins Number of obs = 488
Model VCE: OIM
Expression: Pr(y>5), predict(pr(5,.))
1._at: age = 20
2._at: age = 25
3._at: age = 30
4._at: age = 35
5._at: age = 40
6._at: age = 45
7._at: age = 50

```
\begin{tabular}{c|rrrrrr}
\hline & \multicolumn{5}{|c}{\begin{tabular}{c} 
Delta-method \\
Margin \\
std.err.
\end{tabular}} & z \\
& P>|z| & [95\% conf. interval] \\
\hline -at & & & & & & \\
1 & .8912598 & .0151773 & 58.72 & 0.000 & .8615127 & .9210068 \\
2 & .9104568 & .0103467 & 87.99 & 0.000 & .8901775 & .930736 \\
3 & .9160005 & .0120025 & 76.32 & 0.000 & .892476 & .9395251 \\
4 & .9096667 & .0136693 & 66.55 & 0.000 & .8828753 & .9364581 \\
5 & .8894289 & .0206992 & 42.97 & 0.000 & .8488593 & .9299985 \\
6 & .8491103 & .0447429 & 18.98 & 0.000 & .7614159 & .9368048 \\
7 & .7781644 & .0970557 & 8.02 & 0.000 & .5879387 & .9683902 \\
\hline
\end{tabular}

We can visualize these results by using marginsplot:
```

. marginsplot
Variables that uniquely identify margins: age

```

Predictive margins with \(95 \%\) Cls


The probability increases until age 30 and decreases thereafter.

\section*{Also see}
[R] intreg - Interval regression
[U] 20 Estimation and postestimation commands

\section*{Title}
ivfprobit - Fractional probit model with continuous endogenous covariates
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
ivfprobit fits a model for a fractional dependent variable, such as a rate or proportion, where one or more of the covariates are endogenous. The dependent variable must be greater than or equal to 0 and less than or equal to 1 . ivfprobit assumes all endogenous covariates are continuous.

\section*{Quick start}

Fractional probit regression of y 1 on x and endogenous regressor y 2 that is instrumented using z ivfprobit y1 x (y2 = z)

Same as above, but with endogenous regressors y2 and y3, using \(z 1, z 2\), and \(z 3\) as instruments ivfprobit y1 x (y2 y3 = z1 z2 z3)

\section*{Menu}

Statistics \(>\) Endogenous covariates \(>\) Fractional probit with endogenous covariates

\section*{Syntax}
```

ivfprobit depvar $\left[\right.$ varlist $\left._{1}\right]\left(\right.$ varlist $_{2}=$ varlist $\left._{\mathrm{iv}}\right)$ [if] [in] [weight] [, options $]$

```
varlist \(_{1}\) is the list of exogenous variables.
varlist \(_{2}\) is the list of endogenous variables.
\(v^{v a r l i s t} t_{\mathrm{iv}}\) is the list of exogenous variables used with varlist \(_{1}\) as instruments for varlist \({ }_{2}\).
options
Model
noconstant
constraints (numlist)
SE/Robust
vce (vcetype)

Reporting
level (\#)
first
nocnsreport
display_options

Description
suppress constant term
apply specified linear constraints
vcetype may be robust, cluster clustvar, bootstrap, or jackknife
set confidence level; default is level (95)
report first-stage regression
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

Maximization
maximize_options
coeflegend
control the maximization process
display legend instead of statistics
varlist \(_{1}\) and varlist \(_{\mathrm{iv}}\) may contain factor variables; see [U] 11.4.3 Factor variables.
depvar, varlist \({ }_{1}\), varlist \(t_{2}\), and varlist \({ }_{\mathrm{iv}}\) may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bootstrap, by, collect, fp, jackknife, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands.
Weights are not allowed with the bootstrap prefix; see \([R]\) bootstrap.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed. See [U] 11.1.6 weight.
coeflegend does not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options}
\(\qquad\) Model
noconstant, constraints (numlist); see [R] Estimation options.
\(\qquad\) SE/Robust
vce(vcetype) specifies the type of standard error reported, which includes types that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
\(\qquad\) Reporting
level(\#); see [R] Estimation options.
first requests that the parameters for the reduced-form equations showing the relationships between the endogenous variables and instruments be displayed. The default is not to show these parameter estimates.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.
\(\qquad\) Maximization
maximize_options: difficult, technique (algorithm_spec), iterate (\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from(init_specs); see [R] Maximize.

The following option is available with ivfprobit but is not shown in the dialog box: coeflegend; see [R] Estimation options.

\section*{Remarks and examples}

Remarks are presented under the following headings:

> Model setup
> Model identification

\section*{Model setup}
ivfprobit fits models for fractional dependent variables when one or more of the covariates is endogenous. Fractional variables can take any value in the interval \([0,1]\); thus, ivfprobit is useful for modeling outcomes such as rates and proportions.

Formally, we can write the model fit by ivfprobit as
\[
\begin{align*}
E\left(y_{1 i} \mid \mathbf{x}_{1 i}, \mathbf{x}_{2 i}, \mathbf{y}_{2 i}, u_{i}\right) & =\Phi\left(\mathbf{y}_{2 i} \boldsymbol{\beta}+\mathbf{x}_{1 i} \boldsymbol{\gamma}+u_{i}\right)  \tag{1}\\
\mathbf{y}_{2 i} & =\mathbf{x}_{i 1} \boldsymbol{\Pi}_{1}+\mathbf{x}_{i 2} \boldsymbol{\Pi}_{2}+v_{i}
\end{align*}
\]
where subscript \(i\) denotes the observation, \(\mathbf{y}_{2 i}\) is a \(1 \times p\) vector of continuous endogenous variables, \(\mathbf{x}_{1 i}\) is a \(1 \times k_{1}\) vector of exogenous covariates, and \(\mathbf{x}_{2 i}\) is a \(1 \times k_{2}\) vector of excluded instruments. Endogeneity arises from the possible correlation between \(u_{i}\) and \(v_{i}\). The coefficients in vectors \(\boldsymbol{\beta}\) and \(\gamma\) are the parameters of interest. Matrices \(\Pi_{1}\) and \(\Pi_{2}\) contain the coefficients of the first stage for the reduced-form equation.

To obtain parameter estimates, ivfprobit maximizes the same likelihood as ivprobit but does not require a binary dependent variable and does not require the joint density of the errors in the model to be specified correctly; see [R] ivprobit for more information. ivfprobit fits the model via quasilikelihood estimation rather than maximum likelihood estimation. The key insight behind quasilikelihood estimation is that we do not need to know the true distribution of the entire model to obtain consistent parameter estimates. In fact, the only requirement is the correct specification of the conditional mean given in (1) after integrating over \(u_{i}\). Specifying the full distribution of the model correctly is required only if we want to obtain asymptotically efficient standard errors from maximum likelihood estimation.
ivfprobit does not assume that the true model is a probit model that accounts for endogeneity, such as the one fit by ivprobit. Therefore, the standard errors provided by maximum likelihood estimation are not appropriate. Instead, ivfprobit takes the maximum quasilikelihood approach and reports robust standard errors by default.

For further discussion on quasilikelihood estimation in the context of fractional regression, see Papke and Wooldridge (1996) and Wooldridge (2010).
\(>\) Example 1
We use a corporate \(401(\mathrm{k})\) participation dataset and fit a fractional probit regression of the \(401(\mathrm{k})\) participation rate (prate), on corporate employment size (ltotemp) and its square, an indicator of whether the \(401(\mathrm{k})\) is the sole pension plan (sole), and plan matching rate (mrate). The plan matching rate is endogenous and is instrumented using the age of the plan (age) and its square.

Our outcome variable prate has values between 0 and 1, including 1,351 firms with participation rates of 1 . We assume that the functional form of the expected participation rate, after integrating over \(u_{i}\), is a cumulative normal density as in (1).

We use ivfprobit to fit the fractional probit model, accounting for endogeneity of mrate.
```

. use https://www.stata-press.com/data/r18/401k
(Firm-level data on 401k participation)
. ivfprobit prate c.ltotemp\#\#c.ltotemp i.sole (mrate = c.age\#\#c.age)
Fitting exogenous fractional probit model:
Iteration 0: Log pseudolikelihood = -1769.7046
Iteration 1: Log pseudolikelihood = -1675.4223
Iteration 2: Log pseudolikelihood = -1674.7663
Iteration 3: Log pseudolikelihood = -1674.7661
Iteration 4: Log pseudolikelihood = -1674.7661
Fitting full model:
Iteration 0: Log pseudolikelihood = -3712.498
Iteration 1: Log pseudolikelihood = -3712.4767
Iteration 2: Log pseudolikelihood = -3712.4767
Fractional probit model with endogenous regressors

```
Log pseudolikelihood \(=-3712.4767\)
                                    Number of obs \(=4,075\)
                                    Wald chi2(4) \(=907.06\)
                                    Prob > chi2 \(=0.0000\)
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline & Coefficient & Robust std. err. & z & \(P>|z|\) & [95\% conf & interval] \\
\hline mrate & 1.907922 & . 0946094 & 20.17 & 0.000 & 1.722491 & 2.093353 \\
\hline ltotemp & -. 4229273 & . 0744177 & -5.68 & 0.000 & -. 5687833 & -. 2770713 \\
\hline \begin{tabular}{l}
c. Itotemp\# \\
c.ltotemp
\end{tabular} & . 0217492 & . 0046476 & 4.68 & 0.000 & . 01264 & . 0308583 \\
\hline \[
\begin{gathered}
\text { sole } \\
\text { Only plan }
\end{gathered}
\] & -. 1733119 & . 0366136 & -4.73 & 0.000 & -. 2450733 & -. 1015504 \\
\hline _cons & 1.904103 & . 3199032 & 5.95 & 0.000 & 1.277104 & 2.531102 \\
\hline corr (e.mrate, e.prate) & -. 5690386 & . 0431738 & & & -. 6476498 & -. 4784406 \\
\hline sd(e.mrate) & . 3989664 & . 0061807 & & & . 3870345 & . 4112661 \\
\hline \multicolumn{5}{|l|}{\multirow[t]{2}{*}{Wald test of exogeneity: chi2(1) \(=102.40\) Endogenous: mrate}} & \multicolumn{2}{|l|}{Prob \(>\) chi2 \(=0.0000\)} \\
\hline & & & & & & \\
\hline \multicolumn{7}{|l|}{Exogenous: ltotemp c.ltotemp\#c.ltotemp 1.sole age c.age\#c.age} \\
\hline
\end{tabular}

We find a positive effect of mrate on the participation rate. Additionally, we see that the correlation between the unobservables, corr (e.mrate, e.prate), is different from zero. This means there is evidence to support our endogeneity conjecture.

\section*{Model identification}

As in the basic linear instrumental-variables model, the order condition for identification requires that the number of excluded exogenous variables (that is, the additional instruments) be at least as great as the number of included endogenous variables ( \(k_{2} \geq p\) ). ivfprobit checks this for you and issues an error message if the order condition is not met.
ivfprobit, like probit and ivprobit, checks the exogenous and endogenous variables to see if any of them predict the outcome variable perfectly. It will drop any offending variables and observations and then fit the model on the remaining data. Instruments that are perfect predictors do not affect estimation, so they are not checked. See Model identification in [R] probit for more information.

\section*{Stored results}
ivfprobit stores the following in e():
Scalars
e(N)
e(k)
e(k_eq)
e(k_eq_model)
e(k_dv)
e(df_m)
e(11)
e(N_clust)
e(endog_ct)
e (p)
e(p_exog)
e(chi2)
e(chi2_exog)
e(rank)
e(ic)
e(rc)
e(converged)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(endog)
e(exog)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(chi2type)
e(vce)
e(vcetype)
e(opt)
e(which)
\(e\left(m l \_\right.\)method)
e(user)
e(technique)
e(properties)
e(estat_cmd)
\(e\) (predict)
\(e\) (footnote)
e(marginsok)
e(asbalanced)
e(asobserved)
Matrices
```

e(b)
e(Cns)
e(ilog)
e(gradient)
e(Sigma)
e(V)
e(V_modelbased)

```

Functions
e(sample)
number of observations
number of parameters
number of equations in \(e(b)\)
number of equations in overall model test
number of dependent variables
model degrees of freedom
log likelihood
number of clusters
number of endogenous covariates
model Wald \(p\)-value
exogeneity test Wald \(p\)-value
model Wald \(\chi^{2}\)
Wald \(\chi^{2}\) test of exogeneity
rank of e(V)
number of iterations
return code
1 if converged, 0 otherwise
ivfprobit
command as typed
name of dependent variable
names of endogenous variables
names of exogenous variables
weight type
weight expression
title in estimation output
name of cluster variable
Wald; type of model \(\chi^{2}\) test
vcetype specified in vce()
title used to label Std. err.
type of optimization
max or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement estat
program used to implement predict
program used to implement the footnote display
predictions allowed by margins
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
constraints matrix
iteration \(\log\) (up to 20 iterations)
gradient vector
\(\widehat{\boldsymbol{\Sigma}}\)
variance-covariance matrix of the estimators
model-based variance
marks estimation sample

In addition to the above, the following is stored in r() :
Matrices
\(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

\section*{Methods and formulas}

See Methods and formulas in [R] ivprobit.

\section*{References}

Papke, L. E., and J. M. Wooldridge. 1996. Econometric methods for fractional response variables with an application to \(401(\mathrm{k})\) plan participation rates. Journal of Applied Econometrics 11: 619-632. https://doi.org/10.1002/(SICI) 1099-1255(199611)11:6〈619::AID-JAE418〉3.0.CO;2-1.

Wooldridge, J. M. 2010. Econometric Analysis of Cross Section and Panel Data. 2nd ed. Cambridge, MA: MIT Press.

\section*{Also see}
[R] ivfprobit postestimation - Postestimation tools for ivfprobit
\([R]\) fracreg - Fractional response regression
[R] gmm - Generalized method of moments estimation
[R] ivprobit - Probit model with continuous endogenous covariates
\([R]\) ivregress - Single-equation instrumental-variables regression
[R] ivtobit - Tobit model with continuous endogenous covariates
[R] probit - Probit regression
[ERM] eprobit - Extended probit regression
[SVY] svy estimation - Estimation commands for survey data
[XT] xtprobit - Random-effects and population-averaged probit models
[U] 20 Estimation and postestimation commands

\section*{Title}
ivfprobit postestimation - Postestimation tools for ivfprobit
\begin{tabular}{llll} 
Postestimation commands & predict & margins & estat \\
Remarks and examples & Stored results & Methods and formulas & Also see
\end{tabular}

\section*{Postestimation commands}

The following postestimation commands are of special interest after ivfprobit:
Command Description
estat correlation
estat covariance
report the correlation matrix of the errors of the dependent variable and the endogenous variables
report the covariance matrix of the errors of the dependent variable and the endogenous variables

These commands are not appropriate after the svy prefix.

The following standard postestimation commands are also available:
\begin{tabular}{|c|c|}
\hline Command & Description \\
\hline contrast & contrasts and ANOVA-style joint tests of estimates \\
\hline estat ic & Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) \\
\hline estat summarize & summary statistics for the estimation sample \\
\hline estat vce & variance-covariance matrix of the estimators (VCE) \\
\hline estat (svy) & postestimation statistics for survey data \\
\hline estimates & cataloging estimation results \\
\hline etable & table of estimation results \\
\hline * forecast & dynamic forecasts and simulations \\
\hline lincom & point estimates, standard errors, testing, and inference for linear combinations of coefficients \\
\hline margins & marginal means, predictive margins, marginal effects, and average marginal effects \\
\hline marginsplot & graph the results from margins (profile plots, interaction plots, etc.) \\
\hline nlcom & point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients \\
\hline predict & conditional means, linear predictions, etc. \\
\hline predictnl & point estimates, standard errors, testing, and inference for generalized predictions \\
\hline pwcompare & pairwise comparisons of estimates \\
\hline test & Wald tests of simple and composite linear hypotheses \\
\hline testnl & Wald tests of nonlinear hypotheses \\
\hline
\end{tabular}

\footnotetext{
\({ }^{*}\) forecast is not appropriate with svy estimation results.
}

\section*{predict}

\section*{Description for predict}
predict creates a new variable containing predictions such as conditional means, linear predictions, and standard errors.

\section*{Menu for predict}

Statistics \(>\) Postestimation

\section*{Syntax for predict}
```

predict [type] newvar [if] [in] [, statistic asfmethod]
predict [type] {stub*|newvarlist } [if] [in], scores

```

\section*{statistic Description}

Main
\begin{tabular}{ll}
cm & conditional mean; the default \\
xb & linear prediction \\
stdp & standard error of the linear prediction
\end{tabular}

These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.
asfmethod Description
Main
asf average structural function; the default
fixedasf fixed average structural function

\section*{Options for predict}
\(\qquad\) Main
cm , the default, calculates the conditional mean.
xb calculates the linear prediction.
stdp calculates the standard error of the linear prediction.
asf and fixedasf determine how the average structural function (ASF) of the conditional mean is computed. These options are not allowed with xb or stdp.
asf is the default estimator when the cm statistic is specified. asf computes the ASF of the conditional mean. It is the prediction conditional on the errors of the endogenous variable equations. Put another way, it is the conditional mean accounting for the correlation of the endogenous covariates with the errors of the main equation. Derivatives and contrasts based on asf have a structural interpretation. See margins below for computing derivatives and contrasts.
fixedasf calculates a fixed ASF. It is the prediction using only the coefficients and variables of the outcome equation. fixedasf does not condition on the errors of the endogenous variable equations. Contrasts between two fixed counterfactuals averaged over the whole sample have a potential-outcome interpretation. Intuitively, it is as if the values of the covariates were fixed at a value exogenously by fiat. See margins below for computing derivatives and contrasts.
To be clear, derivatives and contrasts between two fixed counterfactuals using the default asf option also have a potential-outcome interpretation. And, unlike fixedasf, they retain that interpretation when computed over subpopulations for both linear and nonlinear models.
scores calculates the equation-level score variables.
For models with one endogenous regressor, four new variables are created.
The first new variable will contain \(\partial \ln L / \partial\left(\boldsymbol{z}_{i} \delta\right)\).
The second new variable will contain \(\partial \ln L / \partial\left(\boldsymbol{x}_{i} \boldsymbol{\Pi}\right)\).
The third new variable will contain \(\partial \ln L / \partial\) atanh \(\rho\).
The fourth new variable will contain \(\partial \ln L / \partial \ln \sigma\).
For models with \(p\) endogenous regressors, \(p+\{(p+1)(p+2)\} / 2\) new variables are created.
The first new variable will contain \(\partial \ln L / \partial\left(\boldsymbol{z}_{i} \boldsymbol{\delta}\right)\).
The second through \((p+1)\) th new variables will contain \(\partial \ln L / \partial\left(\boldsymbol{x}_{i} \boldsymbol{\Pi}_{k}\right), k=1, \ldots, p\), where \(\Pi_{k}\) is the \(k\) th column of \(\Pi\).
The remaining score variables will contain the partial derivatives of \(\ln L\) with respect to \(s_{21}\), \(s_{31}, \ldots, s_{p+1,1}, s_{22}, \ldots, s_{p+1,2}, \ldots, s_{p+1, p+1}\), where \(s_{m, n}\) denotes the \((m, n)\) element of the Cholesky decomposition of the error covariance matrix.

\section*{margins}

\section*{Description for margins}
margins estimates margins of response for linear predictions and probabilities.

\section*{Menu for margins}

Statistics \(>\) Postestimation

\section*{Syntax for margins}
```

margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...)...] [options]

```
statistic Description

Main
\begin{tabular}{ll}
cm & conditional mean; the default \\
xb & linear prediction \\
stdp & not allowed with margins
\end{tabular}

Statistics not allowed with margins are functions of stochastic quantities other than e(b). For the full syntax, see \([R]\) margins.

\section*{estat}

\section*{Description for estat}
estat correlation displays the correlation matrix of the errors of the dependent variable and the endogenous variables.
estat covariance displays the covariance matrix of the errors of the dependent variable and the endogenous variables.

\section*{Menu for estat}

Statistics \(>\) Postestimation

\section*{Syntax for estat}

Correlation matrix
```

estat correlation [, border(bspec) left(\#) format(%fmt)]

```

Covariance matrix
```

estat covariance [, border(bspec) left(\#) format(%fmt)]

```

\section*{Options for estat}
\(\qquad\) Main
border (bspec) sets the border style of the matrix display. The default is border (all). left(\#) sets the left indent of the matrix display. The default is left(2).
format ( \(\% f m t\) ) specifies the format for displaying the individual elements of the matrix. The default is format \((\% 9.0 \mathrm{~g})\).

\section*{Remarks and examples}

Remarks are presented under the following headings:

\author{
Marginal effects \\ Obtaining predicted values
}

\section*{Marginal effects}

\section*{> Example 1}

We can obtain marginal effects by using the margins command after ivfprobit. We will calculate average marginal effects by using the \(401(\mathrm{k})\) participation model of example 1 in [R] ivfprobit.
```

. use https://www.stata-press.com/data/r18/401k
(Firm-level data on 401k participation)
. ivfprobit prate c.ltotemp\#\#c.ltotemp i.sole (mrate = c.age\#\#c.age)
(output omitted)
. margins, dydx(mrate) predict(cm)

```
Average marginal effects \(\quad\) Number of obs \(=4,075\)
Model VCE: Robust
Expression: Conditional mean of prate, predict(cm)
dy/dx wrt: mrate
\begin{tabular}{r|rrrrrr}
\hline & \multicolumn{4}{|c}{ Delta-method } \\
& dy/dx & std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline mrate & .5288314 & .0442438 & 11.95 & 0.000 & .4421152 & .6155477 \\
\hline
\end{tabular}

The marginal effect from ivfprobit suggests that a \(\$ 1\) increase in the matching rate (per dollar) given by employers can increase participation by approximately \(50 \%\).

\section*{Obtaining predicted values}

After fitting your model with ivfprobit, you can obtain the conditional mean, or the linear prediction and its standard error for both the estimation sample and other samples, by using the predict command; see [U] 20 Estimation and postestimation commands and [R] predict.

\section*{Stored results}
estat correlation stores the following results in \(r()\) :
Matrices
\(r\) (corr) correlation matrix of the errors
estat covariance stores the following results in \(r()\) :
Matrices
\(r\) (cov) covariance matrix of the errors

\section*{Methods and formulas}

Recall that the model fit by ivfprobit is given by two equations:
\[
\begin{aligned}
E\left[y_{1 i} \mid \mathbf{x}_{1 i}, \mathbf{x}_{2 i}, \mathbf{y}_{2 i}, u_{i}\right] & =\Phi\left(\mathbf{y}_{2 i} \boldsymbol{\beta}+\mathbf{x}_{1 i} \boldsymbol{\gamma}+u_{i}\right) \\
\mathbf{y}_{2 i} & =\mathbf{x}_{i 1} \boldsymbol{\Pi}_{1}+\mathbf{x}_{i 2} \boldsymbol{\Pi}_{2}+v_{i}
\end{aligned}
\]

The linear prediction for observation \(i\) is calculated as \(\mathbf{y}_{2 i} \widehat{\beta}+\mathbf{x}_{1 i} \widehat{\gamma}\). The predicted conditional mean is given by \(\Phi\left(\widehat{m}_{i}\right)\), where \(\Phi(\cdot)\) is the standard normal distribution function and \(\widehat{m}_{i}\) is the plugin estimator of
\[
m_{i}=\frac{\mathbf{y}_{2 i} \boldsymbol{\beta}+\mathbf{x}_{1 i} \boldsymbol{\gamma}+\rho v_{i} / \sigma}{\sqrt{1-\rho^{2}}}
\]
where \(\sigma\) is the standard deviation of \(v_{i}\) and \(\rho\) is the correlation coefficient between \(u_{i}\) and \(v_{i}\). The ASF uses \(\widehat{m}_{i}\) instead of \(\mathbf{y}_{2 i} \widehat{\boldsymbol{\beta}}+\mathbf{x}_{1 i} \widehat{\gamma}\) to evaluate \(\Phi(\cdot)\) and account for endogeneity in the model. The fixed ASF is evaluated at \(\mathbf{y}_{2 i} \widehat{\boldsymbol{\beta}}+\mathbf{x}_{1 i} \widehat{\boldsymbol{\gamma}}\).

\section*{Also see}
[R] ivfprobit - Fractional probit model with continuous endogenous covariates
[U] 20 Estimation and postestimation commands

\section*{Title}
ivpoisson - Poisson model with continuous endogenous covariates
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
ivpoisson estimates the parameters of a Poisson regression model in which some of the covariates are endogenous. The model is also known as an exponential conditional mean model in which some of the covariates are endogenous. The model may be specified using either additive or multiplicative error terms. The model is frequently used to model count outcomes and is also used to model nonnegative outcome variables.

\section*{Quick start}

Two-step GMM estimation of the Poisson regression of y 1 on x and endogenous regressor y 2 that is instrumented using \(\mathbf{z}\)
ivpoisson gmm y1 x (y2 = z)
Same as above, but specify multiplicative errors rather than additive
ivpoisson gmm y1 x (y2 = z), multiplicative
Use iterative GMM estimation
ivpoisson gmm y1 x (y2 = z), igmm
Specify a weight matrix that allows for correlation within clusters identified by cvar
ivpoisson gmm y1 x (y2 = z), wmatrix(cluster cvar)
Use the control-function estimator
ivpoisson cfunction y1 x (y2 = z)

\section*{Menu}

Statistics \(>\) Endogenous covariates \(>\) Poisson model with endogenous covariates

\section*{Syntax}

Generalized method of moments estimator
ivpoisson gmm depvar \(\left[\right.\) varlist \(\left._{1}\right]\left[\left(\right.\right.\) varlist \(_{2}=\) varlist \(\left.\left._{\mathrm{iv}}\right)\right][\) if \(][\) in \(][\) weight \(]\)
[, reg_err_opt options]

Control-function estimator
ivpoisson cfunction depvar \(\left[\right.\) varlist \(\left._{1}\right] \quad\left(\right.\) varlist \(_{2}=\) varlist \(\left._{\mathrm{iv}}\right)\) [if][in][ weight]
[, options]
reg_err_opt
Description
Model
additive multiplicative
add regression errors to the conditional mean term; the default multiply regression errors by the conditional mean term

Description

\section*{Model}
noconstant
exposure (varname \(e_{e}\) )
offset ( varname \(_{o}\) )
* twostep
* onestep
* igmm

Weight matrix wmatrix (wmtype)
center winitial (iwtype[, independent])
specify initial weight matrix; iwtype may be unadjusted, identity, or the name of a Stata matrix
(independent may not be specified with ivpoisson gmm)
SE/Robust
suppress constant term
include \(\ln \left(\right.\) varname \(\left._{e}\right)\) in model with coefficient constrained to 1 include varname \(e_{o}\) in model with coefficient constrained to 1 use two-step GMM estimator; the default for ivpoisson gmm use one-step GMM estimator; the default for ivpoisson cfunction use iterative GMM estimator
specify weight matrix; wmtype may be robust, cluster clustvar, or unadjusted
center moments in weight-matrix computation
vce(vcetype)

Reporting
level(\#)
irr
display_options

Optimization
from(initial_values)
\(\dagger\) igmmiterate(\#)
\(\dagger \overline{\text { igmmeps }}(\#)\)
\(\dagger\) igmmweps (\#)
optimization_options
vcetype may be robust, cluster clustvar, bootstrap, jackknife, or unadjusted
set confidence level; default is level (95)
report incidence-rate ratios
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
specify initial values for parameters
specify maximum number of iterations for iterated GMM estimator
specify \# for iterated GMM parameter convergence criterion; default is igmmeps (1e-6)
specify \# for iterated GMM weight-matrix convergence criterion; default is igmmweps (1e-6)
control the optimization process; seldom used
*You can specify at most one of these options.
\(\dagger\) These options may be specified only when igmm is specified.
varlist \(_{1}\) and varlist \(_{\mathrm{iv}}\) may contain factor variables; see [U] 11.4.3 Factor variables.
depvar, varlist \({ }_{1}\), varlist \(t_{2}\), and varlist \({ }_{\mathrm{iv}}\) may contain time-series operators; see [U] 11.4.4 Time-series varlists. bootstrap, by, collect, jackknife, rolling, and statsby are allowed; see [U] 11.1.10 Prefix commands. Weights are not allowed with the bootstrap prefix; see [R] bootstrap.
aweights are not allowed with the jackknife prefix; see \([R]\) jackknife.
aweights, fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options}
\(\qquad\)
noconstant, exposure (varname \()\), offset (varname \({ }_{o}\) ); see [R] Estimation options.
additive, the default, specifies that the regression errors be added to the conditional mean term and have mean 0 .
multiplicative specifies that the regression errors be multiplied by the conditional mean term and have mean 1.
twostep, onestep, and igmm specify which estimator is to be used.
twostep requests the two-step GMM estimator. gmm obtains parameter estimates based on the initial weight matrix, computes a new weight matrix based on those estimates, and then reestimates the parameters based on that weight matrix. twostep is the default for ivpoisson gmm.
onestep requests the one-step GMM estimator. The parameters are estimated based on an initial weight matrix, and no updating of the weight matrix is performed except when calculating the appropriate variance-covariance (VCE) matrix. onestep is the default for ivpoisson cfunction.
igmm requests the iterative GMM estimator. gmm obtains parameter estimates based on the initial weight matrix, computes a new weight matrix based on those estimates, reestimates the parameters based on that weight matrix, computes a new weight matrix, and so on, to convergence. Convergence is declared when the relative change in the parameter vector is less than igmmeps (), the relative change in the weight matrix is less than igmmweps(), or igmmiterate() iterations have been completed. Hall (2005, sec. 2.4 and 3.6) mentions that there may be gains to finite-sample efficiency from using the iterative estimator.

Weight matrix
wmatrix (wmtype) specifies the type of weight matrix to be used in conjunction with the two-step and iterated GMM estimators.

Specifying wmatrix (robust) requests a weight matrix that is appropriate when the errors are independent but not necessarily identically distributed. wmatrix (robust) is the default.
Specifying wmatrix(cluster clustvar) requests a weight matrix that accounts for arbitrary correlation among observations within clusters identified by clustvar.
Specifying wmatrix(unadjusted) requests a weight matrix that is suitable when the errors are homoskedastic.
wmatrix() cannot be specified if onestep is also specified.
center requests that the sample moments be centered (demeaned) when computing GMM weight matrices. By default, centering is not done.
winitial (wmtype [, independent]) specifies the weight matrix to use to obtain the first-step parameter estimates.

Specifying winitial(unadjusted) requests a weighting matrix that assumes the error functions are independent and identically distributed. This matrix is of the form \(\left(\mathbf{Z}^{\prime} \mathbf{Z}\right)^{-1}\), where \(\mathbf{Z}\) represents all the exogenous regressors and instrumental variables.
winitial (identity) requests that the identity matrix be used.
winitial (matname) requests that Stata matrix matname be used.

Including the independent suboption creates a weight matrix that assumes error functions are independent. Elements of the weight matrix corresponding to covariances between any two error functions are set equal to zero. This suboption only applies to ivpoisson cfunction.
winitial (unadjusted) is the default for ivpoisson gmm.
winitial(unadjusted, independent) is the default for ivpoisson cfunction.

SE/Robust
vce (vcetype) specifies the type of standard error reported, which includes types that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
vce(unadjusted) specifies that an unadjusted (nonrobust) VCE matrix be used; this, along with the twostep option, results in the "optimal two-step GMM" estimates often discussed in textbooks. vce(unadjusted) may not be set in ivpoisson cfunction.
The default vcetype is based on the wmtype specified in the wmatrix() option. If wmatrix() is specified but vce() is not, then vcetype is set equal to wmtype. To override this behavior in ivpoisson gmm and obtain an unadjusted (nonrobust) VCE matrix, specify vce(unadjusted). The default vcetype for ivpoisson cfunction is robust.
Specifying vce(bootstrap) or vce (jackknife) results in standard errors based on the bootstrap or jackknife, respectively. See \([R]\) vee_option, \([R]\) bootstrap, and \([R]\) jackknife for more information on these VCEs.

The syntax for vcetypes is identical to those for wmatrix().

\section*{Reporting}

\section*{level (\#); see [R] Estimation options.}
irr reports estimated coefficients transformed to incidence-rate ratios, that is, \(e^{\beta_{i}}\) rather than \(\beta_{i}\). Standard errors and confidence intervals are similarly transformed. This option affects how results are displayed, not how they are estimated or stored. irr may be specified at estimation or when replaying previously estimated results. irr is not allowed with additive.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.

Optimization
from(initial_values) specifies the initial values to begin the estimation. You can specify a \(1 \times k\) matrix, where \(k\) is the number of parameters in the model, or you can specify a parameter name, its initial value, another parameter name, its initial value, and so on. For example, to initialize the coefficient for male to 1.23 and the constant _cons to 4.57 , you would type
```

ivpoisson ..., from(male 1.23 _cons 4.57) ...

```

Initial values declared using this option override any that are declared within substitutable expressions. If you specify a parameter that does not appear in your model, ivpoisson exits with error code 480 . If you specify a matrix, the values must be in the same order in which the parameters are declared in your model. ivpoisson ignores the row and column names of the matrix.
igmmiterate(\#), igmmeps(\#), and igmmweps(\#) control the iterative process for the iterative GMM estimator for ivpoisson. These options can be specified only if you also specify igmm.
igmmiterate(\#) specifies the maximum number of iterations to perform with the iterative GMM estimator. The default is the number set using set maxiter, which is 300 by default.
igmmeps (\#) specifies the convergence criterion used for successive parameter estimates when the iterative GMM estimator is used. The default is igmmeps ( \(1 \mathrm{e}-6\) ). Convergence is declared when the relative difference between successive parameter estimates is less than igmmeps () and the relative difference between successive estimates of the weight matrix is less than igmmweps ().
igmmweps (\#) specifies the convergence criterion used for successive estimates of the weight matrix when the iterative GMM estimator is used. The default is igmmweps ( \(1 \mathrm{e}-6\) ). Convergence is declared when the relative difference between successive parameter estimates is less than igmmeps() and the relative difference between successive estimates of the weight matrix is less than igmmweps().
optimization_options: technique(), conv_maxiter(), conv_ptol(), conv_vtol(),
conv_nrtol(), and tracelevel(). technique() specifies the optimization technique to use; gn (the default), nr , dfp , and bfgs are allowed. conv_maxiter() specifies the maximum number of iterations; conv_ptol(), conv_vtol(), and conv_nrtol() specify the convergence criteria for the parameters, gradient, and scaled Hessian, respectively. tracelevel () allows you to obtain additional details during the iterative process. See [M-5] optimize().

\section*{Remarks and examples}
ivpoisson estimates the parameters of a Poisson regression model in which some of the covariates are endogenous. A regressor is endogenous if it is related to the unobserved error term. The model is also known as an exponential conditional mean model in which some of the covariates are endogenous. The model may be specified using either additive or multiplicative error terms.

The model is frequently used to model count outcomes and is also used to model nonnegative outcome variables. Poisson regression is a special exponential conditional mean model. See \([\mathrm{R}]\) poisson for more information on Poisson regression.

The exponential conditional mean model has an error form representation in which the dependent variable \(y\) is a function of the exogenous covariates \(\mathbf{x}\), endogenous covariates \(\mathbf{y}_{2}\), and an error \(\epsilon\). The covariates \(\mathbf{x}\) are independent of \(\epsilon\), while \(\mathbf{y}_{2}\) are not.
ivpoisson allows \(\epsilon\) to enter either additively,
\[
y_{i}=\exp \left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}_{1}+\mathbf{y}_{2, i}^{\prime} \boldsymbol{\beta}_{2}\right)+\epsilon_{i}
\]
or multiplicatively,
\[
y_{i}=\exp \left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}_{1}+\mathbf{y}_{2, i}^{\prime} \boldsymbol{\beta}_{2}\right) \epsilon_{i}
\]

Mullahy (1997), Cameron and Trivedi (2013), Windmeijer and Santos Silva (1997), and Wooldridge (2010) discuss the generalized method of moments (GMM) estimators implemented in ivpoisson. GMM is frequently used in modern econometrics. Many econometric and statistical models can be expressed as conditions on the population moments. The parameter estimates produced by GMM estimators make the sample-moment conditions as true as possible given the data. See \([R]\) gmm for further information on GMM estimation and how Stata performs it.

The rest of the discussion is presented under the following headings:

\section*{GMM estimator for additive model}

The GMM estimator uses additional variables, known as instruments and denoted by \(\mathbf{z}_{i}\), to specify moment conditions that hold in the population. The GMM parameter estimates make the sample versions of these population-moment conditions as close to true as possible. The instrumental variables are assumed to be correlated with the endogenous covariates \(\mathbf{y}_{2, i}\) but independent of the errors \(\epsilon_{i}\).

Under additive errors, the dependent variable \(y_{i}\) is determined by exogenous covariates \(\mathbf{x}_{i}\), endogenous covariates \(\mathbf{y}_{2, i}\), and zero-mean error \(\epsilon_{i}\) as
\[
y_{i}=\exp \left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}_{1}+\mathbf{y}_{2, i}^{\prime} \boldsymbol{\beta}_{2}\right)+\epsilon_{i}
\]

This leads to the following error function
\[
u\left(y_{i}, \mathbf{x}_{i}, \mathbf{y}_{2, i}, \boldsymbol{\beta}_{1}, \boldsymbol{\beta}_{2}\right)=y_{i}-\exp \left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}_{1}+\mathbf{y}_{2, i}^{\prime} \boldsymbol{\beta}_{2}\right)
\]

The population-moment conditions for GMM estimation are \(E\left\{\widetilde{\mathbf{z}}_{i} u\left(y_{i}, \mathbf{x}_{i}, \mathbf{y}_{2, i}, \boldsymbol{\beta}_{1}, \boldsymbol{\beta}_{2}\right)\right\}=\mathbf{0}\), where the vector \(\widetilde{\mathbf{z}}_{i}\) is partitioned as \(\left(\mathbf{x}_{i}^{\prime}, \mathbf{z}_{i}^{\prime}\right)\). The sample-moment conditions are formed by replacing the expectation with the corresponding sample mean. The GMM estimator solves a minimization problem to make the sample-moment conditions as close to zero as possible. Details on how estimation is performed are given in Methods and formulas.

Now, we will demonstrate how ivpoisson gmm works in the additive error setting with an example.

\section*{> Example 1: ivpoisson gmm with additive errors}

This example uses simulated data based on the following story. A news website randomly samples 500 young adults in a major city. The website wants to model the number of times the sampled individuals visit its website (visits) based on their overall time spent on the Internet (time) and the number of times they receive an ad for the website through email or viewing another website (ad). The website also suspects the gender of the individual may matter, so an exogenous dummy variable, female, is included in the model.

We suspect time spent on the Internet is correlated with unobserved factors that additively affect the number of times an individual visits the website. So we treat time as an endogenous regressor. Two instruments are used for this variable. The time spent on the phone (phone) is one instrument. The other instrument is the time spent interacting with friends and family that live out of town (frfam).

We model the number of visits the website receives using an exponential conditional mean model with additive errors and use ivpoisson gmm to estimate the parameters of the regression in the output below. To allow for heteroskedasticity of the errors, we use robust standard errors, which is the default; see Obtaining standard errors in \([\mathrm{R}] \mathrm{gmm}\) for a discussion of why robust standard errors is the default.
```

. use https://www.stata-press.com/data/r18/website
(Visits to website)
. ivpoisson gmm visits ad female (time = phone frfam)
Step 1
Iteration 0: GMM criterion Q(b) = . 33829416
Iteration 1: GMM criterion Q(b) = .00362656
Iteration 2: GMM criterion Q(b) = .00131886
Iteration 3: GMM criterion Q(b) = .00131876
Step 2
Iteration 0: GMM criterion Q(b) = .00027102
Iteration 1: GMM criterion Q(b) = .00025811
Iteration 2: GMM criterion Q(b) = .00025811

```
\begin{tabular}{lll} 
Exponential mean model with endogenous regressors & & \\
Number of parameters \(=\) & 4 & Number of obs \(=\) \\
Number of moments \(=\) & 5 & \\
Initial weight matrix: Unadjusted & & \\
GMM weight matrix: & Robust &
\end{tabular}
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline visits & Coefficient & Robust std. err. & z & \(P>|z|\) & [95\% conf & interval] \\
\hline time & . 0589294 & . 0107942 & 5.46 & 0.000 & . 0377732 & . 0800857 \\
\hline ad & . 137344 & . 010157 & 13.52 & 0.000 & . 1174366 & . 1572515 \\
\hline female & -. 0247707 & . 0376218 & -0.66 & 0.510 & -. 098508 & . 0489666 \\
\hline _cons & 1.041505 & . 0385848 & 26.99 & 0.000 & . 9658807 & 1.11713 \\
\hline
\end{tabular}

Endogenous: time
Exogenous: ad female phone frfam
We find significant coefficients for all covariates but female. At fixed values of the other covariates, increased time spent on the Internet will raise the expected number of website visits. Receiving additional advertisements will also cause an increase in the expected number of website visits.

\section*{GMM estimator for multiplicative model}

Under multiplicative errors, the dependent variable \(y_{i}\) is determined by exogenous covariates \(\mathbf{x}_{i}\), endogenous covariates \(\mathbf{y}_{2, i}\), and unit-mean errors \(\epsilon_{i}\) as
\[
y_{i}=\exp \left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}_{1}+\mathbf{y}_{2, i}^{\prime} \boldsymbol{\beta}_{2}\right) \epsilon_{i}
\]

This setting yields a different error function than the additive error case. This ratio formulation is
\[
u\left(y_{i}, \mathbf{x}_{i}, \mathbf{y}_{2, i}, \boldsymbol{\beta}_{1}, \boldsymbol{\beta}_{2}\right)=y_{i} / \exp \left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}_{1}+\mathbf{y}_{2, i}^{\prime} \boldsymbol{\beta}_{2}\right)-1
\]

Given the instrumental variables \(\mathbf{z}\), the population-moment conditions for GMM estimation are \(E\left\{\widetilde{\mathbf{z}}_{i} u\left(y_{i}, \mathbf{x}_{i}, \mathbf{y}_{2, i}, \boldsymbol{\beta}_{1}, \boldsymbol{\beta}_{2}\right)\right\}=\mathbf{0}\). The vector \(\widetilde{\mathbf{z}}_{i}\) is partitioned as \(\left(\mathbf{x}_{i}^{\prime}, \mathbf{z}_{i}^{\prime}\right)\). As above, the samplemoment conditions are the sample analogs of the population-moment conditions, and the GMM estimator solves a minimization problem to make the sample-moment conditions as close to zero as possible. Details on how estimation is performed are given in Methods and formulas.
> Example 2: ivpoisson gmm with multiplicative errors
In this example, we observe a simulated random sample of 5,000 households. We model the number of trips taken by members of the household in the 24 -hour period immediately prior to the interview time by using an exponential conditional mean model with multiplicative errors. Exogenous covariates include the distance to the central business district from the household (cbd), the distance from the household to a public transit node (ptn), whether there is a full-time worker in the household (worker), and whether the examined period is on a weekend (weekend). We suspect that the endogenous regressor, the transportation cost of the household in the prior week (tcost), is correlated with unobserved factors that affect the number of trips taken. This transportation cost includes gasoline and bus, train tickets, etc.

The ratio of the cost of a public transit day pass in the sampled area to the national average cost of such a pass ( pt ) is also observed. This is used as an instrument for transportation cost.

In the output below, we estimate the parameters of the regression with ivpoisson gmm. To allow for heteroskedasticity of the errors, we use robust standard errors, which is the default.
```

. use https://www.stata-press.com/data/r18/trip
(Household trips)
. ivpoisson gmm trips cbd ptn worker weekend (tcost = pt), multiplicative
Step 1
Iteration 0: GMM criterion Q(b) = .04949852
Iteration 1: GMM criterion Q(b) = .00011194
Iteration 2: GMM criterion Q(b) = 1.563e-08
Iteration 3: GMM criterion Q(b) = 3.685e-16
Step 2
Iteration 0: GMM criterion Q(b) = 2.287e-16
Iteration 1: GMM criterion Q(b) = 1.275e-31
note: model is exactly identified.
Exponential mean model with endogenous regressors
Number of parameters = 6 Number of obs = 5,000
Number of moments = 6
Initial weight matrix: Unadjusted
GMM weight matrix: Robust

```
\begin{tabular}{r|rcrcrr}
\hline & Robust \\
trips & Coefficient & std. err. & \multicolumn{1}{c}{z} & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline tcost & .0352185 & .0098182 & 3.59 & 0.000 & .0159752 & .0544617 \\
cbd & -.008398 & .0020172 & -4.16 & 0.000 & -.0123517 & -.0044444 \\
ptn & -.0113146 & .0021819 & -5.19 & 0.000 & -.015591 & -.0070383 \\
worker & .6623018 & .0519909 & 12.74 & 0.000 & .5604015 & .764202 \\
weekend & .3009323 & .0362682 & 8.30 & 0.000 & .2298479 & .3720167 \\
_cons & .2654423 & .1550127 & 1.71 & 0.087 & -.0383769 & .5692616 \\
\hline
\end{tabular}

Endogenous: tcost
Exogenous: cbd ptn worker weekend pt
We find that all coefficients are significant. At fixed values of the other covariates, we see that additional mileage from the central business district and public transit nodes reduces the expected number of trips taken. Individuals who live farther away from the central business district may still be out of the house the same amount of time, but they will take fewer trips because the transit time has increased. The situation is similar for those who live farther from public transit.

To interpret the other parameters, we will look at the partial effects of their respective independent variables. The partial effects of a change in an independent variable on the modeled conditional expectation function vary over the data because the model is nonlinear. However, under the multiplicative error model, the ratio of the new value to the old value after a discrete change in an independent variable is constant over the data.

Let \(\mathbf{w}=\left(\mathbf{x}^{\prime}, \mathbf{y}_{2}^{\prime}\right)^{\prime}\). If we add 1 to the \(j\) th independent variable in \(\mathbf{w}\), the functional form of the model implies that
\[
\frac{E\left(y \mid \mathbf{w}, w_{j}+1, \epsilon\right)}{E\left(y \mid \mathbf{w}, w_{j}, \epsilon\right)}=\frac{E\left(y \mid w_{1}, \ldots, w_{j}+1, \ldots, w_{k}, \epsilon\right)}{E\left(y \mid w_{1}, \ldots, w_{j}, \ldots, w_{k}, \epsilon\right)}=e^{\beta_{j}}
\]

When \(y\) is a count variable, this normalized effect is called the incidence-rate ratio (IRR) for a one-unit change in \(w_{j}\).

More generally, the IRR for a \(\Delta w_{j}\) change in \(w_{j}\) is \(e^{\beta_{j} \Delta w_{j}}\) under a multiplicative-error exponential conditional mean model. We can calculate incidence-rate ratios for different changes in the covariates by using lincom; see [R] lincom.

Here we replay the ivpoisson results by typing the command name and we specify the irr option to get the incidence-rate ratios. Each significance test for a coefficient equaling zero becomes a test for the incidence-rate ratio equaling one.
```

. ivpoisson, irr
Exponential mean model with endogenous regressors

| Number of parameters $=$ | 6 | Number of obs $=$ | 5,000 |
| :--- | :--- | :--- | :--- |
| Number of moments $=$ | 6 |  |  |
| Initial weight matrix: Unadjusted |  |  |  |
| GMM weight matrix: | Robust |  |  |

```
\begin{tabular}{r|rcccrr}
\hline & \multicolumn{5}{c}{\begin{tabular}{c} 
Robust \\
trips
\end{tabular}} & IRR \\
std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline tcost & 1.035846 & .0101701 & 3.59 & 0.000 & 1.016103 & 1.055972 \\
cbd & .9916371 & .0020003 & -4.16 & 0.000 & .9877243 & .9955655 \\
ptn & .9887491 & .0021573 & -5.19 & 0.000 & .9845299 & .9929864 \\
worker & 1.939251 & .1008234 & 12.74 & 0.000 & 1.751376 & 2.14728 \\
weekend & 1.351118 & .0490026 & 8.30 & 0.000 & 1.258409 & 1.450657 \\
_cons & 1.304008 & .2021377 & 1.71 & 0.087 & .9623501 & 1.766962 \\
\hline
\end{tabular}

Note: _cons estimates baseline incidence rate.
Endogenous: tcost
Exogenous: cbd ptn worker weekend pt
Holding other covariates and the error constant, the expected number of trips made from houses with a full-time worker is nearly twice that of those houses without a full-time worker. Similarly, the expected number of trips made during a weekend day is close to \(35 \%\) higher than the expected number of trips made on other days. For each additional dollar of weekly transportation cost, the expected number of household trips is increased by approximately \(3.6 \%\).

\section*{CF estimator for multiplicative model}

Control-function (CF) estimators can be used to account for endogenous covariates. As Wooldridge (2010, sec. 18.5) describes, CF estimators assume a certain structural relationship between the endogenous covariates and the exogenous covariates and use functions of first-stage parameter estimates to control for the endogeneity in the second stage.

Wooldridge (2010, sec. 18.5) notes that the VCE of the second-stage estimator must be adjusted to account for estimates from the first stage. ivpoisson cfunction solves this problem by stacking the moment conditions that define each stage and applying a single GMM estimator. See Newey (1984) and Wooldridge (2010, sec. 14.2) for a description of this technique. No adjustment to the VCE is necessary because there is only one stage.

The CF estimator augments the original multiplicative model with an estimated term that controls for the endogeneity of \(\mathbf{y}_{2, i}\). When \(\mathbf{y}_{2, i}\) is exogenous, the coefficient on this control term is zero. Let \(\mathbf{z}\) be instrumental variables, and the vector \(\widetilde{\mathbf{z}}_{i}\) be \(\left(\mathbf{x}_{i}^{\prime}, \mathbf{z}_{i}^{\prime}\right)\).

The augmented model is
\[
y_{i}=\exp \left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}_{1}+\mathbf{y}_{2, i}^{\prime} \boldsymbol{\beta}_{2}+\mathbf{v}_{i}^{\prime} \boldsymbol{\rho}+c_{i}\right)
\]
where
\[
\mathbf{y}_{2, i}=\mathbf{B} \widetilde{\mathbf{z}}_{i}^{\prime}+\mathbf{v}_{i}
\]

The term \(\mathbf{v}_{i}^{\prime} \boldsymbol{\rho}\) controls for the endogeneity of \(\mathbf{y}_{2, i}\), and we normalize \(E\left\{\exp \left(c_{i}\right)\right\}=1\). The coefficient vector \(\boldsymbol{\rho}\) measures the strength of the endogeneity of \(\mathbf{y}_{2, i} ; \mathbf{y}_{2, i}\) is exogenous when \(\boldsymbol{\rho}=\mathbf{0}\).
ivpoisson cfunction estimates \(\boldsymbol{\beta}_{1}\) and \(\boldsymbol{\beta}_{2}\) and the auxiliary parameters \(\boldsymbol{\rho}\) and \(\mathbf{B}\) by GMM; see Methods and formulas for details.

\section*{> Example 3: Control-function estimator}

We return to the previous example, where we estimated the parameters of an exponential conditional mean model for the number of trips taken by a household in a 24 -hour period. We will estimate the parameters of the regression with the CF estimator method and compare our results with those obtained with the GMM estimator in example 2.

In the output below, we estimate the parameters of the regression with the ivpoisson cfunction command.
```

. ivpoisson cfunction trips cbd ptn worker weekend (tcost = pt)
Step 1
Iteration 0: GMM criterion Q(b) = .00056156
Iteration 1: GMM criterion Q(b) = 2.366e-07
Iteration 2: GMM criterion Q(b) = 5.552e-14
Iteration 3: GMM criterion Q(b) = 9.760e-27
note: model is exactly identified.
Exponential mean model with endogenous regressors
Number of parameters = 13 Number of obs = 5,000
Number of moments= = 13
Initial weight matrix: Unadjusted
GMM weight matrix: Robust

```
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline trips & Coefficient & Robust std. err. & z & \(P>|z|\) & [95\% conf & interval] \\
\hline \multicolumn{7}{|l|}{trips} \\
\hline cbd & -. 0082567 & . 0020005 & -4.13 & 0.000 & -. 0121777 & -. 0043357 \\
\hline ptn & -. 0113719 & . 0021625 & -5.26 & 0.000 & -. 0156102 & -. 0071335 \\
\hline worker & . 6903044 & . 0521642 & 13.23 & 0.000 & . 5880645 & . 7925444 \\
\hline weekend & . 2978149 & . 0356474 & 8.35 & 0.000 & . 2279472 & . 3676825 \\
\hline tcost & . 0320718 & . 0092738 & 3.46 & 0.001 & . 0138955 & . 0502481 \\
\hline _cons & . 2145986 & . 1359327 & 1.58 & 0.114 & -. 0518246 & . 4810218 \\
\hline \multicolumn{7}{|l|}{tcost} \\
\hline cbd & . 0165466 & . 0043693 & 3.79 & 0.000 & . 0079829 & . 0251102 \\
\hline ptn & -. 040652 & . 0045946 & -8.85 & 0.000 & -. 0496573 & -. 0316467 \\
\hline worker & 1.550985 & . 0996496 & 15.56 & 0.000 & 1.355675 & 1.746294 \\
\hline weekend & . 0423009 & . 0779101 & 0.54 & 0.587 & -. 1104002 & . 1950019 \\
\hline pt & . 7739176 & . 0150072 & 51.57 & 0.000 & . 7445041 & . 8033312 \\
\hline _cons & 12.13934 & . 1123471 & 108.05 & 0.000 & 11.91915 & 12.35954 \\
\hline /c_tcost & . 1599984 & . 0111752 & 14.32 & 0.000 & . 1380954 & . 1819014 \\
\hline
\end{tabular}

Endogenous: tcost
Exogenous: cbd ptn worker weekend pt
The output table presents results for the estimated coefficients in each of three equations. First, in the trips equation, we see the results for the estimated coefficients in the equation for the dependent variable trips. Second, in the tcost equation, we see the estimated coefficients in the regression of tcost on the instrumental and exogenous variables. Third, the /c_tcost ancillary parameter corresponds to the estimate of \(\rho\), the coefficient on the residual variable included to control for the endogeneity of tcost.

We find that all coefficients are significant in the exponential conditional mean equation, trips. The coefficient estimates in the trips equation are similar to the estimates obtained by the GMM estimator in example 2. That the estimated coefficient on the tcost control variable is significantly different from zero suggests that tcost is endogenous.

\section*{Stored results}
ivpoisson stores the following in e():
Scalars
```

e(N)
e(k)
e(k_eq)
e(k_aux)
e(k_dv)
e(Q)
e(J)
e(J_df)
e(N_clust)
e(rank)
e(ic)
e(converged)

```

Macros
e(cmd)
e(cmdline)
e(depvar)
e(endog)
e(exog)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(offset1)
e(winit)
e(winitname)
e(estimator)
e(additive)
e(multiplicative)
e(gmmestimator)
e(wmatrix)
e(vce)
e(vcetype)
e(technique)
e(properties)
e(estat_cmd)
e(predict)
e(footnote)
e(marginsok)
e(marginsnotok)
e(asbalanced)
e(asobserved)
Matrices
e(b)
e(v)
e(init)
e(Wuser)
e(W)
e(S)
e(V_modelbased)
Functions
e(sample)
number of observations
number of parameters
number of equations
number of auxiliary parameters
number of dependent variables
criterion function
Hansen \(J \chi^{2}\) statistic
\(J\) statistic degrees of freedom
number of clusters
rank of \(e(V)\)
number of iterations used by iterative GMM estimator
1 if converged, 0 otherwise
ivpoisson
command as typed
dependent variable of regression
names of endogenous variables
names of exogenous variables
weight type
weight expression
title in estimation output
name of cluster variable
offset variable for first equation
initial weight matrix used
name of user-supplied initial weight matrix
gmm or cfunction
additive if additive errors specified
multiplicative if multiplicative errors specified
onestep, twostep, or igmm
wmtype specified in wmatrix()
vcetype specified in vce()
title used to label Std. err.
optimization technique
b V
program used to implement estat
program used to implement predict
program used to implement footnote display
predictions allowed by margins
predictions disallowed by margins
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
variance-covariance matrix
initial values of the estimators
user-supplied initial weight matrix
weight matrix used for final round of estimation
moment covariance matrix used in robust VCE computations model-based variance
marks estimation sample

In addition to the above, the following is stored in r() :
Matrices
\(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

\section*{Methods and formulas}

The estimators in ivpoisson are GMM estimators that can be expressed in terms of error functions and the instruments that are used to form the moment conditions. When offsets \(o_{j}^{\beta}\) are used in the outcome variable equation, the following formulas apply with \(\mathbf{x}_{j}^{\prime} \boldsymbol{\beta}_{1}\) changed to \(\mathbf{x}_{j}^{\prime} \boldsymbol{\beta}_{1}+o_{j}^{\beta}\).

The error functions for the GMM estimators are given in the text.
Here we provide some details about the form of the error function used by the CF estimator.
Recall that the multiplicative model is
\[
y_{i}=\exp \left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}_{1}+\mathbf{y}_{2, i}^{\prime} \boldsymbol{\beta}_{2}\right) \epsilon_{i}
\]

We parameterize the endogenous variables in the form
\[
\mathbf{y}_{2, i}=\mathbf{B} \widetilde{\mathbf{z}}_{i}^{\prime}+\mathbf{v}_{i}
\]

This allows us to decompose \(\epsilon_{i}\) as
\[
\epsilon_{i}=\exp \left(\mathbf{v}_{i}^{\prime} \boldsymbol{\rho}+c_{i}\right)
\]

Given this setup, we obtain the following conditional mean:
\[
E\left(y \mid \mathbf{x}_{i}, \mathbf{z}_{i}, \mathbf{y}_{2, i}, \mathbf{v}_{i}\right)=\exp \left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}_{1}+\mathbf{y}_{2, i}^{\prime} \boldsymbol{\beta}_{2}+\mathbf{v}_{i}^{\prime} \boldsymbol{\rho}\right)
\]

We estimate \(\mathbf{v}_{i}\) as the residuals of the linear regression of \(\mathbf{y}_{2, i}\) on \(\widetilde{\mathbf{z}}_{i}\). The estimates of \(\mathbf{v}_{i}\) are used as additional covariates in the exponential conditional mean model for \(y\) to estimate \(\boldsymbol{\beta}_{1}, \boldsymbol{\beta}_{2}\), and \(\rho\). In essence, the estimates of \(\mathbf{v}_{i}\) control for the endogeneity.

The error functions for the endogenous covariates are defined as
\[
\mathbf{u}_{e n, i}\left(\mathbf{y}_{2, i}, \widetilde{\mathbf{z}}_{i}, \mathbf{B}\right)=\mathbf{y}_{2, i}-\mathbf{B} \widetilde{\mathbf{z}}_{i}^{\prime}
\]

Now, we define the error function for the dependent variable as
\[
u_{y}\left(y_{i}, \mathbf{x}_{i}, \mathbf{y}_{2, i}, \mathbf{u}_{e n, i}, \boldsymbol{\beta}_{1}, \boldsymbol{\beta}_{2}, \boldsymbol{\rho}\right)=y_{i} / \exp \left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}_{1}+\mathbf{y}_{2, i}^{\prime} \boldsymbol{\beta}_{2}+\mathbf{u}_{e n, i}^{\prime} \boldsymbol{\rho}\right)-1
\]
\(\mathbf{u}_{e n, i}\) will be vector valued if we have multiple endogenous covariates \(\mathbf{y}_{2, i}\). Call the dimension of \(\mathbf{y}_{2, i} g . \mathbf{u}_{e n, i}\) and \(u_{y, i}\) define \(g+1\) separate error functions. We will use variables \(\widetilde{\mathbf{z}}_{i}\) to instrument each error function in \(\mathbf{u}_{e n, i}\). So for error function \(j=1, \ldots, g\), we have the error function \(u_{e n, i, j}\) and the population-moment conditions \(E\left(\widetilde{\mathbf{z}}_{i} u_{e n, i, j}\right)=\mathbf{0}\).

We calculate \(\widehat{\mathbf{v}}_{o i}\) previous to estimation as the residuals of the linear regression of \(\mathbf{y}_{2, i}\) on \(\widetilde{\mathbf{z}}_{i}\). We use variables \(\mathbf{x}_{i}, \mathbf{y}_{2, i}\), and \(\widehat{\mathbf{v}}_{o i}\) to instrument the error function \(u_{y}\). This leads to the population-moment conditions \(E\left\{\left(\mathbf{x}_{i}^{\prime}, \mathbf{y}_{2, i}^{\prime}, \widehat{\mathbf{v}}_{o i}^{\prime}\right) u_{y, i}\right\}=\mathbf{0}\)

Details of GMM estimation can be found in Methods and formulas of [R] gmm. Determination of the weight matrix \(\mathbf{W}_{N}\) is discussed there.

Under the GMM estimation, the GMM estimators \(\widehat{\boldsymbol{\beta}}_{1}\) and \(\widehat{\boldsymbol{\beta}}_{2}\) are the values of \(\boldsymbol{\beta}_{1}\) and \(\boldsymbol{\beta}_{2}\) that minimize
\[
\begin{equation*}
Q\left(\boldsymbol{\beta}_{1}, \boldsymbol{\beta}_{2}\right)=\left\{\frac{1}{N} \sum_{i} \widetilde{\mathbf{z}}_{i} u_{i}\left(y_{i}, \mathbf{x}_{i}, \mathbf{y}_{2, i}, \boldsymbol{\beta}_{1}, \boldsymbol{\beta}_{2}\right)\right\}^{\prime} \mathbf{W}_{N}\left\{\frac{1}{N} \sum_{i} \widetilde{\mathbf{z}}_{i} u_{i}\left(y_{i}, \mathbf{x}_{i}, \mathbf{y}_{2, i}, \boldsymbol{\beta}_{1}, \boldsymbol{\beta}_{2}\right)\right\} \tag{1}
\end{equation*}
\]
for \(q \times q\) weight matrix \(\mathbf{W}_{N}\), where \(q\) is the dimension of \(\widetilde{\mathbf{z}}_{i}\). The error functions \(u_{i}\) were defined in the text.

In the CF method, we have multiple error functions as defined above. We can stack the moment conditions and write them more compactly as \(\mathbf{Z}_{i}^{\prime} \mathbf{u}_{i}\left(\mathbf{B}, \boldsymbol{\beta}_{1}, \boldsymbol{\beta}_{2}, \boldsymbol{\rho}\right)\), where
\[
\mathbf{Z}_{i}=\left[\begin{array}{ccccc}
\mathbf{x}_{i}^{\prime} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\
\mathbf{0} & \mathbf{y}_{2, i}^{\prime} & \mathbf{0} & \cdots & \mathbf{0} \\
\mathbf{0} & \mathbf{0} & \widetilde{\mathbf{z}}_{i} & \cdots & \mathbf{0} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
\mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \widetilde{\mathbf{z}}_{i}
\end{array}\right]
\]
and
\[
\mathbf{u}_{i}\left(\mathbf{B}, \boldsymbol{\beta}_{1}, \boldsymbol{\beta}_{2}, \boldsymbol{\rho}\right)=\left[\begin{array}{c}
u_{y}\left(y_{i}, \mathbf{x}_{i}, \mathbf{y}_{2, i}, \mathbf{u}_{e n, i}, \boldsymbol{\beta}_{1}, \boldsymbol{\beta}_{2}, \boldsymbol{\rho}\right) \\
\mathbf{u}_{e n}\left(\mathbf{y}_{2, i}, \widetilde{\mathbf{z}}_{i}, \mathbf{B}\right)
\end{array}\right]
\]

The matrix \(\mathbf{Z}_{i}\) has \(g+1\) rows and \(k+g z\) columns, where \(k\) is the number of covariates for \(y_{i}\) and \(z\) is the number of exogenous covariates in \(\widetilde{\mathbf{z}}_{i}\).

The GMM estimators \(\widehat{\mathbf{B}}, \widehat{\boldsymbol{\beta}}_{1}, \widehat{\boldsymbol{\beta}}_{2}\), and \(\widehat{\boldsymbol{\rho}}\) are the values of \(\mathbf{B}, \boldsymbol{\beta}_{1}, \boldsymbol{\beta}_{2}\), and \(\boldsymbol{\rho}\) that minimize
\[
\begin{align*}
Q\left(\mathbf{B}, \boldsymbol{\beta}_{1}, \boldsymbol{\beta}_{2}, \boldsymbol{\rho}\right) & = \\
& \left\{N^{-1} \sum_{i=1}^{N} \mathbf{Z}_{i}^{\prime} \mathbf{u}_{i}\left(\mathbf{B}, \boldsymbol{\beta}_{1}, \boldsymbol{\beta}_{2}, \boldsymbol{\rho}\right)\right\}^{\prime} \mathbf{W}_{N}\left\{N^{-1} \sum_{i=1}^{N} \mathbf{Z}_{i}^{\prime} \mathbf{u}_{i}\left(\mathbf{B}, \boldsymbol{\beta}_{1}, \boldsymbol{\beta}_{2}, \boldsymbol{\rho}\right)\right\} \tag{2}
\end{align*}
\]
for \((k+g z) \times(k+g z)\) weight matrix \(\mathbf{W}_{N}\).
By default, ivpoisson minimizes (1) and (2) using the Gauss-Newton method. See Hayashi (2000, 498) for a derivation. This technique is typically faster than quasi-Newton methods and does not require second-order derivatives.

\section*{References}

Cameron, A. C., and P. K. Trivedi. 2013. Regression Analysis of Count Data. 2nd ed. New York: Cambridge University Press.
Hall, A. R. 2005. Generalized Method of Moments. Oxford: Oxford University Press.
Hayashi, F. 2000. Econometrics. Princeton, NJ: Princeton University Press.

Mullahy, J. 1997. Instrumental-variable estimation of count data models: Applications to models of cigarette smoking behavior. Review of Economics and Statistics 79: 586-593. https://doi.org/10.1162/003465397557169.

Newey, W. K. 1984. A method of moments interpretation of sequential estimators. Economics Letters 14: 201-206. https://doi.org/10.1016/0165-1765(84)90083-1.
Windmeijer, F., and J. M. C. Santos Silva. 1997. Endogeneity in count data models: An application to demand for health care. Journal of Applied Econometrics 12: 281-294. https://doi.org/10.1002/(SICI)1099-1255(199705)12: 3〈281::AID-JAE436〉3.0.CO;2-1.

Wooldridge, J. M. 2010. Econometric Analysis of Cross Section and Panel Data. 2nd ed. Cambridge, MA: MIT Press.

\section*{Also see}
[R] ivpoisson postestimation - Postestimation tools for ivpoisson
[R] gmm - Generalized method of moments estimation
[R] ivprobit - Probit model with continuous endogenous covariates
\([R]\) ivregress - Single-equation instrumental-variables regression
[R] ivtobit - Tobit model with continuous endogenous covariates
[R] nl - Nonlinear least-squares estimation
[R] nlsur - Estimation of nonlinear systems of equations
[R] poisson - Poisson regression
[R] regress - Linear regression
[U] 20 Estimation and postestimation commands

\section*{Title}
ivpoisson postestimation - Postestimation tools for ivpoisson
\begin{tabular}{lll} 
Postestimation commands & predict & margins \\
estat & Remarks and examples & Stored results \\
Methods and formulas & Reference & Also see
\end{tabular}

\section*{Postestimation commands}

The following postestimation command is of special interest after ivpoisson:
Command Description
estat overid perform test of overidentifying restrictions

The following standard postestimation commands are also available:
\begin{tabular}{ll} 
Command & Description \\
\hline contrast & contrasts and ANOVA-style joint tests of estimates \\
estat summarize & summary statistics for the estimation sample \\
estat vce & variance-covariance matrix of the estimators (VCE) \\
estimates & cataloging estimation results \\
etable & table of estimation results \\
forecast & dynamic forecasts and simulations \\
hausman & \begin{tabular}{l} 
Hausman's specification test
\end{tabular} \\
lincom & \begin{tabular}{l} 
point estimates, standard errors, testing, and inference for linear combinations of \\
coefficients
\end{tabular} \\
margins & \begin{tabular}{l} 
marginal means, predictive margins, marginal effects, and average marginal effects \\
marginsplot \\
graph the results from margins (profile plots, interaction plots, etc.)
\end{tabular} \\
predict & point estimates, standard errors, testing, and inference for nonlinear combinations \\
predictnl & number of events, linear predictions, residuals, etc. \\
pwcompare & \begin{tabular}{l} 
point estimates, standard errors, testing, and inference for generalized predictions \\
pairwise comparisons of estimates
\end{tabular} \\
test & Wald tests of simple and composite linear hypotheses \\
testnl & Wald tests of nonlinear hypotheses
\end{tabular}

\section*{predict}

\section*{Description for predict}
predict creates a new variable containing predictions such as numbers of events, linear predictions, and residuals.

\section*{Menu for predict}

Statistics \(>\) Postestimation

\section*{Syntax for predict}
predict [type] newvar [if] [in] [, statistic nooffset ]
statistic Description
Main
n
xbtotal
xb
residuals
number of events; the default
linear prediction, using residual estimates for ivpoisson cfunction linear prediction residuals

These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

\section*{Options for predict}
\(\qquad\) Main
n , the default, calculates the predicted number of events via the exponential-form estimate. This is \(\exp \left(\mathbf{x}_{j}^{\prime} \boldsymbol{\beta}_{1}+\mathbf{y}_{2, j}^{\prime} \boldsymbol{\beta}_{2}\right)\) if neither offset () nor exposure() was specified, \(\exp \left(\mathbf{x}_{j}^{\prime} \boldsymbol{\beta}_{1}+\mathbf{y}_{2, j}^{\prime} \boldsymbol{\beta}_{2}+\right.\) \(\left.\operatorname{offset}_{j}\right)\) if offset() was specified, or \(\exp \left(\mathbf{x}_{j}^{\prime} \boldsymbol{\beta}_{1}+\mathbf{y}_{2, j}^{\prime} \boldsymbol{\beta}_{2}\right) \times\) exposure \(_{j}\) if exposure() was specified.
After generalized method of moments estimation, the exponential-form estimate is not a consistent estimate of the conditional mean of \(y_{j}\), because it is not corrected for \(E\left(\epsilon_{j} \mid \mathbf{y}_{2, j}\right)\). More details are found in Methods and formulas.

After control-function estimation, we correct the exponential-form estimate for \(E\left(\epsilon_{j} \mid \mathbf{y}_{2, j}\right)\) by using the estimated residuals of \(\mathbf{y}_{2, j}\) and the c_* auxiliary parameters. This supplements the direct effect of \(\mathbf{y}_{2, j}\) and \(\mathbf{x}_{j}\) through \(\boldsymbol{\beta}_{1}\) and \(\boldsymbol{\beta}_{2}\) with the indirect effects of \(\mathbf{y}_{2, j}, \mathbf{x}_{j}\), and the instruments \(\mathbf{z}_{j}\) through the endogenous error \(\epsilon_{j}\). Thus, the exponential-form estimate consistently estimates the conditional mean of \(y_{j}\).
xbtotal calculates the linear prediction, which is \(\mathbf{x}_{j}^{\prime} \boldsymbol{\beta}_{1}+\mathbf{y}_{2, j}^{\prime} \boldsymbol{\beta}_{2}\) if neither offset () nor exposure () was specified, \(\mathbf{x}_{j}^{\prime} \boldsymbol{\beta}_{1}+\mathbf{y}_{2, j}^{\prime} \boldsymbol{\beta}_{2}+\) offset \(_{j}\) if offset () was specified, or \(\mathbf{x}_{j}^{\prime} \boldsymbol{\beta}_{1}+\mathbf{y}_{2, j}^{\prime} \boldsymbol{\beta}_{2}+\ln \left(\right.\) exposure \(\left._{j}\right)\) if exposure() was specified.
After control-function estimation, the estimate of the linear form \(\mathbf{x}_{j}^{\prime} \boldsymbol{\beta}_{1}\) includes the estimated residuals of the endogenous regressors with coefficients from the \(c_{-} *\) auxiliary parameters.
xb calculates the linear prediction, which is \(\mathbf{x}_{j}^{\prime} \boldsymbol{\beta}_{1}+\mathbf{y}_{2, j}^{\prime} \boldsymbol{\beta}_{2}\) if neither offset () nor exposure () was specified, \(\mathbf{x}_{j}^{\prime} \boldsymbol{\beta}_{1}+\mathbf{y}_{2, j}^{\prime} \boldsymbol{\beta}_{2}+\) offset \(_{j}\) if offset () was specified, or \(\mathbf{x}_{j}^{\prime} \boldsymbol{\beta}_{1}+\mathbf{y}_{2, j}^{\prime} \boldsymbol{\beta}_{2}+\ln \left(\right.\) exposure \(\left._{j}\right)\) if exposure() was specified. See nooffset below.
residuals calculates the residuals. Under additive errors, these are calculated as \(y_{j}-\exp \left(\mathbf{x}_{j}^{\prime} \boldsymbol{\beta}_{1}+\right.\) \(\left.\mathbf{y}_{2, j}^{\prime} \boldsymbol{\beta}_{2}\right)\). Under multiplicative errors, they are calculated as \(y_{j} / \exp \left(\mathbf{x}_{j}^{\prime} \boldsymbol{\beta}_{1}+\mathbf{y}_{2, j}^{\prime} \boldsymbol{\beta}_{2}\right)-1\).
When offset () or exposure() is specified, \(\mathbf{x}_{j}^{\prime} \boldsymbol{\beta}_{1}\) is not used directly in the residuals. \(\mathbf{x}_{j}^{\prime} \boldsymbol{\beta}_{1}+\) offset \(_{j}\) is used if offset() was specified. \(\mathbf{x}_{j}^{\prime} \boldsymbol{\beta}_{1}+\ln\) (exposure \(_{j}\) ) is used if exposure() was specified. See nooffset below.
After control-function estimation, the estimate of the linear form \(\mathbf{x}_{j}^{\prime} \boldsymbol{\beta}_{1}\) includes the estimated residuals of the endogenous regressors with coefficients from the \(c_{-} *\) auxiliary parameters.
nooffset is relevant only if you specified offset() or exposure() when you fit the model. It modifies the calculations made by predict so that they ignore the offset or exposure variable. nooffset removes the offset from calculations involving both the treat() equation and the dependent count variable.

\section*{margins}

\section*{Description for margins}
margins estimates margins of response for numbers of events and linear predictions.

\section*{Menu for margins}

Statistics \(>\) Postestimation

\section*{Syntax for margins}
```

margins [marginlist] [, options]

```

statistic Description
n
xbtotal
xb
residuals
number of events; the default
linear prediction, using residual estimates for ivpoisson cfunction linear prediction not allowed with margins

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

\section*{estat}

\section*{Description for estat}
estat overid reports Hansen's \(J\) statistic, which is used to determine the validity of the overidentifying restrictions in a GMM model. ivpoisson gmm uses GMM estimation to obtain parameter estimates. Under additive and multiplicative errors, Hansen's \(J\) statistic can be accurately reported when more instruments than endogenous regressors are specified. It is not appropriate to report the \(J\) statistic after ivpoisson cfunction, because a just-identified model is fit.

If the model is correctly specified in the sense that \(E\left\{\widetilde{\mathbf{z}}_{i} u\left(y_{i}, \mathbf{x}_{i}, \mathbf{y}_{2, i}, \boldsymbol{\beta}\right)\right\}=0\), then the sample analog to that condition should hold at the estimated value of \(\boldsymbol{\beta}_{1}\) and \(\boldsymbol{\beta}_{2}\). The \(\widetilde{\mathbf{z}}_{i}\) variables are the exogenous regressors \(\mathbf{x}_{i}\) and instrumental variables \(\mathbf{z}_{i}\) used in ivpoisson gmm. The \(\mathbf{y}_{2, i}\) are the endogenous regressors. The \(u\) function is the error function, which will have a different form for multiplicative and additive errors in the regression.

Hansen's \(J\) statistic is valid only if the weight matrix is optimal, meaning that it equals the inverse of the covariance matrix of the moment conditions. Therefore, estat overid only reports Hansen's \(J\) statistic after two-step or iterated estimation or if you specified winitial (matname) when calling ivpoisson gmm. In the latter case, it is your responsibility to determine the validity of the \(J\) statistic.

\section*{Menu for estat}

Statistics \(>\) Postestimation

\section*{Syntax for estat}
```

estat overid

```

\section*{Remarks and examples}
estat overid reports Hansen's \(J\) statistic, which is used to determine the validity of the overidentifying restrictions in a GMM model. It is not appropriate to use it after ivpoisson cfunction, because a just-identified model is fit.

Recall that the GMM criterion function is
\[
\begin{equation*}
Q(\boldsymbol{\beta})=\left\{\frac{1}{N} \sum_{i} \widetilde{\mathbf{z}}_{i} u\left(y_{i}, \mathbf{x}_{i}, \mathbf{y}_{2, i}, \boldsymbol{\beta}_{1}, \boldsymbol{\beta}_{2}\right)\right\}^{\prime} \mathbf{W}_{N}\left\{\frac{1}{N} \sum_{i} \widetilde{\mathbf{z}}_{i} u\left(y_{i}, \mathbf{x}_{i}, \mathbf{y}_{2, i}, \boldsymbol{\beta}_{1}, \boldsymbol{\beta}_{2}\right)\right\} \tag{A1}
\end{equation*}
\]

Our \(u\) function within this formula will change depending on whether we use additive or multiplicative errors. The \(\widetilde{\mathbf{z}}\) vector contains the exogenous regressors and instrumental variables used. ivpoisson gmm estimates regression coefficients to minimize \(Q\).

Let \(l\) be the dimension of \(\widetilde{\mathbf{z}}\) and \(k\) the number of regressors. If \(\mathbf{W}_{N}\) is an optimal weight matrix, under the null hypothesis \(H_{0}: E\left\{\widetilde{\mathbf{z}}_{i} u\left(y_{i}, \mathbf{x}_{i}, \mathbf{y}_{2, i}, \boldsymbol{\beta}_{1}, \boldsymbol{\beta}_{2}\right)\right\}=\mathbf{0}\), the test statistic \(J=N \times Q \sim \chi^{2}(l-k)\). A large test statistic casts doubt on the null hypothesis.

Because the weight matrix \(\mathbf{W}_{N}\) must be optimal, estat overid works only after the two-step and iterated estimation or if you supplied your own initial weight matrix by using the winitial (matname) option of ivpoisson gmm and used the one-step estimator.

Often, the overidentifying restrictions test is interpreted as a test of the validity of the instruments z. However, other forms of model misspecification can sometimes lead to a significant test statistic. See Hall (2005, sec. 5.1) for a discussion of the overidentifying restrictions test and its behavior in correctly specified and misspecified models.

Note that ivpoisson gmm defaults to the two-step estimator when other options are not specified to override the default. Thus, it is appropriate to perform the \(J\) test after the regression of example 1 in \([R]\) ivpoisson.
\(>\) Example 1: Specification test
Recall example 1 of \([\mathrm{R}]\) ivpoisson. We estimated the parameters of an exponential conditional mean model for the number of visits to a website. Additive errors were used. Exogenous regressors included the gender of an individual and the number of ads received from the website.

An endogenous regressor, time spent on the Internet, was also included in the model. Two instruments were used. One of the instruments measured the time spent interacting with friends and out-of-town family. The other measured the time spent on the phone.

We will reestimate the parameters of the regression here and then test the specification.
```

. use https://www.stata-press.com/data/r18/website
(Visits to website)
. ivpoisson gmm visits ad female (time = phone frfam)
(output omitted)
estat overid
Test of overidentifying restriction:
Hansen's J chi2(1) = .129055 ( }\textrm{p}=0.7194\mathrm{ )

```

We have two instruments for one endogenous variable, so the \(J\) statistic has one degree of freedom. The \(J\) statistic is not significant. We fail to reject the null hypothesis that the model is correctly specified.

\section*{Stored results}
estat overid stores the following in r() :
Scalars
\[
\begin{array}{ll}
\mathrm{r}(\mathrm{~J}) & \text { Hansen's } J \text { statistic } \\
\mathrm{r}\left(\mathrm{~J} \_ \text {df }\right) & J \text { statistic degrees of freedom } \\
\mathrm{r}\left(\mathrm{~J} \_\mathrm{p}\right) & J \text { statistic } p \text {-value }
\end{array}
\]

\section*{Methods and formulas}

The vector \(\mathbf{x}_{i}\) contains the exogenous regressors, and \(\mathbf{z}_{i}\) the instruments. The vector \(\widetilde{\mathbf{z}}_{i}\) is partitioned as \(\left(\mathbf{x}_{i}, \mathbf{z}_{i}\right)\). The vector \(\mathbf{y}_{2, i}\) contains the endogenous regressors.

Under multiplicative errors, the conditional mean of \(y_{i}\) is
\[
\begin{aligned}
E\left(y_{i} \mid \mathbf{y}_{2, i}, \widetilde{\mathbf{z}}_{i}\right) & =E\left\{E\left(y_{i} \mid \mathbf{x}_{i}, \mathbf{y}_{2, i}, \epsilon_{i}\right) \mid \mathbf{y}_{2, i}, \widetilde{\mathbf{z}}_{i}\right\} \\
& =E\left\{\exp \left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}_{1}+\mathbf{y}_{2, i}^{\prime} \boldsymbol{\beta}_{2}\right) \epsilon_{i} \mid \mathbf{y}_{2, i}, \widetilde{\mathbf{z}}_{i}\right\} \\
& =\exp \left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}_{1}+\mathbf{y}_{2, i}^{\prime} \boldsymbol{\beta}_{2}\right) E\left(\epsilon_{i} \mid \mathbf{y}_{2, i}, \widetilde{\mathbf{z}}_{i}\right)
\end{aligned}
\]

Under the CF estimator,
\[
\begin{aligned}
E\left(\epsilon_{i} \mid \mathbf{y}_{2, i}, \widetilde{\mathbf{z}}_{i}\right) & =E\left\{E\left(\epsilon_{i} \mid \mathbf{v}_{i}, c_{i}\right) \mid \mathbf{y}_{2, i}, \widetilde{\mathbf{z}}_{i}\right\} \\
& =E\left\{\exp \left(\mathbf{v}_{i}^{\prime} \boldsymbol{\rho}+c_{i}\right) \mid \mathbf{y}_{2, i}, \widetilde{\mathbf{z}}_{i}\right\} \\
& =\exp \left\{\left(\mathbf{y}_{2, i}-\mathbf{B} \widetilde{\mathbf{z}}_{i}^{\prime}\right)^{\prime} \boldsymbol{\rho}\right\} E\left(c_{i} \mid \mathbf{y}_{2, i}, \widetilde{\mathbf{z}}_{i}\right) \\
& =\exp \left\{\left(\mathbf{y}_{2, i}-\mathbf{B} \widetilde{\mathbf{z}}_{i}^{\prime}\right)^{\prime} \boldsymbol{\rho}\right\}
\end{aligned}
\]

Thus under the CF estimator, we estimate the conditional mean of \(y_{i}\) as
\[
E\left(y_{i} \mid \mathbf{y}_{2, i}, \widetilde{\mathbf{z}}_{i}\right)=\exp \left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}_{1}+\mathbf{y}_{2, i}^{\prime} \boldsymbol{\beta}_{2}+\left(\mathbf{y}_{2, i}-\mathbf{B} \widetilde{\mathbf{z}}_{i}^{\prime}\right)^{\prime} \boldsymbol{\rho}\right)
\]

The CF estimator explicitly models the functional form of the endogeneity of \(\mathbf{y}_{2, i}\) and \(\epsilon_{i}\) with the instruments and exogenous regressors \(\widetilde{\mathbf{z}}_{i}\). This allows it to correct the exponential-form estimator for the \(E\left(\epsilon_{i} \mid \mathbf{y}_{2, i}, \widetilde{\mathbf{z}}_{i}\right)\) term.

In contrast, the GMM estimator does not model the functional form of the endogeneity of \(\mathbf{y}_{2, i}\) and \(\epsilon_{i}\). Therefore, \(E\left(\epsilon_{i} \mid \mathbf{y}_{2, i}, \widetilde{\mathbf{z}}_{i}\right)\) is not estimated, and the exponential-form estimator under GMM estimation simply ignores this term. Noting that because \(\widetilde{\mathbf{z}}_{i}\) and \(\epsilon_{i}\) are independent, \(E\left(\epsilon_{i} \mid \mathbf{y}_{2, i}, \widetilde{\mathbf{z}}_{i}\right)=E\left(\epsilon_{i} \mid \mathbf{y}_{2, i}\right)\), we can obviously see that ignoring the term will lead to inconsistent estimation of the conditional mean of \(y_{i} . y_{2, i}\) and \(\epsilon_{i}\) are not independent, so \(E\left(\epsilon_{i} \mid y_{2, i}\right)\) may vary based on \(y_{2, i}\).

In the additive errors setting, a similar derivation will show that the exponential-form estimator obtained from GMM estimation is inconsistent for the conditional mean of \(y_{i}\).

\section*{Reference}

Hall, A. R. 2005. Generalized Method of Moments. Oxford: Oxford University Press.

\section*{Also see}
[R] ivpoisson - Poisson model with continuous endogenous covariates
[U] 20 Estimation and postestimation commands

\section*{Title}
ivprobit - Probit model with continuous endogenous covariates

Description
Syntax
Remarks and examples
Acknowledgments

Quick start
Options for ML estimator
Stored results
References

\section*{Menu}

Options for two-step estimator Methods and formulas
Also see

\section*{Description}
ivprobit fits models for binary dependent variables where one or more of the covariates are endogenous and errors are normally distributed. By default, ivprobit uses maximum likelihood, but Newey's (1987) minimum \(\chi^{2}\) (two-step) estimator can be requested. Both estimators assume that the endogenous covariates are continuous and so are not appropriate for use with discrete endogenous covariates.

\section*{Quick start}

Probit regression of y 1 on x and endogenous regressor y 2 that is instrumented using z
ivprobit \(\mathrm{y} 1 \mathrm{x}(\mathrm{y} 2=\mathrm{z})\)
With robust standard errors
ivprobit \(y 1 \mathrm{x}(\mathrm{y} 2=\mathrm{z})\), vce(robust)
Use Newey's two-step estimator
ivprobit y1 x (y2 = z), twostep
Same as above, and show first-stage regression results
ivprobit \(y 1 \mathrm{x}(\mathrm{y} 2=\mathrm{z})\), twostep first

\section*{Menu}

Statistics \(>\) Endogenous covariates \(>\) Probit model with endogenous covariates

\section*{Syntax}

Maximum likelihood estimator
ivprobit depvar \(\left[\right.\) varlist \(\left._{1}\right]\) ( varlist \(_{2}=\) varlist \(_{\mathrm{iv}}\) ) [if] [in] [weight] [, mle_options]

Two-step estimator
ivprobit depvar \(\left[\right.\) varlist \(\left._{1}\right]\) (varlist \({ }_{2}=\) varlist \(_{\mathrm{iv}}\) ) [if] [in] [weight], twostep [tse_options]
varlist \(_{1}\) is the list of exogenous variables.
varlist \(_{2}\) is the list of endogenous variables.
\(v^{v a r l i s t} t_{\mathrm{iv}}\) is the list of exogenous variables used with varlist \(_{1}\) as instruments for varlist \({ }_{2}\).
mle_options
Model
mle
asis
constraints (constraints)
SE/Robust
vce (vcetype)

Reporting
level(\#)
first
nocnsreport
display_options

Description
use conditional maximum-likelihood estimator; the default retain perfect predictor variables
apply specified linear constraints
vcetype may be oim, robust, cluster clustvar, opg, bootstrap, or jackknife
set confidence level; default is level (95)
report first-stage regression
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

Maximization
maximize_options
coeflegend
control the maximization process
display legend instead of statistics

\section*{Description}

\section*{Model}
* twostep
asis
SE
vce (vcetype)
Reporting
level (\#)
first
display_options
coeflegend
use Newey's two-step estimator; the default is mle
retain perfect predictor variables
vcetype may be twostep, bootstrap, or jackknife
set confidence level; default is level (95)
report first-stage regression
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
display legend instead of statistics
*twostep is required.
varlist \(_{1}\) and varlist \(_{\mathrm{iv}}\) may contain factor variables; see [U] 11.4.3 Factor variables.
depvar, varlist \({ }_{1}\), varlist \(t_{2}\), and varlist \({ }_{\mathrm{iv}}\) may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bootstrap, by, collect, jackknife, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands. fp is allowed with the maximum likelihood estimator.
Weights are not allowed with the bootstrap prefix; see [R] bootstrap.
vce(), first, twostep, and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed with the maximum likelihood estimator. fweights are allowed with Newey's two-step estimator. See [U] 11.1.6 weight.
coeflegend does not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options for ML estimator}
\(\qquad\) Model
mle requests that the conditional maximum-likelihood estimator be used. This is the default.
asis requests that all specified variables and observations be retained in the maximization process. This option is typically not used and may introduce numerical instability. Normally, ivprobit omits any endogenous or exogenous variables that perfectly predict success or failure in the dependent variable. The associated observations are also excluded. For more information, see Model identification in [R] probit.
constraints (constraints); see [R] Estimation options.

SE/Robust
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.

Reporting
level(\#); see [R] Estimation options.
first requests that the parameters for the reduced-form equations showing the relationships between the endogenous variables and instruments be displayed. For the two-step estimator, first shows the first-stage regressions. For the maximum likelihood estimator, these parameters are estimated jointly with the parameters of the probit equation. The default is not to show these parameter estimates.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\% fint), pformat (\% fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate(\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from(init_specs); see [R] Maximize.

The following option is available with ivprobit but is not shown in the dialog box: coeflegend; see [R] Estimation options.

\section*{Options for two-step estimator}
\(\qquad\) Model
twostep is required and requests that Newey's (1987) efficient two-step estimator be used to obtain the coefficient estimates.
asis requests that all specified variables and observations be retained in the maximization process. This option is typically not used and may introduce numerical instability. Normally, ivprobit omits any endogenous or exogenous variables that perfectly predict success or failure in the dependent variable. The associated observations are also excluded. For more information, see Model identification in [R] probit.

SE
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (twostep) and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
\(\qquad\) Reporting
level(\#); see [R] Estimation options.
first requests that the parameters for the reduced-form equations showing the relationships between the endogenous variables and instruments be displayed. For the two-step estimator, first shows the first-stage regressions. For the maximum likelihood estimator, these parameters are estimated jointly with the parameters of the probit equation. The default is not to show these parameter estimates.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.

The following option is available with ivprobit but is not shown in the dialog box:
coeflegend; see [R] Estimation options.

\section*{Remarks and examples}

Remarks are presented under the following headings:

\author{
Model setup \\ Model identification
}

\section*{Model setup}
ivprobit fits models with dichotomous dependent variables and endogenous covariates. You can use it to fit a probit model when you suspect that one or more of the covariates are correlated with the error term. ivprobit is to probit modeling what ivregress is to linear regression analysis; see [ R ] ivregress for more information.

Formally, the model is
\[
\begin{aligned}
& y_{1 i}^{*}=\boldsymbol{y}_{2 i} \boldsymbol{\beta}+\boldsymbol{x}_{1 i} \boldsymbol{\gamma}+u_{i} \\
& \boldsymbol{y}_{2 i}=\boldsymbol{x}_{1 i} \boldsymbol{\Pi}_{1}+\boldsymbol{x}_{2 i} \boldsymbol{\Pi}_{2}+\boldsymbol{v}_{i}
\end{aligned}
\]
where \(i=1, \ldots, N, \boldsymbol{y}_{2 i}\) is a \(1 \times p\) vector of endogenous variables, \(\boldsymbol{x}_{1 i}\) is a \(1 \times k_{1}\) vector of exogenous variables, \(\boldsymbol{x}_{2 i}\) is a \(1 \times k_{2}\) vector of additional instruments, and the equation for \(\boldsymbol{y}_{2 i}\) is written in reduced form. By assumption, \(\left(u_{i}, \boldsymbol{v}_{i}\right) \sim N(\mathbf{0}, \boldsymbol{\Sigma})\), where \(\sigma_{11}\) is normalized to one to identify the model. \(\boldsymbol{\beta}\) and \(\gamma\) are vectors of structural parameters, and \(\Pi_{1}\) and \(\Pi_{2}\) are matrices of reduced-form parameters. This is a recursive model: \(\mathbf{y}_{2 i}\) appears in the equation for \(y_{1 i}^{*}\), but \(y_{1 i}^{*}\) does not appear in the equation for \(\mathbf{y}_{2 i}\). We do not observe \(y_{1 i}^{*}\); instead, we observe
\[
y_{1 i}= \begin{cases}0 & y_{1 i}^{*}<0 \\ 1 & y_{1 i}^{*} \geq 0\end{cases}
\]

The order condition for identification of the structural parameters requires that \(k_{2} \geq p\). Presumably, \(\boldsymbol{\Sigma}\) is not block diagonal between \(u_{i}\) and \(\boldsymbol{v}_{i}\); otherwise, \(\boldsymbol{y}_{2 i}\) would not be endogenous.
- Technical note

This model is derived under the assumption that \(\left(u_{i}, \boldsymbol{v}_{i}\right)\) is independent and identically distributed multivariate normal for all \(i\). The vce (cluster clustvar) option can be used to control for a lack of independence. As with most probit models, if \(u_{i}\) is heteroskedastic, point estimates will be inconsistent.

\section*{> Example 1}

We have hypothetical data on 500 two-parent households, and we wish to model whether the woman is employed. We have a variable, fem_work, that is equal to 1 if she has a job and 0 otherwise. Her decision to work is a function of the number of children at home (kids), number of years of schooling completed (fem_educ), and other household income measured in thousands of dollars (other_inc). We suspect that unobservable shocks affecting the woman's decision to hold a job also affect the household's other income. Therefore, we treat other_inc as endogenous. As an instrument, we use the number of years of schooling completed by the man (male_educ).

The syntax for specifying the exogenous, endogenous, and instrumental variables is identical to that used in ivregress; see [R] ivregress for details.
```

. use https://www.stata-press.com/data/r18/laborsup
. ivprobit fem_work fem_educ kids (other_inc = male_educ)
Fitting exogenous probit model:
Iteration 0: Log likelihood = -344.63508
Iteration 1: Log likelihood = -252.10819
Iteration 2: Log likelihood = -252.04529
Iteration 3: Log likelihood = -252.04529
Fitting full model:
Iteration 0: Log likelihood = -2368.2142
Iteration 1: Log likelihood = -2368.2062
Iteration 2: Log likelihood = -2368.2062
Probit model with endogenous regressors Number of obs = 500
Log likelihood = -2368.2062

```
Wald chi2(3) \(=163.88\)

Prob > chi2 \(=0.0000\)
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline & Coefficient & Std. err. & z & \(\mathrm{P}>|z|\) & [95\% con & interval] \\
\hline other_inc & -. 0542756 & . 0060854 & -8.92 & 0.000 & -. 0662028 & -. 0423485 \\
\hline fem_educ & . 211111 & . 0268648 & 7.86 & 0.000 & . 1584569 & . 2637651 \\
\hline kids & -. 1820929 & . 0478267 & -3.81 & 0.000 & -. 2758315 & -. 0883542 \\
\hline _cons & . 3672086 & . 4480724 & 0.82 & 0.412 & -. 5109971 & 1.245414 \\
\hline \[
\begin{aligned}
& \text { corr (e.othe~c }, \\
& \text { e.fem_work) }
\end{aligned}
\] & . 3720375 & . 1300518 & & & . 0946562 & . 5958136 \\
\hline sd(e.other_~c) & 16.66621 & . 5270318 & & & 15.66461 & 17.73186 \\
\hline Wald test of ex Endogenous: oth Exogenous: fem & geneity (cor & \(=0):\) ch
e_educ & (1) \(=\) & & \multicolumn{2}{|l|}{Prob > chi2 \(=0.0096\)} \\
\hline
\end{tabular}
ivprobit used the default maximum likelihood estimator. The header of the output contains the sample size as well as a Wald statistic and \(p\)-value for the test of the hypothesis that all the slope coefficients are jointly zero. Below, the table of coefficients, Stata reminds us that the endogenous variable is other_inc and that fem_educ, kids, and male_educ were used as instruments.

At the bottom of the output is a Wald test of the exogeneity of the endogenous variables. We reject the null hypothesis of no endogeneity. If there is no endogeneity, a standard probit regression would be preferable (see [R] probit).

Below we fit our model with Newey's (1987) minimum \(\chi^{2}\) estimator using the twostep option.

\section*{> Example 2}

Refitting our labor-supply model with the two-step estimator yields


All the coefficients have the same signs as their counterparts in the maximum likelihood model. The Wald test at the bottom of the output confirms our earlier finding of endogeneity.

\section*{- Technical note}

In a standard probit model, the error is assumed to have a variance of 1 . In the probit model with endogenous covariates, we assume that \(\left(u_{i}, \boldsymbol{v}_{i}\right)\) is multivariate normal with covariance matrix
\[
\operatorname{Var}\left(u_{i}, \boldsymbol{v}_{i}\right)=\boldsymbol{\Sigma}=\left[\begin{array}{cc}
1 & \boldsymbol{\Sigma}_{21}^{\prime} \\
\boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22}
\end{array}\right]
\]

From the properties of the multivariate normal distribution, it follows that \(\operatorname{Var}\left(u_{i} \mid \boldsymbol{v}_{i}\right)=1-\boldsymbol{\Sigma}_{21}^{\prime} \boldsymbol{\Sigma}_{22}^{-1} \boldsymbol{\Sigma}_{21}\). Newey's estimator and other two-step probit estimators yield estimates of \(\boldsymbol{\beta} / \sigma\) and \(\gamma / \sigma\), where \(\sigma\) is the square root of \(\operatorname{Var}\left(u_{i} \mid \boldsymbol{v}_{i}\right)\), instead of estimates of \(\boldsymbol{\beta}\) and \(\boldsymbol{\gamma}\). Hence, we cannot directly compare the estimates obtained from Newey's estimator with those obtained from maximum likelihood, which estimate \(\boldsymbol{\beta}, \gamma\), and \(\sigma\) separately. See Wooldridge (2010, 585-594) for a discussion about the interpretation of the estimates and the computation of marginal effects of two-step probit estimators under endogeneity.

\section*{Model identification}

As in the linear simultaneous-equation model, the order condition for identification requires that the number of excluded exogenous variables (that is, the additional instruments) be at least as great as the number of included endogenous variables. ivprobit checks this for you and issues an error message if the order condition is not met.

Like probit, logit, and logistic, ivprobit checks the exogenous and endogenous variables to see if any of them predict the outcome variable perfectly. It will then omit offending variables and observations and fit the model on the remaining data. Instruments that are perfect predictors do not affect estimation, so they are not checked. See Model identification in [R] probit for more information.
ivprobit will also occasionally display messages such as
```

Note: 4 failures and O successes completely determined.

```

For an explanation of this message, see [R] logit.

\section*{Stored results}
ivprobit, mle stores the following in e():
Scalars
e(N)
e(N_cds)
e(N_cdf)
e(k)
e(k_eq)
e(k_eq_model)
e(k_dv)
e(df_m)
e(11)
e(N_clust)
e(endog_ct)
\(e(p)\)
e(p_exog)
e(chi2)
e(chi2_exog)
\(e\) (rank)
e(ic)
e(rc)
e(converged)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(endog)
e(exog)
\(e\) (wtype)
e(wexp)
e(title)
e(clustvar)
e(chi2type)
e(vce)
e(vcetype)
e(asis)
\(e\) (method)
e(opt)
e(which)
number of observations
number of completely determined successes
number of completely determined failures
number of parameters
number of equations in \(e(b)\)
number of equations in overall model test
number of dependent variables
model degrees of freedom
log likelihood
number of clusters
number of endogenous covariates
model Wald \(p\)-value
exogeneity test Wald \(p\)-value
model Wald \(\chi^{2}\)
Wald \(\chi^{2}\) test of exogeneity
rank of \(e(V)\)
number of iterations
return code
1 if converged, 0 otherwise
ivprobit
command as typed
name of dependent variable
names of endogenous variables
names of exogenous variables
weight type
weight expression
title in estimation output
name of cluster variable
Wald; type of model \(\chi^{2}\) test
vcetype specified in vce()
title used to label Std. err.
asis, if specified
ml
type of optimization
max or min; whether optimizer is to perform maximization or minimization
```

e(ml_method)
e(user)
e(technique)
e(properties)
e(estat_cmd)
e(predict)
e(footnote)
e(marginsok)
e(marginsprop)
e(asbalanced)
e(asobserved)

```
Matrices
    e(b)
    e(Cns)
    e(rules)
    e(ilog)
    e(gradient)
    e(Sigma)
    e(V)
    e(V_modelbased)
Functions
    e(sample)
type of ml method name of likelihood-evaluator program maximization technique
b V
program used to implement estat
program used to implement predict
program used to implement the footnote display
predictions allowed by margins
signals to the margins command
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
constraints matrix
information about perfect predictors
iteration \(\log\) (up to 20 iterations)
gradient vector
\(\widehat{\Sigma}\)
variance-covariance matrix of the estimators
model-based variance
marks estimation sample

In addition to the above, the following is stored in r() :
```

Matrices
r(table)

```
matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.
ivprobit, twostep stores the following in e():
Scalars
```

    e(N)
    e(N_cds)
    e(N_cdf)
    e(df_m)
    e(df_exog)
    e(p)
    e(p_exog)
    e(chi2)
    e(chi2_exog)
    e(rank)
    ```
Macros
    e(cmd)
    e(cmdline)
    e(depvar)
    e(endog)
    e (exog)
    e(wtype)
    e(wexp)
    e(chi2type)
    e(vce)
    e(asis)
    e(method)
    e(properties)
    e(estat_cmd)
    e(predict)
number of observations
number of completely determined successes
number of completely determined failures
model degrees of freedom
degrees of freedom for \(\chi^{2}\) test of exogeneity
model Wald \(p\)-value
exogeneity test Wald \(p\)-value
model Wald \(\chi^{2}\)
Wald \(\chi^{2}\) test of exogeneity
rank of \(e(V)\)
    ivprobit
    command as typed
    name of dependent variable
    names of endogenous variables
    names of exogenous variables
    weight type
    weight expression
    Wald; type of model \(\chi^{2}\) test
    vcetype specified in vce()
    asis, if specified
    twostep
b V
program used to implement estat
program used to implement predict
```

e(footnote)
e(marginsok)
e(marginsprop)
e(asbalanced)
e(asobserved)

```

Matrices
e(b)
e(rules)
e(V)
Functions
e(sample)
program used to implement the footnote display predictions allowed by margins signals to the margins command factor variables fvset as asbalanced factor variables fvset as asobserved
coefficient vector information about perfect predictors variance-covariance matrix of the estimators
marks estimation sample

In addition to the above, the following is stored in r() :

Matrices
\(r(t a b l e) \quad\) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

\section*{Methods and formulas}

Fitting limited-dependent variable models with endogenous covariates has received considerable attention in the econometrics literature. Building on the results of Amemiya (1978, 1979), Newey (1987) developed an efficient method of estimation that encompasses both Rivers and Vuong's (1988) simultaneous-equations probit model and Smith and Blundell's (1986) simultaneous-equations tobit model. An efficient alternative to two-step estimation, and ivprobit's default, is to use maximum likelihood. For compactness, we write the model as
\[
\begin{align*}
y_{1 i}^{*} & =\boldsymbol{z}_{i} \boldsymbol{\delta}+u_{i}  \tag{1a}\\
\boldsymbol{y}_{2 i} & =\boldsymbol{x}_{i} \boldsymbol{\Pi}+\boldsymbol{v}_{i} \tag{1b}
\end{align*}
\]
where \(\boldsymbol{z}_{i}=\left(\boldsymbol{y}_{2 i}, \boldsymbol{x}_{1 i}\right), \boldsymbol{x}_{i}=\left(\boldsymbol{x}_{1 i}, \boldsymbol{x}_{2 i}\right), \boldsymbol{\delta}=\left(\boldsymbol{\beta}^{\prime}, \boldsymbol{\gamma}^{\prime}\right)^{\prime}\), and \(\boldsymbol{\Pi}=\left(\boldsymbol{\Pi}_{1}^{\prime}, \boldsymbol{\Pi}_{2}^{\prime}\right)^{\prime}\).
Deriving the likelihood function is straightforward because we can write the joint density \(f\left(y_{1 i}, \boldsymbol{y}_{2 i} \mid \boldsymbol{x}_{i}\right)\) as \(f\left(y_{1 i} \mid \boldsymbol{y}_{2 i}, \boldsymbol{x}_{i}\right) f\left(\boldsymbol{y}_{2 i} \mid \boldsymbol{x}_{i}\right)\). When there is an endogenous regressor, the log likelihood for observation \(i\) is
\[
\ln L_{i}=w_{i}\left[y_{1 i} \ln \Phi\left(m_{i}\right)+\left(1-y_{1 i}\right) \ln \left\{1-\Phi\left(m_{i}\right)\right\}+\ln \phi\left(\frac{y_{2 i}-\boldsymbol{x}_{i} \boldsymbol{\Pi}}{\sigma}\right)-\ln \sigma\right]
\]
where
\[
m_{i}=\frac{\boldsymbol{z}_{i} \boldsymbol{\delta}+\rho\left(y_{2 i}-\boldsymbol{x}_{i} \boldsymbol{\Pi}\right) / \sigma}{\left(1-\rho^{2}\right)^{\frac{1}{2}}}
\]
\(\Phi(\cdot)\) and \(\phi(\cdot)\) are the standard normal distribution and density functions, respectively; \(\sigma\) is the standard deviation of \(v_{i} ; \rho\) is the correlation coefficient between \(u_{i}\) and \(v_{i}\); and \(w_{i}\) is the weight for observation \(i\) or one if no weights were specified. Instead of estimating \(\sigma\) and \(\rho\), we estimate \(\ln \sigma\) and atanh \(\rho\), where
\[
\operatorname{atanh} \rho=\frac{1}{2} \ln \left(\frac{1+\rho}{1-\rho}\right)
\]

For multiple endogenous covariates, let
\[
\operatorname{Var}\left(u_{i}, \boldsymbol{v}_{i}\right)=\boldsymbol{\Sigma}=\left[\begin{array}{cc}
1 & \boldsymbol{\Sigma}_{21}^{\prime} \\
\boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22}
\end{array}\right]
\]

As in any probit model, we have imposed the normalization \(\operatorname{Var}\left(u_{i}\right)=1\) to identify the model. The \(\log\) likelihood for observation \(i\) is
\[
\ln L_{i}=w_{i}\left[y_{1 i} \ln \Phi\left(m_{i}\right)+\left(1-y_{1 i}\right) \ln \left\{1-\Phi\left(m_{i}\right)\right\}+\ln f\left(\boldsymbol{y}_{2 i} \mid \boldsymbol{x}_{i}\right)\right]
\]
where
\[
\ln f\left(\boldsymbol{y}_{2 i} \mid \boldsymbol{x}_{i}\right)=-\frac{p}{2} \ln 2 \pi-\frac{1}{2} \ln \left|\boldsymbol{\Sigma}_{22}\right|-\frac{1}{2}\left(\boldsymbol{y}_{2 i}-\boldsymbol{x}_{i} \boldsymbol{\Pi}\right) \boldsymbol{\Sigma}_{22}^{-1}\left(\boldsymbol{y}_{2 i}-\boldsymbol{x}_{i} \boldsymbol{\Pi}\right)^{\prime}
\]
and
\[
m_{i}=\left(1-\boldsymbol{\Sigma}_{21}^{\prime} \boldsymbol{\Sigma}_{22}^{-1} \boldsymbol{\Sigma}_{21}\right)^{-\frac{1}{2}}\left\{\boldsymbol{z}_{i} \boldsymbol{\delta}+\left(\boldsymbol{y}_{2 i}-\boldsymbol{x}_{i} \boldsymbol{\Pi}\right) \boldsymbol{\Sigma}_{22}^{-1} \boldsymbol{\Sigma}_{21}\right\}
\]

With maximum likelihood estimation, this command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce (robust) and vce (cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.

The maximum likelihood version of ivprobit also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

The two-step estimates are obtained using Newey's (1987) minimum \(\chi^{2}\) estimator. The reducedform equation for \(y_{1 i}^{*}\) is
\[
\begin{aligned}
y_{1 i}^{*} & =\left(\boldsymbol{x}_{i} \boldsymbol{\Pi}+\boldsymbol{v}_{i}\right) \boldsymbol{\beta}+\boldsymbol{x}_{1 i} \boldsymbol{\gamma}+u_{i} \\
& =\boldsymbol{x}_{i} \boldsymbol{\alpha}+\boldsymbol{v}_{i} \boldsymbol{\beta}+u_{i} \\
& =\boldsymbol{x}_{i} \boldsymbol{\alpha}+\nu_{i}
\end{aligned}
\]
where \(\nu_{i}=\boldsymbol{v}_{i} \boldsymbol{\beta}+u_{i}\). Because \(u_{i}\) and \(\boldsymbol{v}_{i}\) are jointly normal, \(\nu_{i}\) is also normal. Note that
\[
\boldsymbol{\alpha}=\left[\begin{array}{l}
\boldsymbol{\Pi}_{1} \\
\boldsymbol{\Pi}_{2}
\end{array}\right] \boldsymbol{\beta}+\left[\begin{array}{l}
\boldsymbol{I} \\
\mathbf{0}
\end{array}\right] \boldsymbol{\gamma}=D(\boldsymbol{\Pi}) \boldsymbol{\delta}
\]
where \(D(\boldsymbol{\Pi})=\left(\boldsymbol{\Pi}, \boldsymbol{I}_{1}\right)\) and \(\boldsymbol{I}_{1}\) is defined such that \(\boldsymbol{x}_{i} \boldsymbol{I}_{1}=\boldsymbol{x}_{1 i}\). Letting \(\widehat{\boldsymbol{z}_{i}}=\left(\boldsymbol{x}_{i} \widehat{\boldsymbol{\Pi}}, \boldsymbol{x}_{1 i}\right), \widehat{\boldsymbol{z}_{i}} \boldsymbol{\delta}=\) \(\boldsymbol{x}_{i} D(\widehat{\boldsymbol{\Pi}}) \boldsymbol{\delta}\), where \(D(\widehat{\boldsymbol{\Pi}})=\left(\widehat{\boldsymbol{\Pi}}, \boldsymbol{I}_{1}\right)\). Thus, one estimator of \(\boldsymbol{\alpha}\) is \(D(\widehat{\boldsymbol{\Pi}}) \boldsymbol{\delta}\); denote this estimator by \(\widehat{D} \delta\).
\(\boldsymbol{\alpha}\) could also be estimated directly as the solution to
\[
\begin{equation*}
\max _{\boldsymbol{\alpha}, \boldsymbol{\lambda}} \sum_{i=1}^{N} l\left(y_{1 i}, \boldsymbol{x}_{i} \boldsymbol{\alpha}+\widehat{\boldsymbol{v}_{i}} \boldsymbol{\lambda}\right) \tag{2}
\end{equation*}
\]
where \(l(\cdot)\) is the \(\log\) likelihood for probit. Denote this estimator by \(\widetilde{\boldsymbol{\alpha}}\). The inclusion of the \(\widehat{\boldsymbol{v}}_{\boldsymbol{i}} \boldsymbol{\lambda}\) term follows because the multivariate normality of \(\left(u_{i}, \boldsymbol{v}_{i}\right)\) implies that, conditional on \(\boldsymbol{y}_{2 i}\), the expected value of \(u_{i}\) is nonzero. Because \(\boldsymbol{v}_{i}\) is unobservable, the least-squares residuals from fitting ( \(1 b\) ) are used.

Amemiya (1978) shows that the estimator of \(\boldsymbol{\delta}\) defined by
\[
\max _{\boldsymbol{\delta}}(\widetilde{\boldsymbol{\alpha}}-\widehat{\boldsymbol{D}} \boldsymbol{\delta})^{\prime} \widehat{\boldsymbol{\Omega}}^{-1}(\widetilde{\boldsymbol{\alpha}}-\widehat{\boldsymbol{D}} \boldsymbol{\delta})
\]
where \(\widehat{\boldsymbol{\Omega}}\) is a consistent estimator of the covariance of \(\sqrt{N}(\widetilde{\boldsymbol{\alpha}}-\widehat{\boldsymbol{D}} \boldsymbol{\delta})\), is asymptotically efficient relative to all other estimators that minimize the distance between \(\widetilde{\boldsymbol{\alpha}}\) and \(D(\widehat{\boldsymbol{\Pi}}) \boldsymbol{\delta}\). Thus, an efficient estimator of \(\boldsymbol{\delta}\) is
\[
\begin{equation*}
\widehat{\boldsymbol{\delta}}=\left(\widehat{\boldsymbol{D}}^{\prime} \widehat{\boldsymbol{\Omega}}^{-1} \widehat{\boldsymbol{D}}\right)^{-1} \widehat{\boldsymbol{D}}^{\prime} \widehat{\boldsymbol{\Omega}}^{-1} \widetilde{\boldsymbol{\alpha}} \tag{3}
\end{equation*}
\]
and
\[
\begin{equation*}
\operatorname{Var}(\widehat{\boldsymbol{\delta}})=\left(\widehat{\boldsymbol{D}}^{\prime} \widehat{\boldsymbol{\Omega}}^{-1} \widehat{\boldsymbol{D}}\right)^{-1} \tag{4}
\end{equation*}
\]

To implement this estimator, we need \(\widehat{\boldsymbol{\Omega}}^{-1}\).
Consider the two-step maximum likelihood estimator that results from first fitting (1b) by OLS and computing the residuals \(\widehat{\boldsymbol{v}_{i}}=\boldsymbol{y}_{2 i}-\boldsymbol{x}_{i} \widehat{\boldsymbol{\Pi}}\). The estimator is then obtained by solving
\[
\max _{\boldsymbol{\delta}, \boldsymbol{\lambda}} \sum_{i=1}^{N} l\left(y_{1 i}, \boldsymbol{z}_{i} \boldsymbol{\delta}+\widehat{\boldsymbol{v}}_{i} \boldsymbol{\lambda}\right)
\]

This is the two-step instrumental-variables (2SIV) estimator proposed by Rivers and Vuong (1988), and its role will become apparent shortly.

From Proposition 5 of Newey (1987), \(\sqrt{N}(\widetilde{\boldsymbol{\alpha}}-\widehat{\boldsymbol{D}} \boldsymbol{\delta}) \xrightarrow{d} N(\mathbf{0}, \boldsymbol{\Omega})\), where
\[
\boldsymbol{\Omega}=\boldsymbol{J}_{\alpha \alpha}^{-1}+(\boldsymbol{\lambda}-\boldsymbol{\beta})^{\prime} \boldsymbol{\Sigma}_{22}(\boldsymbol{\lambda}-\boldsymbol{\beta}) \boldsymbol{Q}^{-1}
\]
and \(\boldsymbol{\Sigma}_{22}=E\left\{\boldsymbol{v}_{i}^{\prime} \boldsymbol{v}_{i}\right\} . \boldsymbol{J}_{\alpha \alpha}^{-1}\) is simply the covariance matrix of \(\widetilde{\boldsymbol{\alpha}}\), ignoring that \(\widehat{\boldsymbol{\Pi}}\) is an estimated parameter matrix. Moreover, Newey shows that the covariance matrix from an OLS regression of \(\boldsymbol{y}_{2 i}(\widehat{\boldsymbol{\lambda}}-\widehat{\boldsymbol{\beta}})\) on \(\boldsymbol{x}_{i}\) is a consistent estimator of the second term. \(\widehat{\boldsymbol{\lambda}}\) can be obtained from solving (2), and the 2SIV estimator yields a consistent estimate, \(\widehat{\boldsymbol{\beta}}\).

Mechanically, estimation proceeds in several steps.
1. Each of the endogenous right-hand-side variables is regressed on all the exogenous variables, and the fitted values and residuals are calculated. The matrix \(\widehat{D}=D(\widehat{\boldsymbol{\Pi}})\) is assembled from the estimated coefficients.
2. probit is used to solve (2) and obtain \(\widetilde{\boldsymbol{\alpha}}\) and \(\widehat{\boldsymbol{\lambda}}\). The portion of the covariance matrix corresponding to \(\boldsymbol{\alpha}, \boldsymbol{J}_{\alpha \alpha}^{-1}\), is also saved.
3. The 2 SIV estimator is evaluated, and the parameters \(\widehat{\boldsymbol{\beta}}\) corresponding to \(\boldsymbol{y}_{2 i}\) are collected.
4. \(\boldsymbol{y}_{2 i}(\widehat{\boldsymbol{\lambda}}-\widehat{\boldsymbol{\beta}})\) is regressed on \(\boldsymbol{x}_{i}\). The covariance matrix of the parameters from this regression is added to \(\boldsymbol{J}_{\alpha \alpha}^{-1}\), yielding \(\widehat{\boldsymbol{\Omega}}\).
5. Evaluating (3) and (4) yields the estimates \(\widehat{\boldsymbol{\delta}}\) and \(\operatorname{Var}(\widehat{\boldsymbol{\delta}})\).
6. A Wald test of the null hypothesis \(H_{0}: \boldsymbol{\lambda}=\mathbf{0}\), using the 2SIV estimates, serves as our test of exogeneity.
The two-step estimates are not directly comparable with those obtained from the maximum likelihood estimator or from probit. The argument is the same for Newey's efficient estimator as for Rivers and Vuong's (1988) 2SIV estimator, so we consider the simpler 2SIV estimator. From the properties of the normal distribution,
\[
E\left(u_{i} \mid \boldsymbol{v}_{i}\right)=\boldsymbol{v}_{i} \boldsymbol{\Sigma}_{22}^{-1} \boldsymbol{\Sigma}_{21} \quad \text { and } \quad \operatorname{Var}\left(u_{i} \mid \boldsymbol{v}_{i}\right)=1-\boldsymbol{\Sigma}_{21}^{\prime} \boldsymbol{\Sigma}_{22}^{-1} \boldsymbol{\Sigma}_{21}
\]

We write \(u_{i}\) as \(u_{i}=\boldsymbol{v}_{i} \boldsymbol{\Sigma}_{22}^{-1} \boldsymbol{\Sigma}_{21}+e_{i}=\boldsymbol{v}_{i} \boldsymbol{\lambda}+e_{i}\), where \(e_{i} \sim N\left(0,1-\rho^{2}\right), \rho^{2}=\boldsymbol{\Sigma}_{21}^{\prime} \boldsymbol{\Sigma}_{22}^{-1} \boldsymbol{\Sigma}_{21}\), and \(e_{i}\) is independent of \(\boldsymbol{v}_{i}\). In the second stage of \(2 \operatorname{SIV}\), we use a probit regression to estimate the parameters of
\[
y_{1 i}=\boldsymbol{z}_{i} \boldsymbol{\delta}+\boldsymbol{v}_{i} \boldsymbol{\lambda}+e_{i}
\]

Because \(\boldsymbol{v}_{i}\) is unobservable, we use the sample residuals from the first-stage regressions.
\[
\operatorname{Pr}\left(y_{1 i}=1 \mid \boldsymbol{z}_{i}, \boldsymbol{v}_{i}\right)=\operatorname{Pr}\left(\boldsymbol{z}_{i} \boldsymbol{\delta}+\boldsymbol{v}_{i} \boldsymbol{\lambda}+e_{i}>0 \mid \boldsymbol{z}_{i}, \boldsymbol{v}_{i}\right)=\Phi\left\{\left(1-\rho^{2}\right)^{-\frac{1}{2}}\left(\boldsymbol{z}_{i} \boldsymbol{\delta}+\boldsymbol{v}_{i} \boldsymbol{\lambda}\right)\right\}
\]

Hence, as mentioned previously, 2SIV and Newey's estimator do not estimate \(\boldsymbol{\delta}\) and \(\boldsymbol{\lambda}\) but rather
\[
\boldsymbol{\delta}_{\rho}=\frac{1}{\left(1-\rho^{2}\right)^{\frac{1}{2}}} \boldsymbol{\delta} \quad \text { and } \quad \boldsymbol{\lambda}_{\rho}=\frac{1}{\left(1-\rho^{2}\right)^{\frac{1}{2}}} \boldsymbol{\lambda}
\]

\section*{Acknowledgments}

The two-step estimator is based on the probitiv command written by Jonah Gelbach of the University of California at Berkeley Law School and the ivprob command written by Joe Harkness of the University of Connecticut.

\section*{References}

Amemiya, T. 1978. The estimation of a simultaneous equation generalized probit model. Econometrica 46: 1193-1205. https://doi.org/10.2307/1911443.
-_ 1979. The estimation of a simultaneous-equation tobit model. International Economic Review 20: 169-181. https://doi.org/10.2307/2526423.
Finlay, K., and L. M. Magnusson. 2009. Implementing weak-instrument robust tests for a general class of instrumentalvariables models. Stata Journal 9: 398-421.

Miranda, A., and S. Rabe-Hesketh. 2006. Maximum likelihood estimation of endogenous switching and sample selection models for binary, ordinal, and count variables. Stata Journal 6: 285-308.
Newey, W. K. 1987. Efficient estimation of limited dependent variable models with endogenous explanatory variables. Journal of Econometrics 36: 231-250. https://doi.org/10.1016/0304-4076(87)90001-7.

Rivers, D., and Q. H. Vuong. 1988. Limited information estimators and exogeneity tests for simultaneous probit models. Journal of Econometrics 39: 347-366. https://doi.org/10.1016/0304-4076(88)90063-2.
Smith, R. J., and R. W. Blundell. 1986. An exogeneity test for the simultaneous equation tobit model with an application to labor supply. Econometrica 54: 679-685. https://doi.org/10.2307/1911314.
Wooldridge, J. M. 2010. Econometric Analysis of Cross Section and Panel Data. 2nd ed. Cambridge, MA: MIT Press.

\section*{Also see}
[R] ivprobit postestimation - Postestimation tools for ivprobit
[R] gmm - Generalized method of moments estimation
\([R]\) ivfprobit - Fractional probit model with continuous endogenous covariates
\([\mathrm{R}]\) ivregress - Single-equation instrumental-variables regression
\([R]\) ivtobit - Tobit model with continuous endogenous covariates
[R] probit - Probit regression
[ERM] eprobit - Extended probit regression
[SVY] svy estimation - Estimation commands for survey data
[XT] xtprobit - Random-effects and population-averaged probit models
[U] 20 Estimation and postestimation commands

\section*{Title}
ivprobit postestimation - Postestimation tools for ivprobit
\begin{tabular}{llll} 
Postestimation commands & predict & margins & estat \\
Remarks and examples & Stored results & Methods and formulas & References
\end{tabular}

\section*{Postestimation commands}

The following postestimation commands are of special interest after ivprobit:
Command Description
\begin{tabular}{ll} 
estat classification & \begin{tabular}{l} 
report various summary statistics, including the classification table \\
estat correlation
\end{tabular} \\
\begin{tabular}{l} 
report the correlation matrix of the errors of the dependent variable and \\
the endogenous variables
\end{tabular} \\
estat covariance & \begin{tabular}{l} 
report the covariance matrix of the errors of the dependent variable and \\
the endogenous variables \\
compute area under ROC curve and graph the curve \\
graph sensitivity and specificity versus probability cutoff
\end{tabular} \\
lroc & lsens
\end{tabular}

These commands are not appropriate after the two-step estimator or the svy prefix.

The following standard postestimation commands are also available:
\begin{tabular}{cl} 
Command & Description \\
\hline contrast & \begin{tabular}{l} 
contrasts and ANOVA-style joint tests of estimates \\
* estat ic
\end{tabular} \\
estat summarize & \begin{tabular}{l} 
sumpe's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian \\
information criteria (AIC, CAIC, AICc, and BIC) \\
estat vce \\
variance-covariance matrix of the estimators (VCE)
\end{tabular} \\
estat (svy) & \begin{tabular}{l} 
postestimation statistics for survey data \\
cataloging estimation results
\end{tabular} \\
etable & \begin{tabular}{l} 
table of estimation results
\end{tabular} \\
\(\dagger_{\text {forecast }}\) & \begin{tabular}{l} 
dynamic forecasts and simulations
\end{tabular} \\
lincom & \begin{tabular}{l} 
Hausman's specification test \\
point estimates, standard errors, testing, and inference for linear combi- \\
nations of coefficients
\end{tabular} \\
lrtest & \begin{tabular}{l} 
likelihood-ratio test; not available with two-step estimator \\
margins
\end{tabular} \\
marginsplot means, predictive margins, marginal effects, and average marginal \\
nlcom & \begin{tabular}{l} 
graph the results from margins (profile plots, interaction plots, etc.) \\
point estimates, standard errors, testing, and inference for nonlinear com- \\
binations of coefficients
\end{tabular} \\
predict & \begin{tabular}{l} 
linear predictions and their SEs, probabilities, etc.
\end{tabular}
\end{tabular}
predictnl
pwcompare
* suest
test
testnl
point estimates, standard errors, testing, and inference for generalized predictions
pairwise comparisons of estimates
seemingly unrelated estimation
Wald tests of simple and composite linear hypotheses
Wald tests of nonlinear hypotheses
*estat ic, forecast, and suest are not appropriate after ivprobit, twostep.
\(\dagger_{\text {forecast, hausman, and lest are not appropriate with svy estimation results. }}\)

\section*{predict}

\section*{Description for predict}
predict creates a new variable containing predictions such as structural functions, linear predictions, standard errors, and probabilities.

\section*{Menu for predict}

\author{
Statistics \(>\) Postestimation
}

\section*{Syntax for predict}

After ML
```

predict [type] newvar [if] [in] [, statistic asfmethod rules asif]
predict [type] {stub*|newvarlist } [if] [in], scores

```

After twostep
predict [type] newvar [if][in] [, twostep_statistic]
statistic Description
Main
\(\mathrm{xb} \quad\) linear prediction excluding endogeneity; the default
\(\mathrm{pr} \quad\) probability of a positive outcome
stdp standard error of the linear prediction
asfmethod Description
Main
asf average structural function; the default
fixedasf fixed average structural function

Main
xb
stdp
linear prediction; the default standard error of the linear prediction

These statistics are available both in and out sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

\section*{Options for predict}
\(\qquad\) Main
xb , the default, calculates the linear prediction.
pr calculates the probability of a positive outcome. Results depend on how the endogeneity complication is handled, which is determined by the asf or fixedasf option. pr is not available with the two-step estimator.
stdp calculates the standard error of the linear prediction.
asf and fixedasf determine how the average structural function (ASF) of the specified statistic is computed. These options are not allowed with xb or stdp .
asf is the default for the ML estimator when the pr statistic is specified. asf computes the ASF of the specified statistic. It is the statistic conditional on the errors of the endogenous variable equations. Put another way, it is the statistic accounting for the correlation of the endogenous covariates with the errors of the main equation. Derivatives and contrasts based on asf have a structural interpretation. See margins for computing derivatives and contrasts.
fixedasf calculates a fixed ASF. It is the specified statistic using only the coefficients and variables of the outcome equation. fixedasf does not condition on the errors of the endogenous variable equations. Contrasts between two fixed counterfactuals averaged over the whole sample have a potential-outcome interpretation. Intuitively, it is as if the values of the covariates were fixed at a value exogenously by fiat. See margins for computing derivatives and contrasts.

To be clear, derivatives and contrasts between two fixed counterfactuals using the default asf option also have a potential-outcome interpretation. And, unlike fixedasf, they retain that interpretation when computed over subpopulations for both linear and nonlinear models.
rules requests that Stata use any rules that were used to identify the model when making the prediction. By default, Stata calculates missing for excluded observations. rules is not available with the two-step estimator.
asif requests that Stata ignore the rules and the exclusion criteria and calculate predictions for all observations possible using the estimated parameters from the model. asif is not available with the two-step estimator.
scores, not available with twostep, calculates equation-level score variables.
For models with one endogenous regressor, four new variables are created.
The first new variable will contain \(\partial \ln L / \partial\left(\boldsymbol{z}_{i} \boldsymbol{\delta}\right)\).
The second new variable will contain \(\partial \ln L / \partial\left(\boldsymbol{x}_{i} \boldsymbol{\Pi}\right)\).
The third new variable will contain \(\partial \ln L / \partial\) atanh \(\rho\).
The fourth new variable will contain \(\partial \ln L / \partial \ln \sigma\).

For models with \(p\) endogenous regressors, \(p+\{(p+1)(p+2)\} / 2\) new variables are created. The first new variable will contain \(\partial \ln L / \partial\left(\boldsymbol{z}_{i} \boldsymbol{\delta}\right)\).
The second through \((p+1)\) th new score variables will contain \(\partial \ln L / \partial\left(\boldsymbol{x}_{i} \boldsymbol{\Pi}_{k}\right), k=1, \ldots, p\), where \(\Pi_{k}\) is the \(k\) th column of \(\Pi\).
The remaining score variables will contain the partial derivatives of \(\ln L\) with respect to \(s_{21}\), \(s_{31}, \ldots, s_{p+1,1}, s_{22}, \ldots, s_{p+1,2}, \ldots, s_{p+1, p+1}\), where \(s_{m, n}\) denotes the \((m, n)\) element of the Cholesky decomposition of the error covariance matrix.

\section*{margins}

\section*{Description for margins}
margins estimates margins of response for linear predictions and probabilities.

\section*{Menu for margins}

\author{
Statistics \(>\) Postestimation
}

\section*{Syntax for margins}
```

margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...) ...] [options]

```

\section*{After ML} statistic Description

Main
\(\mathrm{xb} \quad\) linear prediction excluding endogeneity; the default
\(\mathrm{pr} \quad\) probability of a positive outcome
stdp not allowed with margins

After twostep
statistic Description
Main
\(\mathrm{xb} \quad\) linear prediction; the default
stdp
not allowed with margins

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

\section*{estat}

\section*{Description for estat}
estat correlation displays the correlation matrix of the errors of the dependent variable and the endogenous variables.
estat covariance displays the covariance matrix of the errors of the dependent variable and the endogenous variables.
estat correlation and estat covariance are not allowed after the ivprobit two-step estimator.

\section*{Menu for estat}

Statistics \(>\) Postestimation

\section*{Syntax for estat}

Correlation matrix
```

estat correlation [, border(bspec) left(\#) format (% %mt)]

```

Covariance matrix
```

estat covariance [, border(bspec) left(\#) format(%fmt)]

```

\section*{Options for estat}
\(\qquad\) Main
border (bspec) sets border style of the matrix display. The default is border (all).
left(\#) sets the left indent of the matrix display. The default is left(2).
format (\% fmt) specifies the format for displaying the individual elements of the matrix. The default is format ( \(\% 9.0 \mathrm{~g}\) ).

\section*{Remarks and examples}

Remarks are presented under the following headings:

\author{
Marginal effects \\ Obtaining predicted values
}

\section*{Marginal effects}

Below, we discuss the interpretation of the predicted probability, pr, with the asf and fixedasf options for the ML estimator using margins.

The model is defined by two equations. The first is the equation for the outcome of interest, given by
\[
y_{1 i}^{*}=\boldsymbol{y}_{2 i} \boldsymbol{\beta}+\boldsymbol{x}_{1 i} \boldsymbol{\gamma}+u_{i}
\]
where we do not observe \(y_{1 i}^{*}\); instead, we observe
\[
y_{1 i}= \begin{cases}0 & y_{1 i}^{*}<0 \\ 1 & y_{1 i}^{*} \geq 0\end{cases}
\]

The second is the equation for the endogenous covariates, \(\boldsymbol{y}_{2 i}\),
\[
\boldsymbol{y}_{2 i}=\boldsymbol{x}_{1 i} \boldsymbol{\Pi}_{1}+\boldsymbol{x}_{2 i} \boldsymbol{\Pi}_{2}+\boldsymbol{v}_{i}
\]

This last equation is the difference between a standard probit model and the model fit by ivprobit. \(\boldsymbol{y}_{2 i}\) is modeled as an exogenous component, \(\boldsymbol{x}_{1 i} \boldsymbol{\Pi}_{1}+\boldsymbol{x}_{2 i} \boldsymbol{\Pi}_{2}\), and a component that is correlated with \(u_{i}\) and causes the endogeneity problem, \(\boldsymbol{v}_{i}\). The ASF predicted probability conditions on an estimate of \(\widehat{\boldsymbol{v}}_{i}\). It is given by
\[
\begin{aligned}
\Phi\left(\widehat{m}_{i}\right) & =\widehat{P}\left(y_{1 i} \mid \boldsymbol{x}_{1 i}, \boldsymbol{x}_{2 i}, y_{2 i}, \widehat{\boldsymbol{v}}_{i}\right) \\
\widehat{m}_{i} & =\boldsymbol{y}_{2 i} \widehat{\boldsymbol{\theta}}_{1}+\boldsymbol{x}_{1 i} \widehat{\boldsymbol{\theta}}_{2}+\widehat{\boldsymbol{v}}_{i} \widehat{\boldsymbol{\theta}}_{3}
\end{aligned}
\]

Because the correlation between \(\boldsymbol{v}_{i}\) and \(u_{i}\) is the problem we intended to address, conditioning on \(\boldsymbol{v}_{i}\) purges the model of endogeneity. Using the ASF, we can get derivatives and contrast. See Wooldridge (2010) and Blundell and Powell (2003) for an in-depth discussion of ASFs and their interpretation.

The fixed ASF, estimated when the fixedasf option is specified, has a different interpretation. Suppose we wanted to analyze \(1\left(\boldsymbol{y}_{2 i} \boldsymbol{\beta}+\boldsymbol{x}_{1 i} \boldsymbol{\gamma}+u_{i}>0\right)\) at two different values of \(\mathbf{y}_{2}\), the original \(\mathbf{y}_{2}\) and \(\mathbf{y}_{2}+1.1(\cdot)\) is an indicator function that evaluates to 1 if the condition in parentheses is satisfied and 0 otherwise. We want the average difference at these two points for the given values of the other covariates. The values of the covariates are not arrived at via the model; they are fixed by fiat. You can think of them as potential outcomes. The difference of the two values of \(\mathbf{y}_{2}\) is given by
\[
1\left\{\left(\boldsymbol{y}_{2 i}+1\right) \boldsymbol{\beta}+\boldsymbol{x}_{1 i} \boldsymbol{\gamma}+u_{i}>0\right\}-1\left(\boldsymbol{y}_{2 i} \boldsymbol{\beta}+\boldsymbol{x}_{1 i} \boldsymbol{\gamma}+u_{i}>0\right)
\]

If we average over the distribution of \(u\), we obtain
\[
\Phi\left\{\left(\boldsymbol{y}_{2 i}+1\right) \boldsymbol{\beta}+\boldsymbol{x}_{1 i} \boldsymbol{\gamma}\right\}-\Phi\left(\boldsymbol{y}_{2 i} \boldsymbol{\beta}+\boldsymbol{x}_{1 i} \boldsymbol{\gamma}\right)
\]

We do not account for endogeneity because the values of the covariates are given and fixed. If the research question you are pursuing after fitting the model has this interpretation, fixedasf gives you an adequate prediction. If, however, you do not want to treat the covariates as fixed, you should account for endogeneity and use asf predictions.

\section*{\(>\) Example 1}

We can obtain marginal effects by using the margins command after ivprobit. Continuing with example 1 in [R] ivprobit, we calculate the difference in the probability of a woman working, fem_work, if other_inc increases by \(10 \%\) versus the probability when other_inc is unchanged. The effect we get has an ASF interpretation. The probabilities are estimated conditional on the residual from the endogenous variable. In other words, the computed effects condition on the level of endogeneity in the model. See Wooldridge (2010) for a discussion about the interpretation of the estimates and the computation of marginal effects of probit estimators under endogeneity.
```

. use https://www.stata-press.com/data/r18/laborsup
. ivprobit fem_work fem_educ kids (other_inc = male_educ)
(output omitted)
. margins, at(other_inc = generate(other_inc))
> at(other_inc = generate(other_inc*1.10))
> contrast(at(r) nowald) predict(pr)
Contrasts of predictive margins Number of obs = 500
Model VCE: OIM
Expression: Average structural function probabilities, predict(pr)
1._at: other_inc = other_inc
2._at: other_inc = other_inc*1.10

```
\begin{tabular}{c|cccc}
\hline & \multicolumn{4}{|c}{\begin{tabular}{c} 
Delta-method \\
std. err.
\end{tabular}} \\
\hline [95\% conf. interval] \\
\hline (2 vs \(\overline{1})\) & -.0762151 & .0100472 & -.0959073 & -.0565229 \\
\hline
\end{tabular}

Here we see that a \(10 \%\) increase in other_inc leads to an average decrease of 0.076 in the probability of fem_work. The effect we get has an ASF interpretation. The probabilities are estimated conditional on the residual from the endogenous variable. In other words, the computed effects condition on the level of endogeneity in the model. See Wooldridge (2010) for a discussion about the interpretation of the estimates and the computation of marginal effects of probit estimators under endogeneity.

\section*{Obtaining predicted values}

After fitting your model with ivprobit, you can obtain the linear prediction and its standard error for both the estimation sample and other samples by using the predict command; see [U] 20 Estimation and postestimation commands and [R] predict. If you use the maximum likelihood estimator, you can also obtain the predicted probability or the linear prediction with an ASF or fixed ASF interpretation.
predict's pr option calculates the probability of a positive outcome, remembering any rules used to identify the model, and calculates missing for excluded observations. predict's rules option uses the rules in predicting probabilities, whereas predict's asif option ignores both the rules and the exclusion criteria and calculates probabilities for all possible observations by using the estimated parameters from the model. See Obtaining predicted values in \([R]\) probit postestimation for an example.

\section*{Stored results}
estat correlation stores the following results in \(r()\) :
Matrices
\(r\) (corr) correlation matrix of the errors
estat covariance stores the following results in \(r()\) :
Matrices
r(cov) covariance matrix of the errors

\section*{Methods and formulas}

The linear prediction is calculated as \(\boldsymbol{z}_{i} \widehat{\boldsymbol{\delta}}\), where \(\widehat{\boldsymbol{\delta}}\) is the estimated value of \(\boldsymbol{\delta}\), and \(\boldsymbol{z}_{i}\) and \(\boldsymbol{\delta}\) are defined in (1a) of [R] ivprobit. The probability of a positive outcome is evaluated at \(m_{i}, \Phi\left(m_{i}\right)\), where \(\Phi(\cdot)\) is the standard normal distribution function and \(m_{i}\) is defined in Methods and formulas of \([\mathrm{R}]\) ivprobit. The ASF uses \(\widehat{m}_{i}\) instead of \(\mathbf{y}_{2 i} \widehat{\boldsymbol{\beta}}+\mathbf{x}_{1 i} \widehat{\gamma}\) to evaluate \(\Phi(\cdot)\) and account for endogeneity in the model. The fixed ASF is evaluated at \(\mathbf{y}_{2 i} \widehat{\boldsymbol{\beta}}+\mathbf{x}_{1 i} \widehat{\boldsymbol{\gamma}}\).

\section*{References}

Blundell, R. W., and J. L. Powell. 2003. Endogeneity in nonparametric and semiparametric regression models. In Advances in Economics and Econometrics: Theory and Applications, Eighth World Congress, ed. M. Dewatripont, L. P. Hansen, and S. J. Turnovsky, vol. 2, 312-357. Cambridge: Cambridge University Press.

Wooldridge, J. M. 2010. Econometric Analysis of Cross Section and Panel Data. 2nd ed. Cambridge, MA: MIT Press.

\section*{Also see}
[R] ivprobit - Probit model with continuous endogenous covariates
[R] estat classification - Classification statistics and table
\([R]\) Iroc - Compute area under ROC curve and graph the curve
[R] Isens - Graph sensitivity and specificity versus probability cutoff
[U] 20 Estimation and postestimation commands

\section*{Title}
ivqregress - Instrumental-variables quantile regression

Description
Options
Acknowledgments

\section*{Quick start \\ Remarks and examples References}

\author{
Menu \\ Stored results Also see
}

Syntax
Methods and formulas

\section*{Description}
ivqregress fits a linear instrumental-variables quantile regression (IVQR) model that accounts for endogenous covariates using two estimators: the inverse quantile regression (IQR) estimator proposed in Chernozhukov and Hansen (2006) and the smoothed estimating equations (SEE) estimator outlined in Kaplan and Sun (2017).

\section*{Quick start}

Use the IQR estimator to fit the median IVQR model of y 1 on exogenous x 1 and endogenous y 2 with instruments z 1 and z 2
ivqregress iqr y1 x1 (y2 = z1 z2)
Same as above, but estimate the 0.75 quantile
ivqregress iqr \(y 1 \mathrm{x} 1\) ( \(\mathrm{y} 2=\mathrm{z} 1 \mathrm{z} 2\) ), quantile(0.75)
Same as above, but estimate the \(0.1,0.2, \ldots, 0.9\) quantiles
ivqregress iqr y1 x1 (y2 = z1 z2), quantile(10(10)90)
Use the SEE estimator to estimate the 0.6 quantile regression of y 1 on exogenous x 1 and endogenous y 2 and y 3 with instruments z 1 and z 2
ivqregress smooth y1 x1 (y2 y3 = z1 z2), quantile(0.6)
Same as above, but estimate the \(0.1,0.2, \ldots, 0.9\) quantiles
ivqregress smooth y1 x1 (y2 y3 = z1 z2), quantile(10(10)90)

IQR options to control optimization
Use 50 grid points in the IQR estimator to fit the 0.5 and 0.75 IVQR model
ivqregress iqr y1 x 1 ( \(\mathrm{y} 2=\mathrm{z} 1 \mathrm{z2}\) ), ngrid(50) quantile(50 75)
Same as above, but construct grid points between 1 and 5 for all the quantiles
ivqregress iqr y1 x 1 ( \(\mathrm{y} 2=\mathrm{z} 1 \mathrm{z2}\) ), ngrid(50) quantile(50 75) bound(1 5)
Same as above, but construct grid points using different bounds for different quantiles
ivqregress iqr y1 x1 (y2 = z1 z2), ngrid(50) quantile(50 75) /// bound(1 5, at(50)) bound(2 6, at(75))

SEE options to control optimization
Use 2 as the initial bandwidth in the SEE estimator to fit the 0.5 and 0.75 IVQR model
ivqregress smooth y x1 (d1 d2 = z1 z2), quantile(50 75) /// initbwidth(2)

Same as above, but use different initial bandwidths for different quantiles
```

ivqregress smooth y x1 (d1 d2 = z1 z2), quantile(50 75) ///
initbwidth(2, at(50)) initbwidth(1, at(75))

```

\section*{Menu}

Statistics \(>\) Endogenous covariates \(>\) Quantile regression with endogenous covariates

\section*{Syntax}

Inverse quantile regression (IQR) estimator
ivqregress iqr depvar \(\left[\right.\) varlist \(\left._{1}\right]\) (varname \(=\) varlist \(_{\mathrm{iv}}\) ) [if] [in] [, options IQR_options]

Smoothed estimating equations (SEE) estimator
ivqregress smooth depvar \(\left[\right.\) varlist \(\left._{1}\right]\) ( varlist \(_{2}=\) varlist \(_{\mathrm{iv}}\) ) [if] [in] [, options SEE_options]
varlist \(_{1}\) is the list of exogenous variables.
varname is an endogenous variable.
varlist \(_{2}\) is the list of endogenous variables.
varlist \(_{\mathrm{iv}}\) is the list of exogenous variables used with varlist \(t_{1}\) as instruments for varlist \(t_{2}\) and varname.
options
Model
quantile(numlist)

\section*{SE/Robust}
vce \(([\) vcetype \(][\), vceopts \(])\)

Reporting
level(\#)
display_options

Optimization
[no]log
verbose
coeflegend

Description
estimate quantiles specified in numlist; default is quantile(0.5)
technique used to estimate standard errors; vcetype may be robust (the default) or bootstrap
set confidence level; default is level (95)
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
suppress or display the iteration \(\log\)
display a verbose iteration \(\log\)
display legend instead of statistics

Options
bound \(\left(\#_{\min } \#_{\max }\left[, \operatorname{at}\left(\#_{q}\right)\right]\right)\) specify the lower and upper bounds for the grid in the \(\#_{q}\) th quantile estimation; may be repeated
ngrid \(\left(\#_{g}\right)\) use \(\#_{g}\) grid points; default is ngrid(30)

SEE_options
Description
Options
initbwidth \(\left(\#_{b}\left[\right.\right.\), at \(\left.\left(\#_{q}\right)\right]\) ) specify initial bandwidth \(\#_{b}\) to smooth the estimating equations for the \(\#_{q}\) th quantile estimation; default is the theoretical optimal bandwidth; may be repeated
iterate(\#) perform maximum of \# iterations when solving the estimating equation; default is iterate (100)
nosearchbwidth
tolerance(\#)
do not search for feasible bandwidth if the initial bandwidth is not feasible; default is to search for feasible bandwidth specify the tolerance for the coefficient vector; default is tolerance ( \(1 \mathrm{e}-9\) )
ztolerance(\#)
specify the tolerance to determine whether the proposed solution for a zero-finding problem is sufficiently close to 0 ; default is ztolerance(1e-9)

\section*{vceopts}

Description
kernel (kernel)
bwidth (\#|bwrule)
use a nonparametric kernel density estimator; default is epanechnikov
specify the bandwidth to be used by the kernel density estimator; default is silverman, which is Silverman's rule of thumb

\section*{kernel}
epanechnikov
epan2
biweight
cosine
gaussian
parzen
rectangle
triangle

\section*{Description}

Epanechnikov kernel function; the default
alternative Epanechnikov kernel function
biweight kernel function
cosine trace kernel function
Gaussian kernel function
Parzen kernel function
rectangle kernel function
triangle kernel function

\section*{bwrule}
silverman
hsheather
bof inger

Description
Silverman's rule of thumb; the default
Hall-Sheather's bandwidth
Bofinger's bandwidth
varlist \(_{1}\), varname, varlist \({ }_{2}\), and varlist \(_{\mathrm{iv}}\) may contain factor variables; see [U] 11.4.3 Factor variables. bootstrap, by, collect, rolling, and statsby are allowed; see [U] 11.1.10 Prefix commands. coeflegend does not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options}

Model
quantile (numlist) specifies the quantiles to be estimated and should contain numbers between 0 and 1 , exclusive. Numbers larger than 1 are interpreted as percentages. The default is quantile (0.5), which corresponds to the median.

\section*{Options}

The following options apply only to the IQR estimator.
bound \(\left(\#_{\min } \#_{\max }\left[, \operatorname{at}\left(\#_{q}\right)\right]\right)\) specifies the lower bound \(\left(\#_{\min }\right)\) and the upper bound \(\left(\#_{\max }\right)\) for the grid in the \(\#_{q}\) th quantile estimation. By default, the bounds are determined by the two-stage quantile regression, extending the two-stage median regression in Amemiya (1982). This option is repeatable as long as different quantiles \(\#_{q}\) are given in each specification.
The specified bound is required to be wider than the \(\#_{\text {level }}\) confidence interval (CI) that is robust to the weak instruments, which is also known as dual CI. The value of \(\#_{\text {level }}\) can be specified in the level() option; the default is \(95 \%\) CI.

The grid points are \(\#_{g}\) equally spaced points between \(\#_{\text {min }}\) and \(\#_{\text {max }}\), where \(\#_{g}\) is specified by the ngrid() option.
\(\operatorname{ngrid}\left(\#_{g}\right)\) specifies the number of grid points in the IQR estimator. The default is ngrid(30); that is, 30 grid points are used.
The following options apply only to the SEE estimator.
initbwidth \(\left(\#_{b}\left[\right.\right.\), at \(\left.\left.\left(\#_{q}\right)\right]\right)\) specifies initial bandwidth \(\#_{b}\) to smooth the estimating equations for the \(\#_{q}\) th quantile estimation. The default is the theoretical optimal bandwidth that minimizes the mean squared errors of the estimating equations; see Kaplan and Sun (2017). This option is repeatable as long as different quantiles \(\#_{q}\) are given in each specification.
iterate(\#) specifies the maximum number of iterations to perform when solving the estimating equation; the default is iterate (100).
nosearchbwidth specifies to not search for a feasible bandwidth if the initial bandwidth is not estimable; the default is to search for a feasible bandwidth.
tolerance(\#) specifies the tolerance used to determine whether successive estimates of the solution have converged. The default is tolerance (1e-9).
ztolerance (\#) specifies the tolerance used to determine whether the proposed solution to a zero-finding problem is sufficiently close to 0 ; the default is ztolerance(1e-9).
vce ([vcetype \(][\), vceopts \(]\) ) specifies the type of VCE to compute and the density estimation method to use in computing the VCE.
vcetype specifies the type of standard error reported, which includes types that are robust to some kinds of misspecification (robust) and that use bootstrap methods (bootstrap); see [R] vce_option.
vceopts available with vcetype robust are the following:
kernel (kernel) specifies the kernel method to be used by the nonparametric density estimator.
The available kernel functions are epanechnikov, epan2, biweight, cosine, gaussian, parzen, rectangle, and triangle. The default is epanechnikov. See [R] kdensity for the kernel function forms.
bwidth (\#|bwrule) specifies the bandwidth to be used by the nonparametric density estimator. If specified as a number, it is used as the bandwidth for the nonparametric density estimator. Otherwise, bwrule specifies the method used to compute the bandwidth. Available methods are silverman for Silverman's rule of thumb, hsheather for the Hall-Sheather bandwidth, and bofinger for the Bofinger bandwidth.

See [R] kdensity for Silverman's rule of thumb. See Koenker (2005, sec. 4.10) for a description of the Hall-Sheather and Bofinger bandwidth formulas.

\footnotetext{
Reporting
level (\#); see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fmt), and nolstretch; see [R] Estimation options.
}

Optimization
log and nolog specify whether to display the log showing the progress of the estimation. By default, for the IQR estimator, one dot is shown for each grid point; for the SEE estimator, one line is shown for each bandwidth. The iteration \(\log\) is displayed by default unless you used set iterlog off to suppress it; see set iterlog in [R] set iter.
verbose displays a verbose log showing the iterations of each computation step. For the IQR estimator, each line is shown for each grid point. For the SEE estimator, iteration logs are shown when solving the estimating equations.

The following option is available with ivqregress but is not shown in the dialog box:
coeflegend; see [R] Estimation options.

\section*{Remarks and examples}

Remarks are presented under the following headings:

\author{
Overview \\ When quantile regression matters \\ Examples
}

\section*{Overview}
ivqregress fits a linear IVQR model when some covariates are endogenous. The general IVQR model was first proposed by Chernozhukov and Hansen (2005). ivqregress is based on the linear IVQR model described in Chernozhukov and Hansen (2006, 2008). For an introduction to the IVQR model, see Chernozhukov, Hansen, and Wüthrich (2018). ivqregress implements two estimators: the IQR estimator proposed in Chernozhukov and Hansen (2006) and the SEE estimator outlined in Kaplan and Sun (2017).

In empirical applications, we are usually interested in the effects of some covariate on the outcome variable. The traditional linear regression model is an excellent way to model how the covariate affects the outcome's conditional mean. However, sometimes we would like to study features of the outcome distribution other than the mean to have a complete picture of the effects of covariates. For example, a policymaker may want to learn how participation in a 401 (k) would affect the lower-level, median, and upper-level conditional quantiles of net wealth.

Quantile regression in Koenker and Bassett (1978) can help us grasp a better picture than regular linear regression by estimating the effects on different quantiles of the outcome's conditional distribution. For a general discussion, see [R] qreg. For an illustration of when quantile regression matters, see When quantile regression matters below.

In practice, some covariates of interest are often endogenous for reasons such as self-selection, omission of some relevant variable, and measurement error. For example, participation in a voluntary savings plan for retirement, such as participation in a \(401(\mathrm{k})\) program, may be endogenous because the people who do and do not participate may have different saving preferences, which will affect net wealth growth.

Endogenous covariates make quantile regression estimates inconsistent, as is the case for the linear regression model. Analogous to the instrumental-variable least-squares estimator, there are IVQR model estimators to consistently estimate the effects at different quantiles. For a discussion of instrumental-variables estimation, see \([R]\) ivregress.
ivqregress fits a quantile regression model that accounts for endogenous covariates using two estimators: the IQR estimator proposed in Chernozhukov and Hansen (2006) and the SEE estimator outlined in Kaplan and Sun (2017). Intuitively, ivqregress can be thought of as the ivregress version of qreg.

Here we outline the Stata commands to fit, visualize, infer, and diagnose the IVQR model. In particular, these Stata commands can be grouped into the following categories.

Estimation: ivqregress iqr fits the IVQR model by the IQR estimator proposed in Chernozhukov and Hansen (2006, 2008).
ivqregress smooth fits the IVQR model by the SEE estimator proposed in Kaplan and Sun (2017).

Visualization: estat coefplot allows us to visualize how one covariate's effects vary at different quantiles of the outcome.

Inference: estat endogeffects tests if
1. the endogenous variable does not affect the outcome variable,
2. the effects of the endogenous variable do not vary across estimated quantiles,
3. the effects of the endogenous variable are greater than zero across estimated quantiles, and
4. the variable is exogenous instead of endogenous.
estat dualci provides CIs that are robust to weak instruments for the effects of the endogenous variable. It is allowed only after ivqregress iqr.

Diagnosis: estat waldplot helps diagnose the convergence of the IQR estimator (ivqregress iqr). In particular, estat waldplot allows us to visualize the optimization process during the computation in ivqregress iqr and shows if the searching domain contains the true value of the parameter with a predefined probability level.

In addition, some other classical postestimation tools are also available; see \([R]\) ivqregress postestimation.

\section*{When quantile regression matters}

Here is an example illustrating the advantages of quantile regressions. Suppose we have a simple model \(E(y \mid x)=\beta_{0}+x \beta_{1}\), where \(y\) is the outcome variable and \(x\) is a covariate. For simplicity, we assume \(x\) can only take values in \(\{0,1,2,3,4,5,6\}\). By definition, \(\beta_{1}\) fully characterizes the effects of increasing one unit of \(x\) on the conditional mean of outcome \(y\); that is, \(\beta_{1}=E(y \mid x=\) \(a+1)-E(y \mid x=a)\). Now we consider two scenarios of the data-generating process.
1. The probability density function of the outcome conditional on \(x=a+1, f(y \mid x=a+1)\), is only location shifted relative to \(f(y \mid x=a)\). In this case, \(\beta_{1}\) summarizes the effect of \(x\) not only on the conditional mean but also on each conditional quantile of \(y\). This case is illustrated in the left panel of figure 1.
2. The probability density function of the outcome conditional on \(x=a+1, f(y \mid x=a+1)\), is both location shifted and rescaled relative to \(f(y \mid x=a)\). In this case, \(\beta_{1}\) summarizes the effect of \(x\) only on the conditional mean but not on conditional quantiles of \(y\). This case is illustrated in the right panel of figure 1 .


Figure 1.
In the left panel, we see that each conditional density is parallel relative to each other, and only the location has been shifted. In this case, \(\beta_{1}\) captures the shift in both conditional mean and any conditional quantiles of the outcome. As a result, running a linear regression provides as much information on \(\beta_{1}\) as quantile regression.

In contrast, in the right panel, conditional density for each level of \(x\) has different locations and different shapes. Thus, \(\beta_{1}\) can only summarize the shifts in conditional mean, which are generally
different from the shifts in conditional quantiles. Quantile regression becomes necessary to learn about the effects of \(x\) on the conditional quantiles of the outcome.

\section*{Examples}

\section*{> Example 1: IVQR with the IQR estimator}

Suppose that we want to estimate the effect of \(401(\mathrm{k})\) participation ( p 401 k ) on different conditional quantiles of net financial assets (assets). We use data reported by Chernozhukov and Hansen (2004). These data are from a sample of households in the 1990 Survey of Income and Program Participation (SIPP). For the head of household, we have data on income (income), age (age), number of people in the family (familysize), years of education (educ), marital status (married), whether participated in an IRA (ira), whether received a pension benefit (pension), and whether owned a home (ownhome).

We suspect \(401(\mathrm{k})\) participation is endogenous because it may depend on unobserved factors such as saving preference that also impact financial assets. Using \(401(\mathrm{k})\) eligibility (e401k) as an instrument for \(401(\mathrm{k})\) participation, we use ivqregress to estimate how p401k affects the entire range of assets' conditional distribution. One concern about using e401k as an instrument is that choosing to work for a company that offers a \(401(\mathrm{k})\) program is not randomly assigned. Poterba, Venti, and Wise (1995) suggest that after conditioning on income, we can take working for a company that offers a \(401(\mathrm{k})\) plan as exogenous.

The IVQR model we want to fit is
\[
\operatorname{assets}_{i}=\mathrm{p}^{401 \mathrm{k}_{i} \alpha(U)+\operatorname{covariates}_{i}^{\prime} \boldsymbol{\beta}(U)}
\]
where the distribution of \(U\) conditional on the instrument e 401 k and the covariates is assumed to be uniform between 0 and 1 . The covariates income, age, familysize, and educ are included in the model as continuous variables. The covariates i.married, i.ira, i.pension, and i.ownhome are included as categorical (factor) variables. As discussed above, e401k is the instrument for p401k. The coefficients \(\alpha(U)\) and \(\boldsymbol{\beta}(U)\) are random because they depend on the unobserved random variable \(U\). In practice, \(U\) can be considered a ranking variable for the asset. When \(U\) is set to a fixed level \(\tau\), we fit an IVQR model at a specific quantile index \(\tau\). For example, when \(\tau=0.5\), we estimate how 401(k) participation affects the median of net financial assets conditional on other covariates.

The objective of the analysis is to estimate the quantile treatment effects of 401k participation on net financial assets. By definition, the \(\tau\) th conditional quantile of the asset when everyone participates in a \(401(\mathrm{k})\) plan is
\[
\operatorname{assets}_{401(\mathrm{k})}=\alpha(\tau)+\operatorname{covariates}_{i}^{\prime} \boldsymbol{\beta}(\tau)
\]

In contrast, the \(\tau\) th conditional quantile of the asset when everyone does not participate in a \(401(\mathrm{k})\) plan is
\[
\operatorname{assets}_{\mathrm{no}} 401(\mathrm{k})=\operatorname{covariates}_{i}^{\prime} \boldsymbol{\beta}(\tau)
\]

Thus, the coefficient \(\alpha(\tau)\) can fully summarize the quantile treatment effect of p 401 k on assets. That is
\[
\alpha(\tau)=\operatorname{assets}_{401(\mathrm{k})}-\operatorname{asset}_{\mathrm{no}} 401(\mathrm{k})
\]

In this example, we use the IQR estimator (ivqregress iqr) to estimate the effect of \(401(\mathrm{k})\) participation on the conditional median of the net financial assets. The dependent variable is assets. The endogenous variable i.p401k and the instrument i.e401k are specified in parentheses; the other covariates follow as a regular variable list. ivqregress fits the IV median regression model by default. The estimation result is stored as est_iqr for later use.
. use https://www.stata-press.com/data/r18/assets2
(Excerpt from Chernozhukov and Hansen (2004))
. ivqregress iqr assets (i.p401k = i.e401k) income age familysize
> i.married i.ira i.pension i.ownhome educ
Initial grid:
Quantile = 0.50: ..........10..........20........... 30 done
Adaptive grid:

IV median regression \(\quad\) Number of obs \(=9,913\)
Estimator: Inverse quantile regression Wald chi2(9) = 1289.75
Prob > chi2 \(=0.0000\)


Endogenous: 1.p401k
Exogenous: income age familysize 1.married 1.ira 1.pension 1.ownhome educ 1.e401k
. estimates store est_iqr
The coefficient for p 401 k is 5,313 . It means participation in a \(401(\mathrm{k})\) would increase the median net financial assets by \(\$ 5,313\), conditional on other covariates, relative to a scenario where no one participates. We store the estimation result as est_iqr for later use.

After ivqregress iqr, we can use estat dualci to obtain the dual CI, which is robust to weak instruments, for the coefficient on the endogenous variable.
. estat dualci
Dual confidence interval Number of obs \(=9,913\)
\begin{tabular}{c|cccccc}
\hline assets & Coefficient & \begin{tabular}{c} 
Robust \\
std. err.
\end{tabular} & z & \(\mathrm{P}>|\mathrm{z}|\) & {\([95 \%\) conf. interval] } \\
\hline \begin{tabular}{c} 
p401k \\
Yes
\end{tabular} & 5313.397 & 573.2818 & 9.27 & 0.000 & 3683.916 & 7304.986 \\
\hline
\end{tabular}

The dual CI is usually wider than the regular CI, but it provides a more robust inference if the instruments are weak. In this example, we see that the dual \(95 \% \mathrm{CI}\) is [3684, 7305], which is wider than the regular \(95 \%\) CI [4190, 6437].

\section*{> Example 2: IVQR with the smooth estimator}

In this example, we use ivqregress to fit the IVQR model as in example 1 but using the SEE estimator (ivqregress smooth). The model specification is the same as in example 1. The estimation result is stored as est_smooth for later use.
```

. ivqregress smooth assets (i.p401k = i.e401k) income age familysize
> i.married i.ira i.pension i.ownhome educ
Fitting smoothed IV quantile regression:
Quantile = .5:
Step 1: Bandwidth = 1302.9736 GMM criterion Q(b) = 2.617e-08
Step 2: Bandwidth = 6079.6881 GMM criterion Q(b) = 2.391e-12
Step 3: Bandwidth = 1438.3068 GMM criterion Q(b) = 8.068e-13

```
IV median regression \(\quad\) Number of obs \(=9,913\)
Estimator: Smoothed estimating equations Wald chi2(9) = 1243.05
                                    Prob > chi2 \(=0.0000\)
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline assets & Coefficient & Robust std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf & interval] \\
\hline p401k & & & & & & \\
\hline Yes & 5364.468 & 573.3728 & 9.36 & 0.000 & 4240.678 & 6488.258 \\
\hline income & . 1679934 & . 013419 & 12.52 & 0.000 & . 1416925 & . 1942942 \\
\hline age & 113.6318 & 9.352867 & 12.15 & 0.000 & 95.30052 & 131.9631 \\
\hline familysize & -228.7766 & 57.61072 & -3.97 & 0.000 & -341.6916 & -115.8617 \\
\hline \begin{tabular}{l}
married \\
Married
\end{tabular} & -1362.56 & 238.5988 & -5.71 & 0.000 & -1830.205 & -894.9153 \\
\hline \[
\begin{aligned}
& \text { ira } \\
& \text { Yes }
\end{aligned}
\] & 22402.04 & 1043.504 & 21.47 & 0.000 & 20356.81 & 24447.27 \\
\hline \[
\begin{aligned}
& \text { pension } \\
& \text { Receives .. }
\end{aligned}
\] & -713.996 & 220.476 & -3.24 & 0.001 & -1146.121 & -281.8709 \\
\hline ownhome & & & & & & \\
\hline Yes & -12.71396 & 161.3703 & -0.08 & 0.937 & -328.994 & 303.5661 \\
\hline educ & -102.2889 & 34.18527 & -2.99 & 0.003 & -169.2908 & -35.28701 \\
\hline _cons & -5672.645 & 619.7049 & -9.15 & 0.000 & -6887.244 & -4458.045 \\
\hline
\end{tabular}

Endogenous: 1.p401k
Exogenous: income age familysize 1.married 1.ira 1.pension 1.ownhome educ 1.e401k
```

. estimates store est_smooth

```

The interpretation of the coefficient estimates is the same as in example 1. For example, the coefficient for p 401 k is 5,364 . So participation in a \(401(\mathrm{k})\) would increase the median of net financial assets by \(\$ 5,364\), conditional on other covariates, relative to a scenario where no one participates.

Now we can compare the coefficient on p401k between the SEE estimator and the IQR estimator.
. estimates table est_iqr est_smooth, keep(i.p401k) se
\begin{tabular}{c|lr}
\hline Variable & est_iqr & est_smooth \\
\hline p401k & & \\
Yes & 5313.3974 & 5364.468 \\
& 573.28183 & 573.37279 \\
\hline \multicolumn{3}{c}{ Legend: b/se }
\end{tabular}

We see that the point estimates from these two estimators are similar but not the same. It is normal to see different results from the IQR and SEE estimators because these two estimators approximate the original exact estimating equation differently. On one hand, the IQR estimator tries to find the solution by an exhaustive grid search. The estimation result critically depends on the range and finesse of grid points. On the other hand, the SEE estimator uses a kernel method to smooth the original estimating equation. Its result depends on how well the SEE approximates the original, mainly controlled by the bandwidth.

Both the IQR and SEE estimators have their advantages and weaknesses. The IQR estimator is numerically stable, and it allows computing the dual CI, which is robust to weak instruments (use estat dualci). However, the IQR becomes computationally intensive when there is more than one endogenous variable. Thus, ivqregress iqr allows only one endogenous variable. In contrast, the SEE estimator can handle multiple endogenous variables within a reasonable computation time. However, it does not allow estat dualci for inference that is robust to weak instruments. Suppose there is only one endogenous variable in the model. We recommend using both estimators, comparing the results, and using the IQR estimator as a benchmark because it can provide valid inference even if the instrument is weak. If there is more than one endogenous variable, only ivqregress smooth is available.

\section*{\(>\) Example 3: IVQR at different quantiles}

In the first two examples, we estimated the \(401(\mathrm{k})\) participation (p401k) treatment effect on the conditional median of net financial assets (assets). From a policy designer's point of view, we may be more interested in estimating the treatment effect of p 401 k on other conditional quantiles of assets. For example, we can ask questions like 1) how \(401(\mathrm{k})\) participation affects the lower quantile of assets and 2) whether 401(k) participation is unambiguously beneficial for the asset's lower and upper conditional quantiles. In addition, we might also want to know whether the \(401(\mathrm{k})\) participation is endogenous in our model. In this example, we will show how to use ivqregress to fit the IVQR model at different quantiles and how to use the postestimation tools to answer the above questions.

First, we use the IQR estimator to fit the model at different quantiles. In particular, we specify the quantile (10 (10) 90) option to fit the IVQR model at the 10th, 20th, ..., 90th quantiles.

\begin{tabular}{|c|c|c|c|c|c|c|}
\hline assets & Coefficient & Robust std. err. & z & \(P>|z|\) & [95\% conf & interval] \\
\hline \multicolumn{7}{|l|}{q10} \\
\hline \multicolumn{7}{|l|}{p401k} \\
\hline Yes & 3240.08 & 475.6184 & 6.81 & 0.000 & 2307.885 & 4172.275 \\
\hline income & . 0303072 & . 0123138 & 2.46 & 0.014 & . 0061725 & . 0544419 \\
\hline age & 131.5908 & 15.13725 & 8.69 & 0.000 & 101.9223 & 161.2592 \\
\hline familysize & -329.2838 & 123.4665 & -2.67 & 0.008 & -571.2737 & -87.29385 \\
\hline \multicolumn{7}{|l|}{married} \\
\hline Married & -1504.648 & 380.0373 & -3.96 & 0.000 & -2249.508 & -759.7886 \\
\hline \multicolumn{7}{|l|}{ira} \\
\hline \multicolumn{7}{|l|}{pension} \\
\hline Receives . & 63.88643 & 326.6017 & 0.20 & 0.845 & -576.2412 & 704.0141 \\
\hline \multicolumn{7}{|l|}{ownhome} \\
\hline Yes & 969.6861 & 300.4319 & 3.23 & 0.001 & 380.8503 & 1558.522 \\
\hline educ & -301.1635 & 52.02897 & -5.79 & 0.000 & -403.1384 & -199.1885 \\
\hline _cons & -7455.806 & 1192.112 & -6.25 & 0.000 & -9792.302 & -5119.311 \\
\hline
\end{tabular}

\footnotetext{
(output omitted)
}
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline \multicolumn{7}{|l|}{q90} \\
\hline p401k & & & & & & \\
\hline Yes & 15983.42 & 3046.028 & 5.25 & 0.000 & 10013.32 & 21953.53 \\
\hline income & . 8247356 & . 0570029 & 14.47 & 0.000 & . 713012 & . 9364593 \\
\hline age & 485.8734 & 48.99224 & 9.92 & 0.000 & 389.8504 & 581.8965 \\
\hline familysize & -646.4962 & 185.913 & -3.48 & 0.001 & -1010.879 & -282.1134 \\
\hline married & & & & & & \\
\hline Married & -3265.007 & 753.4701 & -4.33 & 0.000 & -4741.782 & -1788.233 \\
\hline \[
\begin{aligned}
& \text { ira } \\
& \text { Yes }
\end{aligned}
\] & 68543.44 & 4952.261 & 13.84 & 0.000 & 58837.18 & 78249.69 \\
\hline \[
\begin{aligned}
& \text { pension } \\
& \text { Receives .. }
\end{aligned}
\] & -4656.177 & 869.4887 & -5.36 & 0.000 & -6360.343 & -2952.01 \\
\hline ownhome & & & & & & \\
\hline Yes & 400.1957 & 680.2776 & 0.59 & 0.556 & -933.124 & 1733.515 \\
\hline educ & 48.4205 & 106.2844 & 0.46 & 0.649 & -159.8931 & 256.7341 \\
\hline _cons & -20594.85 & 2260.983 & -9.11 & 0.000 & -25026.3 & -16163.41 \\
\hline
\end{tabular}

Endogenous: 1.p401k
Exogenous: income age familysize 1.married 1.ira 1.pension 1.ownhome educ 1.e401k

The results show the estimates for the effect of \(401(\mathrm{k})\) participation on each conditional quantile of the asset. The interpretation of the coefficient is similar to example 1, except we are looking at different conditional quantiles. For example, for quantile q90, the estimate for the coefficient on p401k is 15,983 . Thus, \(401(\mathrm{k})\) participation would increase the \(90 \%\) conditional quantile of net financial assets by \(\$ 15,983\).

In addition to looking at the numerical estimates from the coefficient table, we can use estat coefplot to visualize the trend of p401k's treatment effect from the lower to the upper quantile. By default, estat coefplot shows the first endogenous variable, which is 1.p401k in our example. We specify the name () option for later reference of this graph and add a subtitle indicating which estimator we used.
```

. estat coefplot, name(cp_iqr) subtitle(IQR estimator)

```

Coefficient plot for 1.p401k


The dots in the plot show the point estimates of p401k's treatment effect on different conditional quantiles of assets, and the gray bound shows the \(95 \%\) pointwise CI. We see that there is an upward trend of p401k's treatment effect. At lower-level quantiles such as the 10th, 20th, ..., 40th quantiles, the treatment effect is relatively flat. However, we see the treatment effect increases significantly in the upper-level quantiles. The red line shows the two-stage least-squares estimates, which can be used as a benchmark.
estat coefplot is a good way to visualize the treatment effect's trend. If we want to test some hypotheses regarding the trend and the model statistically, we can use estat endogeffects. For example, we are interested in testing the following hypotheses:

No effect: \(\quad 401(\mathrm{k})\) participation does not affect net financial assets for all the estimated quantiles.

Constant effect: 401(k) participation's treatment effect is constant for all the estimated quantiles.

Dominance: \(\quad 401(\mathrm{k})\) participation is unambiguously positive for all the estimated quantiles; that is, the coefficient values are strictly positive.

Exogeneity: \(401(\mathrm{k})\) participation is exogenous.

We will use estat endogeffects to show the Kolmogorov-Smirnov statistic and the \(95 \%\) critical value for each hypothesis. We can reject the null hypothesis if the test statistic is greater than the critical value; otherwise, we cannot reject the null hypothesis. We specify the rseed() option to make the results reproducible because the critical values are generated from a bootstrap sample.
\begin{tabular}{l|rr}
\multicolumn{3}{l}{\begin{tabular}{l}
. estat endogeffects, rseed(12345671) \\
Tests for endogenous effects
\end{tabular}} \\
\hline Null hypothesis & KS statistic & Replications \(=100\) \\
\hline No effect & 11.271 & critical value \\
Constant effect & 5.395 & 2.554 \\
Dominance & 0.000 & 2.446 \\
Exogeneity & 4.145 & 2.467 \\
\hline
\end{tabular}
```

Note: If the KS statistic < critical value, there is
insufficient evidence to reject the null
hypothesis. (KS = Kolmogorov-Smirnov)

```

In particular, we see that the test statistics are greater than the critical values in testing the hypotheses of no effect, constant effect, and exogeneity. Thus, with a \(95 \%\) confidence level, we can reject these three hypotheses. In other words, we find that \(401(\mathrm{k})\) participation has some effect, treatment is not constant across different quantiles, and \(401(\mathrm{k})\) participation is endogenous. In contrast, we cannot reject the dominance hypothesis. Thus, we find that \(401(\mathrm{k})\) participation is unambiguously beneficial for all the estimated quantiles of assets.

The test results are consistent with the result of the coefficient plot produced by estat coefplot, where we saw that the treatment effects are positive (dominance and no effect hypotheses) and upward trended (constant effect hypothesis).

For comparison, we can also use the SEE estimator to fit the model.
```

. ivqregress smooth assets (i.p401k = i.e401k) income age familysize
> i.married i.ira i.pension i.ownhome educ, quantile(10(10)90)
Fitting smoothed IV quantile regression:
Quantile = .1:
Step 1: Bandwidth = 1327.0069 GMM criterion Q(b) = 9.224e-11
Step 2: Bandwidth = 1311.3131 GMM criterion Q(b) = 1.995e-10
Quantile = .2:
Step 1: Bandwidth = 1272.5204
Step 2: Bandwidth = 1237.7195
Quantile = .3:
Step 1: Bandwidth = 1504.4065
Step 2: Bandwidth = 1486.4224
Quantile = .4:
Step 1: Bandwidth = 1362.7753
Step 2: Bandwidth = 1362.6479
Quantile = .5:
Step 1: Bandwidth = 1302.9736
Step 2: Bandwidth = 6079.6881
Step 3: Bandwidth = 1438.3068
Quantile = .6:
Step 1: Bandwidth = 1533.5129
GMM criterion Q(b) = 2.679e-18
GMM criterion Q(b) = 1.141e-19
GMM criterion Q(b) = 1.391e-10
GMM criterion Q(b) = 1.827e-11
GMM criterion Q(b) = 3.623e-10
GMM criterion Q(b) = 2.317e-10
GMM criterion Q(b) = 4.301e-12
GMM criterion Q(b) = 2.932e-10

```
IV quantile regression
Estimator: Smoothed estimating equations

Number of obs \(=9,913\)
Wald chi2 (81) \(=4932.84\)
Prob > chi2 \(=0.0000\)

\begin{tabular}{|c|c|c|c|c|c|c|}
\hline \multicolumn{7}{|l|}{q90} \\
\hline p401k & & & & & & \\
\hline Yes & 15525.23 & 3035.965 & 5.11 & 0.000 & 9574.848 & 21475.61 \\
\hline income & . 8311508 & . 0574108 & 14.48 & 0.000 & . 7186277 & . 9436738 \\
\hline age & 486.9876 & 51.61654 & 9.43 & 0.000 & 385.821 & 588.1541 \\
\hline familysize & -586.2617 & 193.5936 & -3.03 & 0.002 & -965.6983 & -206.8252 \\
\hline married & & & & & & \\
\hline Married & -3877.165 & 781.2296 & -4.96 & 0.000 & -5408.347 & -2345.983 \\
\hline \[
\begin{aligned}
& \text { ira } \\
& \text { Yes }
\end{aligned}
\] & 67888.86 & 4902.106 & 13.85 & 0.000 & 58280.91 & 77496.81 \\
\hline pension & & & & & & \\
\hline Receives . & -4829.506 & 898.9147 & -5.37 & 0.000 & -6591.346 & -3067.665 \\
\hline ownhome & & & & & & \\
\hline Yes & 715.6272 & 722.8727 & 0.99 & 0.322 & -701.1773 & 2132.432 \\
\hline educ & 14.5293 & 110.8781 & 0.13 & 0.896 & -202.7878 & 231.8464 \\
\hline _cons & -19953.21 & 2326.698 & -8.58 & 0.000 & -24513.45 & -15392.96 \\
\hline
\end{tabular}

Endogenous: 1.p401k
Exogenous: income age familysize 1.married 1.ira 1.pension 1.ownhome educ 1.e401k

After ivqregress smooth, we can also use estat coefplot to visualize the treatment effect and estat endogeffects to test some hypotheses of particular interest in the context of the IVQR model.

First, we use estat coefplot to plot the coefficients and then use graph combine so that we can visually compare this plot with the coefficients plot for the IQR estimates.
```

. estat coefplot, name(cp_smooth) subtitle(SEE estimator)

```
. graph combine cp_iqr cp_smooth, xcommon ycommon altshrink


The left and right panels of the figure show the coefficient plots for the IQR and SEE estimates, respectively. We see that both estimators produce similar trends for the coefficients on \(1 . \mathrm{p} 401 \mathrm{k}\) at different quantiles.

Next, we can use estat endogeffects to see if we draw the same conclusion regarding the four hypotheses of interest as we did with the IQR estimator.
\begin{tabular}{l}
\multicolumn{1}{l}{\begin{tabular}{l}
. estat endogeffects, rseed(12345671) \\
Tests for endogenous effects
\end{tabular}} \\
\begin{tabular}{l|rr} 
Null hypothesis & KS statistic & \(95 \%\) \\
& & critical value \\
\hline No effect & 11.507 & \\
Constant effect & 5.351 & 2.593 \\
Dominance & 0.000 & 2.391 \\
Exogeneity & 4.195 & 2.556 \\
\hline
\end{tabular}
\end{tabular}

Note: If the KS statistic < critical value, there is
insufficient evidence to reject the null
hypothesis. (KS = Kolmogorov-Smirnov)
The results align with those produced after ivqregress iqr. That is, the treatment effects are positive (dominance and no effect hypotheses), upward trended (constant effect hypothesis), and endogenous (exogeneity hypothesis).
> Example 4: Robustness checks and diagnostics for the IQR estimator
In this example, we will take a closer look at the IQR estimator and show how to use estat waldplot to inspect the convergence visually. Nevertheless, let's first briefly discuss the intuition and algorithm behind the IQR estimator.

The IVQR model satisfies the following conditional probability:
\[
\operatorname{Pr}\left(y \leq d \alpha(\tau)+\mathbf{x}^{\prime} \boldsymbol{\beta}(\tau) \mid \mathbf{x}, \mathbf{z}\right)=\tau
\]
where \(y\) is the outcome variable, \(d\) is an endogenous variable, \(\mathbf{x}\) is a vector of exogenous covariates, and \(\mathbf{z}\) is a vector of instruments. The coefficients \(\alpha(\tau)\) and \(\boldsymbol{\beta}(\tau)\) are indexed with the quantile level \(\tau\) to indicate that they are for the model of the \(\tau\) conditional quantile of the outcome \(y\). We cannot fit the above model using the regular quantile regression because the conditional set is on \(\mathbf{x}\) and \(\mathbf{z}\) but the covariates contain \(\mathbf{x}\) and \(d\). Now suppose we know the value of \(\alpha(\tau)\). We can then rewrite this conditional probability as
\[
\operatorname{Pr}\left(y-d \alpha(\tau) \leq \mathbf{x}^{\prime} \boldsymbol{\beta}(\tau)+\mathbf{z}^{\prime} 0 \mid \mathbf{x}, \mathbf{z}\right)=\tau
\]

By the definition of quantile regression, we can fit this model by running a quantile regression of the transformed outcome variable, \(y-d \alpha(\tau)\), on the covariates \(\mathbf{x}\) and instruments \(\mathbf{z}\). Notice that if \(\alpha(\tau)\) is the true value, the coefficient on the instruments, which we denote as \(\gamma(\tau)\), should be 0 . In other words, to solve the original moment conditional for the IVQR model, we need to find a \(\alpha(\tau)\) such that the auxiliary quantile regression of \(y-d \alpha(\tau)\) on \(\mathbf{x}\) and \(\mathbf{z}\) produces 0 s for the coefficients on the instrument \(\mathbf{z}\). In practice, we want \(\gamma(\tau)\) as close to 0 as possible, where the closeness to 0 can be measured by the Wald statistic on \(\gamma(\tau)\).

Based on the above intuition, here is an outline of the IQR estimator's algorithm.
1. Define a grid of \(A=\left\{\alpha_{1}, \ldots, \alpha_{J}\right\}\) (see IQR default grid algorithm in Methods and formulas).
2. For each \(\alpha_{j}\) in \(A\), run an auxiliary quantile regression of \(y-d \alpha_{j}\) on covariates \(\mathbf{x}\) and instruments \(\mathbf{z}\).
3. IQR finds \(\alpha_{k} \in A\) as a solution such that the coefficient \(\mathbf{o n} \mathbf{z}\) is as close to 0 as possible in the corresponding auxiliary quantile regression, where the Wald statistic measures the closeness to 0 .
4. The grid points boundary must be wider than the dual CI, which is robust to weak instruments; otherwise, ivqregress iqr will error out. Dual CI means it covers the true value of \(\alpha(\tau)\) with \(95 \%\) probability (see Chernozhukov and Hansen [2008] and estat dualci).

We can use estat waldplot to visualize the above procedure. Using the estimation result in example 1, we first restore the result est_iqr and then use estat waldplot to plot the Wald statistics corresponding to each grid point.
```

. estimates restore est_iqr
(results est_iqr are active now)
. estat waldplot

```

\section*{Convergence diagnostic plot}


The horizontal axis shows the grid points for \(\alpha\), and the vertical axis shows the values of the Wald statistics. The dots in the plot show the Wald statistics corresponding to each grid point. The red line is the \(95 \%\) critical value of the Wald test. Thus, only the Wald statistics below the red line will not reject the hypothesis that \(\gamma_{j}\) equals 0 . Respectively, the \(95 \%\) dual CI corresponds to the \(\alpha\) 's for which the Wald statistics are below the critical value. See example 1 for the use of estat dualci to show the numerical values of the dual CI.

By default, ivqregress iqr uses the dual CI to generate the lower and upper bounds for the grid points to make sure that the grid covers the true value of parameter \(\alpha\) with a large probability. Sometimes, we may want to customize the bounds. For example, suppose we want to search grid points between 3,000 and 6,000 . We can use the bound () option for this purpose.
```

. ivqregress iqr assets (i.p401k = i.e401k) income age familysize
> i.married i.ira i.pension i.ownhome educ, bound(3000 6000)
Initial grid:
Quantile = 0.50: .........10..........20........... }3
convergence not achieved
The grid interval should be wider than the 95% dual confidence interval.
Try to set a wider bound using option bound(). Use estat waldplot for
diagnosis.
r(430);

```

We see that ivqregress iqr stops with a "convergence not achieved" error message. The reason is that the specified bound is too narrow to cover the true value of the parameter with a \(95 \%\) probability.

We can now use estat waldplot to further visualize the issue.
```

. estat waldplot

```


The graph shows that the upper bound of 6,000 is too small because we need the Wald statistics to intersect with the \(95 \%\) critical value at the lower and upper bounds.

We can increase the upper bound and see if the IQR estimator converges. For example, below we increase the upper bound to 8,000 .
```

. ivqregress iqr assets (i.p401k = i.e401k) income age familysize
> i.married i.ira i.pension i.ownhome educ, bound(3000 8000)
Initial grid
quantile = 0.50: .........10...........20............. }3
Adaptive grid
quantile = 0.50: ..........10..........20........... }3
IV median regression Number of obs = 9,913
Estimator: Inverse quantile regression Wald chi2(9) = 1290.41
Prob > chi2 = 0.0000

```
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline assets & Coefficient & Robust std. err. & z & \(P>|z|\) & [95\% conf & interval] \\
\hline p401k & & & & & & \\
\hline Yes & 5332.937 & 574.5175 & 9.28 & 0.000 & 4206.903 & 6458.971 \\
\hline income & . 157381 & . 012478 & 12.61 & 0.000 & . 1329246 & . 1818374 \\
\hline age & 99.78981 & 8.553978 & 11.67 & 0.000 & 83.02432 & 116.5553 \\
\hline familysize & -199.6165 & 54.3519 & -3.67 & 0.000 & -306.1442 & -93.08872 \\
\hline \begin{tabular}{l}
married \\
Married
\end{tabular} & -1351.309 & 227.0824 & -5.95 & 0.000 & -1796.382 & -906.2357 \\
\hline \[
\begin{aligned}
& \text { ira } \\
& \text { Yes }
\end{aligned}
\] & 22631.85 & 1022.023 & 22.14 & 0.000 & 20628.72 & 24634.98 \\
\hline \begin{tabular}{l}
pension \\
Receives ..
\end{tabular} & -694.1447 & 210.533 & -3.30 & 0.001 & -1106.782 & -281.5077 \\
\hline ownhome & & & & & & \\
\hline Yes & -30.67158 & 154.6947 & -0.20 & 0.843 & -333.8676 & 272.5244 \\
\hline educ & -96.30363 & 32.0715 & -3.00 & 0.003 & -159.1626 & -33.44465 \\
\hline _cons & -4983.758 & 569.4043 & -8.75 & 0.000 & -6099.77 & -3867.746 \\
\hline
\end{tabular}

\footnotetext{
Endogenous: 1.p401k
Exogenous: income age familysize 1.married 1.ira 1.pension 1.ownhome educ 1.e401k
}

Now that the IQR estimator converges, we can redraw the Wald plot to confirm that the proposed grid points interval is indeed wider than the dual CI.
```

. estat waldplot

```

\section*{Convergence diagnostic plot}

Quantile \(=.5\)


\section*{Stored results}
ivqregress iqr and ivqregress smooth store the following in e():
Scalars
```

    e(N)
    e(q#)
    e(n_q)
    e(bwidth_q#)
    e(sm_init_bwidth_q#)
    e(sm_bwidth_q#)
    e(convcode)
    e(p)
    e(df_m)
    e(chi2)
    e(rank)
    ```
Macros
    \(e(\mathrm{cmd})\)
    e(cmdline)
    e(depvar)
    e(inst)
    e(bwrule)
    e(kernel)
    e(title)
    e(vce)
    e(vcetype)
    e(estimator)
    e(exogr)
    e(endog)
    e(properties)
    e(estat_cmd)
    e(predict)
        number of observations
        the quantiles requested
        number of quantiles requested
        bandwidth used in standard errors for q_\#th quantile
        initial bandwidth used in smoothing the indicator function in q\#th quantile
        estimation (smooth only)
        bandwidth used in smoothing the indicator function in q\#th quantile estimation
        (smooth only)
    0 if converged; otherwise, return code for why nonconvergence
    \(p\)-value for model test
    model degrees of freedom
    \(\chi^{2}\)
rank of e(V)
ivqregress
command as typed
name of dependent variable
names of instrumental variables
method to compute the bandwidth in standard errors
kernel function
title in estimation output
vcetype specified in vce()
title used to label Std. err.
iqr or smooth
exogenous regressors
endogenous regressors
b V
program used to implement estat
program used to implement predict
```

    e(marginsok) predictions allowed by margins
    e(marginsnotok) predictions disallowed by margins
    Matrices
e(b) coefficient vector
e(V) variance-covariance matrix of the estimators
Functions
e(sample) marks estimation sample

```

In addition to the above, the following is stored in r() :
```

Matrices

```
    \(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values,
    and confidence intervals

Note that results stored in r() are updated when the command is replayed and will be replaced when any r -class command is run after the estimation command.

\section*{Methods and formulas}

Methods and formulas are presented under the following headings:
The model
The IQR estimator
The IQR algorithm
The IQR default grid algorithm
The SEE estimator
The bandwidth selection algorithm
The robust standard errors

\section*{The model}

The general IVQR model was first proposed by Chernozhukov and Hansen (2005). ivqregress fits a linear IVQR model described in Chernozhukov and Hansen (2006, 2008). For notational simplicity, we drop the observational subscript \(i\) to refer to a random variable and add the subscript \(i\) to refer to a realization of a random variable.

We can write the linear IVQR model in the form of a "random coefficients" model as
\[
y=\mathbf{d}^{\prime} \boldsymbol{\alpha}(u)+\mathbf{x}^{\prime} \boldsymbol{\beta}(u)
\]
where
1. \(y\) is a scalar outcome variable, \(\mathbf{d}\) is a vector of endogenous variables, \(\mathbf{x}\) is a vector of exogenous variables, and \(u\) is the unobserved error term;
2. \(\mathbf{d}\) depends on the exogenous covariates \(\mathbf{x}\), and the instrumental variables \(\mathbf{z}\) and unobserved error term are correlated with \(u\);
3. \(u\) is a scalar random variable that characterizes the heterogeneity of the outcome and captures all the unobservables in the outcome from item 1 above. Conditional on \(\mathbf{z}\) and \(\mathbf{x}\), \(u\) is uniformly distributed between 0 and 1 ;
4. \(\boldsymbol{\alpha}(\cdot)\) and \(\boldsymbol{\beta}(\cdot)\) are random coefficient vectors that depend on \(u\);
5. the function \(\tau \rightarrow \mathbf{d}^{\prime} \boldsymbol{\alpha}(\tau)+\mathbf{x}^{\prime} \boldsymbol{\beta}(\tau)\) is strictly increasing in \(\tau\); and
6. the observable variables are \(\left\{y_{i}, \mathbf{x}_{i}, \mathbf{d}_{i}, \mathbf{z}_{i}\right\}_{i=1}^{N}\) with a sample of size \(N\).

Under some regularity conditions (see Chernozhukov and Hansen [2005]), the IVQR model satisfies the conditional probability
\[
\begin{equation*}
\operatorname{Pr}\left\{y \leq \mathbf{d}^{\prime} \boldsymbol{\alpha}(\tau)+\mathbf{x}^{\prime} \boldsymbol{\beta}(\tau) \mid \mathbf{x}, \mathbf{z}\right\}=\tau \tag{1}
\end{equation*}
\]

By the definition of probability and the law of iterated expectation, (1) implies the following unconditional moment condition:
\[
\begin{equation*}
E\left(\left[\tau-I\left\{y \leq \mathbf{d}^{\prime} \boldsymbol{\alpha}(\tau)+\mathbf{x}^{\prime} \boldsymbol{\beta}(\tau)\right\}\right] \Psi\right)=0 \tag{2}
\end{equation*}
\]
where \(I(\cdot)\) is the indicator function, \(\Psi=\left(\widehat{\mathbf{d}}^{\prime}, \mathbf{x}^{\prime}\right)^{\prime}\), and \(\widehat{\mathbf{d}}\) is some function of \(\mathbf{x}\) and \(\mathbf{z}\) and can be treated as instruments for \(\mathbf{d}\). In practice, \(\widehat{\mathbf{d}}\) is the linear prediction of \(\mathbf{d}\) using \(\mathbf{x}\) and \(\mathbf{z}\).

Equation (2) can be used as the estimating equation for the IVQR model. However, the objective function based on (2) is nonconvex and nonsmooth because of the indicator function. Thus, it is computationally challenging to fit the IVQR model by directly using (2).
ivqregress implements two estimators that approximately solve the original moment condition in (2). In particular, ivqregress iqr implements the IQR estimator proposed in Chernozhukov and Hansen (2006), and ivqregress smooth implements the SEE estimator outlined in Kaplan and Sun (2017). Here are the main ideas behind these two estimators.

The IQR estimator reduces the original \(p\)-dimensional (where \(p\) is the dimension of \(\mathbf{x}\) and \(\mathbf{d}\) ) nonconvex problem into a low-dimensional nonconvex problem. Then, it solves the problem by doing an exhaustive grid search over a high-quality grid. The grid is high quality in the sense that it covers the true value of the parameter for \(\alpha(\tau)\) with a high probability (Chernozhukov and Hansen 2008). As a byproduct, the IQR estimator can also provide the CI that is robust to the weak instruments, which is also known as dual CI (see estat dualci). However, the IQR estimator becomes computationally intensive if there is more than one endogenous variable. As a result, ivqregress iqr allows only one endogenous variable.

The SEE estimator smooths the original moment condition in (2) using a kernel method to approximate the indicator function. Thus, the optimization problem reduces to solving a system of smooth nonlinear equations. One advantage of the SEE estimator compared with the IQR estimator is that it can handle more than one endogenous variable. However, it cannot provide the dual CI, which is robust to weak instruments like the IQR estimator.

While the IQR and SEE estimators are consistent for the IVQR model, their results are generally different. The reason is that the two estimators approximate the original moment condition in different ways. On one hand, the IQR estimator tries to find the solution by an exhaustive grid search. The estimation result critically depends on the range and finesse of grid points. On the other hand, the SEE estimator uses a kernel method to smooth the original estimating equation. Its result depends on how well the SEE estimator approximates the original, mainly controlled by the bandwidth.

In practice, suppose there is only one endogenous variable in the model. We recommend using both estimators, comparing the results, and using the IQR estimator as a benchmark because it can provide valid inference even if the instruments are weak.

\section*{The IQR estimator}

Before diving into the details, we discuss the intuition of the IQR estimator. The IVQR model satisfies the conditional probability
\[
\operatorname{Pr}\left\{y \leq d \alpha(\tau)+\mathbf{x}^{\prime} \boldsymbol{\beta}(\tau) \mid \mathbf{x}, \mathbf{z}\right\}=\tau
\]

We cannot fit the above model using regular quantile regression because the conditional set is on \(\mathbf{x}\) and \(\mathbf{z}\), but the covariates contain \(\mathbf{x}\) and \(d\). Now suppose we know the value of \(\alpha(\tau)\). We can then rewrite this conditional probability as
\[
\operatorname{Pr}\left\{y-d \alpha(\tau) \leq \mathbf{x}^{\prime} \boldsymbol{\beta}(\tau)+\mathbf{z}^{\prime} * 0 \mid \mathbf{x}, \mathbf{z}\right\}=\tau
\]

By the definition of quantile regression, we can fit this model by running a quantile regression of the transformed outcome variable, \(y-d \alpha(\tau)\), on the covariates \(\mathbf{x}\) and instruments \(\mathbf{z}\). Notice that if \(\alpha(\tau)\) is the actual value, the coefficient on the instruments, which we denote as \(\gamma(\tau)\), should be 0 . In other words, to solve the original moment conditional for the IVQR model, we need to find a \(\alpha(\tau)\) such that the auxiliary quantile regression of \(y-d \alpha(\tau)\) on \(\mathbf{x}\) and \(\mathbf{z}\) produces zeros for the coefficients on the instruments \(\mathbf{z}\). In practice, we want \(\gamma(\tau)\) as close to 0 as possible, where the closeness to 0 can be measured by the Wald statistic on \(\gamma(\tau)\).

Based on the above intuition, here is an outline of the IQR estimator's algorithm.

\section*{The IQR algorithm}
1. Compute \(\widehat{\mathbf{d}}\), which is the linear projection of \(\mathbf{d}\) on \(\mathbf{x}\) and \(\mathbf{z}\). \(\widehat{\mathbf{d}}\) can be treated as instruments for d .
2. Define a grid of \(A=\left\{\alpha_{1}, \ldots, \alpha_{J}\right\}\). For the algorithm of the default grid generation, see The IQR default grid algorithm.
3. For each \(\alpha_{j}\) in \(A\), run an auxiliary quantile regression of \(y-d \alpha_{j}\) on covariates \(\mathbf{x}\) and instruments \(\widehat{\mathbf{d}}\).
4. IQR finds \(\alpha_{k} \in A\) as a solution such that the coefficient on \(\widehat{\mathbf{d}}\) is as close to 0 as possible in the corresponding auxiliary quantile regression, where the Wald statistic measures the closeness to 0 .
5. The grid points boundary must be wider than the dual CI, which is robust to weak instruments; otherwise, ivqregress iqr will error out. Dual CI means it covers the true value of \(\alpha(\tau)\) with \(\#_{\text {level }}\) probability (see Chernozhukov and Hansen [2008] and estat dualci). The level() option specifies the confidence level \(\#_{\text {level }}\); the default is level (95).

\section*{The IQR default grid algorithm}

The default grid algorithm can be divided into two stages: 1) the initial grid generation based on the two-stage quantile regression, which extends the two-stage median regression in Amemiya (1982); and 2) the adaptive grid that depends on the dual CI , which is robust to weak instruments (Chernozhukov and Hansen 2008).
1. Initial grid based on two-stage quantile regression
a. Run a quantile regression of \(y\) on \(\mathbf{x}\) and \(\widehat{\mathbf{d}}\). Denote \(\widetilde{\alpha}\) as the point estimate for the coefficient on \(\widehat{\mathbf{d}}\) and \(\widetilde{s}\) as its standard errors. \(\widetilde{s}\) is computed by assuming the error term is normally distributed.
b. Compute the lower and upper bounds of the grid. The lower bound is \(\mathrm{lb}=\widetilde{\boldsymbol{\alpha}}-4 \widetilde{s}\), and the upper bound is \(u b=\widetilde{\boldsymbol{\alpha}}+4 \widetilde{s}\).
c. By default, the grid points are \(\#_{g}\) equally spaced points between lb and ub , where the ngrid() option specifies the number of grid points \(\#_{g}\).
2. Adaptive grid based on the dual CI
a. Given the initial grid, go through the steps \(3-5\) in The IQR algorithm.
b. Obtain the dual CI based on the initial grid.
c. Use the dual CI as the bound for the adaptive grid points and generate \(\#_{g}\) equally spaced points.

\section*{The SEE estimator}

The basic idea of the SEE estimator is to replace the indicator function in (2) with a smooth function. To be precise, we replace the moment condition in (2) with
\[
\begin{equation*}
E\left(\left[\tau-\widetilde{I}\left\{y-\mathbf{d}^{\prime} \boldsymbol{\alpha}(\tau)-\mathbf{x}^{\prime} \boldsymbol{\beta}(\tau) \leq 0\right\}\right] \Psi\right)=0 \tag{3}
\end{equation*}
\]
where \(\widetilde{I}(v / h)=\max [0, \min \{1,(1-v / h) / 2\}]\) and \(h\) is the bandwidth. By default, the bandwidth is computed using the theoretical optimal bandwidth that minimizes the mean squared errors of the estimating equations. See proposition 2 in Kaplan and Sun (2017) for the optimal bandwidth.

Because \(\widetilde{I}(\cdot)\) is a smooth function, the SEE estimator reduces to solve a system of smooth nonlinear equations. Let \(F(\theta)\) denote the left-hand side of (3), where \(\theta=\{\boldsymbol{\alpha}(\tau), \boldsymbol{\beta}(\tau)\}\). Let \(\theta(i)\) denote the proposed solution at iteration \(i\), and let \(\theta(i-1)\) denote the proposed solution at the previous iteration. The convergence is declared if mreldif \((\theta(i), \theta(i-1))<\) itol or \(F(\theta)^{\prime} F(\theta)<z\) tol, where itol and ztol can be specified by using the tolerance() and ztolerance() options, respectively. The maximum number of iterations can be specified by using the iterate() option.

By default, the SEE estimator searches for the bandwidth as follows.

\section*{The bandwidth selection algorithm}

Denote \(\widehat{\theta}_{0}\) as the initial values for the parameters \(\alpha(\tau)\) and \(\beta(\tau)\). Denote \(\widehat{\mathbf{h}_{\mathrm{opt}}}\left(\widehat{\theta}_{0}\right)\) as the optimal bandwidths based on the initial values \(\widehat{\theta}_{0} \widehat{\mathbf{h}_{\mathrm{opt}}}\left(\widehat{\theta}_{0}\right)\) is a vector with elements \(\left(h_{1}, h_{2}, h_{3}\right)\), where \(h_{1}\) is a nonparametrically estimated bandwidth, \(h_{2}\) assumes Gaussian distribution, and \(h_{3}\) uses the Silverman rule of thumb. Regardless of the assumption used, each element in \(\widehat{\mathbf{h}_{\mathrm{opt}}}\left(\widehat{\theta}_{0}\right)\) requires initial estimates of the error term \(\epsilon=y-\mathbf{d}^{\prime} \alpha(\tau)-\mathbf{x}^{\prime} \beta(\tau)\). Thus, the optimal bandwidth is a function of the initial estimates for \(\alpha(\tau)\) and \(\beta(\tau)\). For details, see section 5.4 in Kaplan (2022).
1. Let \(\widehat{\theta}_{0}\) be the estimates of a quantile regression of \(y\) on \(\mathbf{d}\) and \(\mathbf{x}\).
2. Based on \(\widehat{\theta}_{0}\), compute the optimal bandwidths \(\widehat{\mathbf{h}_{\mathrm{opt}}}\left(\widehat{\theta}_{0}\right)\).
3. Define the initial bandwidth set as \(\mathbf{h}_{0}=\left\{\widehat{\mathbf{h}_{\mathrm{opt}}}\left(\widehat{\theta}_{0}\right), h_{\mathrm{init}}\right\}\), where \(h_{\text {init }}\) is the bandwidth in the initbwidth () option if specified. \(\mathbf{h}_{0}=\widehat{\mathbf{h}_{\mathrm{opt}}}\left(\widehat{\theta}_{0}\right)\) if initbwidth() is not specified.
4. Find the smallest element in \(\mathbf{h}_{0}\) such that it solves (3). The estimates for \(\alpha(\tau)\) are within the dual CI with \(\#_{\text {level }}\) probability (see Chernozhukov and Hansen [2008]).
a. If a valid bandwidth is found, go to step 5.
b. Otherwise, do a bisection search of the bandwidth with the upper bound as \(100 \times\) \(\min \left(\mathbf{h}_{0}\right)\) and the lower bound as \(\min \left(\mathbf{h}_{0}\right) / 100\).
If a valid bandwidth is found, denote it as \(h^{*}\).
5. Update \(\widehat{\theta}_{0}\) as the solution for the SEE estimator based on bandwidth \(h^{*}\). Repeat steps \(2-4\) based on the updated \(\widehat{\theta_{0}}\).

By default, steps \(1-5\) are used to select the bandwidth. If the nosearchbwidth and initbwidth() options are both specified, steps \(2,4 \mathrm{~b}\), and 5 are omitted. Thus, in this case, ivqregress smooth will try to solve (3) with the specified initial bandwidth without searching for the optimal or feasible bandwidths. If only the nosearchbwidth option is specified, step 4 b is omitted.

\section*{The robust standard errors}

The robust asymptotic variance-covariance estimator for the IQR and SEE estimators can be estimated as follows (see Chernozhukov and Hansen [2006] and de Castro et al. [2019]). Let \(\theta=\{\boldsymbol{\alpha}(\tau), \boldsymbol{\beta}(\tau)\}\) be the true values of parameters and \(\widehat{\theta}=\{\widehat{\boldsymbol{\alpha}(\tau)}, \widehat{\boldsymbol{\beta}(\tau)}\}\) be the IQR or the SEE estimator. For any finite collection of quantile indices \(\tau_{j}, j \in T\)
\[
[\sqrt{n}\{\widehat{\boldsymbol{\theta}}(\tau)-\boldsymbol{\theta}(\tau)\}]_{j \in T} \rightarrow N\left(0,\left[\mathbf{J}\left(\tau_{k}\right)^{-1} \mathbf{S}\left(\tau_{j}, \tau_{k}\right)\left\{\mathbf{J}\left(\tau_{j}\right)^{-1}\right\}^{\prime}\right]_{k, j \in T}\right)
\]
where
\[
\begin{aligned}
\mathbf{J}(\tau) & =E\left\{f_{\epsilon(\tau)}(0 \mid \mathbf{x}, \mathbf{d}, \mathbf{z}) \Psi\left[\mathbf{d}^{\prime}, \mathbf{x}^{\prime}\right]\right\} \\
\mathbf{S}\left(\tau_{j}, \tau_{k}\right) & =\left\{\min \left(\tau_{j}, \tau_{k}\right)-\tau_{j} \tau_{k}\right\} E\left(\Psi \Psi^{\prime}\right)
\end{aligned}
\]
and \(f_{\epsilon(\tau)}(0 \mid \mathbf{x}, \mathbf{d}, \mathbf{z})\) is the conditional density of \(\epsilon(\tau)\) evaluated at 0 , with \(\epsilon_{i}(\tau)=y_{i}-\mathbf{d}_{i}^{\prime} \boldsymbol{\alpha}(\tau)-\) \(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}(\tau)\).

The components in the variance can be obtained by their sample counterparts. In particular, \(\mathbf{S}(\cdot)\) can be estimated as
\[
\widehat{\mathbf{S}}\left(\tau_{k}, \tau_{j}\right)=\left\{\min \left(\tau_{k}, \tau_{j}\right)-\tau_{k} \tau_{j}\right\} \frac{1}{N} \sum_{i=1}^{N} \Psi_{i} \Psi_{i}^{\prime}
\]
\(\mathbf{J}(\cdot)\) can be estimated as
\[
\widehat{\mathbf{J}}(\tau)=\frac{1}{N h_{N}} \sum_{i}^{N} K\left\{\frac{-\widehat{\epsilon_{i}(\tau)}}{h_{n}}\right\} \Psi_{i}\left[\mathbf{d}_{i}^{\prime}, \mathbf{x}_{i}^{\prime}\right]
\]
where \(\widehat{\epsilon_{i}(\tau)}=y_{i}-\mathbf{d}_{i}^{\prime} \widehat{\boldsymbol{\alpha}(\tau)}-\mathbf{x}_{i}^{\prime} \widehat{\boldsymbol{\beta}(\tau)}, K(\cdot)\) is a kernel function, and \(h_{n}\) is the bandwidth.
vce(vcetype, kernel()) specifies the kernel function form \(K(\cdot)\). See [R] kdensity for the function forms of the eight kernels.
vce(vcetype, bwidth()) specifies which bandwidth to use: silverman specifies to use \(h_{\mathrm{s}}\), hsheather specifies to use \(h_{\mathrm{k}}\) with \(h_{1}\) replaced by \(h_{\mathrm{hs}}\), and bofinger specifies to use \(h_{\mathrm{k}}\) with \(h_{1}\) replaced by \(h_{\text {bo }}\).

Silverman's rule of thumb bandwidth is
\[
h_{\mathrm{s}}=0.9 \min \left\{\widehat{\sigma(\epsilon)}, \frac{M}{1.349}\right\} N^{-\frac{1}{5}}
\]
where \(\widehat{\sigma(\epsilon)}\) is the standard deviation of \(\widehat{\epsilon}\) and \(M\) is the interquartile range of \(\widehat{\epsilon}\).

The bandwidth in Koenker \((2005,81)\) is
\[
h_{\mathrm{k}}=\min \left\{\widehat{\sigma(\epsilon)}, \frac{M}{1.349}\right\}\left\{\Phi^{-1}\left(\tau+h_{1}\right)-\Phi^{-1}\left(\tau-h_{1}\right)\right\}
\]
where \(\Phi^{-1}(\cdot)\) is the inverse cumulative standard normal distribution and \(h_{1}\) can be one of the bandwidths in Hall and Sheather (1988) ( \(h_{\mathrm{hs}}\) ) or Bofinger (1975) ( \(h_{\mathrm{bo}}\) ). In particular,
\[
\begin{aligned}
& h_{\mathrm{hs}}=N^{-1 / 3} \Phi^{-1}\left(1-\frac{\alpha}{2}\right)^{2 / 3}\left[\frac{3}{2} \times \frac{\phi\left\{\Phi^{-1}(\tau)\right\}^{2}}{2 \Phi^{-1}(\tau)^{2}+1}\right]^{1 / 3} \\
& h_{\mathrm{bo}}=N^{-1 / 5}\left[\frac{9}{2} \times \frac{\phi\left\{\Phi^{-1}(\tau)\right\}^{4}}{\left\{2 \Phi^{-1}(\tau)^{2}+1\right\}^{2}}\right]^{1 / 5}
\end{aligned}
\]
where \(\phi(\cdot)\) is the standard normal probability density function.

\section*{Acknowledgments}

We thank David M. Kaplan of the Department of Economics at the University of Missouri for the helpful discussions and suggestions.

Stata has an active research community adding features to the area of instrumental-variables quantile regression. We would like to acknowledge their previous and ongoing contributions to this area: sivqr by David M. Kaplan; ivqreg2 by J. A. F. Machado and J. M. C. Santos Silva; cqiv by Victor Chernozhukov, Ivan Fernández-Val, Sukjin Han, and Amanda Kowalski; and many more. Type search instrumental variables quantile regression to see Stata's official and community-contributed features for instrumental-variables quantile regression.

\section*{References}

Amemiya, T. 1982. Two stage least absolute deviations estimators. Econometrica 50: 689-711. https://doi.org/10.2307/1912608.
Bofinger, E. 1975. Estimation of a density function using order statistics. Australian Journal of Statistics 17: 1-7. https://doi.org/10.1111/j.1467-842X.1975.tb01366.x.

Chernozhukov, V., and C. B. Hansen. 2004. The effects of \(401(\mathrm{k})\) participation on the wealth distribution: An instrumental quantile regression analysis. Review of Economics and Statistics 86: 735-751. https://doi.org/10.1162/0034653041811734.
—_. 2005. An IV model of quantile treatment effects. Econometrica 73: 245-261. https://doi.org/10.1111/j.14680262.2005.00570.x.
-. 2006. Instrumental quantile regression inference for structural and treatment effect models. Journal of Econometrics 132: 491-525. https://doi.org/10.1016/j.jeconom.2005.02.009.
-_. 2008. Instrumental variable quantile regression: A robust inference approach. Journal of Econometrics 142: 379-398. https://doi.org/10.1016/j.jeconom.2007.06.005.

Chernozhukov, V., C. B. Hansen, and K. Wüthrich. 2018. Instrumental variable quantile regression. In Handbook of Quantile Regression, ed. R. Koenker, V. Chernozhukov, X. He, and L. Peng, 119-143. Handbooks of Modern Statistical Methods, Boca Raton, FL: Chapman and Hall/CRC. https://doi.org/10.1201/9781315120256.
de Castro, L., A. F. Galvao, D. M. Kaplan, and X. Liu. 2019. Smoothed GMM for quantile models. Journal of Econometrics 213: 121-144. https://doi.org/10.1016/j.jeconom.2019.04.008.

Hall, P., and S. J. Sheather. 1988. On the distribution of a Studentized quantile. Journal of the Royal Statistical Society, Series B 50: 381-391. https://doi.org/10.1111/j.2517-6161.1988.tb01735.x.

Kaplan, D. M. 2022. Smoothed instrumental variables quantile regression. Stata Journal 22: 379-403.
Kaplan, D. M., and Y. Sun. 2017. Smoothed estimating equations for instrumental variables quantile regression. Econometric Theory 33: 105-157. https://doi.org/10.1017/S0266466615000407.

Koenker, R. 2005. Quantile Regression. New York: Cambridge University Press.
Koenker, R., and G. Bassett, Jr. 1978. Regression quantiles. Econometrica 46: 33-50. https://doi.org/10.2307/1913643.
Poterba, J. M., S. F. Venti, and D. A. Wise. 1995. Do 401(k) contributions crowd out other personal saving? Journal of Public Economics 58: 1-32. https://doi.org/10.1016/0047-2727(94)01462-W.

\section*{Also see}
\([R]\) ivqregress postestimation - Postestimation tools for ivqregress
\([R]\) ivregress - Single-equation instrumental-variables regression
[R] qreg - Quantile regression
[U] 20 Estimation and postestimation commands

\section*{Title}
ivqregress postestimation - Postestimation tools for ivqregress
\begin{tabular}{llll} 
Postestimation commands & predict & margins & estat \\
Remarks and examples & Stored results & Methods and formulas & References
\end{tabular}

\section*{Postestimation commands}

The following postestimation commands are of special interest after ivqregress:
\[
\text { Command } \quad \text { Description }
\]
estat coefplot estat endogeffects
*estat dualci
*estat waldplot
plot coefficients and their confidence intervals at different quantiles perform tests of endogeneity report dual confidence intervals for endogenous variable plot Wald statistics corresponding to each grid point
*estat dualci and estat waldplot work only after ivqregress iqr.

The following postestimation commands are also available:
Command Description
\begin{tabular}{|c|c|}
\hline contrast & contrasts and ANOVA-style joint tests of estimates \\
\hline estat summarize & summary statistics for the estimation sample \\
\hline estat vce & variance-covariance matrix of the estimators (VCE) \\
\hline estimates & cataloging estimation results \\
\hline etable & table of estimation results \\
\hline forecast & dynamic forecasts and simulations \\
\hline lincom & point estimates, standard errors, testing, and inference for linear combinations of coefficients \\
\hline margins & marginal means, predictive margins, marginal effects, and average marginal effects \\
\hline marginsplot & graph the results from margins (profile plots, interaction plots, etc.) \\
\hline nlcom & point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients \\
\hline predict & linear predictions and residuals \\
\hline predictnl & point estimates, standard errors, testing, and inference for generalized predictions \\
\hline pwcompare & pairwise comparisons of estimates \\
\hline test & Wald tests of simple and composite linear hypotheses \\
\hline testnl & Wald tests of nonlinear hypotheses \\
\hline
\end{tabular}

\section*{predict}

\section*{Description for predict}
predict creates a new variable containing predictions such as linear predictors and residuals.

\section*{Menu for predict}

\author{
Statistics \(>\) Postestimation
}

\section*{Syntax for predict}
predict \([\) type \(]\) newvar \([\) if \(][i n][\), statistic equation \(([\) eqno \(])]\)
statistic
Description
Main
\begin{tabular}{ll}
xb & linear predictor; the default \\
\(\underline{\text { residuals }}\) & residuals
\end{tabular}

These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

\section*{Options for predict}
\(\qquad\) Main
xb , the default, calculates the linear predictor.
residuals calculates the residuals, that is, \(y_{j}-\mathbf{x}_{j} \mathbf{b}\).
equation ([eqno \(]\) ) specifies the equation to which you are making the calculation.
equation() is filled in with one eqno. equation(\#1) would mean that the calculation is to be made for the first equation, equation(\#2) would mean the second, and so on. You could also refer to the equations by their names. equation (p50) would refer to the equation named p50 and equation (p90) to the equation named p90.
If you do not specify equation(), results are the same as if you had specified equation(\#1).

\section*{margins}

\section*{Description for margins}
margins estimates margins of response for linear predictors.

\section*{Menu for margins}

Statistics \(>\) Postestimation

\section*{Syntax for margins}
```

margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [options]

```
statistic Description
xb linear predictor; the default
residuals not allowed with margins

Statistics not allowed with margins are functions of stochastic quantities other than e(b). For the full syntax, see [R] margins.

\section*{estat}

\section*{Description for estat}
estat coefplot plots the estimated coefficients and their confidence intervals (CIs) after ivqregress.
estat endogeffects tests four hypotheses for the coefficients on the endogenous variables; see Chernozhukov and Hansen (2006). In particular, estat endogeffects provides tests for the following null hypotheses:
1. No effect: the endogenous variables do not affect the outcome variable.
2. Constant effect: the effects of the endogenous variables do not vary across estimated quantiles.
3. Dominance: the effects of the endogenous variables are greater than 0 across estimated quantiles.
4. Exogeneity: the variables are exogenous instead of endogenous.
estat dualci computes the dual CIs for the coefficients on the endogenous variable (one for each quantile) after ivqregress iqr; see Chernozhukov and Hansen (2008). The dual CI is robust to the weak instruments, and it is usually wider than the traditional CI.
estat waldplot plots the Wald statistic corresponding to each grid point after ivqregress iqr.

\section*{Menu for estat}

Statistics \(>\) Postestimation

\section*{Syntax for estat}

Plot coefficients and their CIs at different quantiles
estat coefplot [varname] [, coefplot_options]

Perform tests of endogeneity
estat endogeffects [varlist] [, endogeffects_options]

Report dual CIs for endogenous variable
```

estat dualci [, level(\#) display_options]

```

Plot Wald statistics corresponding to each grid point
```

estat waldplot [, waldplot_options]

```
varname is one of the endogenous regressors or exogenous variables specified when fitting ivqregress. By default, varname is the endogenous variable specified with ivqregress iqr or the first endogenous variable specified with ivqregress smooth.
varlist contains one or more of the endogenous variables specified when fitting a model with ivqregress; the default is the first endogenous variable.
\(\left.\begin{array}{ll}\text { coefplot_options } & \text { Description } \\
\hline \text { noci } & \begin{array}{l}\text { do not plot the CIs } \\
\text { no2sls } \\
\text { do not plot the 2SLS estimates }\end{array} \\
\begin{array}{ll}\text { Plot } \\
\text { connect_options } \\
\text { marker_options }\end{array} & \begin{array}{l}\text { change look of lines or connecting method } \\
\text { change look of markers (color, size, etc.) }\end{array} \\
\begin{array}{ll}\text { Cl plot } \\
\text { ciopts (area_options) }\end{array} & \text { affect rendition of the pointwise CIs }\end{array}\right\}\)\begin{tabular}{l} 
Line options \\
\begin{tabular}{l} 
lineopts (cline_options)
\end{tabular} \\
\begin{tabular}{l}
Y axis, X axis, Titles, Legend, Overall rendition of reference line identifying the 2SLS estimates \\
twoway_options
\end{tabular} \\
\hline
\end{tabular}
endogeffects_options Description
\begin{tabular}{|c|c|}
\hline all & test four hypotheses; the default \\
\hline noeffect & test of no effect \\
\hline constant & test of constant effect \\
\hline dominance & test of stochastic dominance \\
\hline exogeneity & test of exogeneity \\
\hline level(\#) & confidence level of a test; default is level (95) \\
\hline rseed(\#) & set random-number seed to \# \\
\hline reps(\#) & perform \# bootstrap replications; default is reps (100) \\
\hline
\end{tabular}
waldplot_options
Description
quantile(\#) plot Wald statistics for the \#th quantile estimation
level(\#) set confidence level; default is level (95)
Plot
connect_options
marker_options
change look of lines or connecting method
change look of markers (color, size, etc.)
CI plot
ciopts(area_options) affect rendition of the dual CI plot
Y axis, X axis, Titles, Legend, Overall
twoway_options
any options other than by () documented in [G-3] twoway_options

\section*{Options for estat}

Options for estat are presented under the following headings:
Options for estat coefplot
Options for estat endogeffects
Options for estat dualci
Options for estat waldplot

\section*{Options for estat coefplot}
noci removes plots of the pointwise CIs. The default is to plot the CIs.
no2sls removes the plot of the 2SLS estimates. The default is to plot the 2SLS reference line.
\(\qquad\) Plot
connect_options specify how points on a graph are to be connected; [G-3] connect_options.
marker_options affect the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G-3] marker_options.
\(\qquad\) CI plot
ciopts(area_options) affects rendition of the pointwise CIs; see [G-3] area_options.
\(\xrightarrow[{\begin{array}{c}\text { lineopts (cline_options) affects rendition of reference line identifying the 2SLS estimates; see } \\ {[\mathrm{G}-3] \text { cline_options. }}\end{array}}\end{array}]{\text { Line options }}\)
\(\qquad\)
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

\section*{Options for estat endogeffects}
all provides tests for all the four hypotheses, which is the default. If this option is specified with one of the options noeffect, constant, dominance, or exogeneity, then tests for all four hypotheses will be performed.
noeffect provides a test for the null hypothesis of no effect, that is, a test that the specified endogenous variables do not affect the outcome variable.
constant provides a test for the null hypothesis of constant effect, that is, a test that the effects of the specified endogenous variables do not vary across estimated quantiles.
dominance provides a test for the null hypothesis of dominance, that is, a test that the effects of the specified endogenous variables are greater than 0 across estimated quantiles.
exogeneity provides a test for the null hypothesis of exogeneity, that is, a test that the specified variables are exogenous instead of endogenous.
level(\#) specifies the confidence level, as a percentage, for CIs. The default is level(95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals.
rseed (\#) sets the random-number seed. Specifying this option makes the results reproducible because the critical values are drawn from a bootstrap sample.
reps(\#) specifies the number of bootstrap replications to get the critical values of the test. The default is reps (100).

\section*{Options for estat dualci}
level(\#) specifies the confidence level, as a percentage, for the dual CIs. The default is level(95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% firt), and nolstretch; see [R] Estimation options.

\section*{Options for estat waldplot}
quantile(\#) specifies to plot the Wald statistics for each grid points in the \#th quantile estimation; the default is the first equation.
level(\#) specifies the confidence level, as a percentage, for the dual CIs. The default is level(95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals.

\section*{\(\xrightarrow{\text { Plot }}\)}
connect_options specify how points on a graph are to be connected; [G-3] connect_options.
marker_options affect the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G-3] marker_options.
\(\qquad\) Cl plot
ciopts(area_options) affects rendition of the pointwise dual CI plot; see [G-3] area_options.

Y axis, X axis, Titles, Legend, Overall
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

\section*{Remarks and examples}

For examples of using estat coefplot and estat endogeffects, see example 3 in \([R]\) ivqregress. For an example of using estat dualci, see example 1 in [ \(R\) ] ivqregress. For an example of using estat waldplot, see example 4 in [R] ivqregress.

\section*{> Example 1}

In example 1 in [R] ivqregress, we fit an instrumental-variables quantile regression (IVQR) model to estimate the effects of \(401(\mathrm{k})\) participation on the conditional median of the net financial assets. Suppose that now we want to know the median of the net financial assets when everyone does or does not participate in a \(401(\mathrm{k})\) conditional on other covariates; we can use margins to find the answer.

We specify i.p401k immediately after margins to obtain the median of the assets under \(401(\mathrm{k})\) participation and under no \(401(\mathrm{k})\) participation. The at () option specifies the values of other covariates when computing the median. In particular, the continuous variables such as income, age, familysize, and educ are fixed at the sample mean, and people are assumed to be married, participate in an IRA, receive pension benefits, and own a home.
```

. use https://www.stata-press.com/data/r18/assets2
(Excerpt from Chernozhukov and Hansen (2004))
. ivqregress iqr assets (i.p401k = i.e401k) income age familysize
> i.married i.ira i.pension i.ownhome educ
(output omitted)
. margins i.p401k, at((mean) income age familysize educ
> married = 1 ira = 1 pension = 1 ownhome = 1)

```
Adjusted predictions \(\quad\) Number of obs \(=9,913\)
Model VCE: Robust
Expression: Linear predictor, predict()
At: income \(=37208.4\) (mean)
    age \(\quad=41.05891\) (mean)
    familysize \(=2.865328\) (mean)
    married \(=1\)
    ira \(=\quad 1\)
    pension \(=1\)
    ownhome \(=1\)
    educ \(=13.20629\) (mean)
\begin{tabular}{c|cccccc}
\hline & \multicolumn{4}{|c}{\begin{tabular}{c} 
Delta-method \\
std. err.
\end{tabular}} & z & \(\mathrm{P}>|\mathrm{z}|\) \\
\hline Margin & [95\% conf. interval] \\
\hline No & 23681.37 & 1007.612 & 23.50 & 0.000 & 21706.49 & 25656.26 \\
Yes & 28994.77 & 1123.076 & 25.82 & 0.000 & 26793.58 & 31195.96
\end{tabular}

The results show that the conditional median of assets when everyone participates in a \(401(\mathrm{k})\) is \(\$ 28,995\). In contrast, the conditional median of assets when no one participates in a \(401(\mathrm{k})\) is only \(\$ 23,681\). The difference between these two medians is \(\$ 5,313\), which is the quantile treatment effect of p 401 k and is the same as the coefficient's value.

\section*{Stored results}
estat endogeffects stores the following in \(r()\) :
Scalars
\(r\) (N_reps) number of replications
\(r\) (level) confidence level
Macros
\(r\) (endog) tested endogenous regressors
Matrices
\(r(\) table \() \quad\) matrix containing test statistics and critical values
estat dualci stores the following in \(r()\) :
Scalars
\(r(N) \quad\) number of observations
\(r\) (level) confidence level for the dual CI
Matrices
\(r\) (table) matrix containing test statistics, \(p\)-values, and confidence errors, test statistics, \(p\)-values, and confidence intervals

\section*{Methods and formulas}

Methods and formulas are presented under the following headings:
Tests of effects of endogenous variables
Dual CI

\section*{Tests of effects of endogenous variables}
estat endogeffects implements the general inference procedure outlined in section 4 of Chernozhukov and Hansen (2006). It tests the following null hypotheses:
1. No effect: the endogenous variables do not affect the outcome variable.
2. Constant effect: the effects of the endogenous variables do not vary across estimated quantiles.
3. Dominance: the effects of the endogenous variables are greater than 0 across estimated quantiles.
4. Exogeneity: the variables are exogenous instead of endogenous.

It is convenient to write each null hypothesis in the following form:
\[
\mathbf{R}(\tau)\{\boldsymbol{\theta}(\tau)-\mathbf{r}(\tau)\}=0 \quad \text { for each } \tau \in T
\]
where \(\mathbf{R}(\tau)\) is a \(q \times p\) matrix of rank \(q\) when \(q\) is smaller than the dimension of \(\boldsymbol{\theta}(\tau) . \boldsymbol{\theta}(\tau)\) is the \(p \times 1\) coefficient vector for the IVQR model in the \(\tau\) th quantile, \(\mathbf{r}(\tau) \in R^{p}\), and \(T\) is a set of estimated quantile indexes. This form is different from the classical setting because \(\boldsymbol{\theta}(\cdot)\) and \(\mathbf{r}(\cdot)\) are functions, which need to be estimated in some cases.

Based on the IVQR model estimates \(\widehat{\boldsymbol{\theta}}(\cdot)\), we focus on the basic inference process
\[
\mathbf{v}_{n}(\cdot)=\mathbf{R}(\cdot)\{\widehat{\boldsymbol{\theta}}(\cdot)-\widehat{\mathbf{r}}(\cdot)\}
\]
where \(\widehat{\mathbf{r}}(\cdot)\) is either a vector of constants or a vector of estimates from the classical quantile regression. We use the Kolmogorov-Smirnov statistic \(S_{n}=f\left\{\sqrt{n} \mathbf{v}_{n}(\cdot)\right\}\), which is a function of \(\mathbf{v}_{n}(\cdot)\).
\[
S_{n}=\sqrt{n} \sup _{\tau \in T}\left\|\mathbf{v}_{n}(\tau)\right\|_{\widehat{\boldsymbol{\Lambda}}(\tau)}
\]
where \(\|\mathbf{v}\|_{\mathbf{A}}=\sqrt{\mathbf{v}^{\prime} \mathbf{A v}}\). For the choice of \(\widehat{\boldsymbol{\Lambda}}(\tau)\), see section 4.4 in Chernozhukov and Hansen (2006). In essence, \(\widehat{\boldsymbol{\Lambda}}(\tau)\) is the empirical variance of the estimating scores implied by the IVQR model.

The null hypothesis is rejected if
\[
S_{n}>c(1-\alpha)
\]
where the critical value \(c(1-\alpha)\) with confidence level \(1-\alpha\) can be obtained using the bootstrap resampling procedure described in section 4.3 of Chernozhukov and Hansen (2006).

Now we describe the formal definition of the four hypotheses. For notational simplicity, we assume that there is one endogenous variable but the case of multiple endogenous variables can be extended easily. Let \(\alpha(\tau)\) denote the endogenous coefficient for the \(\tau\) th quantile estimation. In this case, \(\mathbf{R}=(1,0, \ldots, 0)\) for the four hypotheses considered.
1. No effect: the null hypothesis is that the endogenous variable has no impact on the outcome: \(\alpha(\tau)=0\). In this case,
\[
\begin{array}{lll}
H_{0}: & \alpha(\tau)=0 & \text { for all } \tau \in T \\
H_{1}: & \alpha(\tau) \neq 0 & \text { for some } \tau \in T
\end{array}
\]
and \(\widehat{r}(\cdot)=0\).
2. Constant effect: the null hypothesis of a constant effect is that the endogenous variable only affects the location of the outcome but not other moments. That is, \(\alpha(\tau)=c\) for all \(\tau \in T\), where \(c\) is a constant. In this case,
\[
\begin{array}{lll}
H_{0}: & \alpha(\tau)=c & \text { for all } \tau \in T \\
H_{1}: & \alpha(\tau) \neq c & \text { for some } \tau \in T
\end{array}
\]
and \(\widehat{r}(\cdot)\) is the estimate of endogenous coefficient for one of the quantile indexes.
3. Dominance: the dominance test tests whether the effects of endogenous variable are unambiguously beneficial. That is, \(\alpha(\tau)>0\) for all \(\tau \in T\). For this hypothesis, we use the one-sided Kolmogorov-Smirnov statistic
\[
S_{n}=\sqrt{n} \sup _{\tau \in T} \max (-\alpha(\tau), 0)
\]

In this case,
\[
\begin{array}{lll}
H_{0}: & \alpha(\tau)>0 & \text { for all } \tau \in T \\
H_{1}: & \alpha(\tau) \leq 0 & \text { for some } \tau \in T
\end{array}
\]
and \(\widehat{r}(\cdot)=0\).
4. Exogeneity: if all the covariates are exogenous, we can fit the model by the regular quantile regression and denote \(\eta(\tau)\) as the quantile regression estimator. The difference between \(\theta(\tau)\) and \(\eta(\tau)\) can be used to formulate a Hausman test of exogeneity. In this case, the null and alternative are defined as
\[
\begin{array}{lll}
H_{0}: & \alpha(\tau)=\mathbf{R} \eta(\tau) & \text { for all } \tau \in T \\
H_{1}: & \alpha(\tau) \neq \mathbf{R} \eta(\tau) & \text { for some } \tau \in T
\end{array}
\]
and \(\widehat{\mathbf{r}}(\tau)=\widehat{\eta}(\tau)\), where \(\widehat{\eta}(\tau)\) is the regular quantile regression estimate.

\section*{Dual \(\mathbf{C l}\)}
estat dualci computes the dual CI proposed in Chernozhukov and Hansen (2008) for the coefficient on the endogenous variable in the IVQR model. The dual CI is robust to the weak instrument, and it is usually wider than the classical CI. estat dualci is allowed only after ivqregress iqr. If you have not read about the methods for the inverse quantile regression (IQR) estimator, see The IQR estimator in Methods and formulas of [R] ivqregress.

Suppose we know the true value of the coefficient on the endogenous covariates, which we denote as \(\alpha(\tau)\), and let \(W_{n}\{\alpha(\tau)\}\) be the Wald statistic for the coefficient on the instruments in the auxiliary quantile regression. Then by proposition 1 in Chernozhukov and Hansen (2008),
\[
W_{n}\{\alpha(\tau)\} \rightarrow_{d} \chi^{2}(1)
\]
and for the confidence region \(\mathrm{CR}_{p}\{\boldsymbol{\alpha}(\tau)\}=\left\{\boldsymbol{\alpha} \in A: W_{n}(\alpha)<c_{p}\right\}\), where \(P\left\{\chi^{2}(1)<c_{p}\right\}=p\), and
\[
P\left[\boldsymbol{\alpha}(\tau) \in \mathrm{CR}_{p}\{\boldsymbol{\alpha}(\tau)\}\right]=P\left[W_{n}\{\boldsymbol{\alpha}(\tau)\}<c_{p}\right]=p
\]

Intuitively, \(W_{n}\{\alpha(\tau)\}\) is the Wald statistic for testing whether the coefficients for the instruments are 0 . When \(\alpha\) equals the true value \(\alpha(\tau), W(\cdot)\) is \(\chi^{2}\) distributed with the degree of freedom of 1 . Thus, a valid CI for \(\alpha\) can be constructed by the inversion of the Wald statistic. That is,
\[
\operatorname{CR}_{p}\{\alpha(\tau)\}=\left\{\alpha \in A: W_{n}(\boldsymbol{\alpha})<c_{p}\right\}
\]
covers the true value of \(\alpha(\tau)\) with probability approaching \(p\).
In practice, the dual CI is constructed by the lower and upper limits of the grid points such that the corresponding Wald statistic is smaller than the critical value \(c_{p}\).

\section*{References}

Chernozhukov, V., and C. B. Hansen. 2006. Instrumental quantile regression inference for structural and treatment effect models. Journal of Econometrics 132: 491-525. https://doi.org/10.1016/j.jeconom.2005.02.009.
-_ 2008. Instrumental variable quantile regression: A robust inference approach. Journal of Econometrics 142: 379-398. https://doi.org/10.1016/j.jeconom.2007.06.005.

\section*{Also see}
[R] ivqregress - Instrumental-variables quantile regression
[U] 20 Estimation and postestimation commands

\section*{Title}
ivregress - Single-equation instrumental-variables regression
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
ivregress fits linear models where one or more of the regressors are endogenously determined. ivregress supports estimation via two-stage least squares (2SLS), limited-information maximum likelihood (LIML), and generalized method of moments (GMM).

\section*{Quick start}

2SLS estimation of a linear regression of y 1 on x 1 and endogenous regressor y2 that is instrumented by z 1
ivregress 2sls y1 x1 ( \(\mathrm{y} 2 \mathrm{=}\) z1)
Same as above, but with two endogenous regressors, y2 and y3 instrumented by z1 and z2
ivregress 2sls y1 x1 (y2 y3 = z1 z2)
With robust standard errors
ivregress 2sls y1 x1 (y2 y3 = z1 z2), vce(robust)
Report small-sample statistics
ivregress 2sls y1 x1 (y2 y3 = z1 z2), small
Use LIML estimation
ivregress liml y1 x1 (y2 y3 = z1 z2)
Use GMM estimation
ivregress gmm y1 x1 (y2 y3 = z1 z2)
Also specify a weighting matrix that allows for correlation within clusters identified by cvar ivregress gmm y1 x1 (y2 y3 = z1 z2), wmatrix (cluster cvar)

\section*{Menu}

Statistics \(>\) Endogenous covariates \(>\) Linear regression with endogenous covariates

\section*{Syntax}
ivregress estimator depvar \(\left[\right.\) varlist \(\left._{1}\right] \quad\left(\right.\) varlist \(_{2}=\) varlist \(\left._{\mathrm{iv}}\right) \quad[\) if \(][\) in \(][\) weight \(]\) [, options]
varlist \(_{1}\) is the list of exogenous variables.
varlist \(_{2}\) is the list of endogenous variables.
varlist \(_{\mathrm{iv}}\) is the list of exogenous variables used with varlist \(t_{1}\) as instruments for varlist \({ }_{2}\).

\section*{estimator Description}
\begin{tabular}{ll} 
2sls & two-stage least squares (2SLS) \\
liml & limited-information maximum likelihood (LIML) \\
gmm & generalized method of moments (GMM)
\end{tabular}

\section*{options Description}

Model
noconstant
hascons
GMM \({ }^{1}\)
wmatrix (wmtype)
center
igmm
eps (\#) \({ }^{2}\)
weps(\#) \({ }^{2}\)
optimization_options \({ }^{2}\)
SE/Robust
vce (vcetype)

Reporting
level(\#)
first
small
noheader
depname (depname)
eform (string)
display_options
suppress constant term
has user-supplied constant
wmtype may be robust, cluster clustvar, hac kernel, or unadjusted center moments in weight matrix computation
use iterative instead of two-step GMM estimator
specify \# for parameter convergence criterion; default is eps (1e-6)
specify \# for weight matrix convergence criterion; default is weps ( \(1 \mathrm{e}-6\) )
control the optimization process; seldom used
vcetype may be unadjusted, robust, cluster clustvar, bootstrap, jackknife, or hac kernel
set confidence level; default is level (95)
report first-stage regression
make degrees-of-freedom adjustments and report small-sample statistics
display only the coefficient table
substitute dependent variable name
report exponentiated coefficients and use string to label them
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
```

perfect do not check for collinearity between endogenous regressors and
excluded instruments
coeflegend
display legend instead of statistics

```
\({ }^{1}\) These options may be specified only when gmm is specified.
\({ }^{2}\) These options may be specified only when igmm is specified.
varlist \(_{1}\), varlist \({ }_{2}\), and varlist \(_{\mathrm{iv}}\) may contain factor variables; see [U] 11.4.3 Factor variables.
depvar, \(^{\text {varlist }}{ }_{1}\), varlist \({ }_{2}\), and varlist \(_{\mathrm{iv}}\) may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bootstrap, by, collect, fmm, jackknife, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands. For more details, see [FMM] fmm: ivregress.
Weights are not allowed with the bootstrap prefix; see \([R]\) bootstrap.
aweights are not allowed with the jackknife prefix; see [R] jackknife.
hascons, vce(), noheader, depname(), and weights are not allowed with the svy prefix; see [SVY] svy.
aweights, fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
perfect and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options}

\section*{Model}
noconstant; see [R] Estimation options.
hascons indicates that a user-defined constant or its equivalent is specified among the independent variables.

GMM
wmatrix (wmtype) specifies the type of weighting matrix to be used in conjunction with the GMM estimator.

Specifying wmatrix (robust) requests a weighting matrix that is optimal when the error term is heteroskedastic. wmatrix (robust) is the default.
Specifying wmatrix(cluster clustvar) requests a weighting matrix that accounts for arbitrary correlation among observations within clusters identified by clustvar.
Specifying wmatrix(hac kernel \#) requests a heteroskedasticity- and autocorrelation-consistent (HAC) weighting matrix using the specified kernel (see below) with \# lags. The bandwidth of a kernel is equal to \(\#+1\).
Specifying wmatrix (hac kernel opt [\#]) requests an HAC weighting matrix using the specified kernel, and the lag order is selected using Newey and West's (1994) optimal lag-selection algorithm. \# is an optional tuning parameter that affects the lag order selected; see the discussion in Methods and formulas.

Specifying wmatrix (hac kernel) requests an HAC weighting matrix using the specified kernel and \(N-2\) lags, where \(N\) is the sample size.
There are three kernels available for HAC weighting matrices, and you may request each one by using the name used by statisticians or the name perhaps more familiar to economists:
bartlett or nwest requests the Bartlett (Newey-West) kernel;
parzen or gallant requests the Parzen (Gallant 1987) kernel; and
quadraticspectral or andrews requests the quadratic spectral (Andrews 1991) kernel.

Specifying wmatrix(unadjusted) requests a weighting matrix that is suitable when the errors are homoskedastic. The GMM estimator with this weighting matrix is equivalent to the 2SLS estimator.
center requests that the sample moments be centered (demeaned) when computing GMM weight matrices. By default, centering is not done.
igmm requests that the iterative GMM estimator be used instead of the default two-step GMM estimator. Convergence is declared when the relative change in the parameter vector from one iteration to the next is less than eps() or the relative change in the weight matrix is less than weps().
eps (\#) specifies the convergence criterion for successive parameter estimates when the iterative GMM estimator is used. The default is eps \((1 e-6)\). Convergence is declared when the relative difference between successive parameter estimates is less than eps() and the relative difference between successive estimates of the weighting matrix is less than weps ().
weps (\#) specifies the convergence criterion for successive estimates of the weighting matrix when the iterative GMM estimator is used. The default is weps \((1 e-6)\). Convergence is declared when the relative difference between successive parameter estimates is less than eps () and the relative difference between successive estimates of the weighting matrix is less than weps().
optimization_options: iterate (\#), [no]log. iterate() specifies the maximum number of iterations to perform in conjunction with the iterative GMM estimator. The default is the number set using set maxiter, which is 300 by default. log/nolog specifies whether to show the iteration log; see set iterlog in [R] set iter. These options are seldom used.

SE/Robust
vce (vcetype) specifies the type of standard error reported, which includes types that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
vce(unadjusted), the default for 2sls and liml, specifies that an unadjusted (nonrobust) VCE matrix be used. The default for gmm is based on the wmtype specified in the wmatrix() option; see wmatrix() above. If wmatrix() is specified with gmm but vce() is not, then vcetype is set equal to wmtype. To override this behavior and obtain an unadjusted (nonrobust) VCE matrix, specify vce(unadjusted).
ivregress also allows the following:
vce (hac kernel [\#|opt [\#]]) specifies that an HAC covariance matrix be used. The syntax used with vce(hac kernel ...) is identical to that used with wmatrix (hac kernel ...); see wmatrix() above.

Reporting
level(\#); see [R] Estimation options.
first requests that the first-stage regression results be displayed.
small requests that the degrees-of-freedom adjustment \(N /(N-k)\) be made to the variance-covariance matrix of parameters and that small-sample \(F\) and \(t\) statistics be reported, where \(N\) is the sample size and \(k\) is the number of parameters estimated. By default, no degrees-of-freedom adjustment is made, and Wald and \(z\) statistics are reported. Even with this option, no degrees-of-freedom adjustment is made to the weighting matrix when the GMM estimator is used.
noheader suppresses the display of the summary statistics at the top of the output, displaying only the coefficient table.
depname (depname) is used only in programs and ado-files that use ivregress to fit models other than instrumental-variables regression. depname () may be specified only at estimation time. depname
is recorded as the identity of the dependent variable, even though the estimates are calculated using depvar. This method affects the labeling of the output - not the results calculated—but could affect later calculations made by predict, where the residual would be calculated as deviations from depname rather than depvar. depname() is most typically used when depvar is a temporary variable (see \([\mathrm{P}]\) macro) used as a proxy for depname.
eform (string) is used only in programs and ado-files that use ivregress to fit models other than instrumental-variables regression. eform() specifies that the coefficient table be displayed in "exponentiated form", as defined in [R] Maximize, and that string be used to label the exponentiated coefficients in the table.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.

The following options are available with ivregress but are not shown in the dialog box:
perfect requests that ivregress not check for collinearity between the endogenous regressors and excluded instruments, allowing one to specify "perfect" instruments. This option cannot be used with the LIML estimator. This option may be required when using ivregress to implement other estimators.
coeflegend; see [R] Estimation options.

\section*{Remarks and examples}
ivregress performs instrumental-variables regression and weighted instrumental-variables regression. For a general discussion of instrumental variables, see Baum (2006), Cameron and Trivedi (2005; 2022, chap. 7) Davidson and MacKinnon (1993), Greene (2018, chap. 8), and Wooldridge (2010, 2020). See Hall (2005) for a lucid presentation of GMM estimation. Angrist and Pischke (2009, chap. 4) offer a casual yet thorough introduction to instrumental-variables estimators, including their use in estimating treatment effects. Some of the earliest work on simultaneous systems can be found in Cowles Commission monographs-Koopmans and Marschak (1950) and Koopmans and Hood (1953) - with the first developments of 2SLS appearing in Theil (1953) and Basmann (1957). However, Stock and Watson (2019, 401-402) present an example of the method of instrumental variables that was first published in 1928 by Philip Wright.

The syntax for ivregress assumes that you want to fit one equation from a system of equations or an equation for which you do not want to specify the functional form for the remaining equations of the system. To fit a full system of equations, using either 2SLS equation-by-equation or three-stage least squares, see [R] reg3. An advantage of ivregress is that you can fit one equation of a multiple-equation system without specifying the functional form of the remaining equations.

Formally, the model fit by ivregress is
\[
\begin{align*}
y_{i} & =\mathbf{y}_{i} \boldsymbol{\beta}_{1}+\mathbf{x}_{1 i} \boldsymbol{\beta}_{2}+u_{i}  \tag{1}\\
\mathbf{y}_{i} & =\mathbf{x}_{1 i} \boldsymbol{\Pi}_{1}+\mathbf{x}_{2 i} \boldsymbol{\Pi}_{2}+\mathbf{v}_{i} \tag{2}
\end{align*}
\]

Here \(y_{i}\) is the dependent variable for the \(i\) th observation, \(\mathbf{y}_{i}\) represents the endogenous regressors (varlist \({ }_{2}\) in the syntax diagram), \(\mathbf{x}_{1 i}\) represents the included exogenous regressors (varlist \({ }_{1}\) in the syntax diagram), and \(\mathbf{x}_{2 i}\) represents the excluded exogenous regressors (varlist \(\mathrm{iv}_{\mathrm{iv}}\) in the syntax diagram). \(\mathbf{x}_{1 i}\) and \(\mathbf{x}_{2 i}\) are collectively called the instruments. \(u_{i}\) and \(\mathbf{v}_{i}\) are zero-mean error terms, and the correlations between \(u_{i}\) and the elements of \(\mathbf{v}_{i}\) are presumably nonzero.

The rest of the discussion is presented under the following headings:

\author{
2SLS and LIML estimators \\ GMM estimator \\ Video example
}

\section*{2SLS and LIML estimators}

The most common instrumental-variables estimator is 2SLS.
\(>\) Example 1: 2SLS estimator
We have state data from the 1980 census on the median dollar value of owner-occupied housing (hsngval) and the median monthly gross rent (rent). We want to model rent as a function of hsngval and the percentage of the population living in urban areas (pcturban):
\[
\operatorname{rent}_{i}=\beta_{0}+\beta_{1} \text { hsngval }_{i}+\beta_{2} \text { pcturban }_{i}+u_{i}
\]
where \(i\) indexes states and \(u_{i}\) is an error term.
Because random shocks that affect rental rates in a state probably also affect housing values, we treat hsngval as endogenous. We believe that the correlation between hsngval and \(u\) is not equal to zero. On the other hand, we have no reason to believe that the correlation between pcturban and \(u\) is nonzero, so we assume that pcturban is exogenous.

Because we are treating hsngval as an endogenous regressor, we must have one or more additional variables available that are correlated with hsngval but uncorrelated with \(u\). Moreover, these excluded exogenous variables must not affect rent directly, because if they do then they should be included in the regression equation we specified above. In our dataset, we have a variable for family income (faminc) and for region of the country (region) that we believe are correlated with hsngval but not the error term. Together, pcturban, faminc, and factor variables 2.region, 3.region, and 4. region constitute our set of instruments.

To fit the equation in Stata, we specify the dependent variable and the list of included exogenous variables. In parentheses, we specify the endogenous regressors, an equal sign, and the excluded exogenous variables. Only the additional exogenous variables must be specified to the right of the equal sign; the exogenous variables that appear in the regression equation are automatically included as instruments.

Here we fit our model with the 2SLS estimator:

\begin{tabular}{r|rccccr}
\hline rent & Coefficient & Std. err. & \(z\) & P>|z| & [95\% conf. interval] \\
\hline hsngval & .0022398 & .0003284 & 6.82 & 0.000 & .0015961 & .0028836 \\
pcturban & .081516 & .2987652 & 0.27 & 0.785 & -.504053 & .667085 \\
_cons & 120.7065 & 15.22839 & 7.93 & 0.000 & 90.85942 & 150.5536 \\
\hline
\end{tabular}

\footnotetext{
Endogenous: hsngval
}

Exogenous: pcturban faminc 2.region 3.region 4.region

As we would expect, states with higher housing values have higher rental rates. The proportion of a state's population that is urban does not have a significant effect on rents.

\section*{Technical note}

In a simultaneous-equations framework, we could write the model we just fit as
\[
\begin{aligned}
\text { hsngval }_{i} & =\pi_{0}+\pi_{1} \text { faminc }_{i}+\pi_{2} 2 . \text { region }_{i}+\pi_{3} 3 . \text { region }_{i}+\pi_{4} 4 . \text { region }_{i}+v_{i} \\
\text { rent }_{i} & =\beta_{0}+\beta_{1} \text { hsngval }_{i}+\beta_{2} \text { pcturban }_{i}+u_{i}
\end{aligned}
\]
which here happens to be recursive (triangular), because hsngval appears in the equation for rent but rent does not appear in the equation for hsngval. In general, however, systems of simultaneous equations are not recursive. Because this system is recursive, we could fit the two equations individually via OLS if we were willing to assume that \(u\) and \(v\) were independent. For a more detailed discussion of triangular systems, see Kmenta (1997, 719-720).

Historically, instrumental-variables estimation and systems of simultaneous equations were taught concurrently, and older textbooks describe instrumental-variables estimation solely in the context of simultaneous equations. However, in recent decades, the treatment of endogeneity and instrumentalvariables estimation has taken on a much broader scope, while interest in the specification of complete systems of simultaneous equations has waned. Most recent textbooks, such as Cameron and Trivedi (2005), Davidson and MacKinnon (1993), and Wooldridge (2010, 2020), treat instrumentalvariables estimation as an integral part of the modern economists' toolkit and introduce it long before shorter discussions on simultaneous equations.

In addition to the 2SLS member of the \(\kappa\)-class estimators, ivregress implements the LIML estimator. Both theoretical and Monte Carlo exercises indicate that the LIML estimator may yield less bias and confidence intervals with better coverage rates than the 2SLS estimator. See Poi (2006) and Stock, Wright, and Yogo (2002) (and the papers cited therein) for Monte Carlo evidence.

\section*{> Example 2: LIML estimator}

Here we refit our model with the LIML estimator:


Endogenous: hsngval
Exogenous: pcturban faminc 2.region 3.region 4.region
These results are qualitatively similar to the 2SLS results, although the coefficient on hsngval is about \(19 \%\) higher.

\section*{GMM estimator}

Since the celebrated paper of Hansen (1982), the GMM has been a popular method of estimation in economics and finance, and it lends itself well to instrumental-variables estimation. The basic principle is that we have some moment or orthogonality conditions of the form
\[
\begin{equation*}
E\left(\mathbf{z}_{i} u_{i}\right)=\mathbf{0} \tag{3}
\end{equation*}
\]

From (1), we have \(u_{i}=y_{i}-\mathbf{y}_{i} \boldsymbol{\beta}_{1}-\mathbf{x}_{1 i} \boldsymbol{\beta}_{2}\). What are the elements of the instrument vector \(\mathbf{z}_{i}\) ? By assumption, \(\mathbf{x}_{1 i}\) is uncorrelated with \(u_{i}\), as are the excluded exogenous variables \(\mathbf{x}_{2 i}\), and so we use \(\mathbf{z}_{i}=\left[\mathbf{x}_{1 i} \mathbf{x}_{2 i}\right]\). The moment conditions are simply the mathematical representation of the assumption that the instruments are exogenous-that is, the instruments are orthogonal to (uncorrelated with) \(u_{i}\).

If the number of elements in \(\mathbf{z}_{i}\) is just equal to the number of unknown parameters, then we can apply the analogy principle to (3) and solve
\[
\begin{equation*}
\frac{1}{N} \sum_{i} \mathbf{z}_{i} u_{i}=\frac{1}{N} \sum_{i} \mathbf{z}_{i}\left(y_{i}-\mathbf{y}_{i} \boldsymbol{\beta}_{1}-\mathbf{x}_{1 i} \boldsymbol{\beta}_{2}\right)=\mathbf{0} \tag{4}
\end{equation*}
\]

This equation is known as the method of moments estimator. Here, where the number of instruments equals the number of parameters, the method of moments estimator coincides with the 2SLS estimator, which also coincides with what has historically been called the indirect least-squares estimator (Judge et al. 1985, 595).

The "generalized" in GMM addresses the case in which the number of instruments (columns of \(\mathbf{z}_{i}\) ) exceeds the number of parameters to be estimated. Here there is no unique solution to the population moment conditions defined in (3), so we cannot use (4). Instead, we define the objective function
\[
\begin{equation*}
Q\left(\boldsymbol{\beta}_{1}, \boldsymbol{\beta}_{2}\right)=\left(\frac{1}{N} \sum_{i} \mathbf{z}_{i} u_{i}\right)^{\prime} \mathbf{W}\left(\frac{1}{N} \sum_{i} \mathbf{z}_{i} u_{i}\right) \tag{5}
\end{equation*}
\]
where \(\mathbf{W}\) is a positive-definite matrix with the same number of rows and columns as the number of columns of \(\mathbf{z}_{i}\). \(\mathbf{W}\) is known as the weighting matrix, and we specify its structure with the wmatrix() option. The GMM estimator of \(\left(\boldsymbol{\beta}_{1}, \boldsymbol{\beta}_{2}\right)\) minimizes \(Q\left(\boldsymbol{\beta}_{1}, \boldsymbol{\beta}_{2}\right)\); that is, the GMM estimator chooses \(\boldsymbol{\beta}_{1}\) and \(\boldsymbol{\beta}_{2}\) to make the moment conditions as close to zero as possible for a given \(\mathbf{W}\). For a more general GMM estimator, see \([\mathrm{R}] \mathrm{gmm}\). gmm does not restrict you to fitting a single linear equation, though the syntax is more complex.

A well-known result is that if we define the matrix \(\mathbf{S}_{0}\) to be the covariance of \(\mathbf{z}_{i} u_{i}\) and set \(\mathbf{W}=\mathbf{S}_{0}^{-1}\), then we obtain the optimal two-step GMM estimator, where by optimal estimator we mean the one that results in the smallest variance given the moment conditions defined in (3).

Suppose that the errors \(u_{i}\) are heteroskedastic but independent among observations. Then
\[
\mathbf{S}_{0}=E\left(\mathbf{z}_{i} u_{i} u_{i} \mathbf{z}_{i}^{\prime}\right)=E\left(u_{i}^{2} \mathbf{z}_{i} \mathbf{z}_{i}^{\prime}\right)
\]
and the sample analogue is
\[
\begin{equation*}
\widehat{\mathbf{S}}=\frac{1}{N} \sum_{i} \widehat{u}_{i}^{2} \mathbf{z}_{i} \mathbf{z}_{i}^{\prime} \tag{6}
\end{equation*}
\]

To implement this estimator, we need estimates of the sample residuals \(\widehat{u}_{i}\). ivregress gmm obtains the residuals by estimating \(\boldsymbol{\beta}_{1}\) and \(\boldsymbol{\beta}_{2}\) by 2SLS and then evaluates (6) and sets \(\mathbf{W}=\widehat{\mathbf{S}}^{-1}\). Equation (6) is the same as the center term of the "sandwich" robust covariance matrix available from most Stata estimation commands through the vce(robust) option.

\section*{> Example 3: GMM estimator}

Here we refit our model of rents by using the GMM estimator, allowing for heteroskedasticity in \(u_{i}\) :
\begin{tabular}{|c|c|c|c|}
\hline \multirow[t]{4}{*}{Instrumental variables GMM regression} & Number of obs & & 50 \\
\hline & Wald chi2(2) & = & 112.09 \\
\hline & Prob > chi2 & = & 0.0000 \\
\hline & R -squared & & 0.6616 \\
\hline GMM weight matrix: Robust & Root MSE & = & 20.358 \\
\hline
\end{tabular}
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline rent & Coefficient & Robust std. err. & z & \(P>|z|\) & [95\% conf. & interval] \\
\hline hsngval & . 0014643 & . 0004473 & 3.27 & 0.001 & . 0005877 & . 002341 \\
\hline pcturban & . 7615482 & . 2895105 & 2.63 & 0.009 & . 1941181 & 1.328978 \\
\hline _cons & 112.1227 & 10.80234 & 10.38 & 0.000 & 90.95052 & 133.2949 \\
\hline
\end{tabular}

Endogenous: hsngval
Exogenous: pcturban faminc 2.region 3.region 4.region
Because we requested that a heteroskedasticity-consistent weighting matrix be used during estimation but did not specify the vce() option, ivregress reported standard errors that are robust to heteroskedasticity. Had we specified vce(unadjusted), we would have obtained standard errors that would be correct only if the weighting matrix \(\mathbf{W}\) does in fact converge to \(\mathbf{S}_{0}^{-1}\).

\section*{- Technical note}

Many software packages that implement GMM estimation use the same heteroskedasticity-consistent weighting matrix we used in the previous example to obtain the optimal two-step estimates but do not use a heteroskedasticity-consistent VCE, even though they may label the standard errors as being "robust". To replicate results obtained from other packages, you may have to use the vce (unadjusted) option. See Methods and formulas below for a discussion of robust covariance matrix estimation in the GMM framework.

By changing our definition of \(\mathbf{S}_{0}\), we can obtain GMM estimators suitable for use with other types of data that violate the assumption that the errors are independent and identically distributed. For example, you may have a dataset that consists of multiple observations for each person in a sample. The observations that correspond to the same person are likely to be correlated, and the estimation technique should account for that lack of independence. Say that in your dataset, people are identified by the variable personid and you type
```

. ivregress gmm ..., wmatrix(cluster personid)

```

Here ivregress estimates \(\mathbf{S}_{0}\) as
\[
\widehat{\mathbf{S}}=\frac{1}{N} \sum_{c \in C} \mathbf{q}_{c} \mathbf{q}_{c}^{\prime}
\]
where \(C\) denotes the set of clusters and
\[
\mathbf{q}_{c}=\sum_{i \in c_{j}} \widehat{u}_{i} \mathbf{z}_{i}
\]
where \(c_{j}\) denotes the \(j\) th cluster. This weighting matrix accounts for the within-person correlation among observations, so the GMM estimator that uses this version of \(\mathbf{S}_{0}\) will be more efficient than the estimator that ignores this correlation.

\section*{> Example 4: GMM estimator with clustering}

We have data from the National Longitudinal Survey on young women's wages as reported in a series of interviews from 1968 through 1988, and we want to fit a model of wages as a function of each woman's age and age squared, job tenure, birth year, and level of education. We believe that random shocks that affect a woman's wage also affect her job tenure, so we treat tenure as endogenous. As additional instruments, we use her union status, number of weeks worked in the past year, and a dummy indicating whether she lives in a metropolitan area. Because we have several observations for each woman (corresponding to interviews done over several years), we want to control for clustering on each person.


Endogenous: tenure
Exogenous: age c.age\#c.age birth_yr grade union wks_work msp
Both job tenure and years of schooling have significant positive effects on wages.

Time-series data are often plagued by serial correlation. In these cases, we can construct a weighting matrix to account for the fact that the error in period \(t\) is probably correlated with the errors in periods \(t-1, t-2\), etc. An HAC weighting matrix can be used to account for both serial correlation and potential heteroskedasticity.

To request an HAC weighting matrix, you specify the wmatrix (hac kernel [\#|opt]) option. kernel specifies which of three kernels to use: bartlett, parzen, or quadraticspectral. kernel determines the amount of weight given to lagged values when computing the HAC matrix, and \# denotes the maximum number of lags to use. Many texts refer to the bandwidth of the kernel instead of the number of lags; the bandwidth is equal to the number of lags plus one. If neither opt nor \# is specified, then \(N-2\) lags are used, where \(N\) is the sample size.

If you specify wmatrix (hac kernel opt), then ivregress uses Newey and West's (1994) algorithm for automatically selecting the number of lags to use. Although the authors' Monte Carlo simulations do show that the procedure may result in size distortions of hypothesis tests, the procedure is still useful when little other information is available to help choose the number of lags.

For more on GMM estimation, see Baum (2006); Baum, Schaffer, and Stillman (2003, 2007); Cameron and Trivedi (2005); Davidson and MacKinnon (1993); Hayashi (2000); or Wooldridge (2010). See Newey and West (1987) and Wang and Wu (2012) for an introduction to HAC covariance matrix estimation.

\section*{Video example}

Instrumental variables regression using Stata

\section*{Stored results}
ivregress stores the following in e() :
Scalars
e(N)
e(mss)
e(df_m)
e(rss)
e(df_r)
e(r2)
e(r2_a)
e(F)
e(rmse)
e(N_clust)
e(chi2)
e(kappa)
e(J)
e(wlagopt)
e(vcelagopt)
e(hac_lag)
\(e\) (rank)
e(k_endog)
e(iterations)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(endog)
e(exog)
e(constant)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(hac_kernel)
e(vce)
e(vcetype)
e(estimator)
e(exogr)
e(wmatrix)
\(e\) (moments)
e(small)
\(e\) (properties)
number of observations
model sum of squares
model degrees of freedom
residual sum of squares
residual degrees of freedom
\(R^{2}\)
adjusted \(R^{2}\)
\(F\) statistic
root mean squared error
number of clusters
\(\chi^{2}\)
\(\kappa\) used in LIML estimator
value of GMM objective function
lags used in HAC weight matrix (if Newey-West algorithm used)
lags used in HAC VCE matrix (if Newey-West algorithm used)
HAC lag
rank of e(V)
number of endogenous regressors (after factor-variable expansion)
number of GMM iterations ( 0 if not applicable)
ivregress
command as typed
name of dependent variable
names of endogenous variables
names of exogenous variables
noconstant or hasconstant if specified
weight type
weight expression
title in estimation output
name of cluster variable
HAC kernel
vcetype specified in vce()
title used to label Std. err.
2 sls , liml, or gmm
exogenous regressors
wmtype specified in wmatrix()
centered if center specified
small if small-sample statistics
b V
```

e(estat_cmd)
e(predict)
e(footnote)
e(marginsok)
e(marginsnotok)
e(asbalanced)
e(asobserved)

```
Matrices
    e(b)
    e(W)
    e(S)
    e(V)
    e(V_modelbased)
Functions
    e(sample)
program used to implement estat
program used to implement predict
program used to implement footnote display
predictions allowed by margins
predictions disallowed by margins
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
weight matrix used to compute GMM estimates
moment covariance matrix used to compute GMM variance-covariance matrix variance-covariance matrix of the estimators model-based variance
marks estimation sample

In addition to the above, the following is stored in r() :

Matrices
r(table)
matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

\section*{Methods and formulas}

Methods and formulas are presented under the following headings:
Notation
2SLS and LIML estimators
GMM estimator

\section*{Notation}

Items printed in lowercase and italicized (for example, \(x\) ) are scalars. Items printed in lowercase and boldfaced (for example, \(\mathbf{x}\) ) are vectors. Items printed in uppercase and boldfaced (for example, \(\mathbf{X )}\) are matrices.

The model is
\[
\begin{gathered}
\mathbf{y}=\mathbf{Y} \boldsymbol{\beta}_{1}+\mathbf{X}_{1} \boldsymbol{\beta}_{2}+\mathbf{u}=\mathbf{X} \boldsymbol{\beta}+\mathbf{u} \\
\mathbf{Y}=\mathbf{X}_{1} \boldsymbol{\Pi}_{1}+\mathbf{X}_{2} \boldsymbol{\Pi}_{2}+\mathbf{V}=\mathbf{Z} \boldsymbol{\Pi}+\mathbf{V}
\end{gathered}
\]
where \(\mathbf{y}\) is an \(N \times 1\) vector of the left-hand-side variable; \(N\) is the sample size; \(\mathbf{Y}\) is an \(N \times p\) matrix of \(p\) endogenous regressors; \(\mathbf{X}_{1}\) is an \(N \times k_{1}\) matrix of \(k_{1}\) included exogenous regressors; \(\mathbf{X}_{2}\) is an \(N \times k_{2}\) matrix of \(k_{2}\) excluded exogenous variables, \(\mathbf{X}=\left[\mathbf{Y} \mathbf{X}_{1}\right], \mathbf{Z}=\left[\mathbf{X}_{1} \mathbf{X}_{2}\right] ; \mathbf{u}\) is an \(N \times 1\) vector of errors; \(\mathbf{V}\) is an \(N \times p\) matrix of errors; \(\boldsymbol{\beta}=\left[\boldsymbol{\beta}_{1} \boldsymbol{\beta}_{2}\right]\) is a \(k=\left(p+k_{1}\right) \times 1\) vector of parameters; and \(\boldsymbol{\Pi}\) is a \(\left(k_{1}+k_{2}\right) \times p\) vector of parameters. If a constant term is included in the model, then one column of \(\mathbf{X}_{1}\) contains all ones.

Let \(\mathbf{v}\) be a column vector of weights specified by the user. If no weights are specified, \(\mathbf{v}=\mathbf{1}\). Let \(\mathbf{w}\) be a column vector of normalized weights. If no weights are specified or if the user specified fweights or iweights, \(\mathbf{w}=\mathbf{v}\); otherwise, \(\mathbf{w}=\left\{\mathbf{v} /\left(\mathbf{1}^{\prime} \mathbf{v}\right)\right\}\left(\mathbf{1}^{\prime} \mathbf{1}\right)\). Let \(\mathbf{D}\) denote the \(N \times N\) matrix with \(\mathbf{w}\) on the main diagonal and zeros elsewhere. If no weights are specified, \(\mathbf{D}\) is the identity matrix.

The weighted number of observations \(n\) is defined as \(\mathbf{1}^{\prime} \mathbf{w}\). For iweights, this is truncated to an integer. The sum of the weights is \(\mathbf{1}^{\prime} \mathbf{v}\). Define \(c=1\) if there is a constant in the regression and zero otherwise.

The order condition for identification requires that \(k_{2} \geq p\) : the number of excluded exogenous variables must be at least as great as the number of endogenous regressors.

In the following formulas, if weights are specified, \(\mathbf{X}_{1}^{\prime} \mathbf{X}_{1}, \mathbf{X}^{\prime} \mathbf{X}, \mathbf{X}^{\prime} \mathbf{y}, \mathbf{y}^{\prime} \mathbf{y}, \mathbf{Z}^{\prime} \mathbf{Z}, \mathbf{Z}^{\prime} \mathbf{X}\), and \(\mathbf{Z}^{\prime} \mathbf{y}\) are replaced with \(\mathbf{X}_{1}^{\prime} \mathbf{D} \mathbf{X}_{1}, \mathbf{X}^{\prime} \mathbf{D X}, \mathbf{X}^{\prime} \mathbf{D} \mathbf{y}, \mathbf{y}^{\prime} \mathbf{D} \mathbf{y}, \mathbf{Z}^{\prime} \mathbf{D Z}, \mathbf{Z}^{\prime} \mathbf{D} \mathbf{X}\), and \(\mathbf{Z}^{\prime} \mathbf{D y}\), respectively. We suppress the \(\mathbf{D}\) below to simplify the notation.

\section*{2SLS and LIML estimators}

Define the \(\kappa\)-class estimator of \(\boldsymbol{\beta}\) as
\[
\mathbf{b}=\left\{\mathbf{X}^{\prime}\left(\mathbf{I}-\kappa \mathbf{M}_{\mathbf{Z}}\right) \mathbf{X}\right\}^{-1} \mathbf{X}^{\prime}\left(\mathbf{I}-\kappa \mathbf{M}_{\mathbf{Z}}\right) \mathbf{y}
\]
where \(\mathbf{M}_{\mathbf{Z}}=\mathbf{I}-\mathbf{Z}\left(\mathbf{Z}^{\prime} \mathbf{Z}\right)^{-1} \mathbf{Z}^{\prime}\). The 2SLS estimator results from setting \(\kappa=1\). The LIML estimator results from selecting \(\kappa\) to be the minimum eigenvalue of \(\left(\mathbf{Y}^{\prime} \mathbf{M}_{\mathbf{Z}} \mathbf{Y}\right)^{-1 / 2} \mathbf{Y}^{\prime} \mathbf{M}_{\mathbf{X}_{1}} \mathbf{Y}\left(\mathbf{Y}^{\prime} \mathbf{M}_{\mathbf{Z}} \mathbf{Y}\right)^{-1 / 2}\), where \(\mathbf{M}_{\mathbf{X}_{1}}=\mathbf{I}-\mathbf{X}_{1}\left(\mathbf{X}_{1}^{\prime} \mathbf{X}_{1}\right)^{-1} \mathbf{X}_{1}^{\prime}\).

The total sum of squares (TSS) equals \(\mathbf{y}^{\prime} \mathbf{y}\) if there is no intercept and \(\mathbf{y}^{\prime} \mathbf{y}-\left\{\left(\mathbf{1}^{\prime} \mathbf{y}\right)^{2} / n\right\}\) otherwise. The degrees of freedom is \(n-c\). The error sum of squares (ESS) is defined as \(\mathbf{y}^{\prime} \mathbf{y}-2 \mathbf{b} \mathbf{X}^{\prime} \mathbf{y}+\mathbf{b}^{\prime} \mathbf{X}^{\prime} \mathbf{X} \mathbf{b}\). The model sum of squares (MSS) equals TSS - ESS. The degrees of freedom is \(k-c\).

The mean squared error, \(s^{2}\), is defined as ESS \(/(n-k)\) if small is specified and ESS \(/ n\) otherwise. The root mean squared error is \(s\), its square root.

If \(c=1\) and small is not specified, a Wald statistic, \(W\), of the joint significance of the \(k-1\) parameters of \(\boldsymbol{\beta}\) except the constant term is calculated; \(W \sim \chi^{2}(k-1)\). If \(c=1\) and small is specified, then an \(F\) statistic is calculated as \(F=W /(k-1) ; F \sim F(k-1, n-k)\).

The \(R^{2}\) is defined as \(R^{2}=1-\mathrm{ESS} / \mathrm{TSS}\).
The adjusted \(R^{2}\) is \(R_{\mathrm{a}}^{2}=1-\left(1-R^{2}\right)(n-c) /(n-k)\).
The unadjusted (default) variance estimate is \(\operatorname{Var}(\mathbf{b})=s^{2}\left\{\mathbf{X}^{\prime}\left(\mathbf{I}-\kappa \mathbf{M}_{\mathbf{Z}}\right) \mathbf{X}\right\}^{-1}\).
For a general discussion of robust variance estimates in regression, see A general notation for the robust variance calculation in \([\mathrm{R}]\) regress. ivregress uses the same definitions for terms discussed in Robust calculation for regress in its robust variance calculation, except for the following.

The vector of scores is given by
\[
\mathbf{u}_{j}=\left(y_{j}-\mathbf{x}_{j} \mathbf{b}\right) \widehat{\mathbf{x}}_{j}
\]
where \(\widehat{\mathbf{x}}_{j}^{\prime}=\mathbf{P z}_{j}{ }^{\prime}\) and \(\mathbf{P}=\left(\mathbf{X}^{\prime} \mathbf{Z}\right)\left(\mathbf{Z}^{\prime} \mathbf{Z}\right)^{-1}\). When the formulas in \([\mathrm{R}]\) regress are applied, \(q_{c}\) is given by its regressionlike definition. If small is not specified, then \(k=0\) in the formulas given in [R] regress.
ivregress 2sls and ivregress liml also support estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

\section*{GMM estimator}

We obtain an initial consistent estimate of \(\boldsymbol{\beta}\) by using the 2SLS estimator; see above. Using this estimate of \(\boldsymbol{\beta}\), we compute the weighting matrix \(\mathbf{W}\) and calculate the GMM estimator
\[
\mathbf{b}_{\mathrm{GMM}}=\left\{\mathbf{X}^{\prime} \mathbf{Z W} \mathbf{Z}^{\prime} \mathbf{X}\right\}^{-1} \mathbf{X}^{\prime} \mathbf{Z} \mathbf{W} \mathbf{Z}^{\prime} \mathbf{y}
\]

The variance of \(\mathbf{b}_{\mathrm{GMM}}\) is
\[
\operatorname{Var}\left(\mathbf{b}_{\mathrm{GMM}}\right)=n\left\{\mathbf{X}^{\prime} \mathbf{Z} \mathbf{W} \mathbf{Z}^{\prime} \mathbf{X}\right\}^{-1} \mathbf{X}^{\prime} \mathbf{Z} \mathbf{W} \widehat{\mathbf{S}} \mathbf{W} \mathbf{Z}^{\prime} \mathbf{X}\left\{\mathbf{X}^{\prime} \mathbf{Z} \mathbf{W} \mathbf{Z}^{\prime} \mathbf{X}\right\}^{-1}
\]
\(\operatorname{Var}\left(\mathbf{b}_{\mathrm{GMM}}\right)\) is of the sandwich form \(\mathbf{D M D}\); see \([\mathrm{P}]\) _robust. If the user specifies the small option, ivregress implements a small-sample adjustment by multiplying the VCE by \(N /(N-k)\).

If vce(unadjusted) is specified, then we set \(\widehat{\mathbf{S}}=\mathbf{W}^{-1}\) and the VCE reduces to the "optimal" GMM variance estimator
\[
\operatorname{Var}\left(\boldsymbol{\beta}_{\mathrm{GMM}}\right)=n\left\{\mathbf{X}^{\prime} \mathbf{Z} \mathbf{W} \mathbf{Z}^{\prime} \mathbf{X}\right\}^{-1}
\]

However, if \(\mathbf{W}^{-1}\) is not a good estimator of \(E\left(\mathbf{z}_{i} u_{i} u_{i} \mathbf{z}_{i}^{\prime}\right)\), then the optimal GMM estimator is inefficient, and inference based on the optimal variance estimator could be misleading.
\(\mathbf{W}\) is calculated using the residuals from the initial 2SLS estimates, whereas \(\mathbf{S}\) is estimated using the residuals based on \(\mathbf{b}_{\mathrm{GMM}}\). The wmatrix () option affects the form of \(\mathbf{W}\), whereas the vce() option affects the form of \(\mathbf{S}\). Except for different residuals being used, the formulas for \(\mathbf{W}^{-1}\) and \(\mathbf{S}\) are identical, so we focus on estimating \(\mathbf{W}^{-1}\).

If wmatrix(unadjusted) is specified, then
\[
\mathbf{W}^{-1}=\frac{s^{2}}{n} \sum_{i} \mathbf{z}_{i} \mathbf{z}_{i}^{\prime}
\]
where \(s^{2}=\sum_{i} u_{i}^{2} / n\). This weight matrix is appropriate if the errors are homoskedastic.
If wmatrix(robust) is specified, then
\[
\mathbf{W}^{-1}=\frac{1}{n} \sum_{i} u_{i}^{2} \mathbf{z}_{i} \mathbf{z}_{i}^{\prime}
\]
which is appropriate if the errors are heteroskedastic.
If wmatrix(cluster clustvar) is specified, then
\[
\mathbf{W}^{-1}=\frac{1}{n} \sum_{c} \mathbf{q}_{c} \mathbf{q}_{c}^{\prime}
\]
where \(c\) indexes clusters,
\[
\mathbf{q}_{c}=\sum_{i \in c_{j}} u_{i} \mathbf{z}_{i}
\]
and \(c_{j}\) denotes the \(j\) th cluster.
If wmatrix (hac kernel [\#]) is specified, then
\[
\mathbf{W}^{-1}=\frac{1}{n} \sum_{i} u_{i}^{2} \mathbf{z}_{i} \mathbf{z}_{i}^{\prime}+\frac{1}{n} \sum_{l=1}^{l=n-1} \sum_{i=l+1}^{i=n} K(l, m) u_{i} u_{i-l}\left(\mathbf{z}_{i} \mathbf{z}_{i-l}^{\prime}+\mathbf{z}_{i-l} \mathbf{z}_{i}^{\prime}\right)
\]
where \(m=\) \# if \# is specified and \(m=n-2\) otherwise. Define \(z=l /(m+1)\). If kernel is nwest, then
\[
K(l, m)= \begin{cases}1-z & 0 \leq z \leq 1 \\ 0 & \text { otherwise }\end{cases}
\]

If kernel is gallant, then
\[
K(l, m)= \begin{cases}1-6 z^{2}+6 z^{3} & 0 \leq z \leq 0.5 \\ 2(1-z)^{3} & 0.5<z \leq 1 \\ 0 & \text { otherwise }\end{cases}
\]

If kernel is quadraticspectral, then
\[
K(l, m)= \begin{cases}1 & z=0 \\ 3\{\sin (\theta) / \theta-\cos (\theta)\} / \theta^{2} & \text { otherwise }\end{cases}
\]
where \(\theta=6 \pi z / 5\).
If wmatrix (hac kernel opt) is specified, then ivregress uses Newey and West's (1994) automatic lag-selection algorithm, which proceeds as follows. Define \(\mathbf{h}\) to be a \(\left(k_{1}+k_{2}\right) \times 1\) vector containing ones in all rows except for the row corresponding to the constant term (if present); that row contains a zero. Define
\[
\begin{aligned}
f_{i} & =\left(u_{i} \mathbf{z}_{i}\right) \mathbf{h} \\
\widehat{\sigma}_{j} & =\frac{1}{n} \sum_{i=j+1}^{n} f_{i} f_{i-j} \quad j=0, \ldots, m^{*} \\
\widehat{s}^{(q)} & =2 \sum_{j=1}^{m^{*}} \widehat{\sigma}_{j} j^{q} \\
\widehat{s}^{(0)} & =\widehat{\sigma}_{0}+2 \sum_{j=1}^{m^{*}} \widehat{\sigma}_{j} \\
\widehat{\gamma} & =c_{\gamma}\left\{\left(\frac{\widehat{s}^{(q)}}{\widehat{s}^{(0)}}\right)^{2}\right\}^{1 / 2 q+1} \\
m & =\widehat{\gamma} n^{1 /(2 q+1)}
\end{aligned}
\]
where \(q, m^{*}\), and \(c_{\gamma}\) depend on the kernel specified:
\begin{tabular}{cccc} 
Kernel & \(q\) & \multicolumn{1}{c}{\(m^{*}\)} & \(c_{\gamma}\) \\
\hline Bartlett & 1 & int \(\left\{20(T / 100)^{2 / 9}\right\}\) & 1.1447 \\
Parzen & 2 & int \(\left\{20(T / 100)^{4 / 25}\right\}\) & 2.6614 \\
Quadratic spectral & 2 & \(\operatorname{int}\left\{20(T / 100)^{2 / 25}\right\}\) & 1.3221
\end{tabular}
where \(\operatorname{int}(x)\) denotes the integer obtained by truncating \(x\) toward zero. For the Bartlett and Parzen kernels, the optimal lag is \(\min \left\{\operatorname{int}(m), m^{*}\right\}\). For the quadratic spectral, the optimal lag is \(\min \left\{m, m^{*}\right\}\).

If wmatrix (hac kernel opt \#) is specified, then ivregress uses \# instead of 20 in the definition of \(m^{*}\) above to select the optimal lag.

If center is specified, when computing weighting matrices ivregress replaces the term \(u_{i} z_{i}\) in the formulas above with \(u_{i} \mathbf{z}_{i}-\overline{u \mathbf{z}}\), where \(\overline{u \mathbf{z}}=\sum_{i} u_{i} \mathbf{z}_{i} / N\).

\section*{References}

Anatolyev, S., and A. Skolkova. 2019. Many instruments: Implementation in Stata. Stata Journal 19: 849-866.
Andrews, D. W. K. 1991. Heteroskedasticity and autocorrelation consistent covariance matrix estimation. Econometrica 59: 817-858. https://doi.org/10.2307/2938229.
Angrist, J. D., and J.-S. Pischke. 2009. Mostly Harmless Econometrics: An Empiricist's Companion. Princeton, NJ: Princeton University Press.

Basmann, R. L. 1957. A generalized classical method of linear estimation of coefficients in a structural equation. Econometrica 25: 77-83. https://doi.org/10.2307/1907743.

Bauldry, S. 2014. miivfind: A command for identifying model-implied instrumental variables for structural equation models in Stata. Stata Journal 14: 60-75.

Baum, C. F. 2006. An Introduction to Modern Econometrics Using Stata. College Station, TX: Stata Press.
Baum, C. F., and A. Lewbel. 2019. Advice on using heteroskedasticity-based identification. Stata Journal 19: 757-767.
Baum, C. F., M. E. Schaffer, and S. Stillman. 2003. Instrumental variables and GMM: Estimation and testing. Stata Journal 3: 1-31.
-_ 2007. Enhanced routines for instrumental variables/generalized method of moments estimation and testing. Stata Journal 7: 465-506.

Benson, D., M. A. Masten, and A. Torgovitsky. 2022. ivcre: An instrumental-variables estimator for the correlated random-coefficients model. Stata Journal 22: 469-495.
Cameron, A. C., and P. K. Trivedi. 2005. Microeconometrics: Methods and Applications. New York: Cambridge University Press.
——. 2022. Microeconometrics Using Stata. 2nd ed. College Station, TX: Stata Press.
Choi, J., and S. Shen. 2019. Two-sample instrumental-variables regression with potentially weak instruments. Stata Journal 19: 581-597.

Clarke, D., and B. Matta. 2018. Practical considerations for questionable IVs. Stata Journal 18: 663-691.
Davidson, R., and J. G. MacKinnon. 1993. Estimation and Inference in Econometrics. New York: Oxford University Press.

Deb, P., E. C. Norton, and W. G. Manning. 2017. Health Econometrics Using Stata. College Station, TX: Stata Press. Desbordes, R., and V. Verardi. 2012. A robust instrumental-variables estimator. Stata Journal 12: 169-181.

D'Haultfæuille, X., A. Maurel, X. Qiu, and Y. Zhang. 2020. Estimating selection models without an instrument with Stata. Stata Journal 20: 297-308.

Dippel, C., A. Ferrara, and S. Heblich. 2020. Causal mediation analysis in instrumental-variables regressions. Stata Journal 20: 613-626.

Du, K., Y. Zhang, and Q. Zhou. 2020. Fitting partially linear functional-coefficient panel-data models with Stata. Stata Journal 20: 976-998.

Finlay, K., and L. M. Magnusson. 2009. Implementing weak-instrument robust tests for a general class of instrumentalvariables models. Stata Journal 9: 398-421.
Gallant, A. R. 1987. Nonlinear Statistical Models. New York: Wiley.
Greene, W. H. 2018. Econometric Analysis. 8th ed. New York: Pearson.
Hall, A. R. 2005. Generalized Method of Moments. Oxford: Oxford University Press.
Hansen, L. P. 1982. Large sample properties of generalized method of moments estimators. Econometrica 50 : 1029-1054. https://doi.org/10.2307/1912775.
Hayashi, F. 2000. Econometrics. Princeton, NJ: Princeton University Press.
Judge, G. G., W. E. Griffiths, R. C. Hill, H. Lütkepohl, and T.-C. Lee. 1985. The Theory and Practice of Econometrics. 2nd ed. New York: Wiley.

Kaplan, D. M. 2022. Smoothed instrumental variables quantile regression. Stata Journal 22: 379-403.
Kmenta, J. 1997. Elements of Econometrics. 2nd ed. Ann Arbor: University of Michigan Press.
Koopmans, T. C., and W. C. Hood. 1953. Studies in Econometric Method. New York: Wiley.

Koopmans, T. C., and J. Marschak. 1950. Statistical Inference in Dynamic Economic Models. New York: Wiley.
Kripfganz, S., and V. Sarafidis. 2021. Instrumental-variable estimation of large-T panel-data models with common factors. Stata Journal 21: 659-686.
Newey, W. K., and K. D. West. 1987. A simple, positive semi-definite, heteroskedasticity and autocorrelation consistent covariance matrix. Econometrica 55: 703-708. https://doi.org/10.2307/1913610.
1994. Automatic lag selection in covariance matrix estimation. Review of Economic Studies 61: 631-653. https://doi.org/10.2307/2297912.
Nichols, A. 2007. Causal inference with observational data. Stata Journal 7: 507-541.
Palmer, T. M., V. Didelez, R. R. Ramsahai, and N. A. Sheehan. 2011. Nonparametric bounds for the causal effect in a binary instrumental-variable model. Stata Journal 11: 345-367.

Pinzon, E. 2016. Estimation under omitted confounders, endogeneity, omitted variable bias, and related problems. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2017/02/20/estimation-under-omitted-confounders-endogeneity-omitted-variable-bias-and-related-problems/.

Poi, B. P. 2006. Jackknife instrumental variables estimation in Stata. Stata Journal 6: 364-376.
Stock, J. H., and M. W. Watson. 2019. Introduction to Econometrics. 4th ed. New York: Pearson.
Stock, J. H., J. H. Wright, and M. Yogo. 2002. A survey of weak instruments and weak identification in generalized method of moments. Journal of Business and Economic Statistics 20: 518-529. https://doi.org/10.1198/073500102288618658.

Sun, L. 2018. Implementing valid two-step identification-robust confidence sets for linear instrumental-variables models. Stata Journal 18: 803-825.

Theil, H. 1953. Repeated Least Squares Applied to Complete Equation Systems. Mimeograph from the Central Planning Bureau, The Hague.

Wang, Q., and N. Wu. 2012. Long-run covariance and its applications in cointegration regression. Stata Journal 12: 515-542.

Wooldridge, J. M. 2010. Econometric Analysis of Cross Section and Panel Data. 2nd ed. Cambridge, MA: MIT Press. -_. 2020. Introductory Econometrics: A Modern Approach. 7th ed. Boston: Cengage.
Wright, P. G. 1928. The Tariff on Animal and Vegetable Oils. New York: Macmillan.

\section*{Also see}
\([\mathrm{R}]\) ivregress postestimation - Postestimation tools for ivregress \({ }^{+}\)
[R] gmm - Generalized method of moments estimation
[R] ivprobit - Probit model with continuous endogenous covariates
[R] ivqregress - Instrumental-variables quantile regression
[R] ivtobit - Tobit model with continuous endogenous covariates
[R] reg3 - Three-stage estimation for systems of simultaneous equations
\([R]\) regress - Linear regression
[ERM] eregress - Extended linear regression
[FMM] fmm: ivregress - Finite mixtures of linear regression models with endogenous covariates
[SEM] Intro 5 - Tour of models
[SP] spivregress - Spatial autoregressive models with endogenous covariates
[SVY] svy estimation - Estimation commands for survey data
[TS] forecast - Econometric model forecasting
[XT] xtivreg - Instrumental variables and two-stage least squares for panel-data models
[U] 20 Estimation and postestimation commands

\section*{Title}
ivregress postestimation - Postestimation tools for ivregress \({ }^{+}\)
\({ }^{+}\)This command includes features that are part of StataNow.
\begin{tabular}{llll} 
Postestimation commands & predict & margins & estat \\
Remarks and examples & Stored results & Methods and formulas & Acknowledgments \\
References & Also see & &
\end{tabular}

\section*{Postestimation commands}

The following postestimation commands are of special interest after ivregress:
Command Description
estat endogenous perform tests of endogeneity
estat firststage report "first-stage" regression statistics
estat overid perform tests of overidentifying restrictions
+estat weakrobust perform inference robust to weak instruments
*estat sbknown perform tests for a structural break with a known break date
*estat single perform tests for a structural break with an unknown break date
These commands are not appropriate with svy estimation results.
\({ }^{+}\)This command is part of StataNow.
*estat sbknown and estat sbsingle work only after ivregress 2 sls.

The following postestimation commands are also available:
\begin{tabular}{|c|c|}
\hline Command & Description \\
\hline contrast & contrasts and ANOVA-style joint tests of estimates \\
\hline estat summarize & summary statistics for the estimation sample \\
\hline estat vce & variance-covariance matrix of the estimators (VCE) \\
\hline estat (svy) & postestimation statistics for survey data \\
\hline estimates & cataloging estimation results \\
\hline etable & table of estimation results \\
\hline \(\dagger\) forecast & dynamic forecasts and simulations \\
\hline \(\dagger\) hausman & Hausman's specification test \\
\hline lincom & point estimates, standard errors, testing, and inference for linear combinations of coefficients \\
\hline margins & marginal means, predictive margins, marginal effects, and average marginal effects \\
\hline marginsplot & graph the results from margins (profile plots, interaction plots, etc.) \\
\hline nlcom & point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients \\
\hline predict & linear predictions and their SEs, probabilities, etc. \\
\hline
\end{tabular}
predictnl
pwcompare
test
testnl
point estimates, standard errors, testing, and inference for generalized predictions pairwise comparisons of estimates
Wald tests of simple and composite linear hypotheses
Wald tests of nonlinear hypotheses
\(\dagger_{\text {forecast }}\) and hausman are not appropriate with svy estimation results.

\section*{predict}

\section*{Description for predict}
predict creates a new variable containing predictions such as linear predictions, residuals, standard errors, probabilities, and expected values.

\section*{Menu for predict}

Statistics \(>\) Postestimation

\section*{Syntax for predict}
\[
\begin{aligned}
& \text { predict }[\text { type }] \text { newvar }[\text { if }][\text { in }][\text {, statistic }] \\
& \text { predict }[\text { type }] \text { stub* }[\text { if }][\text { in }], \text { scores }
\end{aligned}
\]
statistic Description

\section*{Main}
\(\mathrm{xb} \quad\) linear prediction; the default
residuals residuals
stdp standard error of the prediction
stdf standard error of the forecast
\(\operatorname{pr}(a, b) \quad \operatorname{Pr}\left(a<y_{j}<b\right)\) under exogeneity and normal errors
\(\mathrm{e}(a, b) \quad E\left(y_{j} \mid a<y_{j}<b\right)\) under exogeneity and normal errors
ystar \((a, b) \quad E\left(y_{j}^{*}\right), y_{j}^{*}=\max \left\{a, \min \left(y_{j}, b\right)\right\}\) under exogeneity and normal errors
These statistics are available both in and out of sample; type predict ... if e(sample)... if wanted only for the estimation sample.
stdf is not allowed with svy estimation results.
where \(a\) and \(b\) may be numbers or variables; \(a\) missing ( \(a \geq\).) means \(-\infty\), and \(b\) missing ( \(b \geq\).) means \(+\infty\); see [U] 12.2.1 Missing values.

\section*{Options for predict}
\(\qquad\) Main
xb , the default, calculates the linear prediction.
residuals calculates the residuals, that is, \(y_{j}-\mathbf{x}_{j} \mathbf{b}\). These are based on the estimated equation when the observed values of the endogenous variables are used-not the projections of the instruments onto the endogenous variables.
stdp calculates the standard error of the prediction, which can be thought of as the standard error of the predicted expected value or mean for the observation's covariate pattern. This is also referred to as the standard error of the fitted value.
stdf calculates the standard error of the forecast, which is the standard error of the point prediction for 1 observation. It is commonly referred to as the standard error of the future or forecast value. By construction, the standard errors produced by stdf are always larger than those produced by stdp; see Methods and formulas in [R] regress postestimation.
\(\operatorname{pr}(a, b)\) calculates \(\operatorname{Pr}\left(a<\mathbf{x}_{j} \mathbf{b}+u_{j}<b\right)\), the probability that \(y_{j} \mid \mathbf{x}_{j}\) would be observed in the interval \((a, b)\) under exogeneity and assuming errors are normally distributed.
\(a\) and \(b\) may be specified as numbers or variable names; \(l b\) and \(u b\) are variable names;
\(\operatorname{pr}(20,30)\) calculates \(\operatorname{Pr}\left(20<\mathbf{x}_{j} \mathbf{b}+u_{j}<30\right)\);
\(\operatorname{pr}(l b, u b)\) calculates \(\operatorname{Pr}\left(l b<\mathbf{x}_{j} \mathbf{b}+u_{j}<u b\right)\); and
\(\operatorname{pr}(20, u b)\) calculates \(\operatorname{Pr}\left(20<\mathbf{x}_{j} \mathbf{b}+u_{j}<u b\right)\).
\(a\) missing ( \(a \geq\). ) means \(-\infty\); \(\operatorname{pr}(., 30)\) calculates \(\operatorname{Pr}\left(-\infty<\mathbf{x}_{j} \mathbf{b}+u_{j}<30\right)\);
\(\operatorname{pr}(l b, 30)\) calculates \(\operatorname{Pr}\left(-\infty<\mathbf{x}_{j} \mathbf{b}+u_{j}<30\right)\) in observations for which \(l b \geq\).
and calculates \(\operatorname{Pr}\left(l b<\mathbf{x}_{j} \mathbf{b}+u_{j}<30\right)\) elsewhere.
\(b\) missing ( \(b \geq\). ) means \(+\infty\); \(\operatorname{pr}(20,\).\() calculates \operatorname{Pr}\left(+\infty>\mathbf{x}_{j} \mathbf{b}+u_{j}>20\right)\);
\(\operatorname{pr}(20, u b)\) calculates \(\operatorname{Pr}\left(+\infty>\mathbf{x}_{j} \mathbf{b}+u_{j}>20\right)\) in observations for which \(u b \geq\).
and calculates \(\operatorname{Pr}\left(20<\mathbf{x}_{j} \mathbf{b}+u_{j}<u b\right)\) elsewhere.
\(\mathrm{e}(a, b)\) calculates \(E\left(\mathbf{x}_{j} \mathbf{b}+u_{j} \mid a<\mathbf{x}_{j} \mathbf{b}+u_{j}<b\right)\), the expected value of \(y_{j} \mid \mathbf{x}_{j}\) conditional on \(y_{j} \mid \mathbf{x}_{j}\) being in the interval \((a, b)\), meaning that \(y_{j} \mid \mathbf{x}_{j}\) is truncated. \(a\) and \(b\) are specified as they are for pr() . Exogeneity and normally distributed errors are assumed.
ystar \((a, b)\) calculates \(E\left(y_{j}^{*}\right)\), where \(y_{j}^{*}=a\) if \(\mathbf{x}_{j} \mathbf{b}+u_{j} \leq a, y_{j}^{*}=b\) if \(\mathbf{x}_{j} \mathbf{b}+u_{j} \geq b\), and \(y_{j}^{*}=\mathbf{x}_{j} \mathbf{b}+u_{j}\) otherwise, meaning that \(y_{j}^{*}\) is censored. \(a\) and \(b\) are specified as they are for pr() . Exogeneity and normally distributed errors are assumed.
scores calculates the scores for the model. A new score variable is created for each endogenous regressor, as well as an equation-level score that applies to all exogenous variables and constant term (if present).

\section*{margins}

\section*{Description for margins}
margins estimates margins of response for linear predictions, probabilities, and expected values.

\section*{Menu for margins}

Statistics \(>\) Postestimation

\section*{Syntax for margins}
```

margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...)...] [options]

```
statistic Description
\(\mathrm{xb} \quad\) linear prediction; the default
\(\operatorname{pr}(a, b) \quad \operatorname{Pr}\left(a<y_{j}<b\right)\) under exogeneity and normal errors
\(\mathrm{e}(a, b) \quad E\left(y_{j} \mid a<y_{j}<b\right)\) under exogeneity and normal errors
\(\operatorname{ystar}(a, b) \quad E\left(y_{j}^{*}\right), y_{j}^{*}=\max \left\{a, \min \left(y_{j}, b\right)\right\}\) under exogeneity and normal errors
stdp not allowed with margins
stdf not allowed with margins
residuals not allowed with margins

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see \([R]\) margins.

\section*{estat}

\section*{Description for estat}
estat endogenous performs tests to determine whether endogenous regressors in the model are in fact exogenous. After GMM estimation, the \(C\) (difference-in-Sargan) statistic is reported. After 2SLS estimation with an unadjusted VCE, the Durbin (1954) and Wu-Hausman (Wu 1974; Hausman 1978) statistics are reported. After 2SLS estimation with a robust VCE, Wooldridge's (1995) robust score test and a robust regression-based test are reported. In all cases, if the test statistic is significant, then the variables being tested must be treated as endogenous. estat endogenous is not available after LIML estimation.
estat firststage reports various statistics that measure the relevance of the excluded exogenous variables. By default, which statistics are reported depends on whether the equation has one or more than one endogenous regressor.
estat overid performs tests of overidentifying restrictions. If the 2SLS estimator was used, Sargan's (1958) and Basmann's (1960) \(\chi^{2}\) tests are reported, as is Wooldridge's (1995) robust score test; if the LIML estimator was used, Anderson and Rubin's (1950) \(\chi^{2}\) test and Basmann's \(F\) test are reported; and if the GMM estimator was used, Hansen's (1982) \(J\) statistic \(\chi^{2}\) test is reported. In all of these cases, a rejection of the test indicates that the instruments may not be valid.
estat weakrobust is part of StataNow. It performs hypothesis tests on the coefficients of endogenous regressors that are robust to weak instruments. For just-identified models, the AndersonRubin (1949) test is reported. For overidentified models, if a homoskedastic VCE was used in estimation, the conditional likelihood-ratio (CLR) test of Moreira (2003) is reported. If a heteroskedastic or clusterrobust VCE was used instead, an appropriate generalization of the CLR test is used (see Finlay and Magnusson [2009]). When there is only one endogenous regressor, the confidence intervals associated with any of these tests can be requested.

\section*{Menu for estat}

Statistics \(>\) Postestimation

\section*{Syntax for estat}

Perform tests of endogeneity
estat endogenous [varlist] [, lags(\#) forceweights forcenonrobust]

Report "first-stage" regression statistics
```

estat firststage [, all forcenonrobust]

```

\section*{Perform tests of overidentifying restrictions}
```

estat overid [, lags(\#) forceweights forcenonrobust]

```

Perform inference robust to weak instruments (StataNow)
```

estat weakrobust [, weak_options]

```
weak_options
Description
betanull(null_spec)
ar
clr
ci
level(\#)
reps(\#)
rseed(\#)
bound \(\left(\#_{\text {min }} \#_{\text {max }}\right)\)
ngrid(\#)
noadaptgrid
[no]log
iterate(\#)
specify the null hypothesis on the endogenous regressors
perform Anderson-Rubin test
perform conditional likelihood-ratio test
report confidence interval instead of test
set confidence level; default is level (95)
perform \# replications for simulations; default is reps (25000)
set random-number seed for simulations to \#
specify bounds of confidence interval grid
use \# grid points for the confidence interval; default is ngrid(500)
suppress adaptive grid expansion
suppress or display the iteration \(\log\)
perform maximum of \# adaptive iterations; default is iterate(20)
collect is allowed with estat endogenous, estat firststage, estat overid, and estat weakrobust; see [U] 11.1.10 Prefix commands.

\section*{Options for estat}

Options for estat are presented under the following headings:
Options for estat endogenous
Options for estat firststage
Options for estat overid Options for estat weakrobust (StataNow)

\section*{Options for estat endogenous}
lags (\#) specifies the number of lags to use for prewhitening when computing the heteroskedasticityand autocorrelation-consistent (HAC) version of the score test of endogeneity. Specifying lags (0) requests no prewhitening. This option is valid only when the model was fit via 2SLS and an HAC covariance matrix was requested when the model was fit. The default is lags(1).
forceweights requests that the tests of endogeneity be computed even though aweights, pweights, or iweights were used in the previous estimation. By default, these tests are conducted only after unweighted or frequency-weighted estimation. The reported critical values may be inappropriate for weighted data, so the user must determine whether the critical values are appropriate for a given application.
forcenonrobust requests that the Durbin and Wu-Hausman tests be performed after 2SLS estimation even though a robust VCE was used at estimation time. This option is available only if the model was fit by 2SLS.

\section*{Options for estat firststage}
all requests that all first-stage goodness-of-fit statistics be reported regardless of whether the model contains one or more endogenous regressors. By default, if the model contains one endogenous regressor, then the first-stage \(R^{2}\), adjusted \(R^{2}\), partial \(R^{2}\), and \(F\) statistics are reported, whereas if the model contains multiple endogenous regressors, then Shea's partial \(R^{2}\) and adjusted partial \(R^{2}\) are reported instead.
forcenonrobust requests that the minimum eigenvalue statistic and its critical values be reported even though a robust VCE was used at estimation time. The reported critical values assume that the errors are independent and identically distributed (i.i.d.) normal, so the user must determine whether the critical values are appropriate for a given application.

\section*{Options for estat overid}
lags(\#) specifies the number of lags to use for prewhitening when computing the heteroskedasticityand autocorrelation-consistent (HAC) version of the score test of overidentifying restrictions. Specifying lags ( 0 ) requests no prewhitening. This option is valid only when the model was fit via 2SLS and an HAC covariance matrix was requested when the model was fit. The default is lags(1).
forceweights requests that the tests of overidentifying restrictions be computed even though aweights, pweights, or iweights were used in the previous estimation. By default, these tests are conducted only after unweighted or frequency-weighted estimation. The reported critical values may be inappropriate for weighted data, so the user must determine whether the critical values are appropriate for a given application.
forcenonrobust requests that the Sargan and Basmann tests of overidentifying restrictions be performed after 2SLS or LIML estimation even though a robust VCE was used at estimation time. These tests assume that the errors are i.i.d. normal, so the user must determine whether the critical values are appropriate for a given application.

\section*{Options for estat weakrobust (StataNow)}
betanull (null_spec) specifies the null hypothesis of the test on the coefficients of the endogenous regressors. By default, the null hypothesis is that the coefficients of all endogenous regressors are zero. The null hypothesis can be specified using numlist, a list of equalities, or a matrix. When using numlist, you must specify the null values for all endogenous regressors in the order given in the estimation command and specify the copy suboption if there is more than one endogenous regressor. When you use a list of equalities or a matrix, null values are set to zero for endogenous regressors you did not specify. When you use a matrix, null values are assigned to coefficients by matrix column names.
null_spec is one of
\# [\# ..., copy \(]\)
varname \(=\#[\) varname \(=\#[\ldots]][\), skip \(]\)
matname [, skip copy]
skip specifies that any variables that are not endogenous regressors in the model be ignored.
The default action is to issue an error message.
copy specifies that the list of null values be assigned into the null hypothesis by position rather than by name.
ar requests that an Anderson-Rubin test be performed. The associated confidence interval of the test can be requested with option ci. By default, the Anderson-Rubin test is performed when the model is just identified. If both ar and clr are specified, both tests are performed.
clr requests that a CLR test be performed. The associated confidence interval of the test can be requested with option ci. By default, a CLR test is performed when the model is overidentified. If both ar and clr are specified, both tests are performed.
ci requests that a confidence interval be returned instead of a test statistic and a \(p\)-value. This applies to all tests specified by the user. Option ci is available only when there is a single endogenous regressor.
level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level(95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals.
reps (\#) specifies the number of replications to use in simulating critical values when \(p\)-values cannot be computed analytically. The default is reps (25000).
rseed (\#) sets the random-number seed. This option can be used to reproduce results for simulating critical values when \(p\)-values cannot be computed analytically. rseed (\#) is equivalent to typing set seed \# prior to calling estat weakrobust; see [R] set seed.
bound \(\left(\#_{\min } \#_{\max }\right)\) specifies the lower \(\left(\#_{\min }\right)\) and the upper bound \(\left(\#_{\max }\right)\) for the grid used to approximate confidence intervals when they cannot be computed analytically. The specified bounds will be the starting bounds for the adaptive grid expansion procedure or, if noadaptgrid is specified, the fixed bounds of the grid. By default, the lower bound is set to five standard errors below the coefficient estimated by ivregress, and the upper bound is five standard errors above.
ngrid(\#) specifies the starting number of gridpoints for approximating confidence intervals. The default is ngrid (500). Higher values will increase computation time but improve precision, which may be helpful when the confidence interval is wide. If noadaptgrid is specified, ngrid() specifies the fixed number of gridpoints.
noadaptgrid requests that the default method for computing confidence intervals, where the grid is adaptively expanded, be turned off and the interval be estimated with a fixed grid. noadaptgrid is equivalent to specifying iterate ( 0 ).
log and nolog specify whether to display the iteration log showing the progress of the adaptive gridding procedure. The iteration \(\log\) is displayed by default unless you used set iterlog off to suppress it; see set iterlog in [R] set iter.
iterate(\#) specifies the maximum number of iterations for the adaptive grid expansion procedure. Each iteration expands the grid by half the width of the starting bounds in each direction, and the procedure stops when the maximum number of iterations is reached or when the confidence interval has finite endpoints and was unchanged by the most recent expansion. The default is iterate(20).

\section*{Remarks and examples}

Remarks are presented under the following headings:
```

estat endogenous
estat firststage
estat overid
estat weakrobust (StataNow)

```

\section*{estat endogenous}

A natural question to ask is whether a variable presumed to be endogenous in the previously fit model could instead be treated as exogenous. If the endogenous regressors are in fact exogenous, then the OLS estimator is more efficient; and depending on the strength of the instruments and other factors, the sacrifice in efficiency by using an instrumental-variables estimator can be significant. Thus, unless an instrumental-variables estimator is really needed, olS should be used instead. estat endogenous provides several tests of endogeneity after 2SLS and GMM estimation.

\section*{\(>\) Example 1}

In example 1 of \([\mathrm{R}]\) ivregress, we fit a model of the average rental rate for housing in a state as a function of the percentage of the population living in urban areas and the average value of houses. We treated hsngval as endogenous because unanticipated shocks that affect rental rates probably affect house prices as well. We used family income and region dummies as additional instruments for hsngval. Here we test whether we could treat hsngval as exogenous.
```

. use https://www.stata-press.com/data/r18/hsng
(1980 Census housing data)
. ivregress 2sls rent pcturban (hsngval = faminc i.region)
(output omitted)
. estat endogenous
Tests of endogeneity
HO: Variables are exogenous
Durbin (score) chi2(1) = 12.8473 (p = 0.0003)
Wu-Hausman F(1,46) = 15.9067 (p = 0.0002)

```

Because we did not specify any variable names after the estat endogenous command, Stata by default tested all the endogenous regressors (namely, hsngval) in our model. The null hypothesis of the Durbin and \(\mathrm{Wu}-H a u s m a n\) tests is that the variable under consideration can be treated as exogenous. Here both test statistics are highly significant, so we reject the null of exogeneity; we must continue to treat hsngval as endogenous.

The difference between the Durbin and Wu-Hausman tests of endogeneity is that the former uses an estimate of the error term's variance based on the model assuming the variables being tested are exogenous, while the latter uses an estimate of the error variance based on the model assuming the variables being tested are endogenous. Under the null hypothesis that the variables being tested are exogenous, both estimates of the error variance are consistent. What we label the Wu-Hausman statistic is Wu's (1974) " \(T_{2}\) " statistic, which Hausman (1978) showed can be calculated very easily via linear regression. Baum, Schaffer, and Stillman \((2003,2007)\) provide a lucid discussion of these tests.

When you fit a model with multiple endogenous regressors, you can test the exogeneity of a subset of the regressors while continuing to treat the others as endogenous. For example, say you have three endogenous regressors, \(\mathrm{y} 1, \mathrm{y} 2\), and y 3 , and you fit your model by typing
```

. ivregress depvar ... (y1 y2 y3 = ...)

```

Suppose you are confident that y1 must be treated as endogenous, but you are undecided about y2 and y3. To test whether y2 and y3 can be treated as exogenous, you would type
- estat endogenous y2 y3

The Durbin and \(\mathrm{Wu}-H a u s m a n\) tests assume that the error term is i.i.d. Therefore, if you requested a robust VCE at estimation time, estat endogenous will instead report Wooldridge's (1995) score test and a regression-based test of exogeneity. Both these tests can tolerate heteroskedastic and autocorrelated errors, while only the regression-based test is amenable to clustering.
\(>\) Example 2
We refit our housing model, requesting robust standard errors, and then test the exogeneity of hsngval:
```

. use https://www.stata-press.com/data/r18/hsng
(1980 Census housing data)
. ivregress 2sls rent pcturban (hsngval = faminc i.region), vce(robust)
(output omitted)
. estat endogenous
Tests of endogeneity
HO: Variables are exogenous
Robust score chi2(1) = 2.10428 ( }\textrm{p}=0.1469\mathrm{ )
Robust regression F(1,46) = 4.31101 (p = 0.0435)

```

Wooldridge's score test does not reject the null hypothesis that hsngval is exogenous at conventional significance levels \((p=0.1469)\). However, the regression-based test does reject the null hypothesis at the 5\% significance level ( \(p=0.0435\) ). Typically, these two tests yield the same conclusion; the fact that our dataset has only 50 observations could be contributing to the discrepancy. Here we would be inclined to continue to treat hsngval as endogenous. Even if hsngval is exogenous, the 2SLS estimates are still consistent. On the other hand, if hsngval is in fact endogenous, the OLS estimates would not be consistent. Moreover, as we will see in our discussion of the estat overid command, our additional instruments may be invalid. To test whether an endogenous variable can be treated as exogenous, we must have a valid set of instruments to use to fit the model in the first place!

Unlike the Durbin and Wu-Hausman tests, Wooldridge's score and the regression-based tests do not allow you to test a subset of the endogenous regressors in the model; you can test only whether all the endogenous regressors are in fact exogenous.

After GMM estimation, estat endogenous calculates what Hayashi (2000, 220) calls the \(C\) statistic, also known as the difference-in-Sargan statistic. The \(C\) statistic can be made robust to heteroskedasticity, autocorrelation, and clustering; and the version reported by estat endogenous is determined by the weight matrix requested via the wmatrix() option used when fitting the model with ivregress. Additionally, the test can be used to determine the exogeneity of a subset of the endogenous regressors, regardless of the type of weight matrix used.

If you fit your model using the LIML estimator, you can use the hausman command to carry out a traditional Hausman (1978) test between the OLS and LIML estimates.

\section*{estat firststage}

For an excluded exogenous variable to be a valid instrument, it must be sufficiently correlated with the included endogenous regressors but uncorrelated with the error term. In recent decades, researchers have paid considerable attention to the issue of instruments that are only weakly correlated with the endogenous regressors. In such cases, the usual 2SLS, GMM, and LIML estimators are biased toward the OLS estimator, and inference based on the standard errors reported by, for example, ivregress can be severely misleading. For more information on the theory behind instrumental-variables estimation with weak instruments, see Nelson and Startz (1990); Staiger and Stock (1997); Hahn and Hausman (2003); the survey article by Stock, Wright, and Yogo (2002); and Angrist and Pischke (2009, chap. 4).

When the instruments are only weakly correlated with the endogenous regressors, some Monte Carlo evidence suggests that the LIML estimator performs better than the 2SLS and GMM estimators; see, for example, Poi (2006) and Stock, Wright, and Yogo (2002) (and the papers cited therein). On the other hand, the LIML estimator often results in confidence intervals that are somewhat larger than those from the 2SLS estimator.

Moreover, using more instruments is not a solution, because the biases of instrumental-variables estimators increase with the number of instruments. See Hahn and Hausman (2003).
estat firststage produces several statistics for judging the explanatory power of the instruments and is most easily explained with examples.

\section*{> Example 3}

Again building on the model fit in example 1 of \([R]\) ivregress, we now explore the degree of correlation between the additional instruments faminc, 2.region, 3.region, and 4.region and the endogenous regressor hsngval:
```

. use https://www.stata-press.com/data/r18/hsng
(1980 Census housing data)
. ivregress 2sls rent pcturban (hsngval = faminc i.region)
(output omitted)
estat firststage
First-stage regression summary statistics

```
\begin{tabular}{c|ccccc}
\hline Variable & R-sq. & \begin{tabular}{c} 
Adjusted \\
R-sq.
\end{tabular} & \begin{tabular}{c} 
Partial \\
R-sq.
\end{tabular} & \(F(4,44)\) & Prob \(>\) F \\
\hline hsngval & 0.6908 & 0.6557 & 0.5473 & 13.2978 & 0.0000 \\
\hline
\end{tabular}
Minimum eigenvalue statistic \(=13.2978\)
Critical Values \# of endogenous regressors: 1
HO: Instruments are weak \# of excluded instruments: 4
\begin{tabular}{|c|c|c|c|c|}
\hline 2SLS relative bias & \[
\begin{gathered}
5 \% \\
16.85
\end{gathered}
\] & \[
\begin{gathered}
10 \% \\
10.27
\end{gathered}
\] & \[
\begin{aligned}
& 20 \% \\
& 6.71
\end{aligned}
\] & \[
\begin{aligned}
& 30 \% \\
& 5.34
\end{aligned}
\] \\
\hline & 10\% & 15\% & 20\% & 25\% \\
\hline 2SLS size of nominal 5\% Wald test & 24.58 & 13.96 & 10.26 & 8.31 \\
\hline LIML size of nominal \(5 \%\) Wald test & 5.44 & 3.87 & 3.30 & 2.98 \\
\hline
\end{tabular}

To understand these results, recall that the first-stage regression is
hsngval \(_{i}=\pi_{0}+\pi_{1}\) pcturban \(_{i}+\pi_{2}\) faminc \(+\pi_{3}\) 2.region \(+\pi_{4}\) 3.region \(+\pi_{5}\) 4.region \(+v_{i}\)
where \(v_{i}\) is an error term. The column marked " R -sq." is the simple \(R^{2}\) from fitting the first-stage regression by OLS, and the column marked "Adjusted R-sq." is the adjusted \(R^{2}\) from that regression. Higher values purportedly indicate stronger instruments, and instrumental-variables estimators exhibit less bias when the instruments are strongly correlated with the endogenous variable.

Looking at just the \(R^{2}\) and adjusted \(R^{2}\) can be misleading, however. If hsngval were strongly correlated with the included exogenous variable pcturban but only weakly correlated with the additional instruments, then these statistics could be large even though a weak-instrument problem is present.

The partial \(R^{2}\) statistic measures the correlation between hsngval and the additional instruments after partialing out the effect of pcturban. Unlike the \(R^{2}\) and adjusted \(R^{2}\) statistics, the partial \(R^{2}\) statistic will not be inflated because of strong correlation between hsngval and pcturban. Bound, Jaeger, and Baker (1995) and others have promoted using this statistic.

The column marked " \(F(4,44)\) " is an \(F\) statistic for the joint significance of \(\pi_{2}, \pi_{3}, \pi_{4}\), and \(\pi_{5}\), the coefficients on the additional instruments. Its \(p\)-value is listed in the column marked "Prob \(>\mathrm{F}\) ". If the \(F\) statistic is not significant, then the additional instruments have no significant explanatory power for hsngval after controlling for the effect of pcturban. However, Hall, Rudebusch, and Wilcox (1996) used Monte Carlo simulation to show that simply having an \(F\) statistic that is significant at the typical \(5 \%\) or \(10 \%\) level is not sufficient. Stock, Wright, and Yogo (2002) suggest that the \(F\) statistic should exceed 10 for inference based on the 2SLS estimator to be reliable when there is one endogenous regressor.
estat firststage also presents the Cragg and Donald (1993) minimum eigenvalue statistic as a further test of weak instruments. Stock and Yogo (2005) discuss two characterizations of weak instruments: first, weak instruments cause instrumental-variables estimators to be biased; second, hypothesis tests of parameters estimated by instrumental-variables estimators may suffer from severe size distortions. The test statistic in our example is 13.30 , which is identical to the \(F\) statistic just discussed because our model contains one endogenous regressor.

The null hypothesis of each of Stock and Yogo's tests is that the set of instruments is weak. To perform these tests, we must first choose either the largest relative bias of the 2SLS estimator we are willing to tolerate or the largest rejection rate of a nominal \(5 \%\) Wald test we are willing to tolerate. If the test statistic exceeds the critical value, we can conclude that our instruments are not weak.

The row marked "2SLS relative bias" contains critical values for the test that the instruments are weak based on the bias of the 2SLS estimator relative to the bias of the OLS estimator. For example, from past experience we might know that the OLS estimate of a parameter \(\beta\) may be \(50 \%\) too high. Saying that we are willing to tolerate a \(10 \%\) relative bias means that we are willing to tolerate a bias of the 2SLS estimator no greater than \(5 \%\) (that is, \(10 \%\) of \(50 \%\) ). In our rental rate model, if we are willing to tolerate a \(10 \%\) relative bias, then we can conclude that our instruments are not weak because the test statistic of 13.30 exceeds the critical value of 10.27 . However, if we were willing to tolerate only a relative bias of \(5 \%\), we would conclude that our instruments are weak because \(13.30<16.85\).

The rows marked "2SLS Size of nominal 5\% Wald test" and "LIML Size of nominal 5\% Wald test" contain critical values pertaining to Stock and Yogo's (2005) second characterization of weak instruments. This characterization defines a set of instruments to be weak if a Wald test at the 5\% level can have an actual rejection rate of no more than \(10 \%, 15 \%, 20 \%\), or \(25 \%\). Using the current example, suppose that we are willing to accept a rejection rate of at most \(10 \%\). Because \(13.30<24.58\), we
cannot reject the null hypothesis of weak instruments. On the other hand, if we use the LIML estimator instead, then we can reject the null hypothesis because \(13.30>5.44\).

\section*{Technical note}

Stock and Yogo (2005) tabulated critical values for 2SLS relative biases of \(5 \%, 10 \%, 20 \%\), and \(30 \%\) for models with 1 , 2 , or 3 endogenous regressors and between 3 and 30 excluded exogenous variables (instruments). They also provide critical values for worst-case rejection rates of \(5 \%, 10 \%\), \(20 \%\), and \(25 \%\) for nominal \(5 \%\) Wald tests of the endogenous regressors with 1 or 2 endogenous regressors and between 1 and 30 instruments. If the model previously fit by ivregress has more instruments or endogenous regressors than these limits, the critical values are not shown. Stock and Yogo did not consider GMM estimators.

When the model being fit contains more than one endogenous regressor, the \(R^{2}\) and \(F\) statistics described above can overstate the relevance of the excluded instruments. Suppose that there are two endogenous regressors, \(Y_{1}\) and \(Y_{2}\), and that there are two additional instruments, \(z_{1}\) and \(z_{2}\). Say that \(z_{1}\) is highly correlated with both \(Y_{1}\) and \(Y_{2}\) but \(z_{2}\) is not correlated with either \(Y_{1}\) or \(Y_{2}\). Then, the first-stage regression of \(Y_{1}\) on \(z_{1}\) and \(z_{2}\) (along with the included exogenous variables) will produce large \(R^{2}\) and \(F\) statistics, as will the regression of \(Y_{2}\) on \(z_{1}, z_{2}\), and the included exogenous variables. Nevertheless, the lack of correlation between \(z_{2}\) and \(Y_{1}\) and \(Y_{2}\) is problematic. Here, although the order condition indicates that the model is just identified (the number of excluded instruments equals the number of endogenous regressors), the irrelevance of \(z_{2}\) implies that the model is in fact not identified. Even if the model is overidentified, including irrelevant instruments can adversely affect the properties of instrumental-variables estimators, because their biases increase as the number of instruments increases.

\section*{> Example 4}
estat firststage presents different statistics when the model contains multiple endogenous regressors. For illustration, we refit our model of rental rates, assuming that both hsngval and faminc are endogenously determined. We use i.region along with popden, a measure of population density, as additional instruments.
```

. ivregress 2sls rent pcturban (hsngval faminc = i.region popden)
(output omitted)

```
estat firststage
Shea's partial R-squared
\begin{tabular}{r|cc}
\hline \multirow{2}{*}{ Variable } & \begin{tabular}{c} 
Shea's \\
partial R-sq.
\end{tabular} & \begin{tabular}{c} 
adj. phea's \\
partial R-sq.
\end{tabular} \\
\hline hsngval & 0.3477 & 0.2897 \\
faminc & 0.1893 & 0.1173 \\
\hline
\end{tabular}
Minimum eigenvalue statistic \(=2.51666\)
Critical Values \# of endogenous regressors: 2
HO: Instruments are weak \# of excluded instruments: 4
\begin{tabular}{l|cllll}
\hline & & \(5 \%\) & \(10 \%\) & \(20 \%\) & \(30 \%\) \\
2SLS relative bias & 11.04 & 7.56 & 5.57 & 4.73 \\
\hline & & \(10 \%\) & \(15 \%\) & \(20 \%\) & \(25 \%\) \\
2SLS size of nominal \(5 \%\) Wald test & 16.87 & 9.93 & 7.54 & 6.28 \\
LIML size of nominal \(5 \%\) Wald test & 4.72 & 3.39 & 2.99 & 2.79 \\
\hline
\end{tabular}

Consider the endogenous regressor hsngval. Part of its variation is attributable to its correlation with the other regressors pcturban and faminc. The other component of hsngval's variation is peculiar to it and orthogonal to the variation in the other regressors. Similarly, we can think of the instruments as predicting the variation in hsngval in two ways, one stemming from the fact that the predicted values of hsngval are correlated with the predicted values of the other regressors and one from the variation in the predicted values of hsngval that is orthogonal to the variation in the predicted values of the other regressors.

What really matters for instrumental-variables estimation is whether the component of hsngval that is orthogonal to the other regressors can be explained by the component of the predicted value of hsngval that is orthogonal to the predicted values of the other regressors in the model. Shea's (1997) partial \(R^{2}\) statistic measures this correlation. Because the bias of instrumental-variables estimators increases as more instruments are used, Shea's adjusted partial \(R^{2}\) statistic is often used instead, as it makes a degrees-of-freedom adjustment for the number of instruments, analogous to the adjusted \(R^{2}\) measure used in OLS regression. Although what constitutes a "low" value for Shea's partial \(R^{2}\) depends on the specifics of the model being fit and the data used, these results, taken in isolation, do not strike us as being a particular cause for concern.

However, with this specification the minimum eigenvalue statistic is low. We cannot reject the null hypothesis of weak instruments for either of the characterizations we have discussed.

By default, estat firststage determines which statistics to present based on the number of endogenous regressors in the model previously fit. However, you can specify the all option to obtain all the statistics.
- Technical note

If the previous estimation was conducted using aweights, pweights, or iweights, then the first-stage regression summary statistics are computed using those weights. However, in these cases the minimum eigenvalue statistic and its critical values are not available.

If the previous estimation included a robust VCE, then the first-stage \(F\) statistic is based on a robust VCE as well; for example, if you fit your model with an HAC VCE using the Bartlett kernel and four lags, then the \(F\) statistic reported is based on regression results using an HAC VCE using the Bartlett kernel and four lags. By default, the minimum eigenvalue statistic and its critical values are not displayed. You can use the forcenonrobust option to obtain them in these cases; the minimum eigenvalue statistic is computed using the weights, though the critical values reported may not be appropriate.

\section*{estat overid}

In addition to the requirement that instrumental variables be correlated with the endogenous regressors, the instruments must also be uncorrelated with the structural error term. If the model is overidentified, meaning that the number of additional instruments exceeds the number of endogenous regressors, then we can test whether the instruments are uncorrelated with the error term. If the model is just identified, then we cannot perform a test of overidentifying restrictions.

The estimator you used to fit the model determines which tests of overidentifying restrictions estat overid reports. If you used the 2SLS estimator without a robust VCE, estat overid reports Sargan's (1958) and Basmann's (1960) \(\chi^{2}\) tests. If you used the 2SLS estimator and requested a robust VCE, Wooldridge's robust score test of overidentifying restrictions is performed instead; without a robust VCE, Wooldridge's test statistic is identical to Sargan's test statistic. If you used the LIML estimator, estat overid reports the Anderson-Rubin (1950) likelihood-ratio test and Basmann's (1960) \(F\) test. estat overid reports Hansen's (1982) \(J\) statistic if you used the GMM estimator. Davidson and MacKinnon (1993, 235-236) give a particularly clear explanation of the intuition behind tests of overidentifying restrictions. Also see Judge et al. (1985, 614-616) for a summary of tests of overidentifying restrictions for the 2SLS and LIML estimators.

Tests of overidentifying restrictions actually test two different things simultaneously. One, as we have discussed, is whether the instruments are uncorrelated with the error term. The other is that the equation is misspecified and that one or more of the excluded exogenous variables should in fact be included in the structural equation. Thus, a significant test statistic could represent either an invalid instrument or an incorrectly specified structural equation.
\(>\) Example 5
Here we refit the model that treated just hsngval as endogenous using 2SLS, and then we perform tests of overidentifying restrictions:
```

. ivregress 2sls rent pcturban (hsngval = faminc i.region)
(output omitted)
estat overid
Tests of overidentifying restrictions:
Sargan (score) chi2(3) = 11.2877 (p = 0.0103)
Basmann chi2(3) = 12.8294 (p = 0.0050)

```

Both test statistics are significant at the \(5 \%\) test level, which means that either one or more of our instruments are invalid or that our structural model is specified incorrectly.

One possibility is that the error term in our structural model is heteroskedastic. Both Sargan's and Basmann's tests assume that the errors are i.i.d.; if the errors are not i.i.d., then these tests are not valid. Here we refit the model by requesting heteroskedasticity-robust standard errors, and then we use estat overid to obtain Wooldridge's score test of overidentifying restrictions, which is robust to heteroskedasticity.
```

. ivregress 2sls rent pcturban (hsngval = faminc i.region), vce(robust)
(output omitted)
. estat overid
Test of overidentifying restrictions:
Score chi2(3) = 6.8364 (p = 0.0773)

```

Here we no longer reject the null hypothesis that our instruments are valid at the \(5 \%\) significance level, though we do reject the null at the \(10 \%\) level. You can verify that the robust standard error on the coefficient for hsngval is more than twice as large as its nonrobust counterpart and that the robust standard error for pcturban is nearly \(50 \%\) larger.

\section*{- Technical note}

The test statistic for the test of overidentifying restrictions performed after GMM estimation is simply the sample size times the value of the objective function \(Q\left(\boldsymbol{\beta}_{1}, \boldsymbol{\beta}_{2}\right)\) defined in (5) of [R] ivregress, evaluated at the GMM parameter estimates. If the weighting matrix \(\mathbf{W}\) is optimal, meaning that \(\mathbf{W}=\operatorname{Var}\left(\mathbf{z}_{i} u_{i}\right)\), then \(Q\left(\boldsymbol{\beta}_{1}, \boldsymbol{\beta}_{2}\right) \stackrel{A}{\sim} \chi^{2}(q)\), where \(q\) is the number of overidentifying restrictions. However, if the estimated \(\mathbf{W}\) is not optimal, then the test statistic will not have an asymptotic \(\chi^{2}\) distribution.

Like the Sargan and Basmann tests of overidentifying restrictions for the 2SLS estimator, the Anderson-Rubin and Basmann tests after LIML estimation are predicated on the errors' being i.i.d. If the previous LIML results were reported with robust standard errors, then estat overid by default issues an error message and refuses to report the Anderson-Rubin and Basmann test statistics. You can use the forcenonrobust option to override this behavior. You can also use forcenonrobust to obtain the Sargan and Basmann test statistics after 2SLS estimation with robust standard errors.

By default, estat overid issues an error message if the previous estimation was conducted using aweights, pweights, or iweights. You can use the forceweights option to override this behavior, though the test statistics may no longer have the expected \(\chi^{2}\) distributions.

\section*{estat weakrobust (StataNow)}

Instrumental-variables methods require that the instruments be correlated with the endogenous regressors. In principle, this is a low bar: even instruments only weakly correlated with the endogenous regressors allow for valid asymptotic inference using standard instrumental-variables methods. But weak instruments can lead to misleading inference in practice, even in relatively large samples. This is especially the case when models are overidentified or there is a high degree of endogeneity.

When instruments are weak, first-stage coefficients are small relative to the variance of their estimates. This causes the distribution of instrumental-variables estimators to be highly nonnormal and thus poorly approximated by standard inference methods. (See Andrews, Stock, and Sun [2019].)

The linear instrumental-variables model fit by ivregress is written in matrix form as
\[
\begin{aligned}
\mathbf{y} & =\mathbf{Y} \boldsymbol{\beta}_{1}+\mathbf{X}_{1} \boldsymbol{\beta}_{2}+\mathbf{u} \\
\mathbf{Y} & =\mathbf{X}_{1} \boldsymbol{\Pi}_{1}+\mathbf{X}_{2} \boldsymbol{\Pi}_{2}+\mathbf{V}
\end{aligned}
\]
where \(\mathbf{y}\) is a vector with the values of the dependent variable in the sample, \(\mathbf{Y}\) is a matrix of endogenous regressors, \(\mathbf{X}_{1}\) is a matrix of included exogenous regressors, and \(\mathbf{X}_{2}\) is a matrix of excluded exogenous regressors. The first equation is referred to as the structural equation, and the second equation is referred to as the first-stage equation. By substituting the endogenous regressors \(\mathbf{Y}\) into the structural equation and reparameterizing, we derive the reduced-form equation
\[
\mathbf{y}=\mathbf{X}_{1} \boldsymbol{\delta}_{1}+\mathbf{X}_{2} \boldsymbol{\delta}_{2}+\boldsymbol{\epsilon}
\]
where \(\boldsymbol{\delta}_{1} \equiv \boldsymbol{\Pi}_{1} \boldsymbol{\beta}_{1}+\boldsymbol{\beta}_{2}\) and \(\boldsymbol{\delta}_{2} \equiv \boldsymbol{\Pi}_{2} \boldsymbol{\beta}_{1}\).
The principle of the tests performed by estat weakrobust is that the reduced-form and first-stage equations are free of endogeneity. The estimates of \(\boldsymbol{\delta}_{1}, \boldsymbol{\delta}_{2}, \boldsymbol{\Pi}_{1}\), and \(\boldsymbol{\Pi}_{2}\) can be tested for consistency with a null hypothesis about \(\boldsymbol{\beta}_{1}\), our parameter of interest. (In practice, the exogenous covariates \(\mathbf{X}_{2}\) are partialled out for simplicity.) estat weakrobust performs tests of the form \(\boldsymbol{\beta}_{1}=\mathbf{r}_{1}\), where \(\mathbf{r}_{1}\) is a constant vector of the same dimension as \(\boldsymbol{\beta}_{1}\).

When the model is just identified, meaning there are no more instruments than there are endogenous regressors, the test of Anderson and Rubin (1949), or the associated confidence interval, is reported by default. When the model has been fit with a heteroskedastic, cluster-robust, or HAC VCE, an appropriate version of the Anderson-Rubin test or confidence interval is reported. In all cases, the Anderson-Rubin test statistic takes on a \(\chi^{2}\) distribution, and critical values and \(p\)-values can be exactly computed.

When the model is overidentified and has been fit with a homoskedastic VCE, the CLR test of Moreira (2003), or its associated confidence interval, is reported by default. The CLR test has better power properties than the Anderson-Rubin test in the overidentified case. When the model is fit with a nonhomoskedastic VCE, a generalization of the CLR test proposed by Finlay and Magnusson (2009) is used. This test is equivalent to the GMM quasi-LR statistic of Kleibergen (2007) in the linear instrumental-variables case considered here.

Regardless of the VCE specified, the CLR test statistic has a nonstandard distribution. However, when the model has been fit with a homoskedastic VCE and there is only a single endogenous regressor, the \(p\)-value for the CLR statistic can be computed exactly using the method of Andrews, Moreira, and Stock (2007). Otherwise, the \(p\)-value for the CLR test statistic is obtained by simulation.

Both the Anderson-Rubin and the CLR tests (and their CIs) can be requested for any model, even though only one test ( or CI ) is computed and reported by default. In the just-identified case, the CLR test is equivalent to the Anderson-Rubin test.

\section*{> Example 6}

We revisit the model of average rental rate as a function of average housing values and the proportion of the population living in urban areas, this time supposing we do not have access to faminc, the variable for median family income. We may suspect that our region indicator variables are weak instruments for hsngval and therefore perform inference robust to weak instruments using estat weakrobust.
```

. ivregress 2sls rent pcturban (hsngval = i.region), vce(robust)
(output omitted)
. estat weakrobust, rseed(12345)
Test robust to weak instruments
Model VCE: Robust
( 1) hsngval = 0
Cond. likelihood-ratio (CLR) test = 5.48
Prob > CLR = 0.0255
Notes: CLR test reported by default because
model is overidentified.
p-value computed by simulation
(25,000 replications).

```

Here estat weakrobust reports a CLR test because there are more instruments than there are endogenous regressors. We find evidence to reject the null hypothesis that the coefficient on hsngval is zero, but the \(p\)-value is substantially larger than the \(p\)-value of 0.003 that is reported by ivregress. Note that because the model is fit with a nonhomoskedastic VCE and a CLR test is being reported, the \(p\)-value is computed by simulation. We have included a random-number seed using the rseed() option for reproducibility, but this is optional.

We can request a confidence interval that is robust to weak instruments for hsngval with the ci option:
```

. estat weakrobust, ci rseed(12345)
Searching for CI bounds:
Iteration 0: Grid points = 500
Iteration 1: Grid points = 1,000
(CI computed using 1,000 grid points on [-.003591, .006664])
Confidence interval robust to weak instruments
Model VCE: Robust

```
\begin{tabular}{l|ccr}
\hline & \multicolumn{2}{c}{} & \multicolumn{2}{c}{ CLR } \\
& Coefficient & [95\% conf. interval] \\
\hline hsngval & .0015365 & .0002263 & .002857 \\
\hline
\end{tabular}

Notes: CLR CI reported by default because model is overidentified. Computed using simulation (25,000 replications).

The confidence interval comes close to containing zero. Indeed, if we specified a confidence level of \(99 \%\) using level (99), zero would be included. Accounting for the possibility of weak instruments, we may conclude that we cannot rule out a zero coefficient on hsngval.

The ci option, which computes Anderson-Rubin and CLR confidence intervals, is available only in the single-endogenous regressor case. When the model has been fit with a homoskedastic VCE or is just identified, confidence intervals can be computed directly by inverting the relevant test. When the model has been fit with a nonhomoskedastic VCE and is overidentified, there is no closed form for the endpoints of the conditional likelihood confidence interval, so a gridding procedure is used to estimate the confidence intervals.

The coefficient on the endogenous regressor, as fit by ivregress, is reported next to the confidence interval in the output of estat weakrobust, ci for reference. However, there is generally no guarantee that the estimated coefficient will be contained in the reported confidence interval. An exception is
that when the model has been fit with homoskedastic errors, the CLR confidence set will contain the LIML estimator. In the just-identified case, the Anderson-Rubin confidence set will also contain the LIML estimator. See Moreira (2003) for a discussion.

\section*{Technical note}

Unlike conventional confidence intervals, the confidence "intervals" produced by inverting Anderson-Rubin and CLR tests are not always finite intervals and may not be intervals at all. Confidence intervals can take one of five forms:
1. Finite interval \([a, b]\). When instruments are strong and the model is well identified, both Anderson-Rubin and CLR confidence intervals typically take this form.
2. Union of finite intervals \(\left[a_{1}, b_{1}\right] \cup\left[a_{2}, b_{2}\right] \cup \ldots \cup\left[a_{m}, b_{m}\right]\). Confidence intervals can take this form when the model VCE is nonhomoskedastic.
3. Union of (possibly infinite) intervals \(\left(-\infty, b_{1}\right] \cup \ldots \cup\left[a_{m},+\infty\right)\). Confidence sets may take this form when instruments are weak. When the model VCE is homoskedastic, this form may only be the union of two infinite intervals.
4. Real line \((-\infty,+\infty)\). Confidence sets may take this form when instruments are weak.
5. Empty set \(\emptyset\). An empty confidence interval means the test rejects every possible null value for the endogenous regressor, thus rejecting the model. The only case in which the model can be rejected in this way is when the model is overidentified and an Anderson-Rubin confidence interval is used. In other cases, an empty confidence interval means the grid used to estimate the confidence interval was too narrow, and the bounds should be widened using the bound () option.

For detailed discussion of unconventional confidence intervals in inference that is robust to weak instruments, see Mikusheva and Poi (2006), Mikusheva (2010), and Kleibergen (2007).

\section*{Stored results}

After 2SLS estimation, estat endogenous stores the following in r() :
Scalars
```

r(durbin) Durbin }\mp@subsup{\chi}{}{2}\mathrm{ statistic
r(p_durbin) p-value for Durbin }\mp@subsup{\chi}{}{2}\mathrm{ statistic
r(wu) Wu-Hausman F}\mathrm{ statistic
r(p_wu) p
r(df) degrees of freedom
r(wudf_r) denominator degrees of freedom for Wu-Hausman F
r(r_score) robust score statistic
r(p_r_score) p
r(hac_score) HAC score statistic
r(p_hac_score) p-value for HAC score statistic
r(lags) lags used in prewhitening
r(regF) regression-based F statistic
r(p_regF) p-value for regression-based F statistic
r(regFdf_n) regression-based F numerator degrees of freedom
r(regFdf_r) regression-based F denominator degrees of freedom

```

After GMM estimation, estat endogenous stores the following in \(r()\) :
Scalars
```

r(C) C \chi 2 statistic
r(p_C) p-value for C }\mp@subsup{\chi}{}{2}\mathrm{ statistic
r(df) degrees of freedom

```
estat firststage stores the following in \(r()\) :
Scalars \(r\) (mineig) minimum eigenvalue statistic
Matrices
```

        r(mineigcv) critical values for minimum eigenvalue statistic
    r(multiresults) Shea's partial R}\mp@subsup{R}{}{2}\mathrm{ statistics
    r(singleresults) first-stage }\mp@subsup{R}{}{2}\mathrm{ and F statistics
    ```

After 2SLS estimation, estat overid stores the following in r() :
Scalars
```

r(lags) lags used in prewhitening

```

```

    r(score) score }\mp@subsup{\chi}{}{2}\mathrm{ statistic
    r(p_score) }\quadp\mathrm{ -value for score }\mp@subsup{\chi}{}{2}\mathrm{ statistic
    r(basmann) Basmann }\mp@subsup{\chi}{}{2}\mathrm{ statistic
    r(p_basmann) p
    r(sargan) Sargan }\mp@subsup{\chi}{}{2}\mathrm{ statistic
    r(p_sargan) p
    ```

After LIML estimation, estat overid stores the following in \(r()\) :
Scalars
```

r(ar) Anderson-Rubin (1950) }\mp@subsup{\chi}{}{2}\mathrm{ statistic
r(p_ar) p-value for Anderson-Rubin (1950) }\mp@subsup{\chi}{}{2}\mathrm{ statistic
r(ar_df) < < 2 degrees of freedom
r(basmann) Basmann F statistic
r(p_basmann) }\quadp\mathrm{ -value for Basmann }F\mathrm{ statistic
r(basmann_df_n) F numerator degrees of freedom
r(basmann_df_d) F}\mathrm{ denominator degrees of freedom

```

After GMM estimation, estat overid stores the following in \(r()\) :
Scalars
```

r(HansenJ) Hansen's J \chi 2 statistic
r(p_HansenJ) p-value for Hansen's }J\mp@subsup{\chi}{}{2}\mathrm{ statistic
r(J_df) < < 2 degrees of freedom

```

After estimation of a just-identified model, or when ar is specified, estat weakrobust stores the following in \(r()\) :

Scalars
r(archi2)
\(r\) (p_archi2)
r(archi2_df)
\(r\) (arF)
\(r(p-a r F)\)
\(r\left(a r F \_d f \_n\right)\)
\(r\left(a r F \_d f \_d\right)\)
\(r\) (ar_ngrid)
Macros \(r\) (bound)

Matrices
\(r\) (betanull)
r(table)
\(r\left(a r \_c i\right)\)
\(r\) (ar_finalbound)

Anderson-Rubin (1949) \(\chi^{2}\) statistic
\(p\)-value for Anderson-Rubin (1949) \(\chi^{2}\) statistic
\(\chi^{2}\) degrees of freedom
Anderson-Rubin (1949) \(F\) statistic
\(p\)-value for Anderson-Rubin (1949) \(F\) statistic
\(F\) numerator degrees of freedom
\(F\) denominator degrees of freedom
number of gridpoints used for Anderson-Rubin (1949) confidence interval
user-specified grid bounds
values of the null hypothesis for endogenous regressors
matrix containing coefficient with reported confidence interval (omitted if CI is empty) matrix containing the bounds of the Anderson-Rubin (1949) confidence interval matrix containing the final upper and lower bounds used by the gridding procedure

After estimation of an overidentified model, or when clr is specified, estat weakrobust stores the following in \(r()\) :

Scalars
```

r(clr)
r(p_clr)
r(J)
r(rk)
r(reps)
r(rseed)
r(clr_ngrid)

```

Macros
```

        r(bound)
    ```
    r(rngstate)

Matrices
```

        r(betanull)
    ```
    r(Sbar)
    \(r\) (Tbar)
    \(r\) (table)
    r(clr_ci)
    \(r\) (clr_finalbound)

CLR statistic of Moreira (2003)
\(p\)-value for CLR statistic
\(J\) statistic of Finlay and Magnusson (2009) (when model VCE is nonhomoskedastic) rank statistic of Kleibergen and Paap (2006) (when model VCE is nonhomoskedastic) number of simulation replications (when results are simulated)
random seed used (when results are simulated)
number of gridpoints used for CLR confidence interval
user-specified grid bounds
random-number state used (when results are simulated)
values of the null hypothesis for endogenous regressors
\(\bar{S}\) statistic of Moreira (2003) (when model VCE is homoskedastic)
\(\bar{T}\) statistic of Moreira (2003) (when model VCE is homoskedastic)
matrix containing coefficient with reported confidence interval (omitted if CI is empty) matrix containing the bounds of the CLR confidence interval matrix containing the final upper and lower bounds used by the gridding procedure

\section*{Methods and formulas}

Methods and formulas are presented under the following headings:

\section*{Notation}
estat endogenous
estat firststage
estat overid
estat weakrobust (StataNow)
Homoskedastic errors
Nonhomoskedastic errors
Confidence intervals

\section*{Notation}

Recall from [ R ] ivregress that the model is
\[
\begin{gathered}
\mathbf{y}=\mathbf{Y} \boldsymbol{\beta}_{1}+\mathbf{X}_{1} \boldsymbol{\beta}_{2}+\mathbf{u}=\mathbf{X} \boldsymbol{\beta}+\mathbf{u} \\
\mathbf{Y}=\mathbf{X}_{1} \boldsymbol{\Pi}_{1}+\mathbf{X}_{2} \boldsymbol{\Pi}_{2}+\mathbf{V}=\mathbf{Z} \boldsymbol{\Pi}+\mathbf{V}
\end{gathered}
\]
where \(\mathbf{y}\) is an \(N \times 1\) vector of the left-hand-side variable, \(N\) is the sample size, \(\mathbf{Y}\) is an \(N \times p\) matrix of \(p\) endogenous regressors, \(\mathbf{X}_{1}\) is an \(N \times k_{1}\) matrix of \(k_{1}\) included exogenous regressors, \(\mathbf{X}_{2}\) is an \(N \times k_{2}\) matrix of \(k_{2}\) excluded exogenous variables, \(\mathbf{X}=\left[\mathbf{Y} \mathbf{X}_{1}\right], \mathbf{Z}=\left[\mathbf{X}_{1} \mathbf{X}_{2}\right]\), \(\mathbf{u}\) is an \(N \times 1\) vector of errors, \(\mathbf{V}\) is an \(N \times p\) matrix of errors, \(\boldsymbol{\beta}=\left[\boldsymbol{\beta}_{1} \boldsymbol{\beta}_{2}\right]\) is a \(k=\left(p+k_{1}\right) \times 1\) vector of parameters, and \(\boldsymbol{\Pi}\) is a \(\left(k_{1}+k_{2}\right) \times p\) vector of parameters. If a constant term is included in the model, then one column of \(\mathbf{X}_{1}\) contains all ones.

\section*{estat endogenous}

Partition \(\mathbf{Y}\) as \(\mathbf{Y}=\left[\begin{array}{ll}\mathbf{Y}_{1} & \mathbf{Y}_{2}\end{array}\right]\), where \(\mathbf{Y}_{1}\) represents the \(p_{1}\) endogenous regressors whose endogeneity is being tested and \(\mathbf{Y}_{2}\) represents the \(p_{2}\) endogenous regressors whose endogeneity is not being tested. If the endogeneity of all endogenous regressors is being tested, \(\mathbf{Y}=\mathbf{Y}_{\mathbf{1}}\) and \(p_{2}=0\). After GMM estimation, estat endogenous refits the model treating \(\mathbf{Y}_{1}\) as exogenous using the same type of weight matrix as requested at estimation time with the wmatrix () option; denote the Sargan statistic from this model by \(J_{e}\) and the estimated weight matrix by \(\mathbf{W}_{e}\). Let \(\mathbf{S}_{e}=\mathbf{W}_{e}^{-1}\). estat endogenous removes from \(\mathbf{S}_{e}\) the rows and columns corresponding to the variables represented by \(\mathbf{Y}_{1}\); denote the inverse of the resulting matrix by \(\mathbf{W}_{e}^{\prime}\). Next, estat endogenous fits the model treating both \(\mathbf{Y}_{1}\) and \(\mathbf{Y}_{2}\) as endogenous, using the weight matrix \(\mathbf{W}_{e}^{\prime}\); denote the Sargan statistic from this model by \(J_{c}\). Then, \(C=\left(J_{e}-J_{c}\right) \sim \chi^{2}\left(p_{1}\right)\). If one simply used the \(J\) statistic from the original model fit by ivregress in place of \(J_{c}\), then in finite samples \(J_{e}-J\) might be negative. The procedure used by estat endogenous is guaranteed to yield \(C \geq 0\); see Hayashi (2000, 220).

Let \(\widehat{\mathbf{u}}_{c}\) denote the residuals from the model treating both \(\mathbf{Y}_{1}\) and \(\mathbf{Y}_{2}\) as endogenous, and let \(\widehat{\mathbf{u}}_{e}\) denote the residuals from the model treating only \(\mathbf{Y}_{2}\) as endogenous. Then, Durbin's (1954) statistic is
\[
D=\frac{\widehat{\mathbf{u}}_{e}^{\prime} \mathbf{P}_{Z Y_{1}} \widehat{\mathbf{u}}_{e}-\widehat{\mathbf{u}}_{c}^{\prime} \mathbf{P}_{Z} \widehat{\mathbf{u}}_{c}}{\widehat{\mathbf{u}}_{e}^{\prime} \widehat{\mathbf{u}}_{e} / N}
\]
where \(\mathbf{P}_{Z}=\mathbf{Z}\left(\mathbf{Z}^{\prime} \mathbf{Z}\right)^{-1} \mathbf{Z}^{\prime}\) and \(\mathbf{P}_{Z Y_{1}}=\left[\mathbf{Z} \mathbf{Y}_{1}\right]\left(\left[\mathbf{Z} \mathbf{Y}_{1}\right]^{\prime}\left[\mathbf{Z} \mathbf{Y}_{1}\right]\right)^{-1}\left[\mathbf{Z} \mathbf{Y}_{1}\right]^{\prime} D \sim \chi^{2}\left(p_{1}\right)\). The Wu-Hausman (Wu 1974; Hausman 1978) statistic is
\[
W H=\frac{\left(\widehat{\mathbf{u}}_{e}^{\prime} \mathbf{P}_{Z Y_{1}} \widehat{\mathbf{u}}_{e}-\widehat{\mathbf{u}}_{c}^{\prime} \mathbf{P}_{Z} \widehat{\mathbf{u}}_{c}\right) / p_{1}}{\left\{\widehat{\mathbf{u}}_{e}^{\prime} \widehat{\mathbf{u}}_{e}-\left(\widehat{\mathbf{u}}_{e}^{\prime} \mathbf{P}_{Z Y_{1}} \widehat{\mathbf{u}}_{e}-\widehat{\mathbf{u}}_{c}^{\prime} \mathbf{P}_{Z} \widehat{\mathbf{u}}_{c}\right)\right\} /\left(N-k_{1}-p-p_{1}\right)}
\]
\(W H \sim F\left(p_{1}, N-k_{1}-p-p_{1}\right)\). Baum, Schaffer, and Stillman \((2003,2007)\) discuss these tests in more detail.

Next, we describe Wooldridge's (1995) score test. The nonrobust version of Wooldridge's test is identical to Durbin's test. Suppose a robust covariance matrix was used at estimation time. Let \(\widehat{\mathbf{e}}\) denote the sample residuals obtained by fitting the model via OLS, treating \(\mathbf{Y}\) as exogenous. We then regress each variable represented in \(\mathbf{Y}\) on \(\mathbf{Z}\); call the residuals for the \(j\) th regression \(\widehat{\mathbf{r}}_{j}, j=1, \ldots, p\). Define \(\widehat{k}_{i j}=\widehat{e}_{i} \widehat{r}_{i j}, i=1, \ldots, N\). We then run the regression
\[
\mathbf{1}=\theta_{1} \widehat{\mathbf{k}}_{1}+\cdots+\theta_{p} \widehat{\mathbf{k}}_{p}+\boldsymbol{\epsilon}
\]
where 1 is an \(N \times 1\) vector of ones and \(\epsilon\) is a regression error term. \(N-\operatorname{RSS} \sim \chi^{2}(p)\), where RSS is the residual sum of squares from the regression just described. If instead an HAC VCE was used at estimation time, then before running the final regression we prewhiten the \(\widehat{\mathbf{k}}_{j}\) series by using a \(\operatorname{VAR}(q)\) model, where \(q\) is the number of lags specified with the lags() option.

The regression-based test proceeds as follows. Following Hausman (1978, 1259), we regress Y on \(\mathbf{Z}\) and obtain the residuals \(\widehat{\mathbf{V}}\). Next, we fit the augmented regression
\[
\mathbf{y}=\mathbf{Y} \boldsymbol{\beta}_{1}+\mathbf{X}_{1} \boldsymbol{\beta}_{2}+\widehat{\mathbf{V}} \boldsymbol{\gamma}+\boldsymbol{\epsilon}
\]
by OLS regression, where \(\epsilon\) is a regression error term. A test of the exogeneity of \(\mathbf{Y}\) is equivalent to a test of \(\gamma=\mathbf{0}\). As Cameron and Trivedi \((2005,276)\) suggest, this test can be made robust to heteroskedasticity, autocorrelation, or clustering by using the appropriate robust VCE when testing \(\gamma=\mathbf{0}\). When a nonrobust VCE is used, this test is equivalent to the Wu -Hausman test described earlier. One cannot simply fit this augmented regression via 2SLS to test the endogeneity of a subset of the endogenous regressors; Davidson and MacKinnon (1993, 229-231) discuss a test of \(\gamma=\mathbf{0}\) for the homoskedastic version of the augmented regression fit by 2SLS, but an appropriate robust test is not apparent.

\section*{estat firststage}

When the structural equation includes one endogenous regressor, estat firststage fits the regression
\[
\mathbf{Y}=\mathbf{X}_{1} \boldsymbol{\pi}_{1}+\mathbf{X}_{2} \boldsymbol{\pi}_{2}+\mathbf{v}
\]
via OLS. The \(R^{2}\) and adjusted \(R^{2}\) from that regression are reported in the output, as well as the \(F\) statistic from the Wald test of \(H_{0}: \boldsymbol{\pi}_{2}=\mathbf{0}\). To obtain the partial \(R^{2}\) statistic, estat firststage fits the regression
\[
\mathbf{M}_{\mathbf{X}_{1}} \mathbf{y}=\mathbf{M}_{\mathbf{X}_{1}} \mathbf{X}_{2} \boldsymbol{\xi}+\boldsymbol{\epsilon}
\]
by OLS, where \(\boldsymbol{\epsilon}\) is a regression error term, \(\boldsymbol{\xi}\) is a \(k_{2} \times 1\) parameter vector, and \(\mathbf{M}_{\mathbf{X}_{1}}=\mathbf{I}-\) \(\mathbf{X}_{1}\left(\mathbf{X}_{1}^{\prime} \mathbf{X}_{1}\right)^{-1} \mathbf{X}_{1}^{\prime}\); that is, the partial \(R^{2}\) is the \(R^{2}\) between \(\mathbf{y}\) and \(\mathbf{X}_{2}\) after eliminating the effects of \(\mathbf{X}_{1}\). If the model contains multiple endogenous regressors and the all option is specified, these statistics are calculated for each endogenous regressor in turn.

To calculate Shea's partial \(R^{2}\), let \(\mathbf{y}_{1}\) denote the endogenous regressor whose statistic is being calculated and \(\mathbf{Y}_{0}\) denote the other endogenous regressors. Define \(\widetilde{\mathbf{y}}_{1}\) as the residuals obtained from regressing \(\mathbf{y}_{1}\) on \(\mathbf{Y}_{0}\) and \(\mathbf{X}_{1}\). Let \(\widehat{\mathbf{y}}_{1}\) denote the fitted values obtained from regressing \(\mathbf{y}_{1}\) on \(\mathbf{X}_{1}\) and \(\mathbf{X}_{2}\); that is, \(\widehat{\mathbf{y}}_{1}\) are the fitted values from the first-stage regression for \(\mathbf{y}_{1}\), and define the columns of \(\widehat{\mathbf{Y}}_{0}\) analogously. Finally, let \(\widetilde{\hat{\mathbf{y}}}_{1}\) denote the residuals from regressing \(\widehat{\mathbf{y}}_{1}\) on \(\widehat{\mathbf{Y}}_{0}\) and \(\mathbf{X}_{1}\). Shea's partial \(R^{2}\) is the simple \(R^{2}\) from the regression of \(\widetilde{\mathbf{y}}_{1}\) on \(\widetilde{\widehat{\mathbf{y}}}_{1}\); denote this as \(R_{S}^{2}\). Shea's adjusted partial \(R^{2}\) is equal to \(1-\left(1-R_{S}^{2}\right)(N-1) /\left(N-k_{Z}+1\right)\) if a constant term is included and \(1-\left(1-R_{S}^{2}\right)(N-1) /\left(N-k_{Z}\right)\) if there is no constant term included in the model, where \(k_{Z}=k_{1}+k_{2}\). For one endogenous regressor, one instrument, no exogenous regressors, and a constant term, \(R_{S}^{2}\) equals the adjusted \(R_{S}^{2}\).

The Stock and Yogo minimum eigenvalue statistic, first proposed by Cragg and Donald (1993) as a test for underidentification, is the minimum eigenvalue of the matrix
\[
\mathbf{G}=\frac{1}{k_{Z}} \widehat{\boldsymbol{\Sigma}}_{\mathbf{V V}}^{-1 / 2} \mathbf{Y}^{\prime} \mathbf{M}_{\mathbf{X}_{1}}^{\prime} \mathbf{X}_{2}\left(\mathbf{X}_{2}^{\prime} \mathbf{M}_{\mathbf{X}_{1}} \mathbf{X}_{2}\right)^{-1} \mathbf{X}_{2}^{\prime} \mathbf{M}_{\mathbf{X}_{1}} \mathbf{Y} \widehat{\boldsymbol{\Sigma}}_{\mathbf{V V}}^{-1 / 2}
\]
where
\[
\widehat{\boldsymbol{\Sigma}}_{\mathbf{V V}}=\frac{1}{N-k_{Z}} \mathbf{Y}^{\prime} \mathbf{M}_{\mathbf{Z}} \mathbf{Y}
\]
\(\mathbf{M}_{\mathbf{Z}}=\mathbf{I}-\mathbf{Z}\left(\mathbf{Z}^{\prime} \mathbf{Z}\right)^{-1} \mathbf{Z}^{\prime}\), and \(\mathbf{Z}=\left[\mathbf{X}_{1} \mathbf{X}_{2}\right]\). Critical values are obtained from the tables in Stock and Yogo (2005).

\section*{estat overid}

The Sargan (1958) and Basmann (1960) \(\chi^{2}\) statistics are calculated by running the auxiliary regression
\[
\widehat{\mathbf{u}}=\mathbf{Z} \boldsymbol{\delta}+\mathbf{e}
\]
where \(\widehat{\mathbf{u}}\) are the sample residuals from the model and \(\mathbf{e}\) is an error term. Then, Sargan's statistic is
\[
S=N\left(1-\frac{\hat{\mathbf{e}}^{\prime} \widehat{\mathbf{e}}}{\widehat{\mathbf{u}}^{\prime} \widehat{\mathbf{u}}}\right)
\]
where \(\widehat{\mathbf{e}}\) are the residuals from that auxiliary regression. Basmann's statistic is calculated as
\[
B=S \frac{N-k_{Z}}{N-S}
\]

Both \(S\) and \(B\) are distributed \(\chi^{2}(m)\), where \(m\), the number of overidentifying restrictions, is equal to \(k_{Z}-k\), where \(k\) is the number of endogenous regressors.

Wooldridge's (1995) score test of overidentifying restrictions is identical to Sargan's (1958) statistic under the assumption of i.i.d. and therefore is not recomputed unless a robust VCE was used at estimation time. If a heteroskedasticity-robust VCE was used, Wooldridge's test proceeds as follows. Let \(\widehat{\mathbf{Y}}\) denote the \(N \times k\) matrix of fitted values obtained by regressing the endogenous regressors on \(\mathbf{X}_{1}\) and \(\mathbf{X}_{2}\). Let \(\mathbf{Q}\) denote an \(N \times m\) matrix of excluded exogenous variables; the test statistic to be calculated is invariant to whichever \(m\) of the \(k_{2}\) excluded exogenous variables is chosen. Define the \(i\) th element of \(\widehat{\mathbf{k}}_{j}, i=1, \ldots, N, j=1, \ldots, m\), as
\[
k_{i j}=\widehat{q}_{i j} \widehat{u}_{i}
\]
where \(\widehat{q}_{i j}\) is the \(i\) th element of \(\widehat{\mathbf{q}}_{j}\), the residuals from regressing the \(j\) th column of \(\mathbf{Q}\) on \(\widehat{\mathbf{Y}}\) and \(\mathbf{X}_{1}\). Finally, fit the regression
\[
\mathbf{1}=\theta_{1} \widehat{\mathbf{k}}_{1}+\cdots+\theta_{m} \widehat{\mathbf{k}}_{m}+\boldsymbol{\epsilon}
\]
where 1 is an \(N \times 1\) vector of ones and \(\boldsymbol{\epsilon}\) is a regression error term, and calculate the residual sum of squares, RSS. Then, the test statistic is \(W=N-\) RSS. \(W \sim \chi^{2}(m)\). If an HAC VCE was used at estimation, then the \(\widehat{\mathbf{k}}_{j}\) are prewhitened using a \(\operatorname{VAR}(p)\) model, where \(p\) is specified using the lags () option.

The Anderson-Rubin (AR; 1950) test of overidentifying restrictions for use after the LIML estimator is calculated as \(\mathrm{AR}=N(\kappa-1)\), where \(\kappa\) is the minimal eigenvalue of a certain matrix defined in Methods and formulas of [R] ivregress. AR \(\sim \chi^{2}(m)\). (Some texts define this statistic as \(N \ln (\kappa)\) because \(\ln (x) \approx(x-1)\) for \(x\) near 1.) Basmann's \(F\) statistic for use after the LIML estimator is calculated as \(B_{F}=(\kappa-1)\left(N-k_{Z}\right) / m . B_{F} \sim F\left(m, N-k_{Z}\right)\).

Hansen's \(J\) statistic is simply the sample size times the value of the GMM objective function defined in (5) of [R] ivregress, evaluated at the estimated parameter values. Under the null hypothesis that the overidentifying restrictions are valid, \(J \sim \chi^{2}(m)\).

John Denis Sargan (1924-1996) was born in Yorkshire, UK. He pioneered the theory of instrumental-variables (IV) estimation in an article published in 1958. In the article, he also developed overidentification tests, developed significance tests, and discussed possible instruments for applied work. A year later, he wrote an article extending the theory to models containing autoregressive errors. This extension was one of his many contributions to time-series econometric analysis. For example, in 1964 he published a paper in which he developed misspecification tests for dynamic equations, along with an IV estimator for models with nonlinear parameters, and a model with a long-run equilibrium. His paper laid the foundation for other econometric methods, such as cointegration analysis, and established what would be known as the London School of Economics (LSE) approach to econometric modeling. He spent twenty years at this institution, supervising the doctoral work of many econometricians who themselves made important contributions to econometrics. In addition to Sargan's many lasting contributions to econometrics, he also left a lasting impression on his students and colleagues through his generosity.

\section*{estat weakrobust (StataNow)}

In estat weakrobust (StataNow) of Remarks and examples, we reintroduced the model fit by ivregress, written in matrix form as
\[
\begin{aligned}
\mathbf{y} & =\mathbf{Y} \boldsymbol{\beta}_{1}+\mathbf{X}_{1} \boldsymbol{\beta}_{2}+\mathbf{u} \\
\mathbf{Y} & =\mathbf{X}_{1} \boldsymbol{\Pi}_{1}+\mathbf{X}_{2} \boldsymbol{\Pi}_{2}+\mathbf{V}
\end{aligned}
\]

The first equation is referred to as the structural equation, and the second equation is referred to as the first-stage equation.

We then derived the reduced-form equation. Below, we write the model in the form of the reduced-form equation and the first-stage equations
\[
\begin{gathered}
\mathbf{y}=\mathbf{X}_{1} \boldsymbol{\delta}_{1}+\mathbf{X}_{2} \boldsymbol{\delta}_{2}+\boldsymbol{\epsilon} \\
\mathbf{Y}=\mathbf{X}_{1} \boldsymbol{\Pi}_{1}+\mathbf{X}_{2} \boldsymbol{\Pi}_{2}+\mathbf{V}
\end{gathered}
\]
where \(\boldsymbol{\delta}_{1} \equiv \boldsymbol{\Pi}_{1} \boldsymbol{\beta}_{1}+\boldsymbol{\beta}_{2}\) and \(\boldsymbol{\delta}_{2} \equiv \boldsymbol{\Pi}_{2} \boldsymbol{\beta}_{1}\). The reduced-form coefficients \(\boldsymbol{\delta}_{1}\) and \(\boldsymbol{\delta}_{2}\) are \(k_{1} \times 1\) and \(k_{2} \times 1\), respectively. Without loss of generality, we can rewrite the model to omit included exogenous regressors \(\mathbf{X}_{1}\) by considering \(\mathbf{y}, \mathbf{Y}, \mathbf{X}_{2}, \boldsymbol{\epsilon}\), and \(\mathbf{V}\) to have been replaced by their partialled-out equivalents:
\[
\begin{gathered}
\mathbf{y}=\mathbf{X}_{2} \boldsymbol{\delta}_{2}+\boldsymbol{\epsilon} \\
\mathbf{Y}=\mathbf{X}_{2} \boldsymbol{\Pi}_{2}+\mathbf{V}
\end{gathered}
\]

Our null hypothesis \(\boldsymbol{\beta}_{1}=\mathbf{r}_{\mathbf{1}}\) implies that \(\boldsymbol{\delta}_{2}-\boldsymbol{\Pi}_{2} \mathbf{r}_{\mathbf{1}}=0\).

\section*{Homoskedastic errors}

Let \(\widehat{\boldsymbol{\Omega}}=1 / N\left[\begin{array}{cc}\widehat{\epsilon}^{\prime} \widehat{\epsilon} & \widehat{\epsilon}^{\prime} \widehat{\mathbf{V}} \\ \widehat{\mathbf{V}}^{\prime} \widehat{\epsilon} & \widehat{\mathbf{V}}^{\prime} \widehat{\mathbf{V}}\end{array}\right]\) be the estimated \((p+1) \times(p+1)\) covariance matrix of \(\left(\boldsymbol{\epsilon}^{\prime}, \mathbf{V}^{\prime}\right)^{\prime}\). When the model is assumed to have homoskedastic errors, we can proceed by computing the statistics
\[
\bar{S}=\left(\mathbf{X}_{2}^{\prime} \mathbf{X}_{2}\right)^{-1 / 2} \mathbf{X}_{2}^{\prime}\left(\mathbf{y}-\mathbf{Y} \mathbf{r}_{1}\right)\left(\mathbf{b}_{0}^{\prime} \widehat{\mathbf{\Omega}} \mathbf{b}_{0}\right)^{-1 / 2}
\]
and
\[
\bar{T}=\left(\mathbf{X}_{2}^{\prime} \mathbf{X}_{2}\right)^{-1 / 2} \mathbf{X}_{2}^{\prime}\left[\mathbf{y}, \mathbf{Y}^{\prime}\right]^{\prime} \widehat{\boldsymbol{\Omega}}^{-1} \mathbf{A}_{0}\left(\mathbf{A}_{0}^{\prime} \widehat{\boldsymbol{\Omega}}^{-1} \mathbf{A}_{0}\right)^{-1 / 2}
\]
where \(\mathbf{b}_{0}=\left[1,-\mathbf{r}_{1}\right]\) and \(\mathbf{A}_{0}=\left[\mathbf{r}_{1}, \mathbf{I}_{p}\right]^{\prime}\).
The Anderson-Rubin (1949) statistic is computed as AR \(=\bar{S}^{\prime} \bar{S}\) and follows a \(\chi^{2}\left(k_{2}\right)\) distribution. When the model has been fit with the small option in ivregress, an \(F\) version of the statistic is returned, which is distributed \(F\left(k_{2}, N-k_{1}-k_{2}\right)\). This \(F\) statistic is computed using \(N /\left(N-k_{1}-k_{2}\right) \widehat{\boldsymbol{\Omega}}\) in place of \(\widehat{\boldsymbol{\Omega}}\).

The CLR statistic is computed as CLR \(=\bar{S}^{\prime} \bar{S}-\bar{\lambda}^{\min }\), where \(\bar{\lambda}^{\text {min }}\) is the minimum eigenvalue of the matrix \((\bar{S}, \bar{T})^{\prime}(\bar{S}, \bar{T})\). When the model is fit with the small option in ivregress, the statistic is computed using \(N /\left(N-k_{1}-k_{2}\right) \widehat{\boldsymbol{\Omega}}\). When the model is just identified, CLR \(\sim \chi^{2}\left(k_{2}\right)\). When the model is overidentified, CLR has a nonstandard distribution. The numerical method of Andrews, Moreira, and Stock (2007) is used to obtain \(p\)-values in the single-endogenous regressor case. In the multiple-endogenous regressor case, \(p\)-values are obtained by simulation conditional on \(\bar{T}\), as outlined in Moreira (2003).

\section*{Nonhomoskedastic errors}

When the model is fit with a nonhomoskedastic VCE, let
\[
\widehat{\Sigma}=\left[\begin{array}{cc}
\widehat{\Sigma}_{\delta} & \widehat{\Sigma}_{\delta \Pi} \\
\widehat{\Sigma}_{\Pi \delta} & \widehat{\Sigma}_{\Pi}
\end{array}\right]
\]
be the \(k_{2}(p+1) \times k_{2}(p+1)\) estimated covariance matrix of \(\left\{\boldsymbol{\delta}_{2}^{\prime}, \operatorname{Vec}\left(\boldsymbol{\Pi}_{2}\right)^{\prime}\right\}^{\prime}\), where \(\operatorname{Vec}(\cdot)\) is the column-major vectorization operator.

We compute the Anderson-Rubin statistic as
\[
\operatorname{AR}\left(\mathbf{r}_{1}\right)=\left(\mathbf{y}-\mathbf{Y} \mathbf{r}_{\mathbf{1}}\right)^{\prime} \mathbf{X}_{2}\left(\mathbf{X}_{2}^{\prime} \mathbf{X}_{2}\right)^{-1} \widehat{\mathbf{\Psi}}\left(\mathbf{r}_{\mathbf{1}}\right)^{-1}\left(\mathbf{X}_{2}^{\prime} \mathbf{X}_{2}\right)^{-1} \mathbf{X}_{2}^{\prime}\left(\mathbf{y}-\mathbf{Y} \mathbf{r}_{\mathbf{1}}\right)
\]
where \(\widehat{\boldsymbol{\Psi}}\left(\mathbf{r}_{1}\right)=\left(\mathbf{b}_{0} \otimes \mathbf{I}_{k}\right)^{\prime} \widehat{\boldsymbol{\Sigma}}\left(\mathbf{b}_{0} \otimes \mathbf{I}_{k}\right)\).
As in the homoskedastic case, AR takes on a \(\chi^{2}\left(k_{2}\right)\) distribution under the null. When the model is fit with the small option, the statistic is computed using a small-sample adjustment, and an \(F\) version of the statistic is returned.

To compute the CLR test statistic, let \(J\left(\mathbf{r}_{1}\right)\) be defined as in Finlay and Magnusson (2009) (or, equivalently, as in Kleibergen [2007]), and let \(\mathrm{rk}\left(\mathbf{r}_{1}\right)\) be the rank statistic of Kleibergen and Paap (2006) (the forms of these statistics are omitted here for brevity). The test statistic is given by
\[
\operatorname{CLR}\left(\mathbf{r}_{\mathbf{1}}\right)=\frac{1}{2}\left[\operatorname{AR}\left(\mathbf{r}_{\mathbf{1}}\right)-\operatorname{rk}\left(\mathbf{r}_{\mathbf{1}}\right)+\sqrt{\left\{\operatorname{AR}\left(\mathbf{r}_{\mathbf{1}}\right)+\operatorname{rk}\left(\mathbf{r}_{\mathbf{1}}\right)\right\}^{2}-4 J\left(\mathbf{r}_{\mathbf{1}}\right) \operatorname{rk}\left(\mathbf{r}_{\mathbf{1}}\right)}\right]
\]

In the just-identified case, CLR is equal to AR and has a \(\chi^{2}\left(k_{2}\right)\) distribution. Otherwise, CLR has a nonstandard distribution, and \(p\)-values are obtained by simulation, as outlined in Kleibergen (2007). When the model is fit with the small option, the statistic is computed using a small-sample adjustment.

\section*{Confidence intervals}

Confidence intervals are obtained by inverting the relevant tests. When the model is fit with a homoskedastic VCE, or when it is just identified, the Anderson-Rubin test can be inverted analytically, so the bounds of the resulting interval (or union of intervals) are computed as a closed form (see Mikusheva [2010] for details). When the model VCE is nonhomoskedastic and the model is overidentified, the Anderson-Rubin confidence interval is obtained by a gridding procedure.

The CLR test is equivalent to the Anderson-Rubin test in the just-identified case, so its confidence intervals can be computed in the same way. In the overidentified case when errors are homoskedastic, the CLR test is inverted numerically using the method of Mikusheva (2010). When errors are nonhomoskedastic, a gridding procedure is used to obtain a confidence interval.

\section*{Acknowledgments}

We thank Keith Finlay of the US Census Bureau, Leandro Magnusson of the University of Western Australia, and Mark Schaffer of Heriot-Watt University for their Stata commands rivtest, discussed in Finlay and Magnusson (2009), and weakiv.

We thank Anna Mikusheva of MIT and Brian Poi of Poi Consulting LLC for their Stata command condivreg, discussed in Mikusheva and Poi (2006).

\section*{References}

Anderson, T. W., and H. Rubin. 1949. Estimation of the parameters of a single equation in a complete system of stochastic equations. Annals of Mathematical Statistics 20: 46-63. https://doi.org/10.1214/aoms/1177730090.
-_ 1950. The asymptotic properties of estimates of the parameters of a single equation in a complete system of stochastic equations. Annals of Mathematical Statistics 21: 570-582. https://doi.org/10.1214/aoms/1177729752.

Andrews, D. W. K., M. J. Moreira, and J. H. Stock. 2007. Performance of conditional Wald tests in IV regression with weak instruments. Journal of Econometrics 139: 116-132. https://doi.org/10.1016/j.jeconom.2006.06.007.

Andrews, I., J. H. Stock, and L. Sun. 2019. Weak instruments in instrumental variables regression: Theory and practice. Annual Review of Economics 11: 727-753. https://doi.org/10.1146/annurev-economics-080218-025643.
Angrist, J. D., and J.-S. Pischke. 2009. Mostly Harmless Econometrics: An Empiricist's Companion. Princeton, NJ: Princeton University Press.

Basmann, R. L. 1960. On finite sample distributions of generalized classical linear identifiability test statistics. Journal of the American Statistical Association 55: 650-659. https://doi.org/10.2307/2281588.
Baum, C. F., M. E. Schaffer, and S. Stillman. 2003. Instrumental variables and GMM: Estimation and testing. Stata Journal 3: 1-31.
-_. 2007. Enhanced routines for instrumental variables/generalized method of moments estimation and testing. Stata Journal 7: 465-506.

Bound, J., D. A. Jaeger, and R. M. Baker. 1995. Problems with instrumental variables estimation when the correlation between the instruments and the endogenous explanatory variable is weak. Journal of the American Statistical Association 90: 443-450. https://doi.org/10.2307/2291055.

Cameron, A. C., and P. K. Trivedi. 2005. Microeconometrics: Methods and Applications. New York: Cambridge University Press.
Cragg, J. G., and S. G. Donald. 1993. Testing identifiability and specification in instrumental variable models. Econometric Theory 9: 222-240. https://doi.org/10.1017/S0266466600007519.

Davidson, R., and J. G. MacKinnon. 1993. Estimation and Inference in Econometrics. New York: Oxford University Press.

Durbin, J. 1954. Errors in variables. Review of the International Statistical Institute 22: 23-32. https://doi.org/10.2307/1401917.

Finlay, K., and L. M. Magnusson. 2009. Implementing weak-instrument robust tests for a general class of instrumentalvariables models. Stata Journal 9: 398-421.

Hahn, J., and J. A. Hausman. 2003. Weak instruments: Diagnosis and cures in empirical econometrics. American Economic Review Papers and Proceedings 93: 118-125.
Hall, A. R., G. D. Rudebusch, and D. W. Wilcox. 1996. Judging instrument relevance in instrumental variables estimation. International Economic Review 37: 283-298. https://doi.org/10.2307/2527324.

Hansen, L. P. 1982. Large sample properties of generalized method of moments estimators. Econometrica 50: 1029-1054. https://doi.org/10.2307/1912775.
Hausman, J. A. 1978. Specification tests in econometrics. Econometrica 46: 1251-1271. https://doi.org/10.2307/1913827.

Hayashi, F. 2000. Econometrics. Princeton, NJ: Princeton University Press.
Judge, G. G., W. E. Griffiths, R. C. Hill, H. Lütkepohl, and T.-C. Lee. 1985. The Theory and Practice of Econometrics. 2nd ed. New York: Wiley.

Kleibergen, F. 2007. Generalizing weak instrument robust IV statistics towards multiple parameters, unrestricted covariance matrices and identification statistics. Journal of Econometrics 139: 181-216.
https://doi.org/10.1016/j.jeconom.2006.06.010.
Kleibergen, F., and R. Paap. 2006. Generalized reduced rank tests using the singular value decomposition. Journal of Econometrics 133: 97-126. https://doi.org/10.1016/j.jeconom.2005.02.011.

Mikusheva, A. 2010. Robust confidence sets in the presence of weak instruments. Journal of Econometrics 157: 236-247. https://doi.org/10.1016/j.jeconom.2009.12.003.
Mikusheva, A., and B. P. Poi. 2006. Tests and confidence sets with correct size when instruments are potentially weak. Stata Journal 6: 335-347.

Moreira, M. J. 2003. A conditional likelihood ratio test for structural models. Econometrica 71: 1027-1048. https://doi.org/10.1111/1468-0262.00438.
Nelson, C. R., and R. Startz. 1990. The distribution of the instrumental variable estimator and its \(t\) ratio when the instrument is a poor one. Journal of Business 63: S125-S140.
Pflueger, C. E., and S. Wang. 2015. A robust test for weak instruments in Stata. Stata Journal 15: 216-225.
Poi, B. P. 2006. Jackknife instrumental variables estimation in Stata. Stata Journal 6: 364-376.
Sargan, J. D. 1958. The estimation of economic relationships using instrumental variables. Econometrica 26: 393-415. https://doi.org/10.2307/1907619.

Shea, J. S. 1997. Instrument relevance in multivariate linear models: A simple measure. Review of Economics and Statistics 79: 348-352. https://doi.org/10.1162/rest.1997.79.2.348.
Staiger, D. O., and J. H. Stock. 1997. Instrumental variables regression with weak instruments. Econometrica 65: 557-586. https://doi.org/10.2307/2171753.

Stock, J. H., J. H. Wright, and M. Yogo. 2002. A survey of weak instruments and weak identification in generalized method of moments. Journal of Business and Economic Statistics 20: 518-529. https://doi.org/10.1198/073500102288618658.

Stock, J. H., and M. Yogo. 2005. Testing for weak instruments in linear IV regression. In Identification and Inference for Econometric Models: Essays in Honor of Thomas Rothenberg, ed. D. W. K. Andrews and J. H. Stock, 80-108. New York: Cambridge University Press.
Sun, L. 2018. Implementing valid two-step identification-robust confidence sets for linear instrumental-variables models. Stata Journal 18: 803-825.

Wooldridge, J. M. 1995. Score diagnostics for linear models estimated by two stage least squares. In Advances in Econometrics and Quantitative Economics: Essays in Honor of Professor C. R. Rao, ed. G. S. Maddala, P. C. B. Phillips, and T. N. Srinivasan, 66-87. Oxford: Blackwell.
Wu, D.-M. 1974. Alternative tests of independence between stochastic regressors and disturbances: Finite sample results. Econometrica 42: 529-546. https://doi.org/10.2307/1911789.

\section*{Also see}
\([R]\) ivregress - Single-equation instrumental-variables regression
[U] 20 Estimation and postestimation commands

\section*{Title}
ivtobit - Tobit model with continuous endogenous covariates

Description
Syntax
Remarks and examples
Acknowledgments

\section*{Quick start}

Options for ML estimator
Stored results
References

\section*{Menu}

Options for two-step estimator Methods and formulas
Also see

\section*{Description}
ivtobit fits tobit models where one or more of the covariates are endogenously determined. By default, ivtobit uses maximum likelihood estimation, but Newey's (1987) minimum \(\chi^{2}\) (two-step) estimator can be requested. Both estimators assume that the endogenous covariates are continuous and so are not appropriate for use with discrete endogenous covariates.

\section*{Quick start}

Tobit regression of y 1 on x and endogenous regressor y 2 that is instrumented by z where y 1 is left-censored at its observed minimum
ivtobit \(y 1 \mathrm{x}(\mathrm{y} 2=\mathrm{z}), \mathrm{ll}\)
Same as above, but specify that y1 is left-censored at 0 and right-censored at 20
ivtobit \(\mathrm{y} 1 \mathrm{x}(\mathrm{y} 2=\mathrm{z}), \mathrm{ll}(0) \mathrm{ul}(20)\)
Use Newey's two-step estimator
ivtobit y1 x (y2 = z), ll(0) ul(20) twostep
Same as above, and show first-stage regression results
ivtobit \(\mathrm{y} 1 \mathrm{x}(\mathrm{y} 2=\mathrm{z}), \mathrm{ll}(0) \mathrm{ul}(20)\) twostep first

\section*{Menu}

Statistics \(>\) Endogenous covariates \(>\) Tobit model with endogenous covariates

\section*{Syntax}

Maximum likelihood estimator
\[
\begin{aligned}
& \text { ivtobit depvar }\left[\text { varlist }_{1}\right]\left(\text { varlist }_{2}=\text { varlist }_{\mathrm{iv}}\right)[\text { if }][\text { in }][\text { weight }], \quad \mathrm{ll}[(\#)] \mathrm{ul}[(\#)][\text { mle_options }]
\end{aligned}
\]

Two-step estimator
```

ivtobit depvar [\mp@subsup{varlist }{1}{}](\mp@subsup{\mathrm{ varlist }}{2}{}=\mp@subsup{\mathrm{ varlist }}{\textrm{iv}}{})[\mathrm{ if ] [in] [weight], twostep}

```
    \(11[(\#)] \mathrm{ul}[(\#)][\) tse_options \(]\)
varlist \(_{1}\) is the list of exogenous variables.
varlist \(_{2}\) is the list of endogenous variables.
varlist \(_{\mathrm{iv}}\) is the list of exogenous variables used with varlist \(_{1}\) as instruments for varlist \({ }_{2}\).
mle_options
Description
Model
*ll \([(\#)]\)
*ul \([(\#)]\)
mle
constraints (constraints)
left-censoring limit
right-censoring limit
use conditional maximum-likelihood estimator; the default apply specified linear constraints

SE/Robust
vce (vcetype)
vcetype may be oim, robust, cluster clustvar, opg, bootstrap, or jackknife

Reporting
level(\#)
set confidence level; default is level(95)
first
report first-stage regression
nocnsreport
display_options
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

\section*{Maximization}
maximize_options
control the maximization process
coeflegend
display legend instead of statistics

\footnotetext{
*You must specify at least one of \(11[(\#)]\) and \(u l[(\#)]\).
}

\section*{tse_options}

Description

\section*{Model}
* twostep
* \(11[(\#)]\)
*ul \([(\#)]\)
SE
vce(vcetype)
use Newey's two-step estimator; the default is mle
left-censoring limit
right-censoring limit
vcetype may be twostep, bootstrap, or jackknife
Reporting
level (\#)
first
display_options
coeflegend
set confidence level; default is level (95)
report first-stage regression
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
display legend instead of statistics
*twostep is required. You must specify at least one of \(11[(\#)]\) and \(u l[(\#)]\).
varlist \(_{1}\) and varlist \(_{\mathrm{iv}}\) may contain factor variables; see [U] 11.4.3 Factor variables.
depvar, varlist \({ }_{1}\), varlist \({ }_{2}\), and varlist \({ }_{\mathrm{iv}}\) may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bootstrap, by, collect, jackknife, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands.
fp is allowed with the maximum likelihood estimator.
Weights are not allowed with the bootstrap prefix; see [R] bootstrap.
vce(), first, twostep, and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed with the maximum likelihood estimator. fweights are allowed with
Newey's two-step estimator. See [U] 11.1.6 weight.
coeflegend does not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options for ML estimator}
\(11[(\#)]\) and \(u l[\#)]\) indicate the lower and upper limits for censoring, respectively. You may specify one or both. Observations with depvar \(\leq 11\) () are left-censored; observations with depvar \(\geq \mathrm{ul}\) () are right-censored; and remaining observations are not censored. You do not have to specify the censoring values at all. It is enough to type 11 , ul, or both. When you do not specify a censoring value, ivtobit assumes that the lower limit is the minimum observed in the data (if 11 is specified) and that the upper limit is the maximum (if \(u l\) is specified).
mle requests that the conditional maximum-likelihood estimator be used. This is the default.
constraints (constraints); see [R] Estimation options.

\section*{SE/Robust}
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] ve__option.
level(\#); see [R] Estimation options.
first requests that the parameters for the reduced-form equations showing the relationships between the endogenous variables and instruments be displayed. For the two-step estimator, first shows the first-stage regressions. For the maximum likelihood estimator, these parameters are estimated jointly with the parameters of the tobit equation. The default is not to show these parameter estimates.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap (\#), fvwrapon (style), cformat (\% fmt), pformat (\%fmt), sformat (\% fmt), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate(\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from(init_specs); see [R] Maximize.

Setting the optimization type to technique (bhhh) resets the default vcetype to vce (opg).
The following option is available with ivtobit but is not shown in the dialog box:
coeflegend; see [R] Estimation options.

\section*{Options for two-step estimator}
\(\qquad\) Model
twostep is required and requests that Newey's (1987) efficient two-step estimator be used to obtain the coefficient estimates.
\(11[(\#)]\) and \(\mathrm{ul}[(\#)]\) indicate the lower and upper limits for censoring, respectively. You may specify one or both. Observations with depvar \(\leq 11\) () are left-censored; observations with depvar \(\geq \mathrm{ul}\) () are right-censored; and remaining observations are not censored. You do not have to specify the censoring values at all. It is enough to type 11 , \(u l\), or both. When you do not specify a censoring value, ivtobit assumes that the lower limit is the minimum observed in the data (if 11 is specified) and that the upper limit is the maximum (if \(u l\) is specified).

SE
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (twostep) and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
level(\#); see [R] Estimation options.
first requests that the parameters for the reduced-form equations showing the relationships between the endogenous variables and instruments be displayed. For the two-step estimator, first shows the first-stage regressions. For the maximum likelihood estimator, these parameters are estimated jointly with the parameters of the tobit equation. The default is not to show these parameter estimates.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap (\#), fvwrapon (style), cformat (\% fint), pformat (\% fit ), sformat (\% fint), and nolstretch; see [R] Estimation options.

The following option is available with ivtobit but is not shown in the dialog box: coeflegend; see [R] Estimation options.

\section*{Remarks and examples}
ivtobit fits models with censored dependent variables and endogenous covariates. You can use it to fit a tobit model when you suspect that one or more of the covariates is correlated with the error term. ivtobit is to tobit what ivregress is to linear regression analysis; see [R] ivregress for more information.

Formally, the model is
\[
\begin{aligned}
y_{1 i}^{*} & =\boldsymbol{y}_{2 i} \boldsymbol{\beta}+\boldsymbol{x}_{1 i} \boldsymbol{\gamma}+u_{i} \\
\boldsymbol{y}_{2 i} & =\boldsymbol{x}_{1 i} \boldsymbol{\Pi}_{1}+\boldsymbol{x}_{2 i} \boldsymbol{\Pi}_{2}+\boldsymbol{v}_{i}
\end{aligned}
\]
where \(i=1, \ldots, N ; \boldsymbol{y}_{2 i}\) is a \(1 \times p\) vector of endogenous variables; \(\boldsymbol{x}_{1 i}\) is a \(1 \times k_{1}\) vector of exogenous variables; \(\boldsymbol{x}_{2 i}\) is a \(1 \times k_{2}\) vector of additional instruments; and the equation for \(\boldsymbol{y}_{2 i}\) is written in reduced form. By assumption, \(\left(u_{i}, \boldsymbol{v}_{i}\right) \sim N(\mathbf{0}, \boldsymbol{\Sigma}) . \boldsymbol{\beta}\) and \(\gamma\) are vectors of structural parameters, and \(\Pi_{1}\) and \(\Pi_{2}\) are matrices of reduced-form parameters. We do not observe \(y_{1 i}^{*}\); instead, we observe
\[
y_{1 i}= \begin{cases}a & y_{1 i}^{*}<a \\ y_{1 i}^{*} & a \leq y_{1 i}^{*} \leq b \\ b & y_{1 i}^{*}>b\end{cases}
\]

The order condition for identification of the structural parameters is that \(k_{2} \geq p\). Presumably, \(\boldsymbol{\Sigma}\) is not block diagonal between \(u_{i}\) and \(\boldsymbol{v}_{i}\); otherwise, \(\boldsymbol{y}_{2 i}\) would not be endogenous.

\section*{- Technical note}

This model is derived under the assumption that \(\left(u_{i}, \boldsymbol{v}_{i}\right)\) is independent and identically distributed multivariate normal for all \(i\). The vce (cluster clustvar) option can be used to control for a lack of independence. As with the standard tobit model without endogeneity, if \(u_{i}\) is heteroskedastic, point estimates will be inconsistent.
\(>\) Example 1: Estimation and parameter interpretation
We model the number of hours per week that high school boys spend using social media (hsocial). The data collection process caused the observations on the number of hours spent to be censored at 12 hours. A tobit-type model is therefore reasonable for our data.

We model each boy's number of hours spent using social media as a function of whether he has a smartphone (sphone), whether he has a computer at home (computer), the year in high school in which he is enrolled (year), and the hours per week he spends studying (hstudy).

We believe that there are unobservable variables that simultaneously affect hstudy and hsocial, which is to say that hstudy is endogenous. Because hstudy is endogenous, we must model it as well. Our model for the endogenous hstudy always includes the exogenous covariates used to model the outcome hsocial. We must also include at least one covariate in the model for the endogenous hstudy that was not included in the model for the outcome hsocial.

We use ivtobit with the default maximum-likelihood estimator to model the endogenous variable hstudy as a function of the highest educational degree attained by their parents (pedu), the time spent watching television (tvhours), and the exogenous covariates used to model hsocial.
```

. use https://www.stata-press.com/data/r18/smedia
(Fictional data on hours spent on social media)
. ivtobit hsocial i.sphone i.computer i.year (hstudy = tvhours i.pedu), ul(12)
Fitting exogenous tobit model
Fitting full model
Iteration 0: Log likelihood = -3240.5279
Iteration 1: Log likelihood = -3186.8824
Iteration 2: Log likelihood = -3173.1147
Iteration 3: Log likelihood = -3172.8561
Iteration 4: Log likelihood = -3172.856
Tobit model with endogenous regressors Number of obs = 1,324
Uncensored = 928
Limits: Lower = -inf
Upper = 12
Left-censored = 0
Right-censored = 396
Wald chi2(6) = 11610.73
Prob > chi2 = 0.0000

```
\begin{tabular}{r|rrrrrr}
\hline & Coefficient & Std. err. & \multicolumn{1}{c}{z} & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline \begin{tabular}{r} 
hstudy
\end{tabular} & -.9610518 & .0327204 & -29.37 & 0.000 & -1.025183 & -.8969211 \\
1.sphone & 6.041781 & .0625236 & 96.63 & 0.000 & 5.919237 & 6.164325 \\
1.computer & 2.51903 & .0629128 & 40.04 & 0.000 & 2.395723 & 2.642337 \\
year & & & & & & \\
2 & .4439009 & .0802309 & 5.53 & 0.000 & .2866513 & .6011505 \\
3 & .8574705 & .080476 & 10.65 & 0.000 & .6997404 & 1.015201 \\
4 & 1.478215 & .0816582 & 18.10 & 0.000 & 1.318168 & 1.638262 \\
& & & & & & \\
_cons & 8.955813 & .2069144 & 43.28 & 0.000 & 8.550268 & 9.361357 \\
\hline corr(e.hstudy, & & & & & & \\
e.hsocial) & .4667975 & .0340342 & & & .3975006 & .5308 \\
sd(e.hsocial) & .9709836 & .0266364 & & & .9201559 & 1.024619 \\
sd(e.hstudy) & .9701792 & .0188547 & & & .9339197 & 1.007847 \\
\hline
\end{tabular}

Wald test of exogeneity (corr = 0) : chi2 (1) = \(135.19 \quad\) Prob > chi2 \(=0.0000\)
Endogenous: hstudy
Exogenous: 1.sphone 1.computer 2.year 3.year 4.year tvhours 2.pedu 3.pedu
The coefficients in the table tell us how much the linear prediction for the outcome changes when there is a change in a covariate.

Below the table, we see a Wald test for whether the correlation between the residuals from the main equation (predicting hstudy) and the residuals from the auxiliary equation (predicting hsocial) is 0 . The correlation itself is 0.47 and shown in the table as corr (e.hstudy, e.hsocial). If the test statistic is not significant, there is not sufficient information in the sample to reject the null hypothesis of no endogeneity. In our example, we reject the null hypothesis that supports our choice of a tobit model that accounts for endogeneity.
- Technical note

In the tobit model with endogenous covariates, we assume that \(\left(u_{i}, \boldsymbol{v}_{i}\right)\) is multivariate normal with covariance matrix
\[
\operatorname{Var}\left(u_{i}, \boldsymbol{v}_{i}\right)=\boldsymbol{\Sigma}=\left[\begin{array}{cc}
\sigma_{u}^{2} & \boldsymbol{\Sigma}_{21}^{\prime} \\
\boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22}
\end{array}\right]
\]

Using the properties of the multivariate normal distribution, \(\operatorname{Var}\left(u_{i} \mid \boldsymbol{v}_{i}\right) \equiv \sigma_{u \mid v}^{2}=\sigma_{u}^{2}-\boldsymbol{\Sigma}_{21}^{\prime} \boldsymbol{\Sigma}_{22}^{-1} \boldsymbol{\Sigma}_{21}\). Calculating the marginal effects on the conditional expected values of the observed and latent dependent variables and on the probability of censoring requires an estimate of \(\sigma_{u}^{2}\). Unlike the default maximumlikelihood estimator, the two-step estimator identifies only \(\sigma_{u \mid v}^{2}\), not \(\sigma_{u}^{2}\), so only the linear prediction and its standard error are available after you have used the twostep option.

\section*{Stored results}
ivtobit, mle stores the following in e() :
Scalars
e(N)
e(N_unc)
e(N_lc)
e(N_rc)
e(llopt)
e(ulopt)
e(k)
e(k_eq)
e(k_eq_model)
e(k_dv)
e(df_m)
e(11)
e(N_clust)
e(endog_ct)
\(e(p)\)
e(p_exog)
e(chi2)
e(chi2_exog)
e(rank)
e(ic)
\(e(r c)\)
e(converged)
Macros
e(cmd)
e(cmdline)
e(depvar)
\(e\) (endog)
e(exog)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(chi2type)
e(vce)
e(vcetype)
\(e\) (method)
e(opt)
e(which)
e(ml_method)
number of observations
number of uncensored observations
number of left-censored observations
number of right-censored observations
minimum of depvar or contents of 11()
maximum of depvar or contents of ul()
number of parameters
number of equations in \(e(b)\)
number of equations in overall model test
number of dependent variables
model degrees of freedom
log likelihood
number of clusters
number of endogenous covariates
model Wald \(p\)-value
exogeneity test Wald \(p\)-value
model Wald \(\chi^{2}\)
Wald \(\chi^{2}\) test of exogeneity
rank of e(V)
number of iterations
return code
1 if converged, 0 otherwise

\section*{ivtobit}
command as typed
name of dependent variable
names of endogenous variables
names of exogenous variables
weight type
weight expression
title in estimation output
name of cluster variable
Wald; type of model \(\chi^{2}\) test
vcetype specified in vce()
title used to label Std. err.
ml
type of optimization
max or min; whether optimizer is to perform maximization or minimization
type of ml method
```

    e(user) name of likelihood-evaluator program
    e(technique)
    e(properties)
    e(estat_cmd)
    e(predict)
    e(footnote)
    e(marginsok)
    e(marginsprop)
    e(asbalanced)
    e(asobserved)
    Matrices
e(b)
e(Cns)
e(ilog)
e(gradient)
e(Sigma)
e(V)
e(V_modelbased)
Functions
e(sample) marks estimation sample

```

In addition to the above, the following is stored in r() :
Matrices
```

r(table)

```
matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.
ivtobit, twostep stores the following in e():
Scalars
e(N)
e(N_unc)
e(N_lc)
e(N_rc)
e(llopt)
e(ulopt)
e(df_m)
e(df_exog)
\(e(p)\)
e(p_exog)
e(chi2)
e(chi2_exog)
\(e\) (rank)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(endog)
e(exog)
e(wtype)
e(wexp)
e(chi2type)
e(vce)
\(e\) (method)
e(properties)
e(estat_cmd)
e(predict)
number of observations
number of uncensored observations
number of left-censored observations
number of right-censored observations
contents of 11 ()
contents of ul()
model degrees of freedom
degrees of freedom for \(\chi^{2}\) test of exogeneity
model Wald \(p\)-value
exogeneity test Wald \(p\)-value
model Wald \(\chi^{2}\)
Wald \(\chi^{2}\) test of exogeneity
rank of e(V)
ivtobit
command as typed
name of dependent variable
names of endogenous variables
names of exogenous variables
weight type
weight expression
Wald; type of model \(\chi^{2}\) test
vcetype specified in vce()
twostep
b V
program used to implement estat
program used to implement predict
```

    e(footnote)
    e(marginsok)
    e(marginsprop)
    e(asbalanced)
    e(asobserved)
    Matrices
e(b)
e(Cns)
e(V)
e(V_modelbased)

```
Functions
    e(sample)
        program used to implement the footnote display
        predictions allowed by margins
        signals to the margins command
        factor variables fvset as asbalanced
                                factor variables fvset as asobserved
                                coefficient vector
                                constraints matrix
                                variance-covariance matrix of the estimators
                                model-based variance
marks estimation sample

In addition to the above, the following is stored in r() :

\section*{Matrices}
\(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

\section*{Methods and formulas}

The estimation procedure used by ivtobit is similar to that used by ivprobit. For compactness, we write the model as
\[
\begin{align*}
y_{1 i}^{*} & =\boldsymbol{z}_{i} \boldsymbol{\delta}+u_{i}  \tag{1a}\\
\boldsymbol{y}_{2 i} & =\boldsymbol{x}_{i} \boldsymbol{\Pi}+\boldsymbol{v}_{i} \tag{1b}
\end{align*}
\]
where \(\boldsymbol{z}_{i}=\left(\boldsymbol{y}_{2 i}, \boldsymbol{x}_{1 i}\right), \boldsymbol{x}_{i}=\left(\boldsymbol{x}_{1 i}, \boldsymbol{x}_{2 i}\right), \boldsymbol{\delta}=\left(\boldsymbol{\beta}^{\prime}, \boldsymbol{\gamma}^{\prime}\right)^{\prime}\), and \(\boldsymbol{\Pi}=\left(\boldsymbol{\Pi}_{1}^{\prime}, \boldsymbol{\Pi}_{2}^{\prime}\right)^{\prime}\). We do not observe \(y_{1 i}^{*}\); instead, we observe
\[
y_{1 i}= \begin{cases}a & y_{1 i}^{*}<a \\ y_{1 i}^{*} & a \leq y_{1 i}^{*} \leq b \\ b & y_{1 i}^{*}>b\end{cases}
\]
\(\left(u_{i}, \boldsymbol{v}_{i}\right)\) is distributed multivariate normal with mean zero and covariance matrix
\[
\boldsymbol{\Sigma}=\left[\begin{array}{cc}
\sigma_{u}^{2} & \boldsymbol{\Sigma}_{21}^{\prime} \\
\boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22}
\end{array}\right]
\]

Using the properties of the multivariate normal distribution, we can write \(u_{i}=\boldsymbol{v}_{i}^{\prime} \boldsymbol{\alpha}+\epsilon_{i}\), where \(\boldsymbol{\alpha}=\boldsymbol{\Sigma}_{22}^{-1} \boldsymbol{\Sigma}_{21} ; \epsilon_{i} \sim N\left(0 ; \sigma_{u \mid v}^{2}\right)\), where \(\sigma_{u \mid v}^{2}=\sigma_{u}^{2}-\boldsymbol{\Sigma}_{21}^{\prime} \boldsymbol{\Sigma}_{22}^{-1} \boldsymbol{\Sigma}_{21} ;\) and \(\epsilon_{i}\) is independent of \(\boldsymbol{v}_{i}, \boldsymbol{z}_{i}\), and \(\boldsymbol{x}_{i}\).

The likelihood function is straightforward to derive because we can write the joint density \(f\left(y_{1 i}, \boldsymbol{y}_{2 i} \mid \boldsymbol{x}_{i}\right)\) as \(f\left(y_{1 i} \mid \boldsymbol{y}_{2 i}, \boldsymbol{x}_{i}\right) f\left(\boldsymbol{y}_{2 i} \mid \boldsymbol{x}_{i}\right)\). We have that
\[
\ln f\left(\boldsymbol{y}_{2 i} \mid \boldsymbol{x}_{i}\right)=-\frac{1}{2}\left(p \ln 2 \pi+\ln \left|\boldsymbol{\Sigma}_{22}\right|+\boldsymbol{v}_{i}^{\prime} \boldsymbol{\Sigma}_{22}^{-1} \boldsymbol{v}_{i}\right)
\]
and
\[
\ln f\left(y_{1 i} \mid \boldsymbol{y}_{2 i}, \boldsymbol{x}_{i}\right)= \begin{cases}\ln \left\{1-\Phi\left(\frac{m_{i}-a}{\sigma_{u \mid v}}\right)\right\} & y_{1 i}=a \\ -\frac{1}{2}\left\{\ln 2 \pi+\ln \sigma_{u \mid v}^{2}+\frac{\left(y_{1 i}-m_{i}\right)^{2}}{\sigma_{u \mid v}^{2}}\right\} & a<y_{1 i}<b \\ \ln \Phi\left(\frac{m_{i}-b}{\sigma_{u \mid v}}\right) & y_{1 i}=b\end{cases}
\]
where
\[
m_{i}=\boldsymbol{z}_{i} \boldsymbol{\delta}+\left(\boldsymbol{y}_{2 i}-\boldsymbol{x}_{i} \boldsymbol{\Pi}\right) \boldsymbol{\Sigma}_{22}^{-1} \boldsymbol{\Sigma}_{21}
\]
and \(\Phi(\cdot)\) is the normal distribution function so that the log likelihood for observation \(i\) is
\[
\ln L_{i}=w_{i}\left\{\ln f\left(y_{1 i} \mid \boldsymbol{y}_{2 i}, \boldsymbol{x}_{i}\right)+\ln f\left(\boldsymbol{y}_{2 i} \mid \boldsymbol{x}_{i}\right)\right\}
\]
where \(w_{i}\) is the weight for observation \(i\) or one if no weights were specified. Instead of estimating \(\sigma_{u \mid v}\) and \(\sigma_{v}\) directly, we estimate \(\ln \sigma_{u \mid v}\) and \(\ln \sigma_{v}\).

With maximum likelihood estimation, this command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce(robust) and vce (cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.

The maximum likelihood version of ivtobit also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

The two-step estimates are obtained using Newey's (1987) minimum \(\chi^{2}\) estimator. For more details on the minimum \(\chi^{2}\) estimator, see \([R]\) ivprobit.

\section*{Acknowledgments}

The two-step estimator is based on the tobitiv command written by Jonah Gelbach of the University of California at Berkeley Law School and the ivtobit command written by Joe Harkness of the University of Connecticut.

\section*{References}

Finlay, K., and L. M. Magnusson. 2009. Implementing weak-instrument robust tests for a general class of instrumentalvariables models. Stata Journal 9: 398-421.
Miranda, A., and S. Rabe-Hesketh. 2006. Maximum likelihood estimation of endogenous switching and sample selection models for binary, ordinal, and count variables. Stata Journal 6: 285-308.

Newey, W. K. 1987. Efficient estimation of limited dependent variable models with endogenous explanatory variables. Journal of Econometrics 36: 231-250. https://doi.org/10.1016/0304-4076(87)90001-7.

\section*{Also see}
[R] ivtobit postestimation - Postestimation tools for ivtobit
[R] gmm - Generalized method of moments estimation
[R] ivprobit - Probit model with continuous endogenous covariates
\([\mathrm{R}]\) ivregress - Single-equation instrumental-variables regression
\([R]\) regress - Linear regression
[R] tobit - Tobit regression
[ERM] eintreg - Extended interval regression
[SVY] svy estimation - Estimation commands for survey data
[XT] xtintreg - Random-effects interval-data regression models
[XT] xttobit - Random-effects tobit models
[U] 20 Estimation and postestimation commands

\section*{Title}
ivtobit postestimation - Postestimation tools for ivtobit
\begin{tabular}{llll} 
Postestimation commands & predict & margins & estat \\
Remarks and examples & Stored results & Methods and formulas & References
\end{tabular}

\section*{Postestimation commands}

The following postestimation commands are of special interest after ivtobit:
Command Description
estat correlation report the correlation matrix of the errors of the dependent variable and the endogenous variables
estat covariance report the covariance matrix of the errors of the dependent variable and the endogenous variables

These commands are not appropriate after the two-step estimator or with svy estimation results.

The following standard postestimation commands are also available:
Command Description
\begin{tabular}{|c|c|}
\hline contrast & contrasts and ANOVA-style joint tests of estimates \\
\hline *estat ic & Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) \\
\hline estat summarize & summary statistics for the estimation sample \\
\hline estat vce & variance-covariance matrix of the estimators (VCE) \\
\hline estat (svy) & postestimation statistics for survey data \\
\hline estimates & cataloging estimation results \\
\hline etable & table of estimation results \\
\hline \({ }^{\dagger}\) forecast & dynamic forecasts and simulations \\
\hline \(\dagger\) hausman & Hausman's specification test \\
\hline lincom & point estimates, standard errors, testing, and inference for linear combinations of coefficients \\
\hline \(\dagger\) lrtest & likelihood-ratio test; not available with two-step estimator \\
\hline margins & marginal means, predictive margins, marginal effects, and average marginal effects \\
\hline marginsplot & graph the results from margins (profile plots, interaction plots, etc.) \\
\hline nlcom & point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients \\
\hline predict & linear predictions and their SEs, probabilities, etc. \\
\hline predictnl & point estimates, standard errors, testing, and inference for generalized predictions \\
\hline pwcompare & pairwise comparisons of estimates \\
\hline * suest & seemingly unrelated estimation \\
\hline
\end{tabular}
test Wald tests of simple and composite linear hypotheses
testnl Wald tests of nonlinear hypotheses
*estat ic, forecast, and suest are not appropriate after ivtobit, twostep.
\(\dagger_{\text {forecast, hausman, and }}\) lrtest are not appropriate with svy estimation results.

\section*{predict}

\section*{Description for predict}
predict creates a new variable containing predictions such as structural functions, linear predictions, standard errors, probabilities, and expected values.

\section*{Menu for predict}

Statistics \(>\) Postestimation

\section*{Syntax for predict}

After ML
predict [type] newvar [if][in] [, statistic asfmethod]
predict [type] \(\{\) stub*|newvarlist \(\}[\) if \(][\) in \(]\), scores

After twostep
predict [type] newvar [if] [in] [, twostep_statistic]
statistic Description
Main
\(\mathrm{xb} \quad\) linear prediction excluding endogeneity; the default
mean linear prediction accounting for endogeneity
stdp standard error of the linear prediction
stdf standard error of the forecast
\(\operatorname{pr}(a, b) \quad \operatorname{Pr}\left(a<y_{j}<b\right)\) accounting for endogeneity
\(\mathrm{e}(a, b) \quad E\left(y_{j} \mid a<y_{j}<b\right)\) accounting for endogeneity
\(\operatorname{ystar}(a, b) \quad E\left(y_{j}^{*}\right), y_{j}^{*}=\max \left\{a, \min \left(y_{j}, b\right)\right\}\) accounting for endogeneity
stdf is not allowed with svy estimation results.
where \(a\) and \(b\) may be numbers or variables; \(a\) missing ( \(a \geq\).) means \(-\infty\), and \(b\) missing ( \(b \geq\).) means \(+\infty\); see [U] 12.2.1 Missing values.
asfmethod
Description
Main
asf average structural function; the default
fixedasf fixed average structural function

Main
xb
stdp
linear prediction; the default
standard error of the linear prediction

These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

\section*{Options for predict}
\(\qquad\) Main
xb , the default, calculates the linear prediction.
mean calculates the linear prediction. Results depend on how the endogeneity complication is handled, which is determined by the asf or fixedasf option. mean is not available with the two-step estimator.
stdp calculates the standard error of the linear prediction. It can be thought of as the standard error of the predicted expected value or mean for the observation's covariate pattern. The standard error of the prediction is also referred to as the standard error of the fitted value.
stdf calculates the standard error of the forecast, which is the standard error of the point prediction for 1 observation. It is commonly referred to as the standard error of the future or forecast value. By construction, the standard errors produced by stdf are always larger than those produced by stdp; see Methods and formulas in [R] regress postestimation.
\(\operatorname{pr}(a, b)\) calculates \(\operatorname{Pr}\left(a<y_{j}<b \mid \mathbf{z}_{j}\right)\), the probability that \(y_{j} \mid \mathbf{z}_{j}\) would be observed in the interval \((a, b)\).
\(a\) and \(b\) may be specified as numbers or variable names; \(l b\) and \(u b\) are variable names;
\(\operatorname{pr}(20,30)\) calculates \(\operatorname{Pr}\left(20<y_{j}<30 \mid \mathbf{z}_{j}\right)\);
\(\operatorname{pr}(l b, u b)\) calculates \(\operatorname{Pr}\left(l b<y_{j}<u b \mid \mathbf{z}_{j}\right)\); and
\(\operatorname{pr}(20, u b)\) calculates \(\operatorname{Pr}\left(20<y_{j}<u b \mid \mathbf{z}_{j}\right)\).
\(a\) missing \((a \geq\).\() means -\infty\); \(\operatorname{pr}(., 30)\) calculates \(\operatorname{Pr}\left(-\infty<y_{j}<30 \mid \mathbf{z}_{j}\right)\);
\(\operatorname{pr}(l b, 30)\) calculates \(\operatorname{Pr}\left(-\infty<y_{j}<30 \mid \mathbf{z}_{j}\right)\) in observations for which \(l b \geq\).
and calculates \(\operatorname{Pr}\left(l b<y_{j}<30 \mid \mathbf{z}_{j}\right)\) elsewhere.
\(b\) missing ( \(b \geq\).) means \(+\infty\); \(\operatorname{pr}(20,\).\() calculates \operatorname{Pr}\left(+\infty>y_{j}>20 \mid \mathbf{z}_{j}\right)\);
\(\operatorname{pr}(20, u b)\) calculates \(\operatorname{Pr}\left(+\infty>y_{j}>20 \mid \mathbf{z}_{j}\right)\) in observations for which \(u b \geq\).
and calculates \(\operatorname{Pr}\left(20<y_{j}<u b \mid \mathbf{z}_{j}\right)\) elsewhere.
Results depend on how the endogeneity complication is handled, which is determined by the asf or fixedasf option.
\(\operatorname{pr}(a, b)\) is not available with the two-step estimator.
\(\mathrm{e}(a, b)\) calculates \(E\left(y_{j} \mid a<y_{j}<b\right)\), the expected value of \(y_{j} \mid \boldsymbol{z}_{j}\) conditional on \(y_{j} \mid \boldsymbol{z}_{j}\) being in the interval \((a, b)\), meaning that \(y_{j} \mid \boldsymbol{z}_{j}\) is truncated. \(a\) and \(b\) are specified as they are for pr() . Results depend on how the endogeneity complication is handled, which is determined by the asf or fixedasf option. \(\mathrm{e}(a, b)\) is not available with the two-step estimator.
ystar \((a, b)\) calculates \(E\left(y_{j}^{*}\right)\), where \(y_{j}^{*}=a\) if \(\boldsymbol{z}_{i} \boldsymbol{\delta}+u_{j} \leq a\), \(y_{j}^{*}=b\) if \(\boldsymbol{z}_{i} \boldsymbol{\delta}+u_{j} \geq b\), and \(y_{j}^{*}=\boldsymbol{z}_{i} \boldsymbol{\delta}+u_{j}\) otherwise, meaning that \(y_{j}^{*}\) is censored. \(a\) and \(b\) are specified as they are for pr() . Results depend on how the endogeneity complication is handled, which is determined by the asf or fixedasf option. ystar \((a, b)\) is not available with the two-step estimator.
asf and fixedasf determine how the average structural function (ASF) of the specified statistic is computed. These options are not allowed with xb , stdp, or stdf.
asf is the default for the ML estimator when the mean, \(\operatorname{pr}(a, b), \mathrm{e}(a, b)\), or ystar \((a, b)\) statistic is specified. asf computes the ASF of the specified statistic. It is the statistic conditional on the errors of the endogenous variable equations. Put another way, it is the statistic accounting for the correlation of the endogenous covariates with the errors of the main equation. Derivatives and contrasts based on asf have a structural interpretation. See margins for computing derivatives and contrasts.
fixedasf calculates a fixed ASF. It is the specified statistic using only the coefficients and variables of the outcome equation. fixedasf does not condition on the errors of the endogenous variable equations. Contrasts between two fixed counterfactuals averaged over the whole sample have a potential-outcome interpretation. Intuitively, it is as if the values of the covariates were fixed at a value exogenously by fiat. See margins for computing derivatives and contrasts.
To be clear, derivatives and contrasts between two fixed counterfactuals using the default asf option also have a potential-outcome interpretation. And, unlike fixedasf, they retain that interpretation when computed over subpopulations for both linear and nonlinear models.
scores, not available with twostep, calculates equation-level score variables.
For models with one endogenous regressor, five new variables are created.
The first new variable will contain \(\partial \ln L / \partial\left(\boldsymbol{z}_{i} \boldsymbol{\delta}\right)\).
The second new variable will contain \(\partial \ln L / \partial\left(\boldsymbol{x}_{i} \boldsymbol{\Pi}\right)\).
The third new variable will contain \(\partial \ln L / \partial \alpha\).
The fourth new variable will contain \(\partial \ln L / \partial \ln \sigma_{u \mid v}\).
The fifth new variable will contain \(\partial \ln L / \partial \ln \sigma_{v}\).
For models with \(p\) endogenous regressors, \(p+\{(p+1)(p+2)\} / 2+1\) new variables are created.
The first new variable will contain \(\partial \ln L / \partial\left(\boldsymbol{z}_{i} \boldsymbol{\delta}\right)\).
The second through \((p+1)\) th new score variables will contain \(\partial \ln L / \partial\left(\boldsymbol{x}_{i} \boldsymbol{\Pi}_{k}\right), k=1, \ldots, p\), where \(\Pi_{k}\) is the \(k\) th column of \(\Pi\).
The remaining score variables will contain the partial derivatives of \(\ln L\) with respect to the \((p+1)(p+2) / 2\) ancillary parameters.

\section*{margins}

\section*{Description for margins}
margins estimates margins of response for linear predictions, probabilities, and expected values.

\section*{Menu for margins}

Statistics \(>\) Postestimation

\section*{Syntax for margins}
```

margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...) ...] [options]

```

\section*{After ML}
statistic Description

Main
\(\mathrm{xb} \quad\) linear prediction excluding endogeneity; the default
mean linear prediction accounting for endogeneity
stdp not allowed with margins
stdf not allowed with margins
\(\operatorname{pr}(a, b) \quad \operatorname{Pr}\left(a<y_{j}<b\right)\) accounting for endogeneity
\(\mathrm{e}(a, b) \quad E\left(y_{j} \mid a<y_{j}<b\right)\) accounting for endogeneity
\(\underline{\text { ystar }}(a, b) \quad E\left(y_{j}^{*}\right), y_{j}^{*}=\max \left\{a, \min \left(y_{j}, b\right)\right\}\) accounting for endogeneity

After twostep
statistic Description
Main
\(\mathrm{xb} \quad\) linear prediction; the default
stdp not allowed with margins

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see \([R]\) margins.

\section*{estat}

\section*{Description for estat}
estat correlation displays the correlation matrix of the errors of the dependent variable and the endogenous variables.
estat covariance displays the covariance matrix of the errors of the dependent variable and the endogenous variables.
estat correlation and estat covariance are not allowed after the ivprobit two-step estimator.

\section*{Menu for estat}

Statistics \(>\) Postestimation

\section*{Syntax for estat}

Correlation matrix
```

estat correlation [, border(bspec) left(\#) format(% (fmt)]

```

Covariance matrix
```

estat covariance [, border(bspec) left(\#) format(% fmt)]

```

\section*{Options for estat}
\(\qquad\) Main
border (bspec) sets border style of the matrix display. The default is border (all).
left(\#) sets the left indent of the matrix display. The default is left(2).
format (\% fint) specifies the format for displaying the individual elements of the matrix. The default is format ( \(\% 9.0 \mathrm{~g}\) ).

\section*{Remarks and examples}

Remarks are presented under the following headings:

\author{
Marginal effects \\ Obtaining predicted values
}

\section*{Marginal effects}

Below, we discuss the interpretation of predictions with the asf and fixedasf options for the ML estimator using margins.

The model is defined by two equations. The first is the equation for the outcome of interest, given by
\[
y_{1 i}^{*}=\boldsymbol{y}_{2 i} \boldsymbol{\beta}+\boldsymbol{x}_{1 i} \boldsymbol{\gamma}+u_{i}
\]
where we do not observe \(y_{1 i}^{*}\); instead, we observe
\[
y_{1 i}= \begin{cases}a & y_{1 i}^{*}<a \\ y_{1 i}^{*} & a \leq y_{1 i}^{*} \leq b \\ b & y_{1 i}^{*}>b\end{cases}
\]

The second is the equation for the endogenous covariates, \(\boldsymbol{y}_{2 i}\),
\[
\boldsymbol{y}_{2 i}=\boldsymbol{x}_{1 i} \boldsymbol{\Pi}_{1}+\boldsymbol{x}_{2 i} \boldsymbol{\Pi}_{2}+\boldsymbol{v}_{i}
\]

This last equation is the difference between a standard tobit model and the model fit by ivtobit. \(\boldsymbol{y}_{2 i}\) is modeled by an exogenous component, \(\boldsymbol{x}_{1 i} \boldsymbol{\Pi}_{1}+\boldsymbol{x}_{2 i} \boldsymbol{\Pi}_{2}\), and a component that is correlated with \(u_{i}\) and causes the endogeneity problem, \(\boldsymbol{v}_{i}\). The ASF linear prediction conditions on an estimate of \(\widehat{\boldsymbol{v}}_{i}\). It is given by
\[
\begin{aligned}
& \widehat{m}_{i}=\widehat{E}\left(y_{1 i} \mid \boldsymbol{x}_{1 i}, \boldsymbol{x}_{2 i}, y_{2 i}, \widehat{\boldsymbol{v}}_{i}\right) \\
& \widehat{m}_{i}=\boldsymbol{y}_{2 i} \widehat{\boldsymbol{\theta}}_{1}+\boldsymbol{x}_{1 i} \widehat{\boldsymbol{\theta}}_{2}+\widehat{\boldsymbol{v}}_{i} \widehat{\boldsymbol{\theta}}_{3}
\end{aligned}
\]

Because the correlation between \(\boldsymbol{v}_{i}\) and \(u_{i}\) is the problem we intended to address, conditioning on \(\boldsymbol{v}_{i}\) purges the model of endogeneity. Using the ASF, we can get derivatives and contrast. See Wooldridge (2010) and Blundell and Powell (2003) for an in-depth discussion of ASFs and their interpretation.

The fixed ASF, estimated when the fixedasf option is specified, has a different interpretation. Suppose we wanted to analyze \(\boldsymbol{y}_{2 i} \boldsymbol{\beta}+\boldsymbol{x}_{1 i} \boldsymbol{\gamma}+u_{i}\) at two different values of \(\mathbf{y}_{\mathbf{2}}\), the original \(\mathbf{y}_{\mathbf{2}}\) and \(\mathbf{y}_{2}+1\). We want the average difference at these two points for the given values of the other covariates. The values of the covariates are not arrived at via the model; they are fixed by fiat. You can think of them as potential outcomes. The difference of the two values of \(\mathbf{y}_{\mathbf{2}}\) is given by
\[
\left\{\left(\boldsymbol{y}_{2 i}+1\right) \boldsymbol{\beta}+\boldsymbol{x}_{1 i} \boldsymbol{\gamma}+u_{i}\right\}-\left(\boldsymbol{y}_{2 i} \boldsymbol{\beta}+\boldsymbol{x}_{1 i} \boldsymbol{\gamma}+u_{i}\right)
\]

If we average over the distribution of \(u\), we obtain
\[
E\left\{\left(\boldsymbol{y}_{2 i}+1\right) \boldsymbol{\beta}+\boldsymbol{x}_{1 i} \boldsymbol{\gamma}\right\}-E\left(\boldsymbol{y}_{2 i} \boldsymbol{\beta}+\boldsymbol{x}_{1 i} \boldsymbol{\gamma}\right)
\]

We do not account for endogeneity because the values of the covariates are given and fixed. If the research question you are pursuing after fitting the model has this interpretation, fixedasf gives you an adequate prediction. If, however, you do not want to treat the covariates as fixed, you should account for endogeneity and use asf predictions.

\section*{> Example 1: Obtaining marginal effects}

We can obtain average marginal effects by using the margins command after ivtobit. For the social-media model of example 1 in [R] ivtobit, suppose that we wanted to know the average marginal effects on the probability of spending more than 12 hours using social media. Average marginal effects are equivalent to estimating how a change in a covariate affects every individual in our sample and taking the average of these effects. The effect of each covariate is estimated with all other covariates kept at their observed values.
```

. use https://www.stata-press.com/data/r18/smedia
(Fictional data on hours spent on social media)
. ivtobit hsocial i.sphone i.computer i.year (hstudy = tvhours i.pedu), ul(12)
(output omitted)
. margins, dydx(*) predict(p(12, .))
Average marginal effects Number of obs = 1,324
Model VCE: OIM
Expression: Pr(hsocial>12), predict(p(12, .))
dy/dx wrt: hstudy 1.sphone 1.computer 2.year 3.year 4.year

```
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline & \multicolumn{3}{|l|}{\[
\begin{array}{cc} 
& \text { Delta-method } \\
\mathrm{dy} / \mathrm{dx} \quad \text { std. err. }
\end{array}
\]} & \(P>|z|\) & [95\% conf & interval] \\
\hline hstudy & -. 1508219 & . 0044544 & -33.86 & 0.000 & -. 1595523 & -. 1420914 \\
\hline 1.sphone & . 3943071 & . 0085181 & 46.29 & 0.000 & . 3776119 & . 4110022 \\
\hline 1. computer & . 3993489 & . 0082484 & 48.42 & 0.000 & . 3831824 & . 4155154 \\
\hline year & & & & & & \\
\hline 2 & . 0663908 & . 0120746 & 5.50 & 0.000 & . 042725 & . 0900567 \\
\hline 3 & . 1334667 & . 012488 & 10.69 & 0.000 & . 1089908 & . 1579426 \\
\hline 4 & . 2343393 & . 0123729 & 18.94 & 0.000 & . 2100888 & . 2585897 \\
\hline
\end{tabular}

Note: dy/dx for factor levels is the discrete change from the base level.
Having a smartphone increases the probability of spending more than 12 hours on social media by 0.39 , on average. Any additional study time decreases the probability of spending more than 12 hours using social media by 0.15 , on average. The other average marginal effects are interpreted similarly. All effects above have a structural interpretation because we are conditioning on the level of endogeneity. See Wooldridge (2010) and Blundell and Powell (2003) for an in-depth discussion of ASFs and their interpretation.

\section*{Obtaining predicted values}

After fitting your model with ivtobit, you can obtain the linear prediction and its standard error for both the estimation sample and other samples by using the predict command. If you used the ML estimator, you can also obtain the linear prediction, the conditional expected values of the observed and latent dependent variables, and the probability of observing the dependent variable in a specified interval-each of these can be computed with an ASF or a fixed ASF interpretation. In addition, with the ML estimator, you can obtain the standard error of the forecast. See [U] 20 Estimation and postestimation commands and \([\mathrm{R}]\) predict.

\section*{Stored results}
estat correlation stores the following results in \(r()\) :
Matrices
\(r\) (corr) correlation matrix of the errors
estat covariance stores the following results in \(r()\) :
Matrices
\(r\) (cov) covariance matrix of the errors

\section*{Methods and formulas}

The linear prediction is calculated as \(\boldsymbol{z}_{i} \widehat{\boldsymbol{\delta}}\), where \(\widehat{\boldsymbol{\delta}}\) is the estimated value of \(\boldsymbol{\delta}\), and \(\boldsymbol{z}_{i}\) and \(\boldsymbol{\delta}\) are defined in \((1 a)\) of \([\mathrm{R}]\) ivtobit. Expected values and probabilities are calculated using the same formulas as those used by the standard tobit model. However, when we use the default asf option with mean, \(\operatorname{pr}(a, b)\), e \((a, b)\), or ystar \((a, b)\), instead of evaluating the standard normal cumulative density and probability density at the linear prediction, we evaluate expected values and probabilities at \(\widehat{m}_{i}\), where \(\widehat{m}_{i}\) is defined in Methods and formulas of [R] ivtobit. Using \(\widehat{m}_{i}\) instead of \(\boldsymbol{z}_{i} \widehat{\boldsymbol{\delta}}\) in the formulas produces the ASF, which accounts for endogeneity. The fixed ASF, obtained with the fixedasf option, evaluates the statistic at \(\boldsymbol{z}_{i} \widehat{\boldsymbol{\delta}}\).

\section*{References}

Blundell, R. W., and J. L. Powell. 2003. Endogeneity in nonparametric and semiparametric regression models. In Advances in Economics and Econometrics: Theory and Applications, Eighth World Congress, ed. M. Dewatripont, L. P. Hansen, and S. J. Turnovsky, vol. 2, 312-357. Cambridge: Cambridge University Press.

Wooldridge, J. M. 2010. Econometric Analysis of Cross Section and Panel Data. 2nd ed. Cambridge, MA: MIT Press.

\section*{Also see \\ [R] ivtobit - Tobit model with continuous endogenous covariates \\ [U] 20 Estimation and postestimation commands}

\section*{Title}
jackknife - Jackknife estimation
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
jackknife performs jackknife estimation of the specified statistics (or expressions) for a Stata command or a user-written program. Statistics are jackknifed by estimating the command once for each observation or cluster in the dataset, leaving the associated observation or cluster out of the calculations. jackknife is designed for use with nonestimation commands, functions of coefficients, or user-written programs. To jackknife coefficients, we recommend using the vce (jackknife) option when allowed by the estimation command.

\section*{Quick start}

Jackknife estimate of the standard deviation of v1 returned by summarize in \(r\) (sd) jackknife sd=r(sd), rclass: summarize v1

Jackknife estimate of the statistic \(r\) (mystat) returned by rclass program myprog1 that returns the sample size in \(r(N)\)
jackknife stat=r(mystat), rclass: myprog1 v1
Same as above, and save the results from each replication in mydata.dta
jackknife stat=r(mystat), rclass saving(mydata): myprog1 v1
Jackknife estimate of a difference in coefficients estimated by regress
jackknife diff=(_b[x2]-_b[x1]): regress y x1 x2 x3
Jackknife estimate of the statistic e(mystat) returned by eclass program myprog2 that returns the sample size in \(e(N)\)
jackknife stat=e(mystat), eclass: myprog2 y x1 x2 x3
Jackknife estimates of coefficients stored in e(b) by myprog2
jackknife _b, eclass: myprog2 y x1 x2 x3
Add variables containing the pseudovalues of the coefficients to the dataset
jackknife _b, eclass keep: myprog2 y x1 x2 x3

\section*{Menu}

Statistics \(>\) Resampling \(>\) Jackknife estimation

\section*{Syntax}
jackknife exp_list \([\), options eform_option \(]\) : command
options

Main
eclass
rclass
n (exp)
Options
cluster (varlist)
idcluster(newvar)
saving (filename, ...)
keep
mse
Reporting
level(\#)
notable
noheader
nolegend
verbose
nodots
dots(\#)
noisily
trace
title (text)
display_options
eform_option
Advanced
nodrop do not drop observations
reject (exp)
coeflegend
variables identifying sample clusters
create new cluster ID variable
keep pseudovalues
use MSE formula for variance estimation
suppress table of results
suppress table header
suppress table legend
display the full table legend
suppress replication dots
display dots every \# replications
display any output from command
trace command
use text as title for jackknife results factor-variable labeling
identify invalid results
display legend instead of statistics
number of observations used is stored in e ( N )
number of observations used is stored in \(r(N)\)
specify exp that evaluates to the number of observations used
save results to filename; save statistics in double precision;
save results to filename every \# replications
set confidence level; default is level(95)
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and
display coefficient table in exponentiated form
command is any command that follows standard Stata syntax. All weight types supported by command are allowed except aweights; see [U] 11.1.6 weight.
collect and svy are allowed; see [U] 11.1.10 Prefix commands.
coeflegend does not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.
\begin{tabular}{|c|c|}
\hline \multirow[t]{3}{*}{exp_list contains} & (name: elist) \\
\hline & elist \\
\hline & eexp \\
\hline \multirow[t]{2}{*}{elist contains} & newvar \(=(\exp )\) \\
\hline & (exp) \\
\hline \multirow[t]{2}{*}{eexp is} & specname \\
\hline & [eqno]specname \\
\hline \multirow[t]{4}{*}{specname is} & _b \\
\hline & _b [] \\
\hline & _se \\
\hline & _se [] \\
\hline \multirow[t]{2}{*}{eqno is} & \# \# \\
\hline & name \\
\hline
\end{tabular}
exp is a standard Stata expression; see [U] 13 Functions and expressions.
Distinguish between [], which are to be typed, and [], which indicate optional arguments.

\section*{Options}
\(\qquad\) Main
eclass, rclass, and n (exp) specify where command stores the number of observations on which it based the calculated results. We strongly advise you to specify one of these options.
eclass specifies that command store the number of observations in \(e(N)\).
rclass specifies that command store the number of observations in \(r(N)\).
n (exp) specifies an expression that evaluates to the number of observations used. Specifying \(n(r(N))\) is equivalent to specifying the rclass option. Specifying \(n(e(N))\) is equivalent to specifying the eclass option. If command stores the number of observations in \(r(N 1)\), specify \(n(r(N 1))\).

If you specify no options, jackknife will assume eclass or rclass, depending on which of \(e(N)\) and \(r(N)\) is not missing (in that order). If both \(e(N)\) and \(r(N)\) are missing, jackknife assumes that all observations in the dataset contribute to the calculated result. If that assumption is incorrect, the reported standard errors will be incorrect. For instance, say that you specify
```

. jackknife coef=_b[x2]: myreg y x1 x2 x3

```
where myreg uses \(e(n)\) instead of \(e(N)\) to identify the number of observations used in calculations. Further assume that observation 42 in the dataset has \(x 3\) equal to missing. The 42 nd observation plays no role in obtaining the estimates, but jackknife has no way of knowing that and will use the wrong \(N\). If, on the other hand, you specify
```

. jackknife coef=_b[x2], n(e(n)): myreg y x1 x2 x3

```
jackknife will notice that observation 42 plays no role. The \(n(e(n))\) option is specified because myreg is an estimation command but it stores the number of observations used in \(e(n)\) (instead of the standard \(e(N)\) ). When jackknife runs the regression omitting the 42 nd observation, jackknife will observe that \(e(n)\) has the same value as when jackknife previously ran the regression using all the observations. Thus jackknife will know that myreg did not use the observation.
cluster (varlist) specifies the variables identifying sample clusters. If cluster() is specified, one cluster is left out of each call to command, instead of 1 observation.
idcluster (newvar) creates a new variable containing a unique integer identifier for each resampled cluster, starting at 1 and leading up to the number of clusters. This option may be specified only when the cluster() option is specified. idcluster() helps identify the cluster to which a pseudovalue belongs.
saving (filename [, suboptions]) creates a Stata data file (.dta file) consisting of (for each statistic in exp_list) a variable containing the replicates.
double specifies that the results for each replication be saved as doubles, meaning 8-byte reals. By default, they are saved as floats, meaning 4-byte reals. This option may be used without the saving() option to compute the variance estimates by using double precision.
every (\#) specifies that results be written to disk every \#th replication. every () should be specified only in conjunction with saving () when command takes a long time for each replication. This option will allow recovery of partial results should some other software crash your computer. See [ P\(]\) postfile.
replace specifies that filename be overwritten if it exists. This option does not appear in the dialog box.
keep specifies that new variables be added to the dataset containing the pseudovalues of the requested
statistics. For instance, if you typed
```

. jackknife coef=_b[x2], eclass keep: regress y x1 x2 x3

```
new variable coef would be added to the dataset containing the pseudovalues for \(\quad \mathrm{b}\) [x2]. Let \(b\) be the value of \(\_\mathrm{b}[\mathrm{x} 2]\) when all observations are used to fit the model, and let \(b(j)\) be the value when the \(j\) th observation is omitted. The pseudovalues are defined as
\[
\text { pseudovalue }_{j}=N\{b-b(j)\}+b(j)
\]
where \(N\) is the number of observations used to produce \(b\).
When the cluster () option is specified, each cluster is given at most one nonmissing pseudovalue. The keep option implies the nodrop option.
mse specifies that jackknife compute the variance by using deviations of the replicates from the observed value of the statistics based on the entire dataset. By default, jackknife computes the variance by using deviations of the pseudovalues from their mean.

\section*{Reporting}
level(\#); see [R] Estimation options.
notable suppresses the display of the table of results.
noheader suppresses the display of the table header. This option implies nolegend.
nolegend suppresses the display of the table legend. The table legend identifies the rows of the table with the expressions they represent.
verbose specifies that the full table legend be displayed. By default, coefficients and standard errors are not displayed.
nodots and dots(\#) specify whether to display replication dots. By default, one dot character is displayed for each successful replication. An " x " is displayed if command returns an error or if
any value in exp_list is missing. You can also control whether dots are displayed using set dots; see \([R]\) set.
nodots suppresses display of the replication dots.
dots (\#) displays dots every \# replications. dots(0) is a synonym for nodots.
noisily specifies that any output from command be displayed. This option implies the nodots option.
trace causes a trace of the execution of command to be displayed. This option implies the noisily option.
title (text) specifies a title to be displayed above the table of jackknife results; the default title is Jackknife results or what is produced in e(title) by an estimation command.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\% fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.
eform_option causes the coefficient table to be displayed in exponentiated form; see [R] eform_option. command determines which eform_option is allowed (eform(string) and eform are always allowed).
command determines which of the following are allowed (eform(string) and eform are always allowed):
\begin{tabular}{ll} 
eform_option & Description \\
\hline eform(string) & use string for the column title \\
eform & \begin{tabular}{l} 
exponentiated coefficient, string is \(\exp (\mathrm{b})\) \\
hr
\end{tabular} \\
hazard ratio, string is Haz . ratio \\
shr & subhazard ratio, string is SHR \\
irr & \begin{tabular}{l} 
incidence-rate ratio, string is IRR \\
or
\end{tabular} \\
\(\underline{\mathrm{rrr}}\) & \begin{tabular}{l} 
odds ratio, string is Odds ratio \\
relative-risk ratio, string is RRR
\end{tabular} \\
\hline
\end{tabular}

Advanced
nodrop prevents observations outside e(sample) and the if and in qualifiers from being dropped before the data are resampled.
reject (exp) identifies an expression that indicates when results should be rejected. When exp is true, the resulting values are reset to missing values.

The following option is available with jackknife but is not shown in the dialog box:
coeflegend; see [R] Estimation options.

\section*{Remarks and examples}

Remarks are presented under the following headings:

\section*{Introduction}

Although the jackknife-developed in the late 1940s and early 1950s-is of largely historical interest today, it is still useful in searching for overly influential observations. This feature is often forgotten. In any case, the jackknife is
- an alternative, first-order unbiased estimator for a statistic;
- a data-dependent way to calculate the standard error of the statistic and to obtain significance levels and confidence intervals; and
- a way of producing measures called pseudovalues for each observation, reflecting the observation's influence on the overall statistic.

The idea behind the simplest form of the jackknife-the one implemented here-is to repeatedly calculate the statistic in question, each time omitting just one of the dataset's observations. Assume that our statistic of interest is the sample mean. Let \(y_{j}\) be the \(j\) th observation of our data on some measurement \(y\), where \(j=1, \ldots, N\) and \(N\) is the sample size. If \(\bar{y}\) is the sample mean of \(y\) using the entire dataset and \(\bar{y}_{(j)}\) is the mean when the \(j\) th observation is omitted, then
\[
\bar{y}=\frac{(N-1) \bar{y}_{(j)}+y_{j}}{N}
\]

Solving for \(y_{j}\), we obtain
\[
y_{j}=N \bar{y}-(N-1) \bar{y}_{(j)}
\]

These are the pseudovalues that jackknife calculates. To move this discussion beyond the sample mean, let \(\widehat{\theta}\) be the value of our statistic (not necessarily the sample mean) using the entire dataset, and let \(\widehat{\theta}_{(j)}\) be the computed value of our statistic with the \(j\) th observation omitted. The pseudovalue for the \(j\) th observation is
\[
\widehat{\theta}_{j}^{*}=N \widehat{\theta}-(N-1) \widehat{\theta}_{(j)}
\]

The mean of the pseudovalues is the alternative, first-order unbiased estimator mentioned above, and the standard error of the mean of the pseudovalues is an estimator for the standard error of \(\widehat{\theta}\) (Tukey 1958).

The jackknife estimate of variance has been largely replaced by the bootstrap estimate (see \([R]\) bootstrap), which is widely viewed as more efficient and robust. The use of jackknife pseudovalues to detect outliers is too often forgotten and is something the bootstrap does not provide. See Mosteller and Tukey (1977, 133-163) and Mooney and Duval (1993, 22-27) for more information.

\section*{Using jackknife}

\section*{Typing}
. jackknife exp_list: command
executes command once for each observation in the dataset, leaving the associated observation out of the calculations that make up exp_list.
command defines the statistical command to be executed. Most Stata commands and user-written programs can be used with jackknife, as long as they follow standard Stata syntax and allow the if qualifier; see [U] 11 Language syntax. The by prefix may not be part of command.
exp_list specifies the statistics to be collected from the execution of command. If command changes the contents in e(b), exp_list is optional and defaults to _b.

When the cluster() option is given, clusters are omitted instead of observations, and \(N\) is the number of clusters instead of the sample size.

\section*{Example 1}

As our first example, we will show that the jackknife standard error of the sample mean is equivalent to the standard error of the sample mean computed using the classical formula in the ci means command. We use the double option to compute the standard errors with the same precision as the ci means command.
```

. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. jackknife r(mean), double: summarize mpg
(running summarize on estimation sample)
Jackknife replications (74): .........10.........20..........30.......................
> ...50.........60.........70.... done
Jackknife results Number of obs = 74
Replications = 74
Command: summarize mpg
_jk_1: r(mean)
n(): r(N)

```
\begin{tabular}{c|cccccr}
\hline & \multicolumn{3}{|c}{ Jackknife } \\
& Coefficient & std. err. & t & \(\mathrm{P}>\mid \mathrm{tI}\) & [95\% conf. interval] \\
\hline\(-j k_{-} 1\) & 21.2973 & .6725511 & 31.67 & 0.000 & 19.9569 & 22.63769 \\
\hline
\end{tabular}
\begin{tabular}{r|rrrrr}
. ci means mpg \\
Variable & Obs & Mean & Std. err. & [95\% conf. interval] \\
\hline mpg & 74 & 21.2973 & .6725511 & 19.9569 & 22.63769
\end{tabular}

\section*{Jackknifed standard deviation}

\section*{Example 2}

Mosteller and Tukey (1977, 139-140) request a 95\% confidence interval for the standard deviation of the 11 values:
\[
0.1, \quad 0.1, \quad 0.1, \quad 0.4, \quad 0.5, \quad 1.0, \quad 1.1, \quad 1.3, \quad 1.9, \quad 1.9, \quad 4.7
\]

Stata's summarize command calculates the mean and standard deviation and stores them as \(r\) (mean) and \(r\) (sd). To obtain the jackknifed standard deviation of the 11 values and save the pseudovalues as a new variable, sd, we would type
```

. clear
input x
x
1. 0.1
2. 0.1
3. 0.1
4. 0.4
5. 0.5
6. 1.0
7. 1.1
8. 1.3
9. 1.9
10. 1.9
11. 4.7
12. end
. jackknife sd=r(sd), rclass keep: summarize x
(running summarize on estimation sample)
Jackknife replications (11): .........10. done

```
Jackknife results \(\quad\) Number of obs \(=11\)
                                    Replications \(=11\)
```

Command: summarize x
sd: r(sd)
n(): r(N)

```
\begin{tabular}{c|cccccr}
\hline & Coefficient & \begin{tabular}{l} 
Jackknife \\
std. err.
\end{tabular} & \(t\) & P>|t| & [95\% conf. interval] \\
\hline sd & 1.343469 & .624405 & 2.15 & 0.057 & -.047792 & 2.73473 \\
\hline
\end{tabular}

Interpreting the output, the standard deviation reported by summarize mpg is 1.34 . The jackknife standard error is 0.62 . The \(95 \%\) confidence interval for the standard deviation is -0.048 to 2.73 .

By specifying keep, jackknife creates in our dataset a new variable, sd, for the pseudovalues.
\begin{tabular}{|c|c|c|}
\hline & x & sd \\
\hline 1. & . 1 & 1.139977 \\
\hline 2. & . 1 & 1.139977 \\
\hline 3. & . 1 & 1.139977 \\
\hline 4. & . 4 & . 8893147 \\
\hline 5. & . 5 & . 824267 \\
\hline 6. & 1 & . 632489 \\
\hline 7. & 1.1 & . 6203189 \\
\hline 8. & 1.3 & . 6218889 \\
\hline 9. & 1.9 & . 835419 \\
\hline 10. & 1.9 & . 835419 \\
\hline 11. & 4.7 & 7.703949 \\
\hline
\end{tabular}

The jackknife estimate is the average of the sd variable, so sd contains the individual values of our statistic. We can see that the last observation is substantially larger than the others. The last observation is certainly an outlier, but whether that reflects the considerable information it contains or indicates that it should be excluded from analysis depends on the context of the problem. Mosteller and Tukey created the data by sampling from an exponential distribution, so the observation is informative.

\section*{> Example 3}

Let's repeat the example above using the automobile dataset, obtaining the standard error of the standard deviation of mpg.
```

. use https://www.stata-press.com/data/r18/auto, clear
(1978 automobile data)
. jackknife sd=r(sd), rclass keep: summarize mpg
(running summarize on estimation sample)
Jackknife replications (74): .........10.........20...........30...........40........
> ...50........60........70.... done
Jackknife results Number of obs = 74
Replications = 74

```
```

Command: summarize mpg
sd: r(sd)
n(): r(N)

```
\begin{tabular}{c|cccccc}
\hline & Coefficient & \begin{tabular}{l} 
Jackknife \\
std. err.
\end{tabular} & t & \(\mathrm{P}>\mid \mathrm{t\mid}\) & [95\% conf. interval] \\
\hline sd & 5.785503 & .6072509 & 9.53 & 0.000 & 4.575254 & 6.995753 \\
\hline
\end{tabular}

Let's look at sd more carefully:
```

. summarize sd, detail

```
                            pseudovalues: r(sd)
\begin{tabular}{rcclr}
\hline & Percentiles & Smallest & & \\
\(1 \%\) & 2.870471 & 2.870471 & & 74 \\
\(5 \%\) & 2.870471 & 2.870471 & & 74 \\
\(10 \%\) & 2.906255 & 2.870471 & Obs & \\
\(25 \%\) & 3.328489 & 2.870471 & Sum of wgt. & \\
\(50 \%\) & 3.948335 & & Mean & 5.817374 \\
& & Largest & Std. dev. & 5.22377 \\
\(75 \%\) & 6.844418 & 17.34316 & & \\
\(90 \%\) & 9.597018 & 19.7617 & Variance & 27.28777 \\
\(95 \%\) & 17.34316 & 19.7617 & Skewness & 4.07202 \\
\(99 \%\) & 38.60905 & 38.60905 & Kurtosis & 23.37823
\end{tabular}
. list make mpg sd if sd > 30
71.
\begin{tabular}{|lrr|}
\hline make & mpg & sd \\
\hline VW Diesel & 41 & 38.60905 \\
\hline
\end{tabular}

Here the VW Diesel is the only diesel car in our dataset.

\section*{Collecting multiple statistics}

\section*{> Example 4}
jackknife is not limited to collecting just one statistic. For instance, we can use summarize, detail and then obtain the jackknife estimate of the standard deviation and skewness. summarize, detail stores the standard deviation in \(r\) (sd) and the skewness in \(r\) (skewness), so we might type
```

. use https://www.stata-press.com/data/r18/auto, clear
(1978 automobile data)
. jackknife sd=r(sd) skew=r(skewness), rclass: summarize mpg, detail
(running summarize on estimation sample)
Jackknife replications (74): .........10.........20..........30.........40.......
> ...50........60........70.... done
Jackknife results Number of obs = 74
Replications = 74
Command: summarize mpg, detail
sd: r(sd)
skew: r(skewness)
n(): r(N)

|  | Coefficient | Jackknife std. err. | t | $P>\|t\|$ | [95\% conf | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| sd | 5.785503 | . 6072509 | 9.53 | 0.000 | 4.575254 | 6.995753 |
| skew | . 9487176 | . 3367242 | 2.82 | 0.006 | . 2776272 | 1.619808 |

```

\section*{Collecting coefficients}

\section*{> Example 5}
jackknife can also collect coefficients from estimation commands. For instance, using auto.dta, we might wish to obtain the jackknife standard errors of the coefficients from a regression in which we model the mileage of a car by its weight and trunk space. To do this, we could refer to the coefficients as _b[weight], _b[trunk], _se[weight], and _se[trunk] in the exp_list, or we could simply use the extended expressions _b. In fact, jackknife assumes _b by default when used with estimation commands.


If you are going to use jackknife to estimate standard errors of model coefficients, we recommend using the vce (jackknife) option when it is allowed with the estimation command; see [R] vce_option.
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline \multicolumn{5}{|l|}{Linear regression} & \begin{tabular}{l}
Number of obs Replications F(2, 73) \\
Prob > F \\
R-squared \\
Adj R-squared Root MSE
\end{tabular} & \[
\begin{array}{lr}
= & 74 \\
= & 74 \\
= & 78.10 \\
= & 0.0000 \\
= & 0.6543 \\
= & 0.6446 \\
= & 3.4492
\end{array}
\] \\
\hline mpg & Coefficient & Jackknife std. err. & t & \(P>|t|\) & [95\% conf. & interval] \\
\hline weight & -. 0056527 & . 0010216 & -5.53 & 0.000 & -. 0076887 & -. 0036167 \\
\hline trunk & -. 096229 & . 1486236 & -0.65 & 0.519 & -. 3924354 & . 1999773 \\
\hline _cons & 39.68913 & 1.873324 & 21.19 & 0.000 & 35.9556 & 43.42266 \\
\hline
\end{tabular}
- Technical note

When the jackknife prefix is used with a user-defined program and when the expression list is _b, jackknife calls
```

set coeftabresults off

```
before entering the replication loop to prevent Stata from performing unnecessary calculations. This means that, provided option noisily is not specified, estimation commands will not build or post the coefficient table matrix \(r\) (table).

If your program calls an estimation command and needs \(r\) (table) to exist to perform properly, then your program will need to call
```

set coeftabresults on

```
before calling other estimation commands.

John Wilder Tukey (1915-2000) was born in Massachusetts. He studied chemistry at Brown and mathematics at Princeton and afterward worked at both Princeton and Bell Labs, as well as being involved in a great many government projects, consultancies, and committees. He made outstanding contributions to several areas of statistics, including time series, multiple comparisons, robust statistics, and exploratory data analysis. Tukey was extraordinarily energetic and inventive, not least in his use of terminology: he is credited with inventing the terms bit and software, in addition to ANOVA, boxplot, data analysis, hat matrix, jackknife, stem-and-leaf plot, trimming, and winsorizing, among many others. Tukey's direct and indirect impacts mark him as one of the greatest statisticians of all time.

\section*{Stored results}
jackknife stores the following in e():
Scalars
e(N)
e(N_reps)
e(N_misreps)
e(N_clust)
\(e\left(k \_e q\right)\)
e(k_extra)
e(k_exp)
e(k_eexp)
e(df_r)
Macros
e(cmdname)
e(cmd)
e(command)
e(cmdline)
e(prefix)
e(wtype)
e(wexp)
e(title)
e(cluster)
e(pseudo)
e(nfunction)
e(exp\#)
sample size
number of complete replications
number of incomplete replications
number of clusters
number of equations in \(e(b)\)
number of extra equations
number of expressions
number of extended expressions (_b or _se)
degrees of freedom
command name from command
same as e(cmdname) or jackknife
command
command as typed
jackknife
weight type
weight expression
title in estimation output
cluster variables
new variables containing pseudovalues
\(e(N), r(N), n()\) option, or empty
expression for the \#th statistic
```

e(mse)
e(vce)
e(vcetype)
e(properties)
e(b)
e(b-jk)
e(V)
e(V_modelbased)

```
Matrices
from mse option
jackknife
title used to label Std. err.
b V
observed statistics
jackknife estimates
jackknife variance-covariance matrix
model-based variance

In addition to the above, the following is stored in r() :

Matrices
\(r(\) table \()\)
matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.
When exp_list is _b, jackknife will also carry forward most of the results already in e() from command.

\section*{Methods and formulas}

Let \(\widehat{\theta}\) be the observed value of the statistic, that is, the value of the statistic calculated using the original dataset. Let \(\widehat{\theta}_{(j)}\) be the value of the statistic computed by leaving out the \(j\) th observation (or cluster); thus \(j=1,2, \ldots, N\) identifies an individual observation (or cluster), and \(N\) is the total number of observations (or clusters). The \(j\) th pseudovalue is given by
\[
\widehat{\theta}_{j}^{*}=\widehat{\theta}_{(j)}+N\left\{\widehat{\theta}-\widehat{\theta}_{(j)}\right\}
\]

When the mse option is specified, the standard error is estimated as
\[
\widehat{\mathrm{se}}=\left\{\frac{N-1}{N} \sum_{j=1}^{N}\left(\widehat{\theta}_{(j)}-\widehat{\theta}\right)^{2}\right\}^{1 / 2}
\]
and the jackknife estimate is
\[
\bar{\theta}_{(.)}=\frac{1}{N} \sum_{j=1}^{N} \widehat{\theta}_{(j)}
\]

Otherwise, the standard error is estimated as
\[
\widehat{\mathrm{se}}=\left\{\frac{1}{N(N-1)} \sum_{j=1}^{N}\left(\widehat{\theta}_{j}^{*}-\bar{\theta}^{*}\right)^{2}\right\}^{1 / 2} \quad \bar{\theta}^{*}=\frac{1}{N} \sum_{j=1}^{N} \widehat{\theta}_{j}^{*}
\]
where \(\bar{\theta}^{*}\) is the jackknife estimate. The variance-covariance matrix is similarly computed.

\section*{References}

Belotti, F., and F. Peracchi. 2020. Fast leave-one-out methods for inference, model selection, and diagnostic checking. Stata Journal 20: 785-804.

Brillinger, D. R. 2002. John W. Tukey: His life and professional contributions. Annals of Statistics 30: 1535-1575. https://doi.org/10.1214/aos/1043351246.
Canette, I. 2014. Using resampling methods to detect influential points. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2014/05/08/using-resampling-methods-to-detect-influential-points/.
Mooney, C. Z., and R. D. Duval. 1993. Bootstrapping: A Nonparametric Approach to Statistical Inference. Newbury Park, CA: Sage.
Mosteller, C. F., and J. W. Tukey. 1977. Data Analysis and Regression: A Second Course in Statistics. Reading, MA: Addison-Wesley.
Overgaard, M., P. K. Andersen, and E. T. Parner. 2023. Pseudo-observations in a multistate setting. Stata Journal 23: 491-517.
Tukey, J. W. 1958. Bias and confidence in not-quite large samples. Abstract in Annals of Mathematical Statistics 29: 614. https://doi.org/10.1214/aoms/1177706647.

\section*{Also see}
[R] jackknife postestimation - Postestimation tools for jackknife
\([R]\) bootstrap - Bootstrap sampling and estimation
[R] permute - Permutation tests
[R] simulate - Monte Carlo simulations
[SVY] svy jackknife - Jackknife estimation for survey data
[U] 13.5 Accessing coefficients and standard errors
[U] 13.6 Accessing results from Stata commands
[U] 20 Estimation and postestimation commands

\section*{Title}
jackknife postestimation - Postestimation tools for jackknife

Postestimation commands predict margins Also see

\section*{Postestimation commands}

The following postestimation commands are available after jackknife:
\begin{tabular}{ll} 
Command & Description \\
\hline contrast & \begin{tabular}{l} 
contrasts and ANOVA-style joint tests of estimates \\
estat ic
\end{tabular} \\
Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian in- \\
formation criteria (AIC, CAIC, AICc, and BIC) \\
estat summarize & \begin{tabular}{l} 
summary statistics for the estimation sample
\end{tabular} \\
estimates & \begin{tabular}{l} 
variance-covariance matrix of the estimators (VCE) \\
cataloging estimation results \\
etable
\end{tabular} \\
hausman of estimation results
\end{tabular}\(\quad\)\begin{tabular}{l} 
Hausman's specification test \\
lincom
\end{tabular}\(\quad\)\begin{tabular}{l} 
point estimates, standard errors, testing, and inference for linear combinations of \\
coefficients
\end{tabular}

The postestimation command is allowed only if it may be used after command.

\section*{predict}

The syntax of predict (and whether predict is even allowed) following jackknife depends on the command used with jackknife.

\section*{margins}

The syntax of margins (and whether margins is even allowed) following jackknife depends on the command used with jackknife.

\section*{Also see}
[R] jackknife - Jackknife estimation
[U] 20 Estimation and postestimation commands

\section*{Title}
kappa - Interrater agreement
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & & &
\end{tabular}

\section*{Description}
kap and kappa calculate the kappa-statistic measure of interrater agreement. kap calculates the statistic for two unique raters or at least two nonunique raters. kappa calculates only the statistic for nonunique raters, but it handles the case where data have been recorded as rating frequencies. kapwgt defines weights used by kap in measuring the importance of disagreements.

\section*{Quick start}

Calculate interrater agreement for values rater1 and rater2 kap rater1 rater2

Add table of assessments
kap rater1 rater2, tab
Same as above, and apply frequency weights defined by wvar
kap rater1 rater2 [fweight=wvar], tab
Agreement for values from three nonunique raters stored in rater1, rater2, and rater3 kap rater1 rater2 rater3

Add values from an additional three raters stored in rater4, rater5, and rater6 kap rater1-rater6

Use weights \(1-|i-j| /(k-1)\) to weight disagreements between rater 1 and rater 2 kap rater1 rater2, wgt(w)

Number of times each subject classified in categories stored in poor, fair, and good kappa poor fair good

\section*{Menu}

\section*{kap: two unique raters}

Statistics \(>\) Epidemiology and related \(>\) Other \(>\) Interrater agreement, two unique raters

\section*{kapwgt}

Statistics \(>\) Epidemiology and related \(>\) Other \(>\) Define weights for the above (kap)

\section*{kap: nonunique raters}

Statistics \(>\) Epidemiology and related \(>\) Other \(>\) Interrater agreement, nonunique raters

\section*{kappa}

Statistics \(>\) Epidemiology and related \(>\) Other \(>\) Interrater agreement, nonunique raters with frequencies

\section*{Syntax}

Interrater agreement, two unique raters
kap varname varname \(_{2}\) [if] [in] [weight] [, options \(]\)

Weights for weighting disagreements kapwgt wgtid \([1 \backslash \# 1[\backslash \# \# 1 \ldots]]\)

Interrater agreement, nonunique raters, variables record ratings for each rater kap varname \({ }_{1}\) varname \(_{2}\) varname \(_{3}[\ldots]\) [if] [in] [weight]

Interrater agreement, nonunique raters, variables record frequency of ratings kappa varlist [if] [in]
options
Description
Main
\begin{tabular}{ll}
\(\underline{\text { tab }}\) & display table of assessments \\
wgt (wgtid) & specify how to weight disagreements; see Options for alternatives \\
absolute & treat rating categories as absolute
\end{tabular}
collect is allowed with kap and kappa; see [U] 11.1.10 Prefix commands.
fweights are allowed; see [U] 11.1.6 weight.

\section*{Options}
\(\qquad\) Main
tab displays a tabulation of the assessments by the two raters.
wgt (wgtid) specifies that wgtid be used to weight disagreements. You can define your own weights by using kapwgt; wgt () then specifies the name of the user-defined matrix. For instance, you might define
```

. kapwgt mine 1 \. . 1 \ 0 . 8 1 \ 0 0 . 8 1

```
and then
```

. kap rata ratb, wgt(mine)

```

Also, two prerecorded weights are available.
wgt (w) specifies weights \(1-|i-j| /(k-1)\), where \(i\) and \(j\) index the rows and columns of the ratings by the two raters and \(k\) is the maximum number of possible ratings.
wgt(w2) specifies weights \(1-\{(i-j) /(k-1)\}^{2}\).
absolute is relevant only if wgt() is also specified. The absolute option modifies how \(i, j\), and \(k\) are defined and how corresponding entries are found in a user-defined weighting matrix. When absolute is not specified, \(i\) and \(j\) refer to the row and column index, not to the ratings themselves. Say that the ratings are recorded as \(\{0,1,1.5,2\}\). There are four ratings; \(k=4\), and \(i\) and \(j\) are still \(1,2,3\), and 4 in the formulas above. Index 3 , for instance, corresponds to rating \(=1.5\). This system is convenient but can, with some data, lead to difficulties.
When absolute is specified, all ratings must be integers, and they must be coded from the set \(\{1,2,3, \ldots\}\). Not all values need be used; integer values that do not occur are simply assumed to be unobserved.

\section*{Remarks and examples}

Remarks are presented under the following headings:
Two raters
More than two raters
The kappa-statistic measure of agreement is scaled to be 0 when the amount of agreement is what would be expected to be observed by chance and 1 when there is perfect agreement. For intermediate values, Landis and Koch (1977a, 165) suggest the following interpretations:
\begin{tabular}{ll} 
below 0.0 & Poor \\
\(0.00-0.20\) & Slight \\
\(0.21-0.40\) & Fair \\
\(0.41-0.60\) & Moderate \\
\(0.61-0.80\) & Substantial \\
\(0.81-1.00\) & Almost perfect
\end{tabular}

\section*{Two raters}

\section*{Example 1}

Consider the classification by two radiologists of 85 xeromammograms as normal, benign disease, suspicion of cancer, or cancer (a subset of the data from Boyd et al. [1982] and discussed in the context of kappa in Altman [1991, 403-405]).


Our dataset contains two variables: rada, radiologist A's assessment, and radb, radiologist B's assessment. Each observation is a patient.

We can obtain the kappa measure of interrater agreement by typing
. kap rada radb
\begin{tabular}{cccccc} 
Agreement & \begin{tabular}{c} 
Expected \\
agreement
\end{tabular} & Kappa & Std. err. & Z & Prob>Z \\
\hline \(63.53 \%\) & \(30.82 \%\) & 0.4728 & 0.0694 & 6.81 & 0.0000
\end{tabular}

If each radiologist had made his determination randomly (but with probabilities equal to the overall proportions), we would expect the two radiologists to agree on \(30.8 \%\) of the patients. In fact, they agreed on \(63.5 \%\) of the patients, or \(47.3 \%\) of the way between random agreement and perfect agreement. The amount of agreement indicates that we can reject the hypothesis that they are making their determinations randomly.

\section*{> Example 2: Weighted kappa, prerecorded weight w}

There is a difference between two radiologists disagreeing about whether a xeromammogram indicates cancer or the suspicion of cancer and disagreeing about whether it indicates cancer or is normal. The weighted kappa attempts to deal with this. kap provides two "prerecorded" weights, w and w2:
\begin{tabular}{|c|c|c|c|c|c|}
\hline \multicolumn{6}{|l|}{Ratings weighted by:} \\
\hline 1.0000 & 0.6667 & 0.33330 & 000 & & \\
\hline 0.6667 & 1.0000 & 0.66670 & 333 & & \\
\hline 0.3333 & 0.6667 & 1.00000 & 667 & & \\
\hline 0.0000 & 0.3333 & 0.66671 & 000 & & \\
\hline Agreement & Expect agreeme & d Kappa & Std. err. & Z & Prob>Z \\
\hline 86.67\% & 69.11 & 0.5684 & 0.0788 & 7.22 & 0.0000 \\
\hline
\end{tabular}

The w weights are given by \(1-|i-j| /(k-1)\), where \(i\) and \(j\) index the rows of columns of the ratings by the two raters and \(k\) is the maximum number of possible ratings. The weighting matrix is printed above the table. Here the rows and columns of the \(4 \times 4\) matrix correspond to the ratings normal, benign, suspicious, and cancerous.

A weight of 1 indicates that an observation should count as perfect agreement. The matrix has 1s down the diagonals - when both radiologists make the same assessment, they are in agreement. A weight of, say, 0.6667 means that they are in two-thirds agreement. In our matrix, they get that score if they are "one apart" - one radiologist assesses cancer and the other is merely suspicious, or one is suspicious and the other says benign, and so on. An entry of 0.3333 means that they are in one-third agreement, or, if you prefer, two-thirds disagreement. That is the score attached when they are "two apart". Finally, they are in complete disagreement when the weight is zero, which happens only when they are three apart-one says cancer and the other says normal.

\section*{> Example 3: Weighted kappa, prerecorded weight w2}

The other prerecorded weight is \(w 2\), where the weights are given by \(1-\{(i-j) /(k-1)\}^{2}\) :
\begin{tabular}{|c|c|c|c|c|c|}
\hline \multicolumn{6}{|l|}{Ratings weighted by:} \\
\hline 1.0000 & 0.8889 & 0.55560 & \multicolumn{3}{|c|}{0.0000} \\
\hline 0.8889 & 1.0000 & 0.88890 & \multicolumn{3}{|c|}{0.5556} \\
\hline 0.5556 & 0.8889 & 1.00000 & \multicolumn{3}{|c|}{0.8889} \\
\hline 0.0000 & 0.5556 & 0.88891. & \multicolumn{3}{|c|}{1.0000} \\
\hline \multicolumn{5}{|c|}{Expected} & Prob>Z \\
\hline 94.77\% & 84.09 & 0.6714 & 0.1079 & 6.22 & 0.0000 \\
\hline
\end{tabular}

The w2 weight makes the categories even more alike and is probably inappropriate here.

\section*{> Example 4: Weighted kappa, user-defined weights}

In addition to using prerecorded weights, we can define our own weights with the kapwgt command. For instance, we might feel that suspicious and cancerous are reasonably similar, that benign and normal are reasonably similar, but that the suspicious/cancerous group is nothing like the benign/normal group:
```

. kapwgt xm 1 \ . 8 1 \ 0 0 1 \ 0 0 . 8 1
. kapwgt xm
1.0000
0.8000 1.0000
0.0000 0.0000 1.0000
0.0000 0.0000 0.8000 1.0000

```

We name the weights xm , and after the weight name, we enter the lower triangle of the weighting matrix, using \to separate rows. We have four outcomes, so we continued entering numbers until we had defined the fourth row of the weighting matrix. If we type kapwgt followed by a name and nothing else, it shows us the weights recorded under that name. Satisfied that we have entered them correctly, we now use the weights to recalculate kappa:
. kap rada radb, wgt (xm)
Ratings weighted by:
\begin{tabular}{rrrrrrr}
1.0000 & 0.8000 & 0.0000 & 0.0000 & & \\
0.8000 & 1.0000 & 0.0000 & 0.0000 & & \\
0.0000 & 0.0000 & 1.0000 & 0.8000 & & \\
0.0000 & 0.0000 & 0.8000 & 1.0000 & Z & Prob>Z \\
Agreement & Expected & agreement & Kappa & Std. err. & & \\
\hline \(80.47 \%\) & \(52.67 \%\) & 0.5874 & 0.0865 & 6.79 & 0.0000
\end{tabular}

\section*{Technical note}

In addition to using weights for weighting the differences in categories, you can specify Stata's traditional weights for weighting the data. In the examples above, we have 85 observations in our dataset - one for each patient. If we only knew the table of outcomes - that there were 21 patients rated normal by both radiologists, etc.-it would be easier to enter the table into Stata and work from it. The easiest way to enter the data is with tabi; see [R] tabulate twoway.

tabi reported the Pearson \(\chi^{2}\) for this table, but we do not care about it. The important thing is that, with the replace option, tabi left the table in memory:
. list in \(1 / 5\)
\begin{tabular}{l|rrr|}
\cline { 2 - 4 } & row & col & pop \\
\cline { 2 - 4 } 1. & 1 & 1 & 21 \\
2. & 1 & 2 & 12 \\
3. & 1 & 3 & 0 \\
4. & 1 & 4 & 0 \\
5. & 2 & 1 & 4 \\
\cline { 2 - 4 } & & &
\end{tabular}

The variable row is radiologist A's assessment, col is radiologist B's assessment, and pop is the number so assessed by both. Thus,
. kap row col [fweight=pop]
Expected
\begin{tabular}{cccccc} 
Agreement & agreement & Kappa & Std. err. & Z & Prob>Z \\
\hline \(63.53 \%\) & \(30.82 \%\) & 0.4728 & 0.0694 & 6.81 & 0.0000
\end{tabular}

If we are going to keep these data, the names row and col are not indicative of what the data reflect. We could type (see [U] 12.6 Dataset, variable, and value labels)
```

. rename row rada
. rename col radb
. label var rada "Radiologist A's assessment"
. label var radb "Radiologist B's assessment"

- label define assess 1 normal 2 benign 3 suspect 4 cancer
. label values rada assess
. label values radb assess
. label data "Altman, page 403"

```
kap's tab option, which can be used with or without weighted data, shows the table of assessments:
\begin{tabular}{|c|c|c|c|c|c|}
\hline Radiologist A's assessment & \multicolumn{5}{|l|}{Radiologist B's assessment} \\
\hline normal & 21 & 12 & 0 & 0 & 33 \\
\hline benign & 4 & 17 & 1 & 0 & 22 \\
\hline suspect & 3 & 9 & 15 & 2 & 29 \\
\hline cancer & 0 & 0 & 0 & 1 & 1 \\
\hline Total & 28 & 38 & 16 & 3 & 85 \\
\hline & Expected & & & & \\
\hline Agreement & agreement & Kappa & Std. err. & Z & Prob>Z \\
\hline 63.53\% & 30.82\% & 0.4728 & 0.0694 & 6.81 & 0.0000 \\
\hline
\end{tabular}

\section*{- Technical note}

You have data on individual patients. There are two raters, and the possible ratings are \(1,2,3\), and 4 , but neither rater ever used rating 3 :
\begin{tabular}{|c|c|c|c|c|}
\hline \multirow[b]{2}{*}{Rater A} & \multicolumn{4}{|c|}{Rater B} \\
\hline & 1 & 2 & 4 & Total \\
\hline 1 & 6 & 4 & 3 & 13 \\
\hline 2 & 5 & 3 & 3 & 11 \\
\hline 4 & 1 & 1 & 26 & 28 \\
\hline Total & 12 & 8 & 32 & 52 \\
\hline
\end{tabular}

Here kap would determine that the ratings are from the set \(\{1,2,4\}\) because those were the only values observed. kap would expect a user-defined weighting matrix to be \(3 \times 3\), and if it were not, kap would issue an error message. In the formula-based weights, the calculation would be based on \(i, j=1,2,3\) corresponding to the three observed ratings \(\{1,2,4\}\).

Specifying the absolute option would clarify that the ratings are \(1,2,3\), and 4 ; it just so happens that rating 3 was never assigned. If a user-defined weighting matrix were also specified, kap would expect it to be \(4 \times 4\) or larger (larger because we can think of the ratings being \(1,2,3,4,5, \ldots\) and it just so happens that ratings \(5,6, \ldots\) were never observed, just as rating 3 was not observed). In the formula-based weights, the calculation would be based on \(i, j=1,2,4\).
\begin{tabular}{|c|c|c|c|c|c|}
\hline \multicolumn{6}{|l|}{Ratings weighted by:} \\
\hline 1.0000 & 0.5000 0.0 & 0.0000 & & & \\
\hline 0.5000 & 1.0000 & 0.5000 & & & \\
\hline 0.0000 & 0.5000 & 1.0000 & & & \\
\hline Agreement & Expected agreement & Kappa & Std. err. & Z & Prob>Z \\
\hline 79.81\% & 57.17\% & 0.5285 & 0.1169 & 4.52 & 0.0000 \\
\hline \multicolumn{6}{|l|}{. kap ratera raterb, wgt(w) absolute} \\
\hline \multicolumn{6}{|l|}{Ratings weighted by:} \\
\hline 1.0000 & 0.6667 & 0.0000 & & & \\
\hline 0.6667 & 1.0000 & 0.3333 & & & \\
\hline 0.0000 & 0.33331 & 1.0000 & & & \\
\hline Agreement & Expected agreement & Kappa & Std. err. & Z & Prob>Z \\
\hline 81.41\% & 55.08\% & 0.5862 & 0.1209 & 4.85 & 0.0000 \\
\hline
\end{tabular}

If all conceivable ratings are observed in the data, specifying absolute makes no difference. For instance, if rater A assigns ratings \(\{1,2,4\}\) and rater B assigns \(\{1,2,3,4\}\), the complete set of assigned ratings is \(\{1,2,3,4\}\), the same that absolute would specify. Without absolute, it makes no difference whether the ratings are coded \(\{1,2,3,4\},\{0,1,2,3\},\{1,7,9,100\},\{0,1,1.5,2.0\}\), or otherwise.

\section*{More than two raters}

For more than two raters, the mathematics are such that the two raters are not considered unique. For instance, if there are three raters, there is no assumption that the three raters who rate the first subject are the same as the three raters who rate the second. Although we call this the "more than two raters" case, it can be used with two raters when the raters' identities vary.

The nonunique rater case can be usefully broken down into three subcases: 1) there are two possible ratings, which we will call positive and negative; 2) there are more than two possible ratings, but the number of raters per subject is the same for all subjects; and 3) there are more than two possible ratings, and the number of raters per subject varies. kappa handles all these cases. To emphasize that there is no assumption of constant identity of raters across subjects, the variables specified contain counts of the number of raters rating the subject into a particular category.

Jacob Cohen (1923-1998) was born in New York City. After studying psychology at City College of New York and New York University, he worked as a medical psychologist until 1959 when he became a full professor in the Department of Psychology at New York University. He made many contributions to research methods, including the kappa measure. He persistently emphasized the value of multiple regression and the importance of power and of measuring effects rather than testing significance.
> Example 5: Two ratings
Fleiss, Levin, and Paik (2003, 612) offer the following hypothetical ratings by different sets of raters on 25 subjects:
\begin{tabular}{rrrrrr} 
& \begin{tabular}{r} 
No. of \\
Subject
\end{tabular} & \begin{tabular}{r} 
No. of \\
raters
\end{tabular} \\
1 & 2 & 2 & 14 & \begin{tabular}{r} 
No. of \\
pos. ratings
\end{tabular} & \begin{tabular}{r} 
No. of \\
Subject
\end{tabular} \\
2 & 2 & 0 & 15 & 2 & 3 \\
3 & 3 & 2 & 16 & 2 & 0 \\
4 & 4 & 3 & 17 & 3 & 2 \\
5 & 3 & 3 & 18 & 2 & 1 \\
6 & 4 & 1 & 19 & 4 & 1 \\
7 & 3 & 0 & 20 & 5 & 1 \\
8 & 5 & 0 & 21 & 3 & 4 \\
9 & 2 & 0 & 22 & 4 & 2 \\
10 & 4 & 4 & 23 & 3 & 0 \\
11 & 5 & 5 & 24 & 3 & 0 \\
12 & 3 & 3 & 25 & 2 & 3 \\
13 & 4 & 4 & & &
\end{tabular}

We have entered these data into Stata, and the variables are called subject, raters, and pos. kappa, however, requires that we specify variables containing the number of positive ratings and negative ratings, that is, pos and raters-pos:
. use https://www.stata-press.com/data/r18/p612
. generate neg = raters-pos
. kappa pos neg
Two-outcomes, multiple raters:
\begin{tabular}{ccc} 
Kappa & \(Z\) & Prob>Z \\
\hline 0.5415 & 5.28 & 0.0000
\end{tabular}

We would have obtained the same results if we had typed kappa neg pos.
> Example 6: More than two ratings, constant number of raters, kappa
Each of 10 subjects is rated into one of three categories by five raters (Fleiss, Levin, and Paik 2003, 615):
```

. use https://www.stata-press.com/data/r18/p615, clear
. list

```
\begin{tabular}{|c|rrrr|}
\cline { 2 - 5 } & subject & cat1 & cat2 & cat3 \\
\cline { 2 - 5 } 1. & 1 & 1 & 4 & 0 \\
2. & 2 & 2 & 0 & 3 \\
3. & 3 & 0 & 0 & 5 \\
4. & 4 & 4 & 0 & 1 \\
5. & 5 & 3 & 0 & 2 \\
\hline 6. & 6 & 1 & 4 & 0 \\
7. & 7 & 5 & 0 & 0 \\
8. & 8 & 0 & 4 & 1 \\
9. & 9 & 1 & 0 & 4 \\
10. & 10 & 3 & 0 & 2 \\
\hline
\end{tabular}

We obtain the kappa statistic:
. kappa cat1-cat3
\begin{tabular}{r|ccc} 
Outcome & Kappa & Z & Prob>Z \\
\hline Category 1 & 0.2917 & 2.92 & 0.0018 \\
Category 2 & 0.6711 & 6.71 & 0.0000 \\
Category 3 & 0.3490 & 3.49 & 0.0002 \\
\hline combined & 0.4179 & 5.83 & 0.0000
\end{tabular}

The first part of the output shows the results of calculating kappa for each of the categories separately against an amalgam of the remaining categories. For instance, the cat1 line is the two-rating kappa, where positive is cat1 and negative is cat2 or cat3. The test statistic, however, is calculated differently (see Methods and formulas). The combined kappa is the appropriately weighted average of the individual kappas. There is considerably less agreement about the rating of subjects into the first category than there is for the second.
> Example 7: More than two ratings, constant number of raters, kap
Now, suppose that we have the same data as in the previous example but that the data are organized differently:
```

. use https://www.stata-press.com/data/r18/p615b
. list

```
\begin{tabular}{r|rrrrrr|}
\cline { 2 - 6 } & subject & rater1 & rater2 & rater3 & rater4 & rater5 \\
\cline { 2 - 7 } 1. & 1 & 1 & 2 & 2 & 2 & 2 \\
2. & 2 & 1 & 1 & 3 & 3 & 3 \\
3. & 3 & 3 & 3 & 3 & 3 & 3 \\
4. & 4 & 1 & 1 & 1 & 1 & 3 \\
5. & 5 & 1 & 1 & 1 & 3 & 3 \\
\hline 6. & 6 & 1 & 2 & 2 & 2 & 2 \\
7. & 7 & 1 & 1 & 1 & 1 & 1 \\
8. & 8 & 2 & 2 & 2 & 2 & 3 \\
9. & 9 & 1 & 3 & 3 & 3 & 3 \\
10. & 10 & 1 & 1 & 1 & 3 & 3 \\
\hline
\end{tabular}

Here we would use kap rather than kappa because the variables record ratings for each rater.
\begin{tabular}{l}
. kap rater1 rater2 rater3 rater4 rater5 \\
There are 5 raters per subject: \\
\multicolumn{2}{l}{ Outcome } & Kappa & Z & Prob>Z \\
\\
\hline 1
\end{tabular}

It does not matter which rater is which when there are more than two raters.
> Example 8: More than two ratings, varying number of raters, kappa
In this unfortunate case, kappa can be calculated, but there is no test statistic for testing against \(\kappa>0\). We do nothing differently - kappa calculates the total number of raters for each subject, and, if it is not a constant, kappa suppresses the calculation of test statistics.
. use https://www.stata-press.com/data/r18/rvary
. list
\begin{tabular}{|c|rrrr|}
\cline { 2 - 5 } & subject & cat1 & cat2 & cat3 \\
\cline { 2 - 5 } 1. & 1 & 1 & 3 & 0 \\
2. & 2 & 2 & 0 & 3 \\
3. & 3 & 0 & 0 & 5 \\
4. & 4 & 4 & 0 & 1 \\
5. & 5 & 3 & 0 & 2 \\
\hline 6. & 6 & 1 & 4 & 0 \\
7. & 7 & 5 & 0 & 0 \\
8. & 8 & 0 & 4 & 1 \\
9. & 9 & 1 & 0 & 2 \\
10. & 10 & 3 & 0 & 2 \\
\hline
\end{tabular}
. kappa cat1-cat3
\begin{tabular}{r|llr} 
Outcome & Kappa & Z & Prob>Z \\
\hline Category 1 & 0.2685 &. &. \\
Category 2 & 0.6457 &. &. \\
Category 3 & 0.2938 &. &.
\end{tabular}

Note: Number of ratings per subject vary; cannot calculate test statistics.

\section*{Example 9: More than two ratings, varying number of raters, kap}

This case is similar to the previous example, but the data are organized differently:
. use https://www.stata-press.com/data/r18/rvary2
. list
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline & subject & rater1 & rater2 & rater3 & rater4 & rater5 \\
\hline 1. & 1 & 1 & 2 & 2 & . & 2 \\
\hline 2. & 2 & 1 & 1 & 3 & 3 & 3 \\
\hline 3. & 3 & 3 & 3 & 3 & 3 & 3 \\
\hline 4. & 4 & 1 & 1 & 1 & 1 & 3 \\
\hline 5. & 5 & 1 & 1 & 1 & 3 & 3 \\
\hline 6. & 6 & 1 & 2 & 2 & 2 & 2 \\
\hline 7. & 7 & 1 & 1 & 1 & 1 & 1 \\
\hline 8. & 8 & 2 & 2 & 2 & 2 & 3 \\
\hline 9. & 9 & 1 & 3 & . & . & 3 \\
\hline 10. & 10 & 1 & 1 & 1 & 3 & 3 \\
\hline
\end{tabular}

Here we specify kap instead of kappa because the variables record ratings for each rater.


Note: Number of ratings per subject vary; cannot calculate test statistics.

\section*{Stored results}
kap and kappa store the following in \(r()\) :
Scalars
\begin{tabular}{|c|c|c|c|}
\hline \(r\) ( N ) & number of subjects (kap only) & r(kappa) & kappa \\
\hline r(prop_o) & observed proportion of agreement (kap & r \((\mathrm{z})\) & \(z\) statistic \\
\hline \[
r \text { (prop_e) }
\] & \begin{tabular}{l}
only) \\
expected proportion of agreement (kap
\end{tabular} & r (se) & standard error for kappa statistic \\
\hline
\end{tabular} only)

\section*{Methods and formulas}

The kappa statistic was first proposed by Cohen (1960). The generalization for weights reflecting the relative seriousness of each possible disagreement is due to Cohen (1968). The analysis-of-variance approach for \(k=2\) and \(m \geq 2\) is due to Landis and Koch (1977b). See Altman (1991, 403-409) or Dunn (2000, chap. 2) for an introductory treatment and Fleiss, Levin, and Paik (2003, chap. 18) for a more detailed treatment. All formulas below are as presented in Fleiss, Levin, and Paik (2003). Let \(m\) be the number of raters, and let \(k\) be the number of rating outcomes.

Methods and formulas are presented under the following headings:
```

kap: m = 2
kappa: m > 2, k=2
kappa:m>2,k>2

```

\section*{kap: \(\mathbf{m}=\mathbf{2}\)}

Define \(w_{i j}(i=1, \ldots, k\) and \(j=1, \ldots, k)\) as the weights for agreement and disagreement ( \(\mathrm{wgt}(\mathrm{)})\), or, if the data are not weighted, define \(w_{i i}=1\) and \(w_{i j}=0\) for \(i \neq j\). If wgt(w) is specified, \(w_{i j}=1-|i-j| /(k-1)\). If wgt(w2) is specified, \(w_{i j}=1-\{(i-j) /(k-1)\}^{2}\).

The observed proportion of agreement is
\[
p_{o}=\sum_{i=1}^{k} \sum_{j=1}^{k} w_{i j} p_{i j}
\]
where \(p_{i j}\) is the fraction of ratings \(i\) by the first rater and \(j\) by the second. The expected proportion of agreement is
\[
p_{e}=\sum_{i=1}^{k} \sum_{j=1}^{k} w_{i j} p_{i \cdot p \cdot j}
\]
where \(p_{i}=\sum_{j} p_{i j}\) and \(p_{\cdot j}=\sum_{i} p_{i j}\).
Kappa is given by \(\widehat{\kappa}=\left(p_{o}-p_{e}\right) /\left(1-p_{e}\right)\).
The standard error of \(\widehat{\kappa}\) for testing against 0 is
\[
\widehat{s}_{0}=\frac{1}{\left(1-p_{e}\right) \sqrt{n}}\left(\left[\sum_{i} \sum_{j} p_{i \cdot p} \cdot p_{\cdot j}\left\{w_{i j}-\left(\bar{w}_{i} .+\bar{w}_{\cdot j}\right)\right\}^{2}\right]-p_{e}^{2}\right)^{1 / 2}
\]
where \(n\) is the number of subjects being rated, \(\bar{w}_{i}\). \(=\sum_{j} p_{\cdot j} w_{i j}\), and \(\bar{w}_{\cdot j}=\sum_{i} p_{i} \cdot w_{i j}\). The test statistic \(Z=\widehat{\kappa} / \widehat{s}_{0}\) is assumed to be distributed \(N(0,1)\).

\section*{kappa: \(m>2, k=2\)}

Each subject \(i, i=1, \ldots, n\), is found by \(x_{i}\) of \(m_{i}\) raters to be positive (the choice as to what is labeled positive is arbitrary).

The overall proportion of positive ratings is \(\bar{p}=\sum_{i} x_{i} /(n \bar{m})\), where \(\bar{m}=\sum_{i} m_{i} / n\). The between-subjects mean square is (approximately)
\[
B=\frac{1}{n} \sum_{i} \frac{\left(x_{i}-m_{i} \bar{p}\right)^{2}}{m_{i}}
\]
and the within-subject mean square is
\[
W=\frac{1}{n(\bar{m}-1)} \sum_{i} \frac{x_{i}\left(m_{i}-x_{i}\right)}{m_{i}}
\]

Kappa is then defined as
\[
\widehat{\kappa}=\frac{B-W}{B+(\bar{m}-1) W}
\]

The standard error for testing against 0 (Fleiss and Cuzick 1979) is approximately equal to and is calculated as
\[
\widehat{s}_{0}=\frac{1}{(\bar{m}-1) \sqrt{n \bar{m}_{H}}}\left\{2\left(\bar{m}_{H}-1\right)+\frac{\left(\bar{m}-\bar{m}_{H}\right)(1-4 \overline{p q})}{\overline{m p q}}\right\}^{1 / 2}
\]
where \(\bar{m}_{H}\) is the harmonic mean of \(m_{i}\) and \(\bar{q}=1-\bar{p}\).
The test statistic \(Z=\widehat{\kappa} / \widehat{s}_{0}\) is assumed to be distributed \(N(0,1)\).

\section*{kappa: \(m>2, k>2\)}

Let \(x_{i j}\) be the number of ratings on subject \(i, i=1, \ldots, n\), into category \(j, j=1, \ldots, k\). Define \(\bar{p}_{j}\) as the overall proportion of ratings in category \(j, \bar{q}_{j}=1-\bar{p}_{j}\), and let \(\widehat{\kappa}_{j}\) be the kappa statistic given above for \(k=2\) when category \(j\) is compared with the amalgam of all other categories. Kappa is
\[
\bar{\kappa}=\frac{\sum_{j} \bar{p}_{j} \bar{q}_{j} \widehat{\kappa}_{j}}{\sum_{j} \bar{p}_{j} \bar{q}_{j}}
\]
(Landis and Koch 1977b). In the case where the number of raters per subject, \(\sum_{j} x_{i j}\), is a constant \(m\) for all \(i\), Fleiss, Nee, and Landis (1979) derived the following formulas for the approximate standard errors. The standard error for testing \(\widehat{\kappa}_{j}\) against 0 is
\[
\widehat{s}_{j}=\left\{\frac{2}{n m(m-1)}\right\}^{1 / 2}
\]
and the standard error for testing \(\bar{\kappa}\) is
\[
\bar{s}=\frac{\sqrt{2}}{\sum_{j} \bar{p}_{j} \bar{q}_{j} \sqrt{n m(m-1)}}\left\{\left(\sum_{j} \bar{p}_{j} \bar{q}_{j}\right)^{2}-\sum_{j} \bar{p}_{j} \bar{q}_{j}\left(\bar{q}_{j}-\bar{p}_{j}\right)\right\}^{1 / 2}
\]

\section*{References}

Altman, D. G. 1991. Practical Statistics for Medical Research. London: Chapman and Hall/CRC.
Boyd, N. F., C. Wolfson, M. Moskowitz, T. Carlile, M. Petitclerc, H. A. Ferri, E. Fishell, A. Gregoire, M. Kiernan, J. D. Longley, I. S. Simor, and A. B. Miller. 1982. Observer variation in the interpretation of xeromammograms. Journal of the National Cancer Institute 68: 357-363. https://doi.org/10.1093/jnci/68.3.357.
Campbell, M. J., D. Machin, and S. J. Walters. 2007. Medical Statistics: A Textbook for the Health Sciences. 4th ed. Chichester, UK: Wiley.

Cohen, J. 1960. A coefficient of agreement for nominal scales. Educational and Psychological Measurement 20 : 37-46.
_-. 1968. Weighted kappa: Nominal scale agreement with provision for scaled disagreement or partial credit. Psychological Bulletin 70: 213-220. https://doi.org/10.1037/h0026256.
Cox, N. J. 2006. Assessing agreement of measurements and predictions in geomorphology. Geomorphology 76: 332-346. https://doi.org/10.1016/j.geomorph.2005.12.001.
Dunn, G. 2000. Statistics in Psychiatry. London: Arnold.
Fleiss, J. L., and J. Cuzick. 1979. The reliability of dichotomous judgments: Unequal numbers of judges per subject. Applied Psychological Measurement 3: 537-542. https://doi.org/10.1177/014662167900300410.

Fleiss, J. L., B. Levin, and M. C. Paik. 2003. Statistical Methods for Rates and Proportions. 3rd ed. New York: Wiley.
Fleiss, J. L., J. C. M. Nee, and J. R. Landis. 1979. Large sample variance of kappa in the case of different sets of raters. Psychological Bulletin 86: 974-977. https://doi.org/10.1037/0033-2909.86.5.974.

Klein, D. 2018. Implementing a general framework for assessing interrater agreement in Stata. Stata Journal 18: 871-901.

Landis, J. R., and G. G. Koch. 1977a. The measurement of observer agreement for categorical data. Biometrics 33: 159-174. https://doi.org/10.2307/2529310.

1977b. A one-way components of variance model for categorical data. Biometrics 33: 671-679. https://doi.org/10.2307/2529465.

Reichenheim, M. E. 2004. Confidence intervals for the kappa statistic. Stata Journal 4: 421-428.
Shrout, P. E. 2001. Jacob Cohen (1923-1998). American Psychologist 56: 166. https://doi.org/10.1037/0003066X.56.2.166.

\section*{Title}
kdensity - Univariate kernel density estimation
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
Acknowledgments & References & Also see &
\end{tabular}

\section*{Description}
kdensity produces kernel density estimates and graphs the result.

\section*{Quick start}

Graph of the kernel density estimate for v1
kdensity v1
Add a normal curve kdensity v1, normal

With a kernel bandwidth of 2
kdensity v1, bwidth(2)
Gaussian kernel function for v1
kdensity v1, kernel(gaussian)
Kernel density estimate for v 1 and v 2 in the same graph area twoway kdensity v1 || kdensity v2

Separate graphs of kernel density estimate of v1 for each level of catvar twoway kdensity v1, by (catvar)

Kernel density estimates of v1 for catvar \(=0\) and 1 in the same graph area twoway kdensity v1 if catvar==0 || kdensity v1 if catvar==1

\section*{Menu}

Statistics \(>\) Nonparametric analysis \(>\) Kernel density estimation

\section*{Syntax}
kdensity varname [if] [in] [weight] [, options]
options
Description
Main
kernel (kernel)
bwidth(\#)
generate (newvar \(_{x}\) newvar \(_{d}\) )
n(\#)
at ( \(\operatorname{var}_{x}\) )
nograph
Kernel plot
cline_options
Density plots
normal
normopts (cline_options)
student (\#)
stopts (cline_options)
Add plots addplot (plot)

Y axis, X axis, Titles, Legend, Overall twoway_options suppress graph
specify kernel function; default is kernel (epanechnikov) half-width of kernel
store the estimation points in newvar \(x_{x}\) and the density estimate in newvar \({ }_{d}\)
estimate density using \# points; default is \(\min (N, 50)\)
estimate density using the values specified by \(\operatorname{var}_{x}\)
affect rendition of the plotted kernel density estimate
add normal density to the graph
affect rendition of normal density
add Student's \(t\) density with \# degrees of freedom to the graph affect rendition of the Student's \(t\) density
add other plots to the generated graph
any options other than by () documented in [G-3] twoway_options

Description
Epanechnikov kernel function; the default alternative Epanechnikov kernel function
biweight kernel function cosine trace kernel function Gaussian kernel function Parzen kernel function rectangle kernel function triangle kernel function
collect is allowed; see [U] 11.1.10 Prefix commands.
fweights, aweights, and iweights are allowed; see [U] 11.1.6 weight.

\section*{Options}
\(\qquad\) Main
kernel (kernel) specifies the kernel function for use in calculating the kernel density estimate. The default kernel is the Epanechnikov kernel (epanechnikov).
bwidth(\#) specifies the half-width of the kernel, the width of the density window around each point. If bwidth() is not specified, the "optimal" width is calculated and used. The optimal width is the width that would minimize the mean integrated squared error if the data were Gaussian and a Gaussian kernel were used, so it is not optimal in any global sense. In fact, for multimodal and highly skewed densities, this width is usually too wide and oversmooths the density (Silverman 1986).
generate (newvar \(x_{x}\) newvar \(_{d}\) ) stores the results of the estimation. newvar \({ }_{x}\) will contain the points at which the density is estimated. newvar \({ }_{d}\) will contain the density estimate.
n (\#) specifies the number of points at which the density estimate is to be evaluated. The default is \(\min (N, 50)\), where \(N\) is the number of observations in memory.
at \(\left(\mathrm{var}_{x}\right)\) specifies a variable that contains the values at which the density should be estimated. This option allows you to more easily obtain density estimates for different variables or different subsamples of a variable and then overlay the estimated densities for comparison.
nograph suppresses the graph. This option is often used with the generate() option.

Kernel plot
cline_options affect the rendition of the plotted kernel density estimate. See [G-3] cline_options.

Density plots
normal requests that a normal density be overlaid on the density estimate for comparison.
normopts (cline_options) specifies details about the rendition of the normal curve, such as the color and style of line used. See [G-3] cline_options.
student (\#) specifies that a Student's \(t\) density with \# degrees of freedom be overlaid on the density estimate for comparison.
stopts(cline_options) affects the rendition of the Student's \(t\) density. See [G-3] cline_options.

Add plots
addplot (plot) provides a way to add other plots to the generated graph. See [G-3] addplot_option.

Y axis, X axis, Titles, Legend, Overall
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

\section*{Remarks and examples}

Kernel density estimators approximate the density \(f(x)\) from observations on \(x\). Histograms do this, too, and the histogram itself is a kind of kernel density estimate. The data are divided into nonoverlapping intervals, and counts are made of the number of data points within each interval. Histograms are bar graphs that depict these frequency counts - the bar is centered at the midpoint of each interval - and its height reflects the average number of data points in the interval.

In more general kernel density estimates, the range is still divided into intervals, and estimates of the density at the center of intervals are produced. One difference is that the intervals are allowed to overlap. We can think of sliding the interval-called a window-along the range of the data and collecting the center-point density estimates. The second difference is that, rather than merely counting the number of observations in a window, a kernel density estimator assigns a weight between

0 and 1 -based on the distance from the center of the window - and sums the weighted values. The function that determines these weights is called the kernel.

Kernel density estimates have the advantages of being smooth and of being independent of the choice of origin (corresponding to the location of the bins in a histogram).

See Salgado-Ugarte, Shimizu, and Taniuchi (1993) and Fox (1990) for discussions of kernel density estimators that stress their use as exploratory data-analysis tools.

Cox (2007) gives a lucid introductory tutorial on kernel density estimation with several Stata produced examples. He provides tips and tricks for working with skewed or bounded distributions and applying the same techniques to estimate the intensity function of a point process.
> Example 1: Histogram and kernel density estimate
Goeden (1978) reports data consisting of 316 length observations of coral trout. We wish to investigate the underlying density of the lengths. To begin on familiar ground, we might draw a histogram. In [R] histogram, we suggest setting the bins to \(\min \left(\sqrt{n}, 10 \cdot \log _{10} n\right)\), which for \(n=316\) is roughly 18 :
```

. use https://www.stata-press.com/data/r18/trocolen
. histogram length, bins(18)
(bin=18, start=226, width=19.777778)

```


The kernel density estimate, on the other hand, is smooth.
. kdensity length


Kernel density estimators are, however, sensitive to an assumption, just as are histograms. In histograms, we specify a number of bins. For kernel density estimators, we specify a width. In the graph above, we used the default width. kdensity is smarter than twoway histogram in that its default width is not a fixed constant. Even so, the default width is not necessarily best.
kdensity stores the width in the returned scalar bwidth, so typing display r(bwidth) reveals it. Doing this, we discover that the width is approximately 20.

Widths are similar to the inverse of the number of bins in a histogram in that smaller widths provide more detail. The units of the width are the units of \(x\), the variable being analyzed. The width is specified as a half-width, meaning that the kernel density estimator with half-width 20 corresponds to sliding a window of size 40 across the data.

We can specify half-widths for ourselves by using the bwidth() option. Smaller widths do not smooth the density as much:
```

. kdensity length, bwidth(10)

```


kernel \(=\) epanechnikov, bandwidth \(=15.0000\)
> Example 2: Different kernels can produce different results
When widths are held constant, different kernels can produce surprisingly different results. This is really an attribute of the kernel and width combination; for a given width, some kernels are more sensitive than others at identifying peaks in the density estimate.

We can see this when using a dataset with lots of peaks. In the automobile dataset, we characterize the density of weight, the weight of the vehicles. Below, we compare the Epanechnikov and Parzen kernels.
```

. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. kdensity weight, kernel(epanechnikov) nograph generate(x epan)
. kdensity weight, kernel(parzen) nograph generate(x2 parzen)
. label var epan "Epanechnikov density estimate"
. label var parzen "Parzen density estimate"
. line epan parzen x, sort ytitle("Density") legend(pos(6))

```


We did not specify a width, so we obtained the default width. That width is not a function of the selected kernel, but of the data. See Methods and formulas for the calculation of the optimal width.

\section*{Example 3: Density with overlaid normal density}

In examining the density estimates, we may wish to overlay a normal density or a Student's \(t\) density for comparison. Using automobile weights, we can get an idea of the distance from normality by using the normal option.
. kdensity weight, kernel(epanechnikov) normal

> Example 4: Compare two densities
We also may want to compare two or more densities. In this example, we will compare the density estimates of the weights for the foreign and domestic cars.
```

. use https://www.stata-press.com/data/r18/auto, clear
(1978 automobile data)
. kdensity weight, nograph generate(x fx)
. kdensity weight if foreign==0, nograph generate(fx0) at(x)
. kdensity weight if foreign==1, nograph generate(fx1) at(x)
. label var fx0 "Domestic cars"
. label var fx1 "Foreign cars"

```
. line fx0 fx1 \(x\), sort ytitle(Density)


\section*{- Technical note}

Although all the examples we included had densities of less than 1 , the density may exceed 1 .
The probability density \(f(x)\) of a continuous variable, \(x\), has the units and dimensions of the reciprocal of \(x\). If \(x\) is measured in meters, \(f(x)\) has units \(1 /\) meter. Thus, the density is not measured on a probability scale, so it is possible for \(f(x)\) to exceed 1 .

To see this, think of a uniform density on the interval 0 to 1 . The area under the density curve is 1: this is the product of the density, which is constant at 1 , and the range, which is 1 . If the variable is then transformed by doubling, the area under the curve remains 1 and is the product of the density, constant at 0.5 , and the range, which is 2 . Conversely, if the variable is transformed by halving, the area under the curve also remains at 1 and is the product of the density, constant at 2 , and the range, which is 0.5 . (Strictly, the range is measured in certain units, and the density is measured in the reciprocal of those units, so the units cancel on multiplication.)

\section*{Stored results}
kdensity stores the following in r() :
Scalars
r(bwidth) kernel bandwidth
\(r(n) \quad\) number of points at which the estimate was evaluated
\(r\) (scale) density bin width
Macros
\(r\) (kernel) name of kernel

\section*{Methods and formulas}

A kernel density estimate is formed by summing the weighted values calculated with the kernel function \(K\), as in
\[
\widehat{f}_{K}=\frac{1}{q h} \sum_{i=1}^{n} w_{i} K\left(\frac{x-X_{i}}{h}\right)
\]
where \(q=\sum_{i} w_{i}\) if weights are frequency weights (fweight) or analytic weights (aweight), and \(q=1\) if weights are importance weights (iweights). Analytic weights are rescaled so that \(\sum_{i} w_{i}=n\) (see [U] 11 Language syntax). If weights are not used, then \(w_{i}=1\), for \(i=1, \ldots, n\). kdensity includes seven different kernel functions. The Epanechnikov is the default function if no other kernel is specified and is the most efficient in minimizing the mean integrated squared error.
\begin{tabular}{|c|c|c|}
\hline Kernel & Formula & \\
\hline Biweight & \[
K[z]=\left\{\begin{array}{l}
\frac{15}{16}\left(1-z^{2}\right)^{2} \\
0
\end{array}\right.
\] & if \(|z|<1\) otherwise \\
\hline Cosine & \[
K[z]=\left\{\begin{array}{l}
1+\cos (2 \pi z) \\
0
\end{array}\right.
\] & if \(|z|<1 / 2\) otherwise \\
\hline Epanechnikov & \[
K[z]=\left\{\begin{array}{l}
\frac{3}{4}\left(1-\frac{1}{5} z^{2}\right) / \sqrt{5} \\
0
\end{array}\right.
\] & if \(|z|<\sqrt{5}\) otherwise \\
\hline Epan2 & \[
K[z]=\left\{\begin{array}{l}
\frac{3}{4}\left(1-z^{2}\right) \\
0
\end{array}\right.
\] & \begin{tabular}{l}
if \(|z|<1\) \\
otherwise
\end{tabular} \\
\hline Gaussian & \(K[z]=\frac{1}{\sqrt{2 \pi}} e^{-z^{2} / 2}\) & \\
\hline Parzen & \[
K[z]=\left\{\begin{array}{l}
\frac{4}{3}-8 z^{2}+8|z|^{3} \\
8(1-|z|)^{3} / 3 \\
0
\end{array}\right.
\] & \begin{tabular}{l}
if \(|z| \leq 1 / 2\) \\
if \(1 / 2<|z| \leq 1\) \\
otherwise
\end{tabular} \\
\hline Rectangular & \(K[z]=\left\{\begin{array}{l}1 / 2 \\ 0\end{array}\right.\) & if \(|z|<1\) otherwise \\
\hline Triangular & \(K[z]=\left\{\begin{array}{l}1-|z| \\ 0\end{array}\right.\) & if \(|z|<1\) otherwise \\
\hline
\end{tabular}

From the definitions given in the table, we can see that the choice of \(h\) will drive how many values are included in estimating the density at each point. This value is called the window width or bandwidth. If the window width is not specified, it is determined as
\[
\left.\begin{array}{rl}
m & =\min \left(\sqrt{\text { variance }_{x}}, \quad \frac{\text { interquartile range }}{x}\right. \\
1.349
\end{array}\right)
\]
where \(x\) is the variable for which we wish to estimate the kernel and \(n\) is the number of observations.
Most researchers agree that the choice of kernel is not as important as the choice of bandwidth. There is a great deal of literature on choosing bandwidths under various conditions; see, for example, Parzen (1962) or Tapia and Thompson (1978). Also see Newton (1988) for a comparison with sample spectral density estimation in time-series applications.

\section*{Acknowledgments}

We gratefully acknowledge the previous work by Isaías H. Salgado-Ugarte of the Universidad Nacional Autónoma de México, Makoto Shimizu of the University of Tokyo, and Toru Taniuchi formerly of the University of Tokyo; see Salgado-Ugarte, Shimizu, and Taniuchi (1993). Their article provides a good overview of the subject of univariate kernel density estimation and presents arguments for its use in exploratory data analysis.

\section*{References}

Cox, N. J. 2005. Speaking Stata: Density probability plots. Stata Journal 5: 259-273.
——. 2007. Kernel estimation as a basic tool for geomorphological data analysis. Earth Surface Processes and Landforms 32: 1902-1912. https://doi.org/10.1002/esp. 1518.
Fiorio, C. V. 2004. Confidence intervals for kernel density estimation. Stata Journal 4: 168-179.
Fox, J. 1990. Describing univariate distributions. In Modern Methods of Data Analysis, ed. J. Fox and J. S. Long, 58-125. Newbury Park, CA: Sage.

Goeden, G. B. 1978. A monograph of the coral trout, Plectropomus leopardus (Lacépède). Queensland Fisheries Services Research Bulletin 1: 1-42.

Kohler, U., and F. Kreuter. 2012. Data Analysis Using Stata. 3rd ed. College Station, TX: Stata Press.
López-de-Ullibarri, I. 2015. Bandwidth selection in kernel distribution function estimation. Stata Journal 15: 784-795.
Newton, H. J. 1988. TIMESLAB: A Time Series Analysis Laboratory. Belmont, CA: Wadsworth.
Parzen, E. 1962. On estimation of a probability density function and mode. Annals of Mathematical Statistics 33: 1065-1076. https://doi.org/10.1214/aoms/1177704472.

Royston, P., and N. J. Cox. 2005. A multivariable scatterplot smoother. Stata Journal 5: 405-412.
Salgado-Ugarte, I. H., and M. A. Pérez-Hernández. 2003. Exploring the use of variable bandwidth kernel density estimators. Stata Journal 3: 133-147.

Salgado-Ugarte, I. H., M. Shimizu, and T. Taniuchi. 1993. snp6: Exploring the shape of univariate data using kernel density estimators. Stata Technical Bulletin 16: 8-19. Reprinted in Stata Technical Bulletin Reprints, vol. 3, pp. 155-173. College Station, TX: Stata Press.
Scott, D. W. 2015. Multivariate Density Estimation: Theory, Practice, and Visualization. 2nd ed. Hoboken, NJ: Wiley. Silverman, B. W. 1986. Density Estimation for Statistics and Data Analysis. London: Chapman and Hall.
Simonoff, J. S. 1996. Smoothing Methods in Statistics. New York: Springer.
Tapia, R. A., and J. R. Thompson. 1978. Nonparametric Probability Density Estimation. Baltimore: Johns Hopkins University Press.

Van Kerm, P. 2003. Adaptive kernel density estimation. Stata Journal 3: 148-156.
-_. 2012. Kernel-smoothed cumulative distribution function estimation with akdensity. Stata Journal 12: 543-548.
Wand, M. P., and M. C. Jones. 1995. Kernel Smoothing. London: Chapman and Hall.

\footnotetext{
Also see
[R] histogram - Histograms for continuous and categorical variables
[R] makespline - Spline generation
\([R]\) npregress kernel - Nonparametric kernel regression
[ R ] npregress series - Nonparametric series regression
}

\section*{Title}
ksmirnov - Kolmogorov-Smirnov equality-of-distributions test

Description
Syntax
Stored results
Also see

\section*{Quick start}

Options for two-sample test Methods and formulas

\section*{Menu}

Remarks and examples
References

\section*{Description}
ksmirnov performs one- and two-sample Kolmogorov-Smirnov tests of the equality of distributions. A one-sample test compares the distribution of the tested variable with the specified distribution. A two-sample test tests the equality of the distributions of two samples.

When testing for normality, please see \([R]\) sktest and \([R]\) swilk.

\section*{Quick start}

One-sample test comparing the distribution of v1 with a Student's \(t\) distribution with 5 degrees of freedom
ksmirnov v1 \(=\mathrm{t}(5, \mathrm{v} 1)\)
Two-sample test comparing distributions of v2 in two groups defined by catvar
ksmirnov v2, by(catvar)
Same as above, but calculate an exact \(p\)-value
```

ksmirnov v2, by(catvar) exact

```

\section*{Menu}

Statistics \(>\) Nonparametric analysis \(>\) Tests of hypotheses \(>\) Kolmogorov-Smirnov test

\section*{Syntax}

One-sample Kolmogorov-Smirnov test
ksmirnov varname \(=\exp [\) if \(][\) in \(]\)
Two-sample Kolmogorov-Smirnov test
ksmirnov varname [if] [in], by (groupvar) [ exact ]
In the first syntax, varname is the variable whose distribution is being tested, and exp must evaluate to the corresponding (theoretical) cumulative. In the second syntax, groupvar must take on two distinct values. The distribution of varname for the first value of groupvar is compared with that of the second value.
collect is allowed; see [U] 11.1.10 Prefix commands.

\section*{Options for two-sample test}
\(\qquad\) Main
by (groupvar) is required. It specifies a binary variable that identifies the two groups. exact specifies that the exact \(p\)-value be computed.

\section*{Remarks and examples}
> Example 1: Two-sample test
Say that we have data on x that resulted from two different experiments, labeled as group==1 and group \(=2\). Our data contain
. use https://www.stata-press.com/data/r18/ksxmpl
. list
\begin{tabular}{l|rr|}
\cline { 2 - 3 } & group & x \\
\hline 1. & 2 & 2 \\
2. & 1 & 0 \\
3. & 2 & 3 \\
4. & 1 & 4 \\
5. & 1 & 5 \\
\cline { 2 - 3 } 6. & 2 & 8 \\
7. & 2 & 10 \\
\hline
\end{tabular}

We wish to use the two-sample Kolmogorov-Smirnov test to determine if there are any differences in the distribution of x for these two groups:
```

. ksmirnov x, by(group)
Two-sample Kolmogorov-Smirnov test for equality of distribution functions
Smaller group D p-value

```
\begin{tabular}{lrr}
1 & 0.5000 & 0.424 \\
2 & -0.1667 & 0.909 \\
Combined K-S & 0.5000 & 0.785
\end{tabular}

The first line tests the hypothesis that x for group 1 contains smaller values than for group 2 . The largest difference between the distribution functions is 0.5 . The approximate asymptotic \(p\)-value for this is 0.424 , which is not significant.

The second line tests the hypothesis that x for group 1 contains larger values than for group 2 . The largest difference between the distribution functions in this direction is 0.1667 . The approximate asymptotic \(p\)-value for this small difference is 0.909 .

Finally, the approximate asymptotic \(p\)-value for the combined test is 0.785 . The approximate \(p\)-values ksmirnov calculates are based on the five-term approximation of the asymptotic distributions derived by Smirnov (1933). These approximations are not good for small samples \((n<50)\). They are too conservative.

An exact \(p\)-value can be calculated using the exact option:
\begin{tabular}{lrcr} 
C ksmirnov x, by (group) exact \\
Two-sample Kolmogorov-Smirnov test for equality of distribution functions \\
Smaller group & D & p-value & Exact \\
\hline 1 & 0.5000 & 0.424 & \\
2 & -0.1667 & 0.909 & \\
Combined K-S & 0.5000 & 0.785 & 0.657
\end{tabular}

\section*{> Example 2: One-sample test}

Let's now test whether x in the example above is distributed normally. Kolmogorov-Smirnov is not a particularly powerful test in testing for normality, and we do not endorse such use of it; see \([R]\) sktest and \([R]\) swilk for better tests.

In any case, we will test against a normal distribution with the same mean and standard deviation:
\begin{tabular}{|c|c|c|c|c|c|}
\hline Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline x & 7 & 4.571429 & 3.457222 & 0 & 10 \\
\hline \multicolumn{6}{|l|}{. ksmirnov \(\mathrm{x}=\) normal ( \((\mathrm{x}-4.571429) / 3.457222)\)} \\
\hline \multicolumn{6}{|l|}{One-sample Kolmogorov-Smirnov test against theoretical distribution normal ( ( \(\mathrm{x}-4.571429\) )/3.457222)} \\
\hline Smaller group & D & p-valu & & & \\
\hline x & 0.1650 & 0.68 & & & \\
\hline Cumulative & -0.1250 & 0.80 & & & \\
\hline Combined K-S & 0.1650 & 0.99 & & & \\
\hline
\end{tabular}

Because Stata has no way of knowing that we based this calculation on the calculated mean and standard deviation of x , the \(p\)-values will be slightly conservative in addition to being approximations. Nevertheless, they clearly indicate that the data cannot be distinguished from normally distributed data.

\section*{Stored results}
ksmirnov stores the following in r() :
Scalars
```

r(D_1) D from line 1
r(p_1) }\quadp\mathrm{ -value from line 1
r(D_2) D from line 2
r(p_2) p-value from line 2
r(D) combined D
r(p) combined p-value
r(p_exact) exact combined p-value
r(group1) name of group from line 1
r(group2) name of group from line 2

```
Macros

\section*{Methods and formulas}

In general, the Kolmogorov-Smirnov test (Kolmogorov 1933; Smirnov 1933; also see Conover [1999], 428-465) is not very powerful against differences in the tails of distributions. In return for this, it is fairly powerful for alternative hypotheses that involve lumpiness or clustering in the data.

The directional hypotheses are evaluated with the statistics
\[
\begin{aligned}
D^{+} & =\max _{x}\{F(x)-G(x)\} \\
D^{-} & =\min _{x}\{F(x)-G(x)\}
\end{aligned}
\]
where \(F(x)\) and \(G(x)\) are the empirical distribution functions for the sample being compared. The combined statistic is
\[
D=\max \left(\left|D^{+}\right|,\left|D^{-}\right|\right)
\]

The \(p\)-value for this statistic may be obtained by evaluating the asymptotic limiting distribution. Let \(m\) be the sample size for the first sample, and let \(n\) be the sample size for the second sample. Smirnov (1933) shows that
\[
\lim _{m, n \rightarrow \infty} \operatorname{Pr}\left\{\sqrt{m n /(m+n)} D_{m, n} \leq z\right\}=1-2 \sum_{i=1}^{\infty}(-1)^{i-1} \exp \left(-2 i^{2} z^{2}\right)
\]

The first five terms form the approximation \(P_{a}\) used by Stata. The exact \(p\)-value is calculated by a counting algorithm; see Gibbons and Chakraborti (2011, 236-238).

Andrei Nikolayevich Kolmogorov (1903-1987), of Russia, was one of the great mathematicians of the twentieth century, making outstanding contributions in many different branches, including set theory, measure theory, probability and statistics, approximation theory, functional analysis, classical dynamics, and theory of turbulence. He was a faculty member at Moscow State University for more than 60 years.
Nikolai Vasilyevich Smirnov (1900-1966) was a Russian statistician whose work included contributions in nonparametric statistics, order statistics, and goodness of fit. After army service and the study of philosophy and philology, he turned to mathematics and eventually rose to be head of mathematical statistics at the Steklov Mathematical Institute in Moscow.

\section*{References}

Aivazian, S. A. 1997. Smirnov, Nikolai Vasil'yevich. In Leading Personalities in Statistical Sciences: From the Seventeenth Century to the Present, ed. N. L. Johnson and S. Kotz, 208-210. New York: Wiley.

Conover, W. J. 1999. Practical Nonparametric Statistics. 3rd ed. New York: Wiley.
Gibbons, J. D., and S. Chakraborti. 2011. Nonparametric Statistical Inference. 5th ed. Boca Raton, FL: Chapman and Hall/CRC.

Goerg, S. J., and J. Kaiser. 2009. Nonparametric testing of distributions-the Epss-Singleton two-sample test using the empirical characteristic function. Stata Journal 9: 454-465.
Jann, B. 2008. Multinomial goodness-of-fit: Large-sample tests with survey design correction and exact tests for small samples. Stata Journal 8: 147-169.

Johnson, N. L., and S. Kotz. 1997. Kolmogorov, Andrei Nikolayevich. In Leading Personalities in Statistical Sciences: From the Seventeenth Century to the Present, ed. N. L. Johnson and S. Kotz, 255-256. New York: Wiley.
Kaplan, D. M. 2019. distcomp: Comparing distributions. Stata Journal 19: 832-848.
Kolmogorov, A. N. 1933. Sulla determinazione empirica di una legge di distribuzione. Giornale dell’ Istituto Italiano degli Attuari 4: 83-91.
Riffenburgh, R. H. 2012. Statistics in Medicine. 3rd ed. San Diego, CA: Academic Press.
Smirnov, N. V. 1933. Estimate of deviation between empirical distribution functions in two independent samples. Bulletin Moscow University 2: 3-16.

\section*{Also see}
[R] runtest - Test for random order
[R] sktest - Skewness and kurtosis tests for normality
[R] swilk - Shapiro-Wilk and Shapiro-Francia tests for normality

\section*{Title}
kwallis - Kruskal-Wallis equality-of-populations rank test
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Option & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
kwallis performs a Kruskal-Wallis test of the hypothesis that several samples are from the same population. This test is a multisample generalization of the two-sample Wilcoxon (Mann-Whitney) rank-sum test.

\section*{Quick start}

Test the equality of distribution of v1 across all levels of categorical variable cvar kwallis v1, by(cvar)

Same as above, but perform the test only for observations with \(\mathrm{v} 2=1\)
```

kwallis v1 if v2==1, by(cvar)

```

\section*{Menu}

Statistics \(>\) Nonparametric analysis \(>\) Tests of hypotheses \(>\) Kruskal-Wallis rank test

\section*{Syntax}
kwallis varname \([i f][i n]\), by (groupvar)
collect is allowed; see [U] 11.1.10 Prefix commands.

\section*{Option}
by (groupvar) is required. It specifies a variable that identifies the groups.

\section*{Remarks and examples}
> Example 1
We have data on the 50 states. The data contain the median age of the population, medage, and the region of the country, region, for each state. We wish to test for the equality of the median age distribution across all four regions simultaneously:
```

. use https://www.stata-press.com/data/r18/census
(1980 Census data by state)
. kwallis medage, by(region)
Kruskal-Wallis equality-of-populations rank test

```
\begin{tabular}{|l|r|r|}
\hline region & \multicolumn{1}{|c|}{ Obs } & Rank sum \\
\hline NE & 9 & 376.50 \\
N Cntrl & 12 & 294.00 \\
South & 16 & 398.00 \\
West & 13 & 206.50 \\
\hline
\end{tabular}
```

chi2(3) = 17.041
Prob = 0.0007
chi2(3) with ties = 17.062
Prob = 0.0007

```

From the output, we see that we can reject the hypothesis that the populations are the same at any level below \(0.07 \%\).

\section*{Stored results}
kwallis stores the following in \(r()\) :
Scalars
```

    r(df) degrees of freedom
    r(chi2)
    r(chi2_adj) }\mp@subsup{\chi}{}{2}\mathrm{ adjusted for ties
    ```

\section*{Methods and formulas}

The Kruskal-Wallis test (Kruskal and Wallis 1952, 1953; also see Altman [1991, 213-215]; Conover [1999, 288-297]; and Riffenburgh [2012, sec. 11.6]) is a multiple-sample generalization of the two-sample Wilcoxon (also called Mann-Whitney) rank-sum test (Wilcoxon 1945; Mann and Whitney 1947). Samples of sizes \(n_{j}, j=1, \ldots, m\), are combined and ranked in ascending order of magnitude. Tied values are assigned the average ranks. Let \(n\) denote the overall sample size, and let \(R_{j}=\sum_{i=1}^{n_{j}} R\left(X_{j i}\right)\) denote the sum of the ranks for the \(j\) th sample. The Kruskal-Wallis one-way analysis-of-variance test, \(H\), is defined as
\[
H=\frac{1}{S^{2}}\left\{\sum_{j=1}^{m} \frac{R_{j}^{2}}{n_{j}}-\frac{n(n+1)^{2}}{4}\right\}
\]
where
\[
S^{2}=\frac{1}{n-1}\left\{\sum_{\text {all ranks }} R\left(X_{j i}\right)^{2}-\frac{n(n+1)^{2}}{4}\right\}
\]

If there are no ties, this equation simplifies to
\[
H=\frac{12}{n(n+1)} \sum_{j=1}^{m} \frac{R_{j}^{2}}{n_{j}}-3(n+1)
\]

The sampling distribution of \(H\) is approximately \(\chi^{2}\) with \(m-1\) degrees of freedom.

William Henry Kruskal (1919-2005) was born in New York City. He studied mathematics and statistics at Antioch College, Harvard, and Columbia, and joined the University of Chicago in 1951. He made many outstanding contributions to linear models, nonparametric statistics, government statistics, and the history and methodology of statistics.

Wilson Allen Wallis (1912-1998) was born in Philadelphia. He studied psychology and economics at the Universities of Minnesota and Chicago and at Columbia. He taught at Yale, Stanford, and Chicago, before moving as president (later chancellor) to the University of Rochester in 1962. He also served in several Republican administrations. Wallis served as editor of the Journal of the American Statistical Association, coauthored a popular introduction to statistics, and contributed to nonparametric statistics.

\section*{References}

Altman, D. G. 1991. Practical Statistics for Medical Research. London: Chapman and Hall/CRC.
Conover, W. J. 1999. Practical Nonparametric Statistics. 3rd ed. New York: Wiley.
Dinno, A. 2015. Nonparametric pairwise multiple comparisons in independent groups using Dunn's test. Stata Journal 15: 292-300.

Fienberg, S. E., S. M. Stigler, and J. M. Tanur. 2007. The William Kruskal Legacy: 1919-2005. Statistical Science 22: 255-261. https://doi.org/10.1214/088342306000000420.
Kruskal, W. H., and W. A. Wallis. 1952. Use of ranks in one-criterion variance analysis. Journal of the American Statistical Association 47: 583-621. https://doi.org/10.2307/2280779.
-_. 1953. Errata: Use of ranks in one-criterion variance analysis. Journal of the American Statistical Association 48: 907-911. https://doi.org/10.2307/2281082.

Mann, H. B., and D. R. Whitney. 1947. On a test of whether one of two random variables is stochastically larger than the other. Annals of Mathematical Statistics 18: 50-60. https://doi.org/10.1214/aoms/1177730491.

Newson, R. B. 2006. Confidence intervals for rank statistics: Somers' D and extensions. Stata Journal 6: 309-334.
Olkin, I. 1991. A conversation with W. Allen Wallis. Statistical Science 6: 121-140. https://doi.org/10.1214/ss/1177011818.
Riffenburgh, R. H. 2012. Statistics in Medicine. 3rd ed. San Diego, CA: Academic Press.
Wilcoxon, F. 1945. Individual comparisons by ranking methods. Biometrics 1: 80-83. https://doi.org/10.2307/3001968.
Zabell, S. L. 1994. A conversation with William Kruskal. Statistical Science 9: 285-303. https://doi.org/10.1214/ss/1177010498.

\section*{Also see}
[R] nptrend - Tests for trend across ordered groups
[R] oneway - One-way analysis of variance
[R] ranksum - Equality tests on unmatched data
[R] sdtest - Variance-comparison tests
[R] signrank - Equality tests on matched data

\section*{Title}
ladder - Ladder of powers
\begin{tabular}{lll} 
Description & Quick start & Menu \\
Syntax & Options for ladder & Options for gladder \\
Options for qladder & Remarks and examples & Stored results \\
Methods and formulas & Acknowledgment & References \\
Also see & &
\end{tabular}

\section*{Description}
ladder searches a subset of the ladder of powers (Tukey 1977) for a transform that converts varname into a normally distributed variable.
gladder and qladder each display a graph matrix. gladder displays nine histograms of transforms of varname according to the ladder of powers. qladder displays the quantiles of transforms of varname according to the ladder of powers against the quantiles of a normal distribution.

\section*{Quick start}

Table showing Tukey's ladder of powers transformations for v ladder v

Same as above, but with separate tables for each level of the categorical variable catvar by catvar: ladder v

Display transformations graphically using histograms gladder v

Same as above, but using quantile plots
qladder v

\section*{Menu}

\section*{ladder}

Statistics \(>\) Summaries, tables, and tests \(>\) Distributional plots and tests \(>\) Ladder of powers

\section*{gladder}

Statistics \(>\) Summaries, tables, and tests \(>\) Distributional plots and tests \(>\) Ladder-of-powers histograms

\section*{qladder}

Statistics \(>\) Summaries, tables, and tests \(>\) Distributional plots and tests \(>\) Ladder-of-powers quantile-normal plots

\section*{Syntax}

Ladder of powers
ladder varname [if] [in] [, generate(newvar) noadjust]

Ladder-of-powers histograms
gladder varname [if] [in] [, histogram_options combine_options]

Ladder-of-powers quantile-normal plots
qladder varname [if] [in] [, qnorm_options combine_options]
by and collect are allowed with ladder; see [U] 11.1.10 Prefix commands.

\section*{Options for ladder}

Main
generate (newvar) saves the transformed values corresponding to the minimum \(\chi^{2}\) value from the table. We do not recommend using generate() because it is literal in interpreting the minimum, thus ignoring nearly equal but perhaps more interpretable transforms.
noadjust is the noadjust option to sktest; see [R] sktest.

\section*{Options for gladder}
histogram_options affect the rendition of the histograms across all relevant transformations; see [R] histogram. Here the normal option is assumed, so you must supply the nonormal option to suppress the overlaid normal density. Also, gladder does not allow the width(\#) option of histogram.
combine_options are any of the options documented in [G-2] graph combine. These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

\section*{Options for qladder}
qnorm_options affect the rendition of the quantile-normal plots across all relevant transformations. See [R] Diagnostic plots.
combine_options are any of the options documented in [G-2] graph combine. These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

\section*{Remarks and examples}
\(>\) Example 1: ladder
We have data on the mileage rating of 74 automobiles and wish to find a transform that makes the variable normally distributed:
\begin{tabular}{|c|c|c|c|}
\hline Transformation & Formula & chi2(2) & Prob > chi2 \\
\hline Cubic & \(\mathrm{mpg}^{\wedge} 3\) & 43.59 & 0.000 \\
\hline Square & \(\mathrm{mpg}{ }^{\text {2 }}\) & 27.03 & 0.000 \\
\hline Identity & mpg & 10.95 & 0.004 \\
\hline Square root & sqrt (mpg) & 4.94 & 0.084 \\
\hline Log & \(\log\) (mpg) & 0.87 & 0.647 \\
\hline 1/(Square root) & 1/sqrt (mpg) & 0.20 & 0.905 \\
\hline Inverse & \(1 / \mathrm{mpg}\) & 2.36 & 0.307 \\
\hline 1/Square & 1/(mpg \({ }^{\text {2 }}\) ) & 11.99 & 0.002 \\
\hline 1/Cubic & 1/(mpg \({ }^{\text { }}\) ) & 24.30 & 0.000 \\
\hline
\end{tabular}

If we had typed ladder \(m p g\), gen (mpgx), the variable mpgx containing \(1 / \sqrt{m p g}\) would have been automatically generated for us. This is the perfect example of why you should not, in general, specify the generate() option. We also cannot reject the hypothesis that the inverse of mpg is normally distributed and that \(1 / \mathrm{mpg}\) - gallons per mile - has a better interpretation. It is a measure of energy consumption.

\section*{> Example 2: gladder}
gladder explores the same transforms as ladder but presents results graphically:
- gladder mpg, fraction


\section*{- Technical note}
gladder is useful pedagogically, but be careful when using it for research work, especially with many observations. For instance, consider the following data on the average July temperature in degrees Fahrenheit for 954 US cities:
\begin{tabular}{|c|c|c|c|}
\hline Transformation & Formula & chi2(2) & Prob > chi2 \\
\hline Cubic & tempjuly^3 & 47.49 & 0.000 \\
\hline Square & tempjuly^2 & 19.70 & 0.000 \\
\hline Identity & tempjuly & 3.83 & 0.147 \\
\hline Square root & sqrt (tempjuly) & 1.83 & 0.400 \\
\hline Log & log (tempjuly) & 5.40 & 0.067 \\
\hline 1/(Square root) & 1/sqrt (tempjuly) & 13.72 & 0.001 \\
\hline Inverse & 1/tempjuly & 26.36 & 0.000 \\
\hline 1/Square & 1/(tempjuly^2) & 64.44 & 0.000 \\
\hline 1/Cubic & 1/(tempjuly^3) & 116.16 & 0.000 \\
\hline
\end{tabular}

From the table, we see that there is certainly a difference in normality between the square and square-root transform. If, however, you can see the difference between the transforms in the diagram below, you have better eyes than we do:
```

. gladder tempjuly, l1title("") ylabel(none) xlabel(none)

```


\section*{Average July temperature}

Histograms by transformation

\section*{Example 3: qladder}

A better graph for seeing normality is the quantile-normal graph, which can be produced by qladder.
. qladder tempjuly, ylabel(none) xlabel(none)


Quantile-normal plots by transformation
This graph shows that for the square transform, the upper tail—and only the upper tail—diverges from what would be expected. This divergence is detected by sktest (see [R] sktest) as a problem with skewness, as we would learn from using sktest to examine tempjuly squared and square rooted.

\section*{Stored results}
ladder stores the following in \(r()\) :
Scalars
```

r(N) number of observations
r(invcube) }\mp@subsup{\chi}{}{2}\mathrm{ for inverse-cubic transformation
r(P_invcube) p-value for normality test after inverse-cubic transformation
r(invsq) }\quad\mp@subsup{\chi}{}{2}\mathrm{ for inverse-square transformation
r(P_invsq) p-value for normality test after inverse-square transformation
r(inv) }\mp@subsup{\chi}{}{2}\mathrm{ for inverse transformation
r(P_inv) p-value for normality test after inverse transformation
r(invsqrt) }\mp@subsup{\chi}{}{2}\mathrm{ for inverse-root transformation
r(P_invsqrt) p-value for normality test after inverse-root transformation
r(log) }\quad\mp@subsup{\chi}{}{2}\mathrm{ for log transformation
r(P_log) p-value for normality test after log transformation
r(sqrt) }\quad\mp@subsup{\chi}{}{2}\mathrm{ for square-root transformation
r(P_sqrt) p-value for normality test after square-root transformation
r(ident) }\quad\mp@subsup{\chi}{}{2}\mathrm{ for untransformed data
r(P_ident) p-value for normality test of untransformed data
r(square) }\quad\mp@subsup{\chi}{}{2}\mathrm{ for square transformation
r(P_square) p-value for normality test after square transformation
r(cube) }\quad\mp@subsup{\chi}{}{2}\mathrm{ for cubic transformation
r(P_cube) p-value for normality test after cubic transformation

```

\section*{Methods and formulas}

For ladder, results are as reported by sktest; see [R] sktest. If generate() is specified, the transform with the minimum \(\chi^{2}\) value is chosen.
gladder sets the number of bins to \(\min \left(\sqrt{n}, 10 \log _{10} n\right)\), rounded to the closest integer, where \(n\) is the number of unique values of varname. See [R] histogram for a discussion of the optimal number of bins.

Also see Findley (1990) for a ladder-of-powers variable transformation program that produces one-way graphs with overlaid box plots, in addition to histograms with overlaid normals. Buchner and Findley (1990) discuss ladder-of-powers transformations as one aspect of preliminary data analysis. Also see Hamilton (1992, 18-23) and Hamilton (2013, 129-132).

\section*{Acknowledgment}
qladder was written by Jeroen Weesie of the Department of Sociology at Utrecht University, The Netherlands.

\section*{References}

Buchner, D. M., and T. W. Findley. 1990. Research in physical medicine and rehabilitation: VIII. Preliminary data analysis. American Journal of Physical Medicine and Rehabilitation 69: 154-169. https://doi.org/10.1097/00002060-199006000-00011.

Cox, N. J. 2005. Speaking Stata: Density probability plots. Stata Journal 5: 259-273.
Findley, T. W. 1990. sed3: Variable transformation and evaluation. Stata Technical Bulletin 2: 15. Reprinted in Stata Technical Bulletin Reprints, vol. 1, pp. 85-86. College Station, TX: Stata Press.

Hamilton, L. C. 1992. Regression with Graphics: A Second Course in Applied Statistics. Belmont, CA: Duxbury.
——. 2013. Statistics with Stata: Updated for Version 12. 8th ed. Boston: Brooks/Cole.
Tukey, J. W. 1977. Exploratory Data Analysis. Reading, MA: Addison-Wesley.

\section*{Also see}
[R] boxcox - Box-Cox regression models
[R] Diagnostic plots - Distributional diagnostic plots
[R] Inskew0 - Find zero-skewness log or Box-Cox transform
[R] lv - Letter-value displays
[R] sktest - Skewness and kurtosis tests for normality

\section*{Title}
level - Set default confidence level

Description Syntax Option Remarks and examples Also see

\section*{Description}
set level specifies the default confidence level for confidence intervals for all commands that report confidence intervals. The initial value is 95 , meaning \(95 \%\) confidence intervals.

\section*{Syntax}
set level \# [, permanently]
\# is any number between 10.00 and 99.99 and may be specified with at most two digits after the decimal point.

\section*{Option}
permanently specifies that, in addition to making the change right now, the level setting be remembered and become the default setting when you invoke Stata.

\section*{Remarks and examples}

To change the level of confidence intervals reported by a particular command, you need not reset the default confidence level. All commands that report confidence intervals have a level (\#) option. When you do not specify the option, the confidence intervals are calculated for the default level set by set level, or for \(95 \%\) if you have not reset set level.

\section*{> Example 1}

We use the ci means command to obtain the confidence interval for the mean of mpg:
```

. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. ci means mpg

| Variable | Obs | Mean | Std. err. | [95\% conf. interval] |  |
| ---: | :---: | :---: | :---: | :---: | ---: |
| mpg | 74 | 21.2973 | .6725511 | 19.9569 | 22.63769 |

```

To obtain \(90 \%\) confidence intervals, we would type
. ci means mpg, level(90)
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. err. & [90\% conf. interval] \\
\hline mpg & 74 & 21.2973 & .6725511 & 20.17683 & 22.41776
\end{tabular}
or
```

. set level 90

```
. ci means mpg
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. err. & [90\% conf. interval] \\
\hline mpg & 74 & 21.2973 & .6725511 & 20.17683 & 22.41776
\end{tabular}

If we opt for the second alternative, the next time that we fit a model (say, with regress), \(90 \%\) confidence intervals will be reported. If we wanted \(95 \%\) confidence intervals, we could specify level(95) on the estimation command, or we could reset the default by typing set level 95.

The current setting of level() is stored as the c-class value c (level); see [P] creturn.

\section*{Also see}
[R] query - Display system parameters
[P] creturn - Return c-class values
[U] 20 Estimation and postestimation commands
[U] 20.8 Specifying the width of confidence intervals

\section*{Title}

Limits - Quick reference for limits

\section*{Description Remarks and examples Also see}

\section*{Description}

This entry provides a quick reference for the size limits in Stata. Note that most of these limits are so high that you will never encounter them.

\section*{Remarks and examples}

Remarks are presented under the following headings:
Maximum size limits
Determining which edition of Stata you are running

\section*{Maximum size limits}

\begin{tabular}{|c|c|c|}
\hline continued & Stata/BE & Stata/MP and Stata/SE \\
\hline \# of nested do-files & 64 & 64 \\
\hline \# of lines in a program & 3,500 & 3,500 \\
\hline \# of bytes in a program & 135,600 & 135,600 \\
\hline length of a variable name (characters) & 32 & 32 \\
\hline length of ado-command name (characters) & 32 & 32 \\
\hline length of a global macro name (characters) & 32 & 32 \\
\hline length of a local macro name (characters) & 31 & 31 \\
\hline length of a str\# variable (bytes) & 2,045 & 2,045 \\
\hline length of a strL variable (bytes) 2,00, & 2,000,000,000 & 2,000,000,000 \\
\hline \multicolumn{3}{|l|}{anova} \\
\hline \# of variables in one anova term & 8 & 8 \\
\hline \# of terms in the repeated() option & 4 & 4 \\
\hline \multicolumn{3}{|l|}{char} \\
\hline length of one characteristic (bytes) & 67,784 & 67,784 \\
\hline \multicolumn{3}{|l|}{constraint} \\
\hline \# of constraints & 1,999 & 1,999 \\
\hline \multicolumn{3}{|l|}{encode and decode} \\
\hline \# of unique values & 65,536 & 65,536 \\
\hline \multicolumn{3}{|l|}{_estimates hold} \\
\hline \# of stored estimation results & 300 & 300 \\
\hline \multicolumn{3}{|l|}{estimates store} \\
\hline \# of stored estimation results & 300 & 300 \\
\hline \multicolumn{3}{|l|}{exlogistic and expoisson} \\
\hline \multicolumn{3}{|l|}{frames} \\
\hline \# of frames & 100 & 100 \\
\hline \multicolumn{3}{|l|}{grmeanby} \\
\hline \# of unique values in varlist & -N/2 & -N/2 \\
\hline \multicolumn{3}{|l|}{graph} \\
\hline minimum graph size (inches) & 1 & 1 \\
\hline maximum graph size (inches) & 100 & 100 \\
\hline \multicolumn{3}{|l|}{graph twoway} \\
\hline \# of variables in a plot & 100 & 100 \\
\hline \# of styles in an option's stylelist & 20 & 20 \\
\hline \multicolumn{3}{|l|}{import sas} \\
\hline \# of variables & 2,048 & 32,766 \\
\hline \multicolumn{3}{|l|}{import spss} \\
\hline \# of variables & 2,048 & 32,766 \\
\hline
\end{tabular}
\begin{tabular}{|c|c|c|}
\hline continued & Stata/BE & Stata/MP and Stata/SE \\
\hline infile (free format) record length without dictionary & none & none \\
\hline infile (fixed format) record length with a dictionary & 524,275 & 524,275 \\
\hline infix (fixed format) record length with a dictionary & 524,275 & 524,275 \\
\hline label & & \\
\hline length of dataset label (characters) & 80 & 80 \\
\hline length of variable label (characters) & 80 & 80 \\
\hline length of value label string (bytes) & 32,000 & 32,000 \\
\hline length of name of value label (characters) & 32 & 32 \\
\hline \# of codings within one value label & 65,536 & 65,536 \\
\hline label language & & \\
\hline \# of different languages & 100 & 100 \\
\hline macro & & \\
\hline \# of nested macros & 20 & 20 \\
\hline manova & & \\
\hline \# of variables in single manova term & 8 & 8 \\
\hline matrix (3) dimension of single matrix & \(800 \times 800\) & \[
\begin{array}{r}
65,534 \times 65,534(\mathrm{MP}) \\
11,000 \times 11,000(\mathrm{SE})
\end{array}
\] \\
\hline maximize options & & \\
\hline iterate() maximum & 16,000 & 16,000 \\
\hline mprobit & & \\
\hline \# of categories in a depvar & 30 & 30 \\
\hline net & & \\
\hline \# of description lines in . pkg file & 100 & 100 \\
\hline \begin{tabular}{l}
nlogit and nlogittree \\
\# of levels in model
\end{tabular} & 8 & 8 \\
\hline notes & & \\
\hline length of one note (bytes) & 67,784 & 67,784 \\
\hline \# of notes attached to _dta & 9,999 & 9,999 \\
\hline \# of notes attached to each variable & 9,999 & 9,999 \\
\hline numlist & & \\
\hline \# of elements in the numeric list & 2,500 & 2,500 \\
\hline \multicolumn{3}{|l|}{_pctile} \\
\hline \# of percentiles & 4,096 & 4,096 \\
\hline
\end{tabular}
putdocx and putpdf
    \# of tables added to a document
reg3, sureg, and other system estimators
    \# of equations800
```

set adosize

```
    memory ado-files may consume 1000K
    memory for Results window buffer
                2000K2000k

slogit
    \# of categories in a depvar
    3030

snapshot
    length of label (characters) 8080
\# of saved snapshots ..... 1,000 ..... 1,000
stcox
\# of variables in strata() option ..... 5
stcurve45
\# of curves plotted on the same graph 10 ..... 10 ..... 1010,000

10,000
```

set scrollbufsize

```
set scrollbufsizememory for Results window buffer2000K3080
,
tabdisp\# of by variables 4\# of margins, i.e., sum of rows,columns, supercolumns, and by groups
tabulate oneway
\# of rows in one-way table ..... 3,00012,000
tabulate twoway
\# of rows \& cols in two-way table
tabulate, summarize()
\# of cells (rows X cols) ..... 375
\(1,200 \times 80\) \(300 \times 20\)4
3,000
3,000teffects
\# of treatments ..... 2020
xt estimation commands (e.g., xtgee,
xtgls, xtpoisson, xtprobit, xtreg
with mle option, and xtpcse whenneither option hetonly nor option
independent is specified)
\# of time periods within panel80065,534 (MP)
\# of integration points accepted ..... 195195 by intpoints(\#)
(1) For Stata/MP, the maximum number of observations is \(1,099,511,627,775\), and for Stata/SE, the maximum number is \(2,147,483,619\). In practice, both editions are limited by memory.
(2) The maximum length of the contents of a macro are fixed in Stata/BE and settable in Stata/SE and Stata/MP. The currently set maximum length is recorded in \(c\) (macrolen); type display c (macrolen). The maximum length can be changed with set maxvar. If you set maxvar to a larger value, the maximum length increases; if you set maxvar to a smaller value, the maximum length decreases. The relationship between them is maximum_length \(=129 \times\) maxvar +200 .
(3) In Mata, matrices are limited only by the amount of memory on your computer.

\section*{Determining which edition of Stata you are running}

Type
. about
The response will be Stata/MP, Stata/SE, or Stata/BE. Other information is also shown, including your serial number. See \([R]\) about.

\section*{Also see}
[R] about - Display information about your Stata
[D] compress - Compress data in memory
[D] Data types - Quick reference for data types
[D] import - Overview of importing data into Stata
[D] infile (fixed format) - Import text data in fixed format with a dictionary
[D] infile (free format) - Import unformatted text data
[D] memory - Memory management
[D] obs - Increase the number of observations in a dataset

\section*{Title}
lincom - Linear combinations of parameters
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & References \\
Also see & & &
\end{tabular}

\section*{Description}
lincom computes point estimates, standard errors, \(t\) or \(z\) statistics, \(p\)-values, and confidence intervals for linear combinations of coefficients after any estimation command, including survey estimation. Results can optionally be displayed as odds ratios, hazard ratios, incidence-rate ratios, or relative-risk ratios.

\section*{Quick start}

Point estimate and confidence interval for sum of coefficients of x1 and x2
lincom x1 + x2
Same as above, but report results as a relative-risk ratio
lincom x 1 + x 2 , rrr
Same as above, but use coefficients from second equation of a multiequation model
lincom [2]x1 + [2]x2, rrr
Difference between coefficients of first and third level of categorical variable a
lincom 1.a - 3.a
Sum of coefficients of x 1 and x 2 after a model adjusted for complex survey design lincom x1 + x2

\section*{Menu}

Statistics \(>\) Postestimation

\section*{Syntax}
lincom exp [, options]
options Description
\begin{tabular}{ll}
\(\underline{\mathrm{ef}} \mathrm{orm}\) & \begin{tabular}{l} 
generic label; \(\exp (b)\) \\
or
\end{tabular} \\
hr & odds ratio \\
shr & hazard ratio \\
\(\underline{\mathrm{irr} r}\) & subhazard ratio \\
\(\underline{\mathrm{rr} r}\) & incidence-rate ratio \\
\(\underline{\text { level (\#) }}\) & relative-risk ratio \\
display_options & set confidence level; default is level(95) \\
control column formats
\end{tabular}
\(\mathrm{df}(\#) \quad\) use \(t\) distribution with \# degrees of freedom for computing \(p\)-values and confidence intervals
exp is any linear combination of coefficients that is a valid syntax for test; see [R] test. exp must not contain an equal sign.
collect is allowed; see [U] 11.1.10 Prefix commands.
\(\mathrm{df}(\#)\) does not appear in the dialog box.

\section*{Options}
eform, or, hr, shr, irr, and rrr all report coefficient estimates as \(\exp (\widehat{\beta})\) rather than \(\widehat{\beta}\). Standard errors and confidence intervals are similarly transformed. or is the default after logistic. The only difference in these options is how the output is labeled.
\begin{tabular}{llll} 
Option & Label & Explanation & Example commands \\
\hline eform & exp(b) & Generic label & cloglog \\
or & Odds ratio & Odds ratio & logistic, logit \\
hr & Haz. ratio & Hazard ratio & stcox, streg \\
shr & SHR & Subhazard ratio & stcrreg \\
irr & IRR & Incidence-rate ratio & poisson \\
rrr & RRR & Relative-risk ratio & mlogit \\
\hline
\end{tabular}
exp may not contain any additive constants when you use the eform, or, hr, irr, or rrr option.
level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level(95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals. display_options: cformat (\%fmt), pformat (\%fmt), and sformat(\%fmt); see [R] Estimation options.

The following option is available with lincom but is not shown in the dialog box:
\(\mathrm{df}(\#)\) specifies that the \(t\) distribution with \# degrees of freedom be used for computing \(p\)-values and confidence intervals. The default is to use e(df_r) degrees of freedom or the standard normal distribution if \(e\left(d f \_r\right)\) is missing.

\section*{Remarks and examples}

Remarks are presented under the following headings:
Using lincom
Odds ratios and incidence-rate ratios
Multiple-equation models

\section*{Using lincom}

After fitting a model and obtaining estimates for coefficients \(\beta_{1}, \beta_{2}, \ldots, \beta_{k}\), you may want to view estimates for linear combinations of the \(\beta_{i}\), such as \(\beta_{1}-\beta_{2}\). lincom can display estimates for any linear combination of the form \(c_{0}+c_{1} \beta_{1}+c_{2} \beta_{2}+\cdots+c_{k} \beta_{k}\).
lincom works after any estimation command for which test works. Any valid expression for test syntax 1 (see \([R]\) test) is a valid expression for lincom.
lincom is useful for viewing odds ratios, hazard ratios, etc., for one group (that is, one set of covariates) relative to another group (that is, another set of covariates). See the examples below.

\section*{Example 1}

We perform a linear regression:
```

. use https://www.stata-press.com/data/r18/regress

```
. regress y x1 x2 x3
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline Source & SS & df & MS & Number of obs & = & 148 \\
\hline & & & & \(\mathrm{F}(3,144)\) & = & 96.12 \\
\hline Model & 3259.3561 & 3 & 1086.45203 & Prob > F & = & 0.0000 \\
\hline Residual & 1627.56282 & 144 & 11.3025196 & R -squared & = & 0.6670 \\
\hline & & & & Adj R-squared & = & 0.6600 \\
\hline Total & 4886.91892 & 147 & 33.2443464 & Root MSE & = & 3.3619 \\
\hline
\end{tabular}
\begin{tabular}{r|rrcrrr}
\hline y & Coefficient & Std. err. & t & \(\mathrm{P}>|\mathrm{t}|\) & [95\% conf. interval] \\
\hline x1 & 1.457113 & 1.07461 & 1.36 & 0.177 & -.666934 & 3.581161 \\
x2 & 2.221682 & .8610358 & 2.58 & 0.011 & .5197797 & 3.923583 \\
x3 & -.006139 & .0005543 & -11.08 & 0.000 & -.0072345 & -.0050435 \\
_cons & 36.10135 & 4.382693 & 8.24 & 0.000 & 27.43863 & 44.76407 \\
\hline
\end{tabular}

To see the difference of the coefficients of \(x 2\) and \(x 1\), we type
```

. lincom x2 - x1
(1) - x1 + x2 = 0

```
\begin{tabular}{r|rrrrrr}
\hline y & Coefficient & Std. err. & t & \(\mathrm{P}>|\mathrm{t}|\) & [95\% conf. interval] \\
\hline\((1)\) & .7645682 & .9950282 & 0.77 & 0.444 & -1.20218 & 2.731316 \\
\hline
\end{tabular}

The expression can be any linear combination.
```

. lincom 3*x1 + 500*x3
( 1) 3*x1 + 500*x3 = 0

```
\begin{tabular}{r|rlcccr}
\hline y & Coefficient & Std. err. & \(t\) & P>|t| & [95\% conf. interval] \\
\hline\((1)\) & 1.301825 & 3.396624 & 0.38 & 0.702 & -5.411858 & 8.015507 \\
\hline
\end{tabular}

Nonlinear expressions are not allowed.
```

. lincom x2/x1
not possible with test
r(131);

```

For information about estimating nonlinear expressions, see \([R]\) nlcom.

\section*{- Technical note}
lincom uses the same shorthands for coefficients as does test (see [R] test). When you type x1, for instance, lincom knows that you mean the coefficient of x1. The formal syntax for referencing this coefficient is actually \(\_\mathrm{b}\) [x1], or alternatively, _coef [x1]. So, more formally, in the last example we could have typed
```

. lincom 3*_b[x1] + 500*_b[x3]
(output omitted)

```

\section*{Odds ratios and incidence-rate ratios}

After logistic regression, the or option can be specified with lincom to display odds ratios for any effect. Incidence-rate ratios after commands such as poisson can be similarly obtained by specifying the irr option.

\section*{> Example 2}

Consider the low birthweight dataset from Hosmer, Lemeshow, and Sturdivant (2013, 24). We fit a logistic regression model of low birthweight (variable low) on the following variables:
\begin{tabular}{lll} 
Variable & Description & Coding \\
\hline age & age in years & \\
race & race & 1 if white, 2 if black, 3 if other \\
smoke & smoking status & 1 if smoker, 0 if nonsmoker \\
ht & history of hypertension & 1 if yes, 0 if no \\
ui & materine irritability weight before pregnancy & 1 if yes, 0 if no \\
lwd & 1 if weight \(<110\) lb., 0 otherwise \\
ptd & \begin{tabular}{l} 
history of premature labor \\
c.age\#\#lwd
\end{tabular} & \begin{tabular}{l} 
age main effects, lwd main effects, \\
and their interaction
\end{tabular} \\
smoke\#\#lwd & \begin{tabular}{l} 
smoke main effects, lwd main effects, \\
and their interaction
\end{tabular} \\
\hline
\end{tabular}

We first fit a model without the interaction terms by using logit.
```

. use https://www.stata-press.com/data/r18/lbw3
(Hosmer \& Lemeshow data)

```
. logit low age lwd i.race smoke ptd ht ui
Iteration 0: Log likelihood \(=-117.336\)
Iteration 1: Log likelihood \(=-99.3982\)
Iteration 2: Log likelihood = -98.780418
Iteration 3: Log likelihood \(=-98.777998\)
Iteration 4: Log likelihood \(=-98.777998\)
Logistic regression
Log likelihood = -98.777998

Number of obs = 189
LR chi2(8) \(=37.12\)
Prob > chi2 \(=0.0000\)
Pseudo R2 \(=0.1582\)
\begin{tabular}{r|rrrrrr}
\hline low & Coefficient & Std. err. & \multicolumn{1}{c}{z} & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline age & -.0464796 & .0373888 & -1.24 & 0.214 & -.1197603 & .0268011 \\
lwd & .8420615 & .4055338 & 2.08 & 0.038 & .0472299 & 1.636893 \\
race & & & & & & \\
Black & 1.073456 & .5150753 & 2.08 & 0.037 & .0639273 & 2.082985 \\
Other & .815367 & .4452979 & 1.83 & 0.067 & -.0574008 & 1.688135 \\
& & & & & & \\
smoke & .8071996 & .404446 & 2.00 & 0.046 & .0145001 & 1.599899 \\
ptd & 1.281678 & .4621157 & 2.77 & 0.006 & .3759478 & 2.187408 \\
ht & 1.435227 & .6482699 & 2.21 & 0.027 & .1646414 & 2.705813 \\
ui & .6576256 & .4666192 & 1.41 & 0.159 & -.2569313 & 1.572182 \\
_cons & -1.216781 & .9556797 & -1.27 & 0.203 & -3.089878 & .656317 \\
\hline
\end{tabular}

To get the odds ratio for black smokers relative to white nonsmokers (the reference group), we type
. lincom 2.race + smoke, or
( 1) [low]2.race \(+[\) low \(]\) smoke \(=0\)
\begin{tabular}{r|rccccc}
\hline low & Odds ratio & Std. err. & \(z\) & \(P>|z|\) & [95\% conf. interval] \\
\hline\((1)\) & 6.557805 & 4.744692 & 2.60 & 0.009 & 1.588176 & 27.07811 \\
\hline
\end{tabular}
lincom computed \(\exp \left(\beta_{2 \text {.race }}+\beta_{\text {smoke }}\right)=6.56\). To see the odds ratio for white smokers relative to black nonsmokers, we type
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline low & Odds ratio & Std. err. & z & \(P>|z|\) & [95\% conf & interval] \\
\hline (1) & . 7662425 & . 4430176 & -0.46 & 0.645 & . 2467334 & 2.379603 \\
\hline
\end{tabular}

Now let's add the interaction terms to the model (Hosmer and Lemeshow 1989, table 4.10). This time, we will use logistic rather than logit. By default, logistic displays odds ratios.
. logistic low i.race ht ui ptd c.age\#\#lwd smoke\#\#lwd
\(\begin{array}{llr}\text { Logistic regression } & \text { Number of obs }=189 \\ & \text { LR chi2 }(10)=42.66 \\ \text { Log likelihood }=-96.00616 & \text { Prob }>\text { chi2 }=0.0000 \\ & \text { Pseudo R2 }=0.1818\end{array}\)
\begin{tabular}{r|rrrrrr}
\hline low & Odds ratio & Std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline race & & & & & & \\
Black & 2.95383 & 1.532789 & 2.09 & 0.037 & 1.068277 & 8.167465 \\
Other & 2.137589 & .9919138 & 1.64 & 0.102 & .8608708 & 5.307752 \\
ht & 3.893141 & 2.575201 & 2.05 & 0.040 & 1.064768 & 14.2346 \\
ui & 2.071284 & .9931388 & 1.52 & 0.129 & .8092926 & 5.301192 \\
ptd & 3.426633 & 1.615282 & 2.61 & 0.009 & 1.360252 & 8.632089 \\
age & .9194513 & .041896 & -1.84 & 0.065 & .8408967 & 1.005344 \\
1.lwd & .1772934 & .3312384 & -0.93 & 0.354 & .0045539 & 6.902367 \\
lwd\#c.age & & & & & & \\
1
\end{tabular}

Note: _cons estimates baseline odds.
Hosmer and Lemeshow (1989, table 4.13) consider the effects of smoking (smoke \(=1\) ) and low maternal weight before pregnancy ( \(1 \mathrm{wd}=1\) ). The effect of smoking among non-low-weight mothers \((l w d=0)\) is given by the odds ratio 3.17 for smoke in the logistic output. The effect of smoking among low-weight mothers is given by
```

. lincom 1.smoke + 1.smoke\#1.lwd
( 1) [low]1.smoke + [low]1.smoke\#1.lwd = 0

```
\begin{tabular}{r|rccccr}
\hline low & Odds ratio & Std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & {\([95 \%\) conf. interval] } \\
\hline\((1)\) & .7755022 & .574951 & -0.34 & 0.732 & .1813465 & 3.316323 \\
\hline
\end{tabular}

We did not have to specify the or option. After logistic, lincom assumes or by default.
The effect of low weight ( \(1 \mathrm{wd}=1\) ) is more complicated because we fit an age \(\times\) lwd interaction. We must specify the age of mothers for the effect. The effect among 30-year-old nonsmokers is given by
```

. lincom 1.1wd + 30*1.1wd\#c.age
( 1) [low]1.lwd + 30*[low]1.1wd\#c.age = 0

```
\begin{tabular}{r|rccccr}
\hline low & Odds ratio & Std. err. & z & P>|z| & [95\% conf. interval] \\
\hline\((1)\) & 14.7669 & 13.5669 & 2.93 & 0.003 & 2.439264 & 89.39633 \\
\hline
\end{tabular}
lincom computed \(\exp \left(\beta_{\text {lwd }}+30 \beta_{\text {agelwd }}\right)=14.8\). It may seem odd that we entered it as 1.1 wd \(+30 * 1 . l w d \# c . a g e\), but remember that these terms are just lincom's (and test's) shorthands for _b[1.1wd] and _b[1.1wd\#c.age]. We could have typed
. lincom _b[1.lwd] + 30*_b[1.lwd\#c.age]
( 1) [low] 1.1wd \(+30 *[\) low] \(1.1 \mathrm{wd} \# \mathrm{c}\). age \(=0\)
\begin{tabular}{r|rccccr}
\hline low & Odds ratio & Std. err. & \(z\) & \(P>|z|\) & [95\% conf. interval] \\
\hline\((1)\) & 14.7669 & 13.5669 & 2.93 & 0.003 & 2.439264 & 89.39633 \\
\hline
\end{tabular}

\section*{Multiple-equation models}
lincom also works with multiple-equation models. The only difference is how you refer to the coefficients. Recall that for multiple-equation models, coefficients are referenced using the syntax

\section*{[eqno] varname}
where eqno is the equation number or equation name and varname is the corresponding variable name for the coefficient; see [U] 13.5 Accessing coefficients and standard errors and [R] test for details.

\section*{\(>\) Example 3}

Let's consider example 4 from [R] mlogit (Tarlov et al. 1989; Wells et al. 1989).
```

. use https://www.stata-press.com/data/r18/sysdsn1
(Health insurance data)
. mlogit insure age male nonwhite i.site, nolog
Multinomial logistic regression $\quad$ Number of obs $=615$

```
LR chi2(10) \(=42.99\)
Prob > chi2 \(=0.0000\)
Pseudo R2 \(=0.0387\)
\begin{tabular}{r|clllll}
\hline insure & Coefficient & Std. err. & \(z\) & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline Indemnity & (base outcome) & & & & \\
\hline Prepaid & & & & & & \\
age & -.011745 & .0061946 & -1.90 & 0.058 & -.0238862 & .0003962 \\
male & .5616934 & .2027465 & 2.77 & 0.006 & .1643175 & .9590693 \\
nonwhite & .9747768 & .2363213 & 4.12 & 0.000 & .5115955 & 1.437958 \\
site & & & & & & \\
2 & .1130359 & .2101903 & 0.54 & 0.591 & -.2989296 & .5250013 \\
3 & -.5879879 & .2279351 & -2.58 & 0.010 & -1.034733 & -.1412433 \\
_cons & .2697127 & .3284422 & 0.82 & 0.412 & -.3740222 & .9134476 \\
age & -.0077961 & .0114418 & -0.68 & 0.496 & -.0302217 & .0146294 \\
male & .4518496 & .3674867 & 1.23 & 0.219 & -.268411 & 1.17211 \\
Uninsure & .2170589 & .4256361 & 0.51 & 0.610 & -.6171725 & 1.05129 \\
nonwhite & & & & & & \\
site & -1.211563 & .4705127 & -2.57 & 0.010 & -2.133751 & -.2893747 \\
2 & -.2078123 & .3662926 & -0.57 & 0.570 & -.9257327 & .510108 \\
3 & & & & & & \\
\hline
\end{tabular}

To see the estimate of the sum of the coefficient of male and the coefficient of nonwhite for the Prepaid outcome, we type
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline insure & Coefficient & Std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. & interval] \\
\hline (1) & 1.53647 & . 3272489 & 4.70 & 0.000 & . 8950741 & 2.177866 \\
\hline
\end{tabular}

To view the estimate as a ratio of relative risks (see \([\mathrm{R}]\) mlogit for the definition and interpretation), we specify the rrr option.
. lincom [Prepaid]male + [Prepaid]nonwhite, rrr
(1) [Prepaid]male + [Prepaid]nonwhite \(=0\) \begin{tabular}{r|rlllll} 
\\
\hline insure & RRR & Std. err. & z & P>|z| & [95\% conf. interval] \\
\hline\((1)\) & 4.648154 & 1.521103 & 4.70 & 0.000 & 2.447517 & 8.827451
\end{tabular}

\section*{Stored results}
lincom stores the following in r() :
Scalars
\(r\) (estimate) point estimate
\(r\) (se) estimate of standard error
\(r(d f) \quad\) degrees of freedom
\(\mathrm{r}(\mathrm{t})\) or \(\mathrm{r}(\mathrm{z}) \quad t\) or \(z\) statistic
\(r(p) \quad p\)-value
r(lb) lower bound of confidence interval
\(r(u b) \quad\) upper bound of confidence interval
\(r\) (level) confidence level

\section*{References}

Hosmer, D. W., Jr., and S. A. Lemeshow. 1989. Applied Logistic Regression. New York: Wiley.
Hosmer, D. W., Jr., S. A. Lemeshow, and R. X. Sturdivant. 2013. Applied Logistic Regression. 3rd ed. Hoboken, NJ: Wiley.

Tarlov, A. R., J. E. Ware, Jr., S. Greenfield, E. C. Nelson, E. Perrin, and M. Zubkoff. 1989. The medical outcomes study. An application of methods for monitoring the results of medical care. Journal of the American Medical Association 262: 925-930. https://doi.org/10.1001/jama.1989.03430070073033.
Wells, K. B., R. D. Hays, M. A. Burnam, W. H. Rogers, S. Greenfield, and J. E. Ware, Jr. 1989. Detection of depressive disorder for patients receiving prepaid or fee-for-service care. Results from the Medical Outcomes Survey. Journal of the American Medical Association 262: 3298-3302. https://doi.org/10.1001/jama.1989.03430230083030.

\section*{Also see}
[R] nlcom - Nonlinear combinations of parameters
\([R]\) test - Test linear hypotheses after estimation
[R] testnl - Test nonlinear hypotheses after estimation
[SVY] svy postestimation - Postestimation tools for svy
[U] 13.5 Accessing coefficients and standard errors
[U] 20 Estimation and postestimation commands

\section*{Title}
linktest - Specification link test for single-equation models
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Option & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
linktest performs a link test for model specification.

\section*{Quick start}

Specification link test after a single-equation estimation command without options
linktest
After tobit estimated with right-censoring limit at 24
linktest, ul(24)
After stcox estimated with Efron method for tied failures
linktest, efron
Perform test on full dataset when estimation used a subset of observations
linktest if e(sample) < .

\section*{Menu}

Statistics \(>\) Postestimation

\section*{Syntax}
```

linktest [if][in] [,cmd_options]

```

When if and in are not specified, the link test is performed on the same sample as the previous estimation. collect is allowed; see [U] 11.1.10 Prefix commands.

\section*{Option}
\(\qquad\) Main
cmd_options must be the same options specified with the underlying estimation command, except the display_options may differ.

\section*{Remarks and examples}

The form of the link test implemented here is based on an idea of Tukey (1949), which was further described by Pregibon (1980), elaborating on work in his unpublished thesis (Pregibon 1979). See Methods and formulas below for more details.

\section*{> Example 1}

We want to explain the mileage ratings of cars in our automobile dataset by using the weight, engine displacement, and whether the car is manufactured outside the United States:
```

. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. regress mpg weight displ foreign

```
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline Source & SS & df & MS & Number of obs & = & 74 \\
\hline & & & & F (3, 70) & = & 45.88 \\
\hline Model & 1619.71935 & 3 & 539.906448 & Prob > F & = & 0.0000 \\
\hline Residual & 823.740114 & 70 & 11.7677159 & R -squared & = & 0.6629 \\
\hline & & & & Adj R-squared & = & 0.6484 \\
\hline Total & 2443.45946 & 73 & 33.4720474 & Root MSE & = & 3.4304 \\
\hline
\end{tabular}
\begin{tabular}{r|rcccrr}
\hline mpg & Coefficient & Std. err. & t & P>|t| & [95\% conf. interval] \\
\hline weight & -.0067745 & .0011665 & -5.81 & 0.000 & -.0091011 & -.0044479 \\
displacement & .0019286 & .0100701 & 0.19 & 0.849 & -.0181556 & .0220129 \\
foreign & -1.600631 & 1.113648 & -1.44 & 0.155 & -3.821732 & .6204699 \\
_cons & 41.84795 & 2.350704 & 17.80 & 0.000 & 37.15962 & 46.53628 \\
\hline
\end{tabular}

On the basis of the \(R^{2}\), we are reasonably pleased with this model.
If our model really is specified correctly, then if we were to regress mpg on the prediction and the prediction squared, the prediction squared would have no explanatory power. This is what linktest does:
. linktest
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline Source & SS & df & MS & Number of obs & = & 74 \\
\hline & & & & F (2, 71) & = & 76.75 \\
\hline Model & 1670.71514 & 2 & 835.357572 & Prob > F & = & 0.0000 \\
\hline Residual & 772.744316 & 71 & 10.8837228 & R -squared & = & 0.6837 \\
\hline & & & & Adj R-squared & = & 0.6748 \\
\hline Total & 2443.45946 & 73 & 33.4720474 & Root MSE & = & 3.299 \\
\hline
\end{tabular}
\begin{tabular}{r|rrcrrr}
\hline mpg & Coefficient & Std. err. & t & \(\mathrm{P}>\mid \mathrm{t\mid}\) & [95\% conf. interval] \\
\hline _hat & -.4127198 & .6577736 & -0.63 & 0.532 & -1.724283 & .8988434 \\
_hatsq & .0338198 & .015624 & 2.16 & 0.034 & .0026664 & .0649732 \\
_cons & 14.00705 & 6.713276 & 2.09 & 0.041 & .6211539 & 27.39294 \\
\hline
\end{tabular}

We find that the prediction squared does have explanatory power, so our specification is not as good as we thought.

Although linktest is formally a test of the specification of the dependent variable, it is often interpreted as a test that, conditional on the specification, the independent variables are specified incorrectly. We will follow that interpretation and now include weight squared in our model:
. regress mpg weight c.weight\#c.weight displ foreign
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline Source & SS & df & MS & Number of obs & = & 74 \\
\hline & & & & \(\mathrm{F}(4,69)\) & = & 39.37 \\
\hline Model & 1699.02634 & 4 & 424.756584 & Prob > F & = & 0.0000 \\
\hline Residual & 744.433124 & 69 & 10.7888859 & R -squared & = & 0.6953 \\
\hline & & & & Adj R-squared & = & 0.6777 \\
\hline Total & 2443.45946 & 73 & 33.4720474 & Root MSE & \(=\) & 3.2846 \\
\hline
\end{tabular}
\begin{tabular}{r|rcccrr}
\hline mpg & Coefficient & Std. err. & t & \(\mathrm{P}>|\mathrm{t}|\) & [95\% conf. interval] \\
\hline weight & -.0173257 & .0040488 & -4.28 & 0.000 & -.0254028 & -.0092486 \\
c.weight\# & & & & & & \\
c.weight & \(1.87 e-06\) & \(6.89 e-07\) & 2.71 & 0.008 & \(4.93 e-07\) & \(3.24 \mathrm{e}-06\) \\
& & & & & & \\
displacement & -.0101625 & .0106236 & -0.96 & 0.342 & -.031356 & .011031 \\
foreign & -2.560016 & 1.123506 & -2.28 & 0.026 & -4.801349 & -.3186832 \\
_cons & 58.23575 & 6.449882 & 9.03 & 0.000 & 45.36859 & 71.10291 \\
\hline
\end{tabular}

Now, we perform the link test on our new model:
\begin{tabular}{|c|c|c|c|c|c|c|c|}
\hline Source & SS & df & MS & \multicolumn{2}{|r|}{\multirow[t]{2}{*}{Number of obs F (2, 71)}} & & 74 \\
\hline & & & & & & & 81.08 \\
\hline Model & 1699.39489 & 2 & 849.697445 & & Prob > F & & 0.0000 \\
\hline Residual & 744.06457 & 71 & 10.4797827 & \multicolumn{2}{|r|}{R -squared} & & 0.6955 \\
\hline & & & \multirow[b]{2}{*}{33.4720474} & \multicolumn{2}{|r|}{\multirow[t]{2}{*}{Adj R-squared Root MSE}} & & 0.6869 \\
\hline Total & 2443.45946 & 73 & & & & & 3.2372 \\
\hline mpg & Coefficient & Std. err. & t & \(P>|t|\) & tl [95\% con & f. & interval] \\
\hline _hat & 1.141987 & . 7612218 & 1.50 & 0.138 & \(38-.3758456\) & & 2.659821 \\
\hline _hatsq & -. 0031916 & . 0170194 & -0.19 & 0.852 & \(52-.0371272\) & & . 0307441 \\
\hline _cons & -1.50305 & 8.196444 & -0.18 & 0.855 & \(55-17.84629\) & & 14.84019 \\
\hline
\end{tabular}

We now pass the link test.

\section*{- Example 2}

Above, we followed a standard misinterpretation of the link test - when we discovered a problem, we focused on the explanatory variables of our model. We might consider varying exactly what the link test tests. The link test told us that our dependent variable was misspecified. For those with an engineering background, mpg is indeed a strange measure. It would make more sense to model energy consumption-gallons per mile-in terms of weight and displacement:
```

. gen gpm = 1/mpg

```
. regress gpm weight displ foreign
\begin{tabular}{r|crllll} 
Source & SS & df & \multicolumn{1}{c}{ MS } & \multicolumn{2}{c}{\begin{tabular}{l} 
Number of obs \\
\(\mathrm{F}(3,70)\)
\end{tabular}} & \(=\)
\end{tabular} \begin{tabular}{rl}
74 \\
Model & .009157962
\end{tabular}
\begin{tabular}{r|rlccrr}
\hline gpm & Coefficient & Std. err. & \(t\) & P>|t| & [95\% conf. interval] \\
\hline weight & .0000144 & \(2.15 e-06\) & 6.72 & 0.000 & .0000102 & .0000187 \\
displacement & .0000186 & .0000186 & 1.00 & 0.319 & -.0000184 & .0000557 \\
foreign & .0066981 & .0020531 & 3.26 & 0.002 & .0026034 & .0107928 \\
_cons & .0008917 & .0043337 & 0.21 & 0.838 & -.0077515 & .009535 \\
\hline
\end{tabular}

This model looks every bit as reasonable as our original model:


Specifying the model in terms of gallons per mile also solves the specification problem and results in a more parsimonious specification.

\section*{> Example 3}

The link test can be used with any single-equation estimation procedure, not solely regression. Let's turn our problem around and attempt to explain whether a car is manufactured outside the United States by its mileage rating and weight. To save paper, we will specify logit's nolog option, which suppresses the iteration log:
. logit foreign mpg weight, nolog
\(\begin{array}{ll}\text { Logistic regression } & \text { Number of obs }=74 \\ & \text { LR chi2 (2) } \\ & =35.72 \\ \text { Log likelihood }=-27.175156 & \text { Prob }>\text { chi2 }\end{array}\)
\begin{tabular}{r|rcccrr}
\hline foreign & Coefficient & Std. err. & z & P>|z| & [95\% conf. interval] \\
\hline mpg & -.1685869 & .0919175 & -1.83 & 0.067 & -.3487418 & .011568 \\
weight & -.0039067 & .0010116 & -3.86 & 0.000 & -.0058894 & -.001924 \\
_cons & 13.70837 & 4.518709 & 3.03 & 0.002 & 4.851859 & 22.56487 \\
\hline
\end{tabular}

When we run linktest after logit, the result is another logit specification:
\begin{tabular}{lll} 
- linktest, nolog & & \\
Logistic regression & Number of obs \(=\) & 74 \\
& LR chi2 \((2)\) & \(=36.83\) \\
& Prob \(>\) chi2 \(=0.0000\) \\
Log likelihood \(=-26.615714\) & Pseudo R2 & \(=0.4090\)
\end{tabular}
\begin{tabular}{r|rcccrr}
\hline foreign & Coefficient & Std. err. & z & P>|z| & [95\% conf. interval] \\
\hline _hat & .8438531 & .2738759 & 3.08 & 0.002 & .3070661 & 1.38064 \\
_hatsq & -.1559115 & .1568642 & -0.99 & 0.320 & -.4633596 & .1515366 \\
_cons & .2630557 & .4299598 & 0.61 & 0.541 & -.57965 & 1.105761 \\
\hline
\end{tabular}

The link test reveals no problems with our specification.
If there had been a problem, we would have been virtually forced to accept the misinterpretation of the link test-we would have reconsidered our specification of the independent variables. When using logit, we have no control over the specification of the dependent variable other than to change likelihood functions.

We admit to having seen a dataset once for which the link test rejected the logit specification. We did change the likelihood function, refitting the model using probit, and satisfied the link test. Probit has thinner tails than logit. In general, however, you will not be so lucky.

\section*{Technical note}

You should specify the same options with linktest that you do with the estimation command, although you do not have to follow this advice as literally as we did in the preceding example. logit's nolog option merely suppresses a part of the output, not what is estimated. We specified nolog both times to save space.

If you are testing a tobit model, you must specify the censoring points just as you do with the tobit command.

If you are not sure which options are important, duplicate exactly what you specified on the estimation command.

If you do not specify if exp or in range with linktest, Stata will by default perform the link test on the same sample as the previous estimation. Suppose that you omitted some data when performing your estimation but want to calculate the link test on all the data, which you might do if you believe the model is appropriate for all the data. You would type linktest if e(sample) < . to do this.

\section*{Stored results}
linktest stores the following in \(r()\) :
Scalars
\(r(\mathrm{t}) \quad t\) statistic on _hatsq
\(r(d f)\) degrees of freedom
linktest is not an estimation command in the sense that it leaves previous estimation results unchanged. For instance, after running a regression and performing the link test, typing regress without arguments after the link test still replays the original regression.

For integrating an estimation command with linktest, linktest assumes that the name of the estimation command is stored in \(e(\mathrm{cmd})\) and that the name of the dependent variable is stored in e (depvar). After estimation, it assumes that the number of degrees of freedom for the \(t\) test is given by \(e\left(d f \_m\right)\) if the macro is defined.

If the estimation command reports \(z\) statistics instead of \(t\) statistics, linktest will also report \(z\) statistics. The \(z\) statistic, however, is still returned in \(r(t)\), and \(r(d f)\) is set to a missing value.

\section*{Methods and formulas}

The link test is based on the idea that if a regression or regression-like equation is properly specified, you should be able to find no additional independent variables that are significant except by chance. One kind of specification error is called a link error. In regression, this means that the dependent variable needs a transformation or "link" function to properly relate to the independent variables. The idea of a link test is to add an independent variable to the equation that is especially likely to be significant if there is a link error.

Let
\[
\mathbf{y}=f(\mathbf{X} \boldsymbol{\beta})
\]
be the model and \(\widehat{\boldsymbol{\beta}}\) be the parameter estimates. linktest calculates
\[
\text { _hat }=\mathbf{X} \widehat{\boldsymbol{\beta}}
\]
and
\[
\text { _hatsq }=\text { hat }^{2}
\]

The model is then refit with these two variables, and the test is based on the significance of _hatsq. This is the form suggested by Pregibon (1979) based on an idea of Tukey (1949). Pregibon (1980) suggests a slightly different method that has come to be known as "Pregibon's goodness-of-link test". We prefer the older version because it is universally applicable, straightforward, and a good second-order approximation. It can be applied to any single-equation estimation technique, whereas Pregibon's more recent tests are estimation-technique specific.

\section*{References}

Pregibon, D. 1979. Data analytic methods for generalized linear models. PhD diss., University of Toronto.
-_ 1980. Goodness of link tests for generalized linear models. Journal of the Royal Statistical Society, Series C 29: 15-24. https://doi.org/10.2307/2346405.
Tukey, J. W. 1949. One degree of freedom for non-additivity. Biometrics 5: 232-242. https://doi.org/10.2307/3001938.

\section*{Also see}
\([\mathrm{R}]\) regress postestimation - Postestimation tools for regress

\section*{Title}

Inskew0 — Find zero-skewness log or Box-Cox transform
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
Acknowledgment & Reference & Also see &
\end{tabular}

\section*{Description}
lnskew0 creates newvar \(=\ln ( \pm \exp -k)\), choosing \(k\) and the sign of exp so that the skewness of newvar is zero.
bcskew0 creates newvar \(=\left(\exp ^{\lambda}-1\right) / \lambda\), the Box - Cox power transformation (Box and Cox 1964), choosing \(\lambda\) so that the skewness of newvar is zero. \(\exp\) must be strictly positive.

\section*{Quick start}

Generate newv1, the zero-skewness log transform of continuous variable v1
lnskew0 newv1 = v1
Same as above, but transform ratio of v1 to v2
lnskew0 newv1 = v1/v2
Zero-skewness Box-Cox transform, newv2, of v2
bcskew0 newv2 = v2
Same as above, and change the value for convergence to 0.0001 from the default 0.001 bcskew0 newv2 = v2, zero(.0001)

\section*{Menu}

\section*{Inskew0}

Data \(>\) Create or change data \(>\) Other variable-creation commands \(>\) Zero-skewness log transform

\section*{bcskew0}

Data \(>\) Create or change data \(>\) Other variable-creation commands \(>\) Box-Cox transform

\section*{Syntax}

Zero-skewness log transform
Inskew0 newvar \(=\exp [\) if \(][\) in \(][\), options \(]\)

Zero-skewness Box-Cox transform
bcskew0 newvar \(=\exp [\) if \(][\) in \(][\), options \(]\)
options
Description
Main
delta(\#) increment for derivative of skewness function; default is delta(0.02) for lnskew0 and delta(0.01) for bcskew0
zero(\#) value for determining convergence; default is zero (0.001)
level (\#) compute the confidence interval at confidence level \#; by default, no confidence interval is calculated
collect is allowed with lnskew0 and bcskew0; see [U] 11.1.10 Prefix commands.

\section*{Options}
\(\stackrel{\text { Main }}{ }\)
delta(\#) specifies the increment used for calculating the derivative of the skewness function with respect to \(k\) (lnskew0) or \(\lambda\) (bcskew 0 ). The default values are 0.02 for Inskew 0 and 0.01 for bcskew0.
zero(\#) specifies a value for skewness to determine convergence that is small enough to be considered zero and is, by default, 0.001.
level(\#) specifies the confidence level for the confidence interval for \(k\) (lnskew0) or \(\lambda\) (bcskew0). The confidence interval is calculated only if level() is specified. \# is specified as an integer; 95 means \(95 \%\) confidence intervals. The level () option is honored only if the number of observations exceeds 7.

\section*{Remarks and examples}
> Example 1: Inskew0
Using our automobile dataset (see [U] 1.2.2 Example datasets), we want to generate a new variable equal to \(\ln (\mathrm{mpg}-k)\) to be approximately normally distributed. mpg records the miles per gallon for each of our cars. One feature of the normal distribution is that it has skewness 0 .
```

. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. lnskew0 lnmpg = mpg

| Transform | k | $[95 \%$ conf. interval] | Skewness |
| ---: | ---: | :---: | ---: |
| $\ln (\mathrm{mpg}-\mathrm{k})$ | 5.383659 | (not calculated) | $-7.05 \mathrm{e}-06$ |

```

This created the new variable \(\operatorname{lnmpg}=\ln (\mathrm{mpg}-5.384)\) :
\begin{tabular}{lrrrl}
\multicolumn{2}{c}{ describe } & lnmpg & & \\
\begin{tabular}{r} 
Variable \\
name
\end{tabular} & \begin{tabular}{r} 
Storage \\
type
\end{tabular} & \begin{tabular}{r} 
Display \\
format
\end{tabular} & \begin{tabular}{c} 
Value \\
label
\end{tabular} & Variable label \\
\hline lnmpg & float & \(\% 9.0 \mathrm{~g}\) & & \(\ln (\mathrm{mpg}-5.383659)\)
\end{tabular}

Because we did not specify the level() option, no confidence interval was calculated. At the outset, we could have typed
```

. use https://www.stata-press.com/data/r18/auto, clear
(1978 automobile data)
. lnskewO lnmpg = mpg, level(95)

| Transform | k | $[95 \%$ conf. interval] | Skewness |  |
| ---: | ---: | ---: | ---: | ---: |
| $\ln (\mathrm{mpg}-\mathrm{k})$ | 5.383659 | -17.12339 | 9.892416 | $-7.05 \mathrm{e}-06$ |

```

The confidence interval is calculated under the assumption that \(\ln (\mathrm{mpg}-k)\) really does have a normal distribution. It would be perfectly reasonable to use lnskew0, even if we did not believe that the transformed variable would have a normal distribution-if we literally wanted the zero-skewness transform - although, then the confidence interval would be an approximation of unknown quality to the true confidence interval. If we now wanted to test the believability of the confidence interval, we could also test our new variable lnmpg by using swilk (see [R] swilk) with the lnnormal option.

\section*{- Technical note}

Inskew0 and bcskew0 report the resulting skewness of the variable merely to reassure you of the accuracy of its results. In our example above, lnskew 0 found \(k\) such that the resulting skewness was \(-7 \times 10^{-6}\), a number close enough to zero for all practical purposes. If we wanted to make it even smaller, we could specify the zero() option. Typing lnskew0 new=mpg, zero (1e-8) changes the estimated \(k\) to 5.383552 from 5.383659 and reduces the calculated skewness to \(-2 \times 10^{-11}\).

When you request a confidence interval, Inskew0 may report the lower confidence interval as '.', which should be taken as indicating the lower confidence limit \(k_{L}=-\infty\). (This cannot happen with bcskew0.)

As an example, consider a sample of size \(n\) on \(x\) and assume that the skewness of \(x\) is positive, but not significantly so, at the desired significance level-say, 5\%. Then, no matter how large and negative you make \(k_{L}\), there is no value extreme enough to make the skewness of \(\ln \left(x-k_{L}\right)\) equal the corresponding percentile ( 97.5 for a \(95 \%\) confidence interval) of the distribution of skewness in a normal distribution of the same sample size. You cannot do this because the distribution of \(\ln \left(x-k_{L}\right)\) tends to that of \(x\)-apart from location and scale shift—as \(x \rightarrow \infty\). This "problem" never applies to the upper confidence limit, \(k_{U}\), because the skewness of \(\ln \left(x-k_{U}\right)\) tends to \(-\infty\) as \(k\) tends upward to the minimum value of \(x\).

\section*{> Example 2: bcskew0}

In example 1, using lnskew0 with a variable such as mpg is probably undesirable. mpg has a natural zero, and we are shifting that zero arbitrarily. On the other hand, use of lnskew 0 with a variable such as temperature measured in Fahrenheit or Celsius would be more appropriate because the zero is indeed arbitrary.

For a variable like mpg, it makes more sense to use the Box-Cox power transform (Box and Cox 1964):
\[
y^{(\lambda)}=\frac{y^{\lambda}-1}{\lambda}
\]
\(\lambda\) is free to take on any value, but \(y^{(1)}=y-1, y^{(0)}=\ln (y)\), and \(y^{(-1)}=1-1 / y\).
bcskew0 works like lnskew0:
. bcskew0 bcmpg = mpg, level(95)
\begin{tabular}{c|cccc} 
Transform & L & {\([95 \%\) conf. interval] } & Skewness \\
\hline\(\left(\mathrm{mpg}^{\wedge} \mathrm{L}-1\right) / \mathrm{L}\) & -.3673283 & -1.212752 & .4339645 & .0001898
\end{tabular}

The \(95 \%\) confidence interval includes \(\lambda=-1\) ( \(\lambda\) is labeled L in the output), which has a rather more pleasing interpretation—gallons per mile—than \(\left(\mathrm{mpg}^{-0.3673}-1\right) /(-0.3673)\). The confidence interval, however, is calculated assuming that the power transformed variable is normally distributed. It makes perfect sense to use bcskew0, even when you do not believe that the transformed variable will be normally distributed, but then the confidence interval is an approximation of unknown quality. If you believe that the transformed data are normally distributed, you can alternatively use boxcox to estimate \(\lambda\); see \([\mathrm{R}]\) boxcox.

\section*{Stored results}
lnskew 0 and bcskew0 store the following in \(r()\) :
Scalars
```

    r(gamma) k (lnskew0)
    r(lambda) \lambda (bcskew0)
    r(lb) lower bound of confidence interval
    r(ub) upper bound of confidence interval
    r(skewness) resulting skewness of transformed variable
    ```

\section*{Methods and formulas}

Skewness is as calculated by summarize; see [R] summarize. Newton's method with numeric, uncentered derivatives is used to estimate \(k\) (lnskew0) and \(\lambda\) (bcskew0). For lnskew0, the initial value is chosen so that the minimum of \(x-k\) is 1 , and thus \(\ln (x-k)\) is 0 . bcskew 0 starts with \(\lambda=1\).

\section*{Acknowledgment}

Inskew0 and bcskew0 were written by Patrick Royston of the MRC Clinical Trials Unit, London, and coauthor of the Stata Press book Flexible Parametric Survival Analysis Using Stata: Beyond the Cox Model.

\section*{Reference}

Box, G. E. P., and D. R. Cox. 1964. An analysis of transformations. Journal of the Royal Statistical Society, Series B 26: 211-252.

\section*{Also see}
[R] boxcox - Box-Cox regression models
[ R ] ladder - Ladder of powers
[R] swilk - Shapiro-Wilk and Shapiro-Francia tests for normality

\section*{Title}
\(\mathbf{l o g}\) — Echo copy of session to file

\author{
Description \\ Menu \\ Options for use with both log and cmdlog \\ Option for use with set logtype \\ Stored results \\ Also see
}

\section*{Quick start}

Syntax
Options for use with log
Remarks and examples
Reference

\section*{Description}
log and its subcommands tell Stata to open a \(\log\) file and create a record of what you type and any output that appears in the Results window, to suspend or resume logging, to check logging status, and to close the log file. The default format is Stata Markup and Control Language (SMCL) but can be plain text. You can have up to five SMCL and five text logs open at a time. cmdlog and its subcommands are similar to log but create a command log recording only what you type and can be only plain text. That is, cmdlog is designed only for interactive use. You can have only one command \(\log\) open at a time.
set logtype, set logmsg, and set linesize are commands to control system parameters that relate to logs.

\section*{Quick start}

Begin recording your Stata session in mylog.smcl
log using mylog
Same as above, but use a text format that can be read by a word processor
log using mylog, text
Save a subset of output to mylog2.smcl while mylog.smcl is still open
log using mylog2, name(mylog2)
Close mylog2.smcl and keep mylog.smcl open
log close mylog2
Create a do-file from commands typed interactively
```

cmdlog using mydo.do

```

\section*{Menu}

File \(>\) Log

\section*{Syntax}

Report status of log file
\(\log\)
log query [logname|_all]

Open log file
log using filename \([\), append replace \([\underline{\text { text }} \mid \underline{\text { smcl }}]\) name (logname) [nomsg \(\mid \mathrm{msg}]\) ]

Close log
log close [logname|_all]

Temporarily suspend logging or resume logging
\(\log \{\) off \(\mid\) on \(\}[\) logname \(]\)

Report status of command log file
cmdlog

Open command log file
cmdlog using filename [, append replace]

Close command log, temporarily suspend logging, or resume logging cmdlog \(\{\underline{\text { close }} \mid\) on \(\mid\) off \(\}\)

Set default format for logs set logtype \(\{\underline{\text { text }} \mid \underline{s m c l}\}[\), permanently \(]\)

Set default for display of log messages
set logmsg \{on|off \}

Specify screen width
set linesize \#

In addition to using the log command, you may access the capabilities of log by selecting File \(>\mathbf{L o g}\) from the menu and choosing one of the options in the list.
collect is allowed with log query, log, and cmdlog; see [U] 11.1.10 Prefix commands.

\section*{Options for use with both log and cmdlog}
append specifies that results be appended to an existing file. If the file does not already exist, a new file is created.
replace specifies that filename, if it already exists, be overwritten. When you do not specify either replace or append, the file is assumed to be new. If the specified file already exists, an error message is issued and logging is not started.

\section*{Options for use with log}
text and smcl specify the format in which the log is to be recorded. The default is complicated to describe but is what you would expect:

If you specify the file as filename.smcl, the default is to write the log in SMCL format (regardless of the value of set logtype).
If you specify the file as filename.log, the default is to write the \(\log\) in text format (regardless of the value of set logtype).
If you type filename without an extension and specify neither the smcl option nor the text option, the default is to write the file according to the value of set logtype. If you have not set logtype, then the default is SMCL. Also, the filename you specified will be fixed to read filename. smcl if a SMCL \(\log\) is being created or filename. \(\log\) if a text \(\log\) is being created.

If you specify either the text or smcl option, then what you specify determines how the log is written. If filename was specified without an extension, the appropriate extension is added for you.
If you open multiple log files, you may choose a different format for each file.
name (logname) specifies an optional name you may use to refer to the log while it is open. You can start multiple log files, give each a different logname, and then close, temporarily suspend, or resume them each individually. The default logname is <unnamed>.
nomsg and msg specify whether to display or suppress the default message at the top and bottom of the \(\log\) file.
nomsg suppresses the default message displayed at the top and bottom of the \(\log\) file. This message consists of the log name (if specified in name (), otherwise unnamed), log path, log type, and date opened or closed.
msg displays the default message at the top and bottom of the log file. This option is for use when set logmsg is turned off.

\section*{Option for use with set logtype}
permanently specifies that, in addition to making the change right now, the logtype setting be remembered and become the default setting when you invoke Stata.

\section*{Remarks and examples}

A full \(\log\) is a file containing what you type and Stata's output that is shown in the Results window. To begin logging your session, you type log using filename. If filename contains embedded spaces, remember to enclose it in double quotation marks.

When the default log format is SMCL, log will add the extension . smcl if filename is specified without one. If text is specified or the default log type is changed to text, log adds the extension .log.

We recommend using SMCL because it preserves fonts and colors. SMCL logs can be viewed and printed from the Viewer window, as can any text file; see \([\mathrm{R}]\) view. Users of console Stata can use translate to produce printable versions of log files. translate also converts SMCL logs to text or other formats, such as PostScript or PDF; see [R] translate.

When you open a full log, the default is to show the name of the file and a time and date stamp:
. log using myfile
```

    name: <unnamed>
    log: C:\data\proj1\myfile.smcl
    log type: smcl
opened on: 12 Jan 2023, 12:28:23

```

The above information will appear in the log. If you do not want this information to appear, precede the command by quietly:
. quietly log using myfile
quietly will not suppress any error messages or anything else you need to know.
Similarly, when you close a full \(\log\), the default is to show the full information,
```

. log close
name: <unnamed>
log: C:\data\proj1\myfile.smcl
log type: smcl
closed on: 12 Jan 2023, 12:32:41

```
and that information will also appear in the log. If you want to suppress that, type quietly log close. Alternatively, specifying nomsg with log using will suppress these messages.

If you do not specify name (logname), Stata will use the name <unnamed>, as shown above. However, you can start multiple log files by specifying the name () option with each new log using command. To control a specific log, type, for example, log close logname; to close all log files at one time, type log close _all.

Stata also lets you log only your commands using cmdlog. Command logs are always text files, which makes them easy to convert to do-files. The default extension is .txt instead of . do to keep you from accidentally overwriting your important do-files. However, cmdlog will allow you to specify .do as the extension of filename.

You can have only one command \(\log\) open at a time. However, you can have full logs open while logging your commands. Moreover, the text file you create for your command \(\log\) does not count against the limit of five text logs.
set logtype specifies the default format in which full logs are to be recorded. Initially, full logs are recorded in SMCL format.
set logmsg specifies whether the default message is displayed at the top and bottom of the log file. The default is on. Specifying set logmsg off suppresses the default message from both log using and log close just as if the nomsg option were used with log using.
set linesize specifies the maximum width, in characters, of Stata output. Most commands in Stata do not respect linesize, because it is not important for most commands. Most users never
need to set linesize, because it will automatically be reset if you resize your Results window. This is also why there is no permanently option allowed with set linesize. set linesize is for use with commands such as list and display and is typically used by programmers who wish the output of those commands to be wider or narrower than the current width of the Results window.

\section*{Stored results}
log and cmdlog store the following in \(r()\) :
Macros
\begin{tabular}{ll} 
r(name) & logname \\
r(filename) & name of file \\
r(status) & on or off \\
r(type) & smcl or text
\end{tabular}
log query _all stores the following in \(r()\) :
Scalars
\(r\) (numlogs) number of open \(\log\) files
For each open \(\log\) file, log query _all also stores
```

r(name\#) logname
r(filename\#) name of file
r(status\#) on or off
r(type\#) smcl or text

```
where \# varies between 1 and the value of \(r\) (numlogs). Be aware that \# will not necessarily represent the order in which the \(\log\) files were first opened, nor will it necessarily remain constant for a given log file upon multiple calls to log query.

\section*{Reference}

Hansen, M. R. 2015. graphlog: Creating log files with embedded graphics. Stata Journal 15: 594-596.

\title{
Also see \\ [R] query - Display system parameters \\ [R] translate - Print and translate logs \\ [GSM] 16 Saving and printing results by using logs \\ [GSW] 16 Saving and printing results by using logs \\ [GSU] 16 Saving and printing results by using logs \\ [U] 15 Saving and printing output-log files
}

\section*{Title}
logistic - Logistic regression, reporting odds ratios
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
logistic fits a logistic regression model of depvar on indepvars, where depvar is a \(0 / 1\) variable (or, more precisely, a 0/non-0 variable). Without arguments, logistic redisplays the last logistic estimates. logistic displays estimates as odds ratios; to view coefficients, type logit after running logistic. To obtain odds ratios for any covariate pattern relative to another, see [R] lincom.

\section*{Quick start}

Report odds ratios from logistic regression of y on x 1 and x 2
logistic y x1 x2
Add indicators for values of categorical variable a
logistic y x1 x2 i.a
Same as above, and apply frequency weights defined by wvar
logistic y x1 x2 i.a [fweight=wvar]
Show base level of a
logistic y x1 x2 i.a, baselevels

\section*{Menu}

Statistics \(>\) Binary outcomes \(>\) Logistic regression

\section*{Syntax}
logistic depvar indepvars \([\) if \(][\) in \(]\) [weight] [, options]
options
Model
noconstant
offset (varname)
asis
constraints (constraints)
SE/Robust
vce (vcetype)

Reporting
level(\#)
coef
nocnsreport
display_options

Description
suppress constant term
include varname in model with coefficient constrained to 1
retain perfect predictor variables
apply specified linear constraints
vcetype may be oim, opg, robust, cluster clustvar, bootstrap, or jackknife
set confidence level; default is level (95)
report estimated coefficients
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

Maximization
maximize_options
collinear
coeflegend
control the maximization process; seldom used
keep collinear variables
display legend instead of statistics
indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
depvar and indepvars may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bayes, bootstrap, by, collect, fp, jackknife, mfp, mi estimate, nestreg, rolling, statsby, stepwise,
and svy are allowed; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: logistic.
vce(bootstrap) and vce(jackknife) are not allowed with the mi estimate prefix; see [MI] mi estimate.
Weights are not allowed with the bootstrap prefix; see \([R]\) bootstrap.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options}

Model
noconstant, offset(varname), constraints (constraints); see [R] Estimation options.
asis forces retention of perfect predictor variables and their associated perfectly predicted observations and may produce instabilities in maximization; see [R] probit.
vce(vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.

Reporting
level (\#); see [R] Estimation options.
coef causes logistic to report the estimated coefficients rather than the odds ratios (exponentiated coefficients). coef may be specified when the model is fit or may be used later to redisplay results. coef affects only how results are displayed and not how they are estimated.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\% fint), pformat (\% fmt), sformat (\% \(\% m t\) ), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate (\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from(init_specs); see [R] Maximize. These options are seldom used.

The following options are available with logistic but are not shown in the dialog box:
collinear, coeflegend; see [R] Estimation options.

\section*{Remarks and examples}

Remarks are presented under the following headings:
logistic and logit
Robust estimate of variance
Video examples

\section*{logistic and logit}
logistic provides an alternative and preferred way to fit maximum-likelihood logit models, the other choice being logit ([R] logit).

First, let's dispose of some confusing terminology. We use the words logit and logistic to mean the same thing: maximum likelihood estimation. To some, one or the other of these words connotes transforming the dependent variable and using weighted least squares to fit the model, but that is not how we use either word here. Thus, the logit and logistic commands produce the same results.

The logistic command is generally preferred to the logit command because logistic presents the estimates in terms of odds ratios rather than coefficients. To some people, this may seem disadvantageous, but you can type logit without arguments after logistic to see the underlying coefficients. You should be cautious when interpreting the odds ratio of the constant term. Usually, this odds ratio represents the baseline odds of the model when all predictor variables are set to zero. However, you must verify that a zero value for all predictor variables in the model actually makes sense before continuing with this interpretation.

Nevertheless, \([R]\) logit is still worth reading because logistic shares the same features as logit, including omitting variables due to collinearity or one-way causation.

For an introduction to logistic regression, see Lemeshow and Hosmer (2005), Pagano and Gauvreau (2022, 455-478), or Pampel (2021); for a complete but nonmathematical treatment, see Kleinbaum and Klein (2010); and for a thorough discussion, see Hosmer, Lemeshow, and Sturdivant (2013). See Gould (2000) for a discussion of the interpretation of logistic regression. See Dupont (2009) or Hilbe (2009) for a discussion of logistic regression with examples using Stata. For a discussion using Stata with an emphasis on model specification, see Vittinghoff et al. (2012).

Stata has a variety of commands for performing estimation when the dependent variable is dichotomous or polytomous. See Long and Freese (2014) for a book devoted to fitting these models with Stata. See help estimation commands for a complete list of all of Stata's estimation commands.

\section*{> Example 1}

Consider the following dataset from a study of risk factors associated with low birthweight described in Hosmer, Lemeshow, and Sturdivant (2013, 24).


Sorted by:
We want to investigate the causes of low birthweight. Here race is a categorical variable indicating whether a person is white (race \(=1\) ), black (race \(=2\) ), or some other race (race \(=3\) ). We want indicator (dummy) variables for race included in the regression, so we will use factor variables.
. logistic low age lwt i.race smoke ptl ht ui
Logistic regression

Log likelihood \(=-100.724\)
\begin{tabular}{llr} 
Number of obs & \(=189\) \\
LR chi2(8) & \(=33.22\) \\
Prob > chi2 & \(=0.0001\) \\
Pseudo R2 & \(=0.1416\)
\end{tabular}
\begin{tabular}{r|rrcccr}
\hline low & Odds ratio & Std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline age & .9732636 & .0354759 & -0.74 & 0.457 & .9061578 & 1.045339 \\
lwt & .9849634 & .0068217 & -2.19 & 0.029 & .9716834 & .9984249 \\
race & & & & & & \\
Black & 3.534767 & 1.860737 & 2.40 & 0.016 & 1.259736 & 9.918406 \\
Other & 2.368079 & 1.039949 & 1.96 & 0.050 & 1.001356 & 5.600207 \\
& & & & & & \\
smoke & 2.517698 & 1.00916 & 2.30 & 0.021 & 1.147676 & 5.523162 \\
ptl & 1.719161 & .5952579 & 1.56 & 0.118 & .8721455 & 3.388787 \\
ht & 6.249602 & 4.322408 & 2.65 & 0.008 & 1.611152 & 24.24199 \\
ui & 2.1351 & .9808153 & 1.65 & 0.099 & .8677528 & 5.2534 \\
_cons & 1.586014 & 1.910496 & 0.38 & 0.702 & .1496092 & 16.8134 \\
\hline
\end{tabular}

Note: _cons estimates baseline odds.
The odds ratios are for a one-unit change in the variable. If we wanted the odds ratio for age to be in terms of 4-year intervals, we would type
- generate age4 = age/4
. logistic low age4 lwt i.race smoke ptl ht ui
(output omitted)
After logistic, we can type logit to see the model in terms of coefficients and standard errors:
. logit
Logistic regression \(\quad\) Number of obs \(=189\)
LR chi2 (8) \(=33.22\)
Prob > chi2 \(=0.0001\)
Log likelihood \(=-100.724 \quad\) Pseudo R2 \(=0.1416\)
\begin{tabular}{r|rrrrrr}
\hline low & Coefficient & Std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline age4 & -.1084012 & .1458017 & -0.74 & 0.457 & -.3941673 & .1773649 \\
lwt & -.0151508 & .0069259 & -2.19 & 0.029 & -.0287253 & -.0015763 \\
race & & & & & & \\
Black & 1.262647 & .5264101 & 2.40 & 0.016 & .2309024 & 2.294392 \\
Other & .8620792 & .4391532 & 1.96 & 0.050 & .0013548 & 1.722804 \\
smoke & .9233448 & .4008266 & 2.30 & 0.021 & .137739 & 1.708951 \\
ptl & .5418366 & .346249 & 1.56 & 0.118 & -.136799 & 1.220472 \\
ht & 1.832518 & .6916292 & 2.65 & 0.008 & .4769494 & 3.188086 \\
ui & .7585135 & .4593768 & 1.65 & 0.099 & -.1418484 & 1.658875 \\
_cons & .4612239 & 1.20459 & 0.38 & 0.702 & -1.899729 & 2.822176 \\
\hline
\end{tabular}

If we wanted to see the logistic output again, we would type logistic without arguments.

\section*{> Example 2}

We can specify the confidence interval for the odds ratios with the level() option, and we can do this either at estimation time or when replaying the model. For instance, to see our first model in example 1 with narrower, \(90 \%\) confidence intervals, we might type
```

. logistic, level(90)
Logistic regression Number of obs = 189
LR chi2(8) = 33.22
Prob > chi2 = 0.0001
Pseudo R2 = 0.1416

```
\begin{tabular}{r|rccccc}
\hline low & Odds ratio & Std. err. & \(z\) & \(\mathrm{P}>|\mathrm{z}|\) & [90\% conf. interval] \\
\hline age4 & .8972675 & .1308231 & -0.74 & 0.457 & .7059409 & 1.140448 \\
lwt & .9849634 & .0068217 & -2.19 & 0.029 & .9738063 & .9962483 \\
race & & & & & & \\
Black & 3.534767 & 1.860737 & 2.40 & 0.016 & 1.487028 & 8.402379 \\
Other & 2.368079 & 1.039949 & 1.96 & 0.050 & 1.149971 & 4.876471 \\
& & & & & & \\
smoke & 2.517698 & 1.00916 & 2.30 & 0.021 & 1.302185 & 4.867819 \\
ptl & 1.719161 & .5952579 & 1.56 & 0.118 & .9726876 & 3.038505 \\
ht & 6.249602 & 4.322408 & 2.65 & 0.008 & 2.003487 & 19.49478 \\
ui & 2.1351 & .9808153 & 1.65 & 0.099 & 1.00291 & 4.545424 \\
_cons & 1.586014 & 1.910496 & 0.38 & 0.702 & .2186791 & 11.50288 \\
\hline
\end{tabular}

Note: _cons estimates baseline odds.

\section*{Robust estimate of variance}

If you specify vce (robust), Stata reports the robust estimate of variance described in [U] \(\mathbf{2 0 . 2 2} \mathbf{~ O b}\) taining robust variance estimates. Here is the model previously fit with the robust estimate of variance:
\begin{tabular}{|c|c|c|}
\hline \multirow[t]{3}{*}{Logistic regression} & Number of obs & \(=189\) \\
\hline & Wald chi2(8) & \(=29.02\) \\
\hline & Prob > chi2 & \(=0.0003\) \\
\hline Log pseudolikelihood \(=-100.724\) & Pseudo R2 & \(=0.1416\) \\
\hline
\end{tabular}
\begin{tabular}{r|rcccccc}
\hline & & Robust \\
low & Odds ratio & std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline age & .9732636 & .0329376 & -0.80 & 0.423 & .9108015 & 1.040009 \\
lwt & .9849634 & .0070209 & -2.13 & 0.034 & .9712984 & .9988206 \\
race & & & & & & \\
Black & 3.534767 & 1.793616 & 2.49 & 0.013 & 1.307504 & 9.556051 \\
Other & 2.368079 & 1.026563 & 1.99 & 0.047 & 1.012512 & 5.538501 \\
& & & & & & \\
smoke & 2.517698 & .9736417 & 2.39 & 0.017 & 1.179852 & 5.372537 \\
ptl & 1.719161 & .7072902 & 1.32 & 0.188 & .7675715 & 3.850476 \\
ht & 6.249602 & 4.102026 & 2.79 & 0.005 & 1.726445 & 22.6231 \\
ui & 2.1351 & 1.042775 & 1.55 & 0.120 & .8197749 & 5.560858 \\
_cons & 1.586014 & 1.939482 & 0.38 & 0.706 & .144345 & 17.42658 \\
\hline
\end{tabular}

Note: _cons estimates baseline odds.

Also, you can specify vce(cluster clustvar) and then, within cluster, relax the assumption of independence. To illustrate this, we have made some fictional additions to the low-birthweight data.

Say that these data are not a random sample of mothers but instead are a random sample of mothers from a random sample of hospitals. In fact, that may be true-we do not know the history of these data.

Hospitals specialize, and it would not be too incorrect to say that some hospitals specialize in more difficult cases. We are going to show two extremes. In one, all hospitals are alike, but we are going to estimate under the possibility that they might differ. In the other, hospitals are strikingly different. In both cases, we assume that patients are drawn from 20 hospitals.

In both examples, we will fit the same model, and we will type the same command to fit it. Below are the same data we have been using but with a new variable, hospid, that identifies from which of the 20 hospitals each patient was drawn (and which we have made up):
```

. use https://www.stata-press.com/data/r18/hospid1, clear
. logistic low age lwt i.race smoke ptl ht ui, vce(cluster hospid)
Logistic regression Number of obs = 189
Wald chi2(8) = 49.67
Prob > chi2 = 0.0000
Pseudo R2 = 0.1416
(Std. err. adjusted for 20 clusters in hospid)

```
\begin{tabular}{r|rccccr}
\hline & & Robust & & & \\
low & Odds ratio & std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline age & .9732636 & .0397476 & -0.66 & 0.507 & .898396 & 1.05437 \\
lwt & .9849634 & .0057101 & -2.61 & 0.009 & .9738352 & .9962187 \\
race & & & & & & \\
Black & 3.534767 & 2.013285 & 2.22 & 0.027 & 1.157563 & 10.79386 \\
Other & 2.368079 & .8451325 & 2.42 & 0.016 & 1.176562 & 4.766257 \\
& & & & & & \\
smoke & 2.517698 & .8284259 & 2.81 & 0.005 & 1.321062 & 4.79826 \\
ptl & 1.719161 & .6676221 & 1.40 & 0.163 & .8030814 & 3.680219 \\
ht & 6.249602 & 4.066275 & 2.82 & 0.005 & 1.74591 & 22.37086 \\
ui & 2.1351 & 1.093144 & 1.48 & 0.138 & .7827337 & 5.824014 \\
_cons & 1.586014 & 1.661913 & 0.44 & 0.660 & .2034094 & 12.36639 \\
\hline
\end{tabular}

Note: _cons estimates baseline odds.
The standard errors are similar to the standard errors we have previously obtained, whether we used the robust or conventional estimators. In this example, we invented the hospital IDs randomly.

Here are the results of the estimation with the same data but with a different set of hospital IDs:
. use https://www.stata-press.com/data/r18/hospid2
. logistic low age lwt i.race smoke ptl ht ui, vce(cluster hospid)
Logistic regression


Log pseudolikelihood \(=-100.724\)\(\quad\)\begin{tabular}{r} 
Number of obs
\end{tabular}
(Std. err. adjusted for 20 clusters in hospid)
\begin{tabular}{r|rccccr}
\hline & & Robust \\
low & Odds ratio & std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\%conf. interval] \\
\hline age & .9732636 & .0293064 & -0.90 & 0.368 & .9174862 & 1.032432 \\
lwt & .9849634 & .0106123 & -1.41 & 0.160 & .9643817 & 1.005984 \\
race & & & & & & \\
Black & 3.534767 & 3.120338 & 1.43 & 0.153 & .6265521 & 19.9418 \\
Other & 2.368079 & 1.297738 & 1.57 & 0.116 & .8089594 & 6.932114 \\
smoke & 2.517698 & 1.570287 & 1.48 & 0.139 & .7414969 & 8.548655 \\
ptl & 1.719161 & .6799153 & 1.37 & 0.171 & .7919045 & 3.732161 \\
ht & 6.249602 & 7.165454 & 1.60 & 0.110 & .660558 & 59.12808 \\
ui & 2.1351 & 1.411977 & 1.15 & 0.251 & .5841231 & 7.804266 \\
_cons & 1.586014 & 1.946253 & 0.38 & 0.707 & .1431423 & 17.573 \\
\hline
\end{tabular}

Note: _cons estimates baseline odds.
Note the strikingly larger standard errors. What happened? In these data, women most likely to have low-birthweight babies are sent to certain hospitals, and the decision on likeliness is based not just on age, smoking history, etc., but on other things that doctors can see but that are not recorded in our data. Thus, merely because a woman is at one of the centers identifies her to be more likely to have a low-birthweight baby.

\section*{Video examples}

Logistic regression, part 1: Binary predictors
Logistic regression, part 2: Continuous predictors
Logistic regression, part 3: Factor variables

\section*{Stored results}
logistic stores the following in e() :

Scalars
e(N)
e(N_cds)
e(N_cdf)
e(k)
\(e\left(k \_e q\right)\)
e(k_eq_model)
e(k_dv)
e(df_m)
e(r2_p)
e(ll)
e(11_0)
e(N_clust)
e(chi2)
e(p)
e(rank)
e(ic)
e(rc)
e(converged)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(offset)
e(chi2type)
e(vce)
e(vcetype)
e(opt)
\(e\) (which)
e(ml_method)
e(user)
e(technique)
e(properties)
e(estat_cmd)
\(e\) (predict)
e(marginsok)
e(marginsnotok)
\(e\) (asbalanced)
e(asobserved)
Matrices
e(b)
e(Cns)
e(ilog)
e(gradient)
e(mns)
e(rules)
e(V)
e(V_modelbased)
Functions
e(sample)
number of observations
number of completely determined successes
number of completely determined failures
number of parameters
number of equations in \(e(b)\)
number of equations in overall model test
number of dependent variables
model degrees of freedom
pseudo- \(R^{2}\)
log likelihood
log likelihood, constant-only model
number of clusters
\(\chi^{2}\)
\(p\)-value for model test
rank of e(V)
number of iterations
return code
1 if converged, 0 otherwise
logistic
command as typed
name of dependent variable
weight type
weight expression
title in estimation output
name of cluster variable
linear offset variable
Wald or LR; type of model \(\chi^{2}\) test
vcetype specified in vce()
title used to label Std. err.
type of optimization
max or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement estat
program used to implement predict
predictions allowed by margins
predictions disallowed by margins
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
constraints matrix
iteration \(\log\) (up to 20 iterations)
gradient vector
vector of means of the independent variables
information about perfect predictors
variance-covariance matrix of the estimators
model-based variance
marks estimation sample

In addition to the above, the following is stored in r() :
Matrices \(r\) (table)
matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r -class command is run after the estimation command.

\section*{Methods and formulas}

Define \(\mathbf{x}_{j}\) as the (row) vector of independent variables, augmented by 1 , and \(\mathbf{b}\) as the corresponding estimated parameter (column) vector. The logistic regression model is fit by logit; see [R] logit for details of estimation.

The odds ratio corresponding to the \(i\) th coefficient is \(\psi_{i}=\exp \left(b_{i}\right)\). The standard error of the odds ratio is \(s_{i}^{\psi}=\psi_{i} s_{i}\), where \(s_{i}\) is the standard error of \(b_{i}\) estimated by logit.

Define \(I_{j}=\mathbf{x}_{j} \mathbf{b}\) as the predicted index of the \(j\) th observation. The predicted probability of a positive outcome is
\[
p_{j}=\frac{\exp \left(I_{j}\right)}{1+\exp \left(I_{j}\right)}
\]

This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce(robust) and vce(cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.
logistic also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

\section*{References}

Archer, K. J., and S. A. Lemeshow. 2006. Goodness-of-fit test for a logistic regression model fitted using survey sample data. Stata Journal 6: 97-105.

Buis, M. L. 2010a. Direct and indirect effects in a logit model. Stata Journal 10: 11-29.
——. 2010b. Stata tip 87: Interpretation of interactions in nonlinear models. Stata Journal 10: 305-308.
Dupont, W. D. 2009. Statistical Modeling for Biomedical Researchers: A Simple Introduction to the Analysis of Complex Data. 2nd ed. Cambridge: Cambridge University Press.

Fernandez-Felix, B. M., E. García-Esquinas, A. Muriel, A. Royuela, and J. Zamora. 2021. Bootstrap internal validation command for predictive logistic regression models. Stata Journal 21: 498-509.
Freese, J. 2002. Least likely observations in regression models for categorical outcomes. Stata Journal 2: 296-300.
Gould, W. W. 2000. sg124: Interpreting logistic regression in all its forms. Stata Technical Bulletin 53: 19-29. Reprinted in Stata Technical Bulletin Reprints, vol. 9, pp. 257-270. College Station, TX: Stata Press.

Hilbe, J. M. 2009. Logistic Regression Models. Boca Raton, FL: Chapman and Hall/CRC.
Hosmer, D. W., Jr., S. A. Lemeshow, and R. X. Sturdivant. 2013. Applied Logistic Regression. 3rd ed. Hoboken, NJ: Wiley.

Kleinbaum, D. G., and M. Klein. 2010. Logistic Regression: A Self-Learning Text. 3rd ed. New York: Springer.
Lalanne, C., and M. Mesbah. 2016. Biostatistics and Computer-based Analysis of Health Data Using Stata. London: ISTE Press.

Lemeshow, S. A., and J.-R. L. Gall. 1994. Modeling the severity of illness of ICU patients: A systems update. Journal of the American Medical Association 272: 1049-1055. https://doi.org/10.1001/jama.1994.03520130087038.

Lemeshow, S. A., and D. W. Hosmer, Jr. 2005. Logistic regression. In Vol. 2 of Encyclopedia of Biostatistics, ed. P. Armitage and T. Colton, 2870-2880. Chichester, UK: Wiley.

Long, J. S., and J. Freese. 2014. Regression Models for Categorical Dependent Variables Using Stata. 3rd ed. College Station, TX: Stata Press.
Mehmetoglu, M., and T. G. Jakobsen. 2022. Applied Statistics Using Stata: A Guide for the Social Sciences. 2nd ed. Thousand Oaks, CA: Sage.

Miranda, A., and S. Rabe-Hesketh. 2006. Maximum likelihood estimation of endogenous switching and sample selection models for binary, ordinal, and count variables. Stata Journal 6: 285-308.
Mitchell, M. N., and X. Chen. 2005. Visualizing main effects and interactions for binary logit models. Stata Journal 5: 64-82.

Pagano, M., and K. Gauvreau. 2022. Principles of Biostatistics. 3rd ed. Boca Raton, FL: CRC Press.
Pampel, F. C. 2021. Logistic Regression: A Primer. 2nd ed. Thousand Oaks, CA: Sage.
Pregibon, D. 1981. Logistic regression diagnostics. Annals of Statistics 9: 705-724.
https://doi.org/10.1214/aos/1176345513.
Schonlau, M. 2005. Boosted regression (boosting): An introductory tutorial and a Stata plugin. Stata Journal 5: 330-354.

Uberti, L. J. 2022. Interpreting logit models. Stata Journal 22: 60-76.
Vittinghoff, E., D. V. Glidden, S. C. Shiboski, and C. E. McCulloch. 2012. Regression Methods in Biostatistics: Linear, Logistic, Survival, and Repeated Measures Models. 2nd ed. New York: Springer.
Xu, J., and J. S. Long. 2005. Confidence intervals for predicted outcomes in regression models for categorical outcomes. Stata Journal 5: 537-559.

\section*{Also see}
[R] logistic postestimation - Postestimation tools for logistic
[R] brier - Brier score decomposition
[R] cloglog - Complementary log-log regression
\([R]\) exlogistic - Exact logistic regression
[R] logit - Logistic regression, reporting coefficients
[R] npregress kernel - Nonparametric kernel regression
[R] npregress series - Nonparametric series regression
\([R]\) roc - Receiver operating characteristic (ROC) analysis
[BAYES] bayes: logistic - Bayesian logistic regression, reporting odds ratios
[FMM] fmm: logit - Finite mixtures of logistic regression models
[LASSO] Lasso intro - Introduction to lasso
[MI] Estimation - Estimation commands for use with mi estimate
[SVY] svy estimation - Estimation commands for survey data
[XT] xtlogit - Fixed-effects, random-effects, and population-averaged logit models
[U] 20 Estimation and postestimation commands

\section*{Title}
logistic postestimation - Postestimation tools for logistic

Postestimation commands
Remarks and examples
Also see
predict margins
Methods and formulas References

\section*{Postestimation commands}

The following postestimation commands are of special interest after logistic:
Command Description
\begin{tabular}{ll} 
estat classification & report various summary statistics, including the classification table \\
estat gof & Pearson or Hosmer-Lemeshow goodness-of-fit test \\
lroc & compute area under ROC curve and graph the curve \\
lsens & graph sensitivity and specificity versus probability cutoff
\end{tabular}

These commands are not appropriate with svy estimation results.

The following standard postestimation commands are also available:
Command Description
\begin{tabular}{|c|c|}
\hline contrast & contrasts and ANOVA-style joint tests of estimates \\
\hline estat ic & Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) \\
\hline estat summarize & summary statistics for the estimation sample \\
\hline estat vce & variance-covariance matrix of the estimators (VCE) \\
\hline estat (svy) & postestimation statistics for survey data \\
\hline estimates & cataloging estimation results \\
\hline etable & table of estimation results \\
\hline * forecast & dynamic forecasts and simulations \\
\hline *hausman & Hausman's specification test \\
\hline lincom & point estimates, standard errors, testing, and inference for linear combinations of coefficients \\
\hline linktest & link test for model specification \\
\hline * lrtest & likelihood-ratio test \\
\hline margins & marginal means, predictive margins, marginal effects, and average marginal effects \\
\hline marginsplot & graph the results from margins (profile plots, interaction plots, etc.) \\
\hline nlcom & point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients \\
\hline predict & probabilities, influence statistics, residuals, etc. \\
\hline predictnl & point estimates, standard errors, testing, and inference for generalized predictions \\
\hline
\end{tabular}
pwcompare
suest
test
testnl
pairwise comparisons of estimates
seemingly unrelated estimation
Wald tests of simple and composite linear hypotheses
Wald tests of nonlinear hypotheses
*forecast, hausman, and lrtest are not appropriate with svy estimation results. forecast is also not appropriate with mi estimation results.

\section*{predict}

\section*{Description for predict}
predict creates a new variable containing predictions such as probabilities, linear predictions, standard errors, influence statistics, deviance residuals, leverages, sequential numbers, Pearson residuals, and the equation-level score.

\section*{Menu for predict}

\author{
Statistics \(>\) Postestimation
}

\section*{Syntax for predict}
\[
\text { predict }[\text { type }] \text { newvar }[\text { if }][\text { in }][\text {, statistic nooffset rules asif }]
\]

\section*{statistic Description}

Main
pr probability of a positive outcome; the default
\(\mathrm{xb} \quad\) linear prediction
stdp standard error of the prediction
* dbeta
* deviance
* dx2
*ddeviance
* hat
* number
*residuals
*rstandard score

Pregibon (1981) \(\Delta \widehat{\beta}\) influence statistic
deviance residual
Hosmer, Lemeshow, and Sturdivant (2013) \(\Delta \chi^{2}\) influence statistic
Hosmer, Lemeshow, and Sturdivant (2013) \(\Delta D\) influence statistic
Pregibon (1981) leverage
sequential number of the covariate pattern
Pearson residuals; adjusted for number sharing covariate pattern standardized Pearson residuals; adjusted for number sharing covariate pattern first derivative of the \(\log\) likelihood with respect to \(\mathbf{x}_{j} \boldsymbol{\beta}\)

Unstarred statistics are available both in and out of sample; type predict ... if e(sample)... if wanted only for the estimation sample. Starred statistics are calculated only for the estimation sample, even when if e(sample) is not specified. \(\mathrm{pr}, \mathrm{xb}\), stdp, and score are the only options allowed with svy estimation results.

\section*{Options for predict}
\(\qquad\) Main
pr , the default, calculates the probability of a positive outcome.
xb calculates the linear prediction.
stdp calculates the standard error of the linear prediction.
dbeta calculates the Pregibon (1981) \(\Delta \widehat{\beta}\) influence statistic, a standardized measure of the difference in the coefficient vector that is due to deletion of the observation along with all others that share the same covariate pattern. In Hosmer, Lemeshow, and Sturdivant (2013, 154-155) jargon, this statistic is \(M\)-asymptotic; that is, it is adjusted for the number of observations that share the same covariate pattern.
deviance calculates the deviance residual.
dx2 calculates the Hosmer, Lemeshow, and Sturdivant \((2013,191) \Delta \chi^{2}\) influence statistic, reflecting the decrease in the Pearson \(\chi^{2}\) that is due to deletion of the observation and all others that share the same covariate pattern.
ddeviance calculates the Hosmer, Lemeshow, and Sturdivant (2013, 191) \(\Delta D\) influence statistic, which is the change in the deviance residual that is due to deletion of the observation and all others that share the same covariate pattern.
hat calculates the Pregibon (1981) leverage or the diagonal elements of the hat matrix adjusted for the number of observations that share the same covariate pattern.
number numbers the covariate patterns-observations with the same covariate pattern have the same number. Observations not used in estimation have number set to missing. The first covariate pattern is numbered 1 , the second 2 , and so on.
residuals calculates the Pearson residual as given by Hosmer, Lemeshow, and Sturdivant (2013, 155) and adjusted for the number of observations that share the same covariate pattern.
rstandard calculates the standardized Pearson residual as given by Hosmer, Lemeshow, and Sturdivant \((2013,191)\) and adjusted for the number of observations that share the same covariate pattern.
score calculates the equation-level score, \(\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)\).

Options
nooffset is relevant only if you specified offset (varname) for logistic. It modifies the calculations made by predict so that they ignore the offset variable; the linear prediction is treated as \(\mathbf{x}_{j} \mathbf{b}\) rather than as \(\mathbf{x}_{j} \mathbf{b}+\) offset \(_{j}\).
rules requests that Stata use any rules that were used to identify the model when making the prediction. By default, Stata calculates missing for excluded observations. See example 1 in [R] logit postestimation.
asif requests that Stata ignore the rules and the exclusion criteria and calculate predictions for all observations possible by using the estimated parameter from the model. See example 1 in \([R]\) logit postestimation.

\section*{margins}

\section*{Description for margins}
margins estimates margins of response for probabilities and linear predictions.

\section*{Menu for margins}

Statistics \(>\) Postestimation

\section*{Syntax for margins}
```

margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...) ...] [options]]

```
statistic Description
\(\mathrm{pr} \quad\) probability of a positive outcome; the default
\(\mathrm{xb} \quad\) linear prediction
stdp not allowed with margins
dbeta not allowed with margins
deviance not allowed with margins
dx2 not allowed with margins
ddeviance not allowed with margins
hat not allowed with margins
number not allowed with margins
residuals not allowed with margins
rstandard not allowed with margins
score not allowed with margins

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

\section*{Remarks and examples}
predict is used after logistic to obtain predicted probabilities, residuals, and influence statistics for the estimation sample. The suggested diagnostic graphs below are from Hosmer, Lemeshow, and Sturdivant (2013), where they are more elaborately explained. Also see Collett (2003, 129-168) for a thorough discussion of model checking.

Remarks are presented under the following headings:
```

predict without options
predict with the xb and stdp options
predict with the residuals option
predict with the number option
predict with the deviance option
predict with the rstandard option
predict with the hat option
predict with the dx2 option
predict with the ddeviance option
predict with the dbeta option

```

\section*{predict without options}

Typing predict newvar after estimation calculates the predicted probability of a positive outcome.
In example 1 of \([R]\) logistic, we ran the model logistic low age lwt i.race smoke ptl ht ui. We obtain the predicted probabilities of a positive outcome by typing
```

. use https://www.stata-press.com/data/r18/lbw
(Hosmer \& Lemeshow data)
. logistic low age lwt i.race smoke ptl ht ui
(output omitted)
. predict p
(option pr assumed; Pr(low))
. summarize p low

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| p | 189 | .3121693 | .1913915 | .0272559 | .8391283 |
| low | 189 | .3121693 | .4646093 | 0 | 1 |

```

\section*{predict with the \(\mathbf{x b}\) and stdp options}
predict with the xb option calculates the linear combination \(x_{j} \mathbf{b}\), where \(x_{j}\) are the independent variables in the \(j\) th observation and \(\mathbf{b}\) is the estimated parameter vector. This is sometimes known as the index function because the cumulative distribution function indexed at this value is the probability of a positive outcome.

With the stdp option, predict calculates the standard error of the prediction, which is not adjusted for replicated covariate patterns in the data. The influence statistics described below are adjusted for replicated covariate patterns in the data.

\section*{predict with the residuals option}
predict can calculate more than predicted probabilities. The Pearson residual is defined as the square root of the contribution of the covariate pattern to the Pearson \(\chi^{2}\) goodness-of-fit statistic, signed according to whether the observed number of positive responses within the covariate pattern is less than or greater than expected. For instance,
. predict r, residuals
. summarize r, detail
Pearson residual
\begin{tabular}{rcrlr}
\hline & Percentiles & Smallest & & \\
\(1 \%\) & -1.750923 & -2.283885 & & 189 \\
\(5 \%\) & -1.129907 & -1.750923 & & 189 \\
\(10 \%\) & -.9581174 & -1.636279 & Obs & \\
\(25 \%\) & -.6545911 & -1.636279 & Sum of wgt. & -.0242299 \\
\(50 \%\) & -.3806923 & & Mean & .9970949 \\
& & Largest & Std. dev. & \\
\(75 \%\) & .8162894 & 2.23879 & & .9941981 \\
\(90 \%\) & 1.510355 & 2.317558 & Variance & .8618271 \\
\(95 \%\) & 1.747948 & 3.002206 & Skewness & 3.038448
\end{tabular}

We notice the prevalence of a few large positive residuals:
```

. sort r
. list id r low p age race in -5/l

```
\begin{tabular}{l|rrrrrr|}
\cline { 2 - 7 } & id & \(r\) & low & p & age & race \\
\cline { 2 - 7 } & 185. & 33 & 2.224501 & 1 & .1681123 & 19 \\
186. & 57 & 2.23879 & 1 & .166329 & 15 & White \\
187. & 16 & 2.317558 & 1 & .1569594 & 27 & Other \\
188. & 77 & 3.002206 & 1 & .0998678 & 26 & White \\
189. & 36 & 3.126763 & 1 & .0927932 & 24 & White \\
\hline
\end{tabular}

\section*{predict with the number option}

Covariate patterns play an important role in logistic regression. Two observations are said to share the same covariate pattern if the independent variables for the two observations are identical. Although we might think of having individual observations, the statistical information in the sample can be summarized by the covariate patterns, the number of observations with that covariate pattern, and the number of positive outcomes within the pattern. Depending on the model, the number of covariate patterns can approach or be equal to the number of observations, or it can be considerably less.

Stata calculates all the residual and diagnostic statistics in terms of covariate patterns, not observations. That is, all observations with the same covariate pattern are given the same residual and diagnostic statistics. Hosmer, Lemeshow, and Sturdivant (2013, 154-155) argue that such " \(M\) asymptotic" statistics are more useful than " \(N\)-asymptotic" statistics.

To understand the difference, think of an observed positive outcome with predicted probability of 0.8 . Taking the observation in isolation, the residual must be positive-we expected 0.8 positive responses and observed 1 . This may indeed be the correct residual, but not necessarily. Under the \(M\)-asymptotic definition, we ask how many successes we observed across all observations with this covariate pattern. If that number were, say, six, and there were a total of 10 observations with this covariate pattern, then the residual is negative for the covariate pattern - we expected eight positive outcomes but observed six. predict makes this kind of calculation and then attaches the same residual to all observations in the covariate pattern.

Occasionally, you might want to find all observations sharing a covariate pattern. number allows you to do this:
```

. predict pattern, number

```
. summarize pattern
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline pattern & 189 & 89.2328 & 53.16573 & 1 & 182
\end{tabular}

We previously fit the model logistic low age lwt i.race smoke ptl ht ui over 189 observations. There are 182 covariate patterns in our data.

\section*{predict with the deviance option}

The deviance residual is defined as the square root of the contribution to the likelihood-ratio test statistic of a saturated model versus the fitted model. It has slightly different properties from the Pearson residual (see Hosmer, Lemeshow, and Sturdivant [2013, 155-157]):
```

predict d, deviance
. summarize d, detail

```
deviance residual
\begin{tabular}{rcrlr}
\hline & Percentiles & Smallest & & \\
\(1 \%\) & -1.843472 & -1.911621 & & \\
\(5 \%\) & -1.33477 & -1.843472 & & 189 \\
\(10 \%\) & -1.148316 & -1.843472 & Obs & 189 \\
\(25 \%\) & -.8445325 & -1.674869 & Sum of wgt. & -.1228811 \\
\(50 \%\) & -.5202702 & & Mean & 1.049237 \\
& & Largest & Std. dev. & \\
\(75 \%\) & .9129041 & 1.894089 & & 1.100898 \\
\(90 \%\) & 1.541558 & 1.924457 & Variance & .6598857 \\
\(95 \%\) & 1.673338 & 2.146583 & Skewness & 2.036938
\end{tabular}

\section*{predict with the rstandard option}

Pearson residuals do not have a standard deviation equal to 1. rstandard generates Pearson residuals normalized to have an expected standard deviation equal to 1 .
```

. predict rs, rstandard
. summarize r rs

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| r | 189 | -.0242299 | .9970949 | -2.283885 | 3.126763 |
| rs | 189 | -.0279135 | 1.026406 | -2.4478 | 3.149081 |
| correlate (obs $=189)$ |  |  |  |  |  |


|  | $r$ | $r s$ |
| ---: | ---: | ---: |
| $r$ | 1.0000 |  |
| $r s$ | 0.9998 | 1.0000 |

```

Remember that we previously created \(r\) containing the (unstandardized) Pearson residuals. In these data, whether we use standardized or unstandardized residuals does not matter much.

\section*{predict with the hat option}
hat calculates the leverage of a covariate pattern - a scaled measure of distance in terms of the independent variables. Large values indicate covariate patterns far from the average covariate pattern that can have a large effect on the fitted model even if the corresponding residual is small. Consider the following graph:
```

. predict h, hat
. scatter h r, xline(0) ytitle("Leverage")

```


The points to the left of the vertical line are observed negative outcomes; here our data contain almost as many covariate patterns as observations, so most covariate patterns are unique. In such unique patterns, we observe either 0 or 1 success and expect \(p\), thus forcing the sign of the residual. If we had fewer covariate patterns-if we did not have continuous variables in our model-there would be no such interpretation, and we would not have drawn the vertical line at 0 .

Points on the left and right edges of the graph represent large residuals - covariate patterns that are not fit well by our model. Points at the top of our graph represent high leverage patterns. When analyzing the influence of observations on the model, we are most interested in patterns with high leverage and small residuals-patterns that might otherwise escape our attention.

\section*{predict with the dx2 option}

There are many ways to measure influence, and hat is one example. dx 2 measures the decrease in the Pearson \(\chi^{2}\) goodness-of-fit statistic that would be caused by deleting an observation (and all others sharing the covariate pattern):
. predict dx2, dx2
. scatter dx2 p


Paraphrasing Hosmer, Lemeshow, and Sturdivant (2013, 195-197), the points going from the top left to the bottom right correspond to covariate patterns with the number of positive outcomes equal to the number in the group; the points on the other curve correspond to 0 positive outcomes. In our data, most of the covariate patterns are unique, so the points tend to lie along one or the other curves; the points that are off the curves correspond to the few repeated covariate patterns in our data in which all the outcomes are not the same.

We examine this graph for large values of dx2-there are two at the top left.

\section*{predict with the ddeviance option}

Another measure of influence is the change in the deviance residuals due to deletion of a covariate pattern:
```

predict dd, ddeviance

```

As with dx2, we typically graph ddeviance against the probability of a positive outcome. We direct you to Hosmer, Lemeshow, and Sturdivant \((2013,195)\) for an example and for the interpretation of this graph.

\section*{predict with the dbeta option}

One of the more direct measures of influence of interest to model fitters is the Pregibon (1981) dbeta measure, a measure of the change in the coefficient vector that would be caused by deleting an observation (and all others sharing the covariate pattern):
- predict db, dbeta
. scatter db p


One observation has a large effect on the estimated coefficients. We can easily find this point:
```

. sort db
. list in l

```
189.
\begin{tabular}{|c|c|c|c|c|c|c|c|c|c|c|}
\hline \[
\begin{array}{r}
\text { id } \\
188
\end{array}
\] & \[
\begin{array}{r}
\text { low } \\
0
\end{array}
\] & \[
\begin{array}{r}
\text { age } \\
25
\end{array}
\] & \[
\begin{array}{r}
\text { lwt } \\
95
\end{array}
\] & \begin{tabular}{l}
race \\
White
\end{tabular} & smoke Smoker & \[
\begin{array}{r}
\text { ptl } \\
3
\end{array}
\] & \[
\begin{array}{r}
\mathrm{ht} \\
0
\end{array}
\] & ui
1 & \[
\begin{array}{r}
\mathrm{ftv} \\
0
\end{array}
\] & \[
\begin{array}{r}
\text { bwt } \\
3637
\end{array}
\] \\
\hline . 839 & \[
\begin{array}{r}
p \\
283
\end{array}
\] & -2.28 & \[
\begin{array}{r}
r \\
385
\end{array}
\] & pattern
\[
117
\] & -1.91 & & -2. & rs & & \\
\hline \multicolumn{4}{|c|}{\[
\begin{array}{r}
\mathrm{dx} 2 \\
5.991726
\end{array}
\]} & \multicolumn{3}{|c|}{\[
\begin{array}{r}
\mathrm{dd} \\
4.197658
\end{array}
\]} & \multicolumn{4}{|c|}{\[
\begin{array}{r}
\mathrm{db} \\
.8909163
\end{array}
\]} \\
\hline
\end{tabular}

Hosmer, Lemeshow, and Sturdivant \((2013,196)\) suggest a graph that combines two of the influence measures:
```

. scatter dx2 p [aw=db], title("Symbol size proportional to dBeta") mfcolor(none)

```

Symbol size proportional to dBeta


We can easily spot the most influential points by the dbeta and dx 2 measures.

\section*{Methods and formulas}

Let \(j\) index observations. Define \(M_{j}\) for each observation as the total number of observations sharing \(j\) 's covariate pattern. Define \(Y_{j}\) as the total number of positive responses among observations sharing \(j\) 's covariate pattern.

The Pearson residual for the \(j\) th observation is defined as
\[
r_{j}=\frac{Y_{j}-M_{j} p_{j}}{\sqrt{M_{j} p_{j}\left(1-p_{j}\right)}}
\]

For \(M_{j}>1\), the deviance residual \(d_{j}\) is defined as
\[
d_{j}= \pm\left(2\left[Y_{j} \ln \left(\frac{Y_{j}}{M_{j} p_{j}}\right)+\left(M_{j}-Y_{j}\right) \ln \left\{\frac{M_{j}-Y_{j}}{M_{j}\left(1-p_{j}\right)}\right\}\right]\right)^{1 / 2}
\]
where the sign is the same as the sign of \(\left(Y_{j}-M_{j} p_{j}\right)\). In the limiting cases, the deviance residual is given by
\[
d_{j}= \begin{cases}-\sqrt{2 M_{j}\left|\ln \left(1-p_{j}\right)\right|} & \text { if } Y_{j}=0 \\ \sqrt{2 M_{j}\left|\ln p_{j}\right|} & \text { if } Y_{j}=M_{j}\end{cases}
\]

The unadjusted diagonal elements of the hat matrix \(h_{\mathrm{U} j}\) are given by \(h_{\mathrm{U} j}=\left(\mathbf{X V X}^{\prime}\right)_{j j}\), where \(V\) is the estimated covariance matrix of parameters. The adjusted diagonal elements \(h_{j}\) created by hat are then \(h_{j}=M_{j} p_{j}\left(1-p_{j}\right) h_{\mathrm{U} j}\).

The standardized Pearson residual \(r_{\mathrm{S} j}\) is \(r_{j} / \sqrt{1-h_{j}}\).

The Pregibon (1981) \(\Delta \widehat{\beta}_{j}\) influence statistic is
\[
\Delta \widehat{\beta}_{j}=\frac{r_{j}^{2} h_{j}}{\left(1-h_{j}\right)^{2}}
\]

The corresponding change in the Pearson \(\chi^{2}\) is \(r_{\mathrm{S} j}^{2}\). The corresponding change in the deviance residual is \(\Delta D_{j}=d_{j}^{2} /\left(1-h_{j}\right)\).

\section*{References}

Collett, D. 2003. Modelling Binary Data. 2nd ed. London: Chapman and Hall/CRC.
Hosmer, D. W., Jr., S. A. Lemeshow, and R. X. Sturdivant. 2013. Applied Logistic Regression. 3rd ed. Hoboken, NJ: Wiley.

Kasza, J. 2015. Stata tip 125: Binned residual plots for assessing the fit of regression models for binary outcomes. Stata Journal 15: 599-604.

Mitchell, M. N., and X. Chen. 2005. Visualizing main effects and interactions for binary logit models. Stata Journal 5: 64-82.

Newson, R. B. 2013. Attributable and unattributable risks and fractions and other scenario comparisons. Stata Journal 13: 672-698.

Powers, D. A., H. Yoshioka, and M.-S. Yun. 2011. mvdcmp: Multivariate decomposition for nonlinear response models. Stata Journal 11: 556-576.

Pregibon, D. 1981. Logistic regression diagnostics. Annals of Statistics 9: 705-724. https://doi.org/10.1214/aos/1176345513.

Wang, Z. 2007. Two postestimation commands for assessing confounding effects in epidemiological studies. Stata Journal 7: 183-196.

\section*{Also see}
[R] logistic - Logistic regression, reporting odds ratios
[R] estat classification - Classification statistics and table
[R] estat gof - Pearson or Hosmer-Lemeshow goodness-of-fit test
\([\mathrm{R}]\) Iroc - Compute area under ROC curve and graph the curve
\([R]\) Isens - Graph sensitivity and specificity versus probability cutoff
[U] 20 Estimation and postestimation commands

\section*{Title}
logit — Logistic regression, reporting coefficients
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
logit fits a logit model for a binary response by maximum likelihood; it models the probability of a positive outcome given a set of regressors. depvar equal to nonzero and nonmissing (typically depvar equal to one) indicates a positive outcome, whereas depvar equal to zero indicates a negative outcome.

Also see [R] logistic; logistic displays estimates as odds ratios. Many users prefer the logistic command to logit. Results are the same regardless of which you use-both are the maximumlikelihood estimator. Several auxiliary commands that can be run after logit, probit, or logistic estimation are described in \([R]\) logistic postestimation.

\section*{Quick start}

Logit model of y on x 1 and x 2
logit y x1 x2
Add indicators for categorical variable a
logit y x1 x2 i.a
With cluster-robust standard errors for clustering by levels of cvar
logit y x1 x2 i.a, vce(cluster cvar)
Save separate coefficient estimates for each level of cvar to myresults.dta
statsby _b, by(cvar) saving(myresults): logit y x1 x2 i.a
Adjust for complex survey design using svyset data
svy: logit y x1 x2 i.a

\section*{Menu}

Statistics \(>\) Binary outcomes \(>\) Logistic regression

\section*{Syntax}
logit depvar [indepvars] [if] [in] [weight] [, options]
options
Model
noconstant
offset (varname)
asis
constraints (constraints)
SE/Robust
vce (vcetype)

Reporting
level(\#)
or
nocnsreport
display_options
suppress constant term
include varname in model with coefficient constrained to 1
retain perfect predictor variables
apply specified linear constraints
vcetype may be oim, opg, robust, cluster clustvar, bootstrap, or jackknife
set confidence level; default is level (95)
report odds ratios
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

Maximization
maximize_options
nocoef
collinear
coeflegend
Description
indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
depvar and indepvars may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bayes, bootstrap, by, collect, fmm, fp, jackknife, mfp, mi estimate, nestreg, rolling, statsby, stepwise, and svy are allowed; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: logit and [FMM] fmm: logit.
vce(bootstrap) and vce(jackknife) are not allowed with the mi estimate prefix; see [MI] mi estimate.
Weights are not allowed with the bootstrap prefix; see \([R]\) bootstrap.
vce(), nocoef, and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
nocoef, collinear, and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options}

SE/Robust
vce(vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
level(\#); see [R] Estimation options.
or reports the estimated coefficients transformed to odds ratios, that is, \(e^{b}\) rather than \(b\). Standard errors and confidence intervals are similarly transformed. This option affects how results are displayed, not how they are estimated. or may be specified at estimation or when replaying previously estimated results.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.
\(\qquad\) Maximization
maximize_options: difficult, technique (algorithm_spec), iterate(\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance (\#), nonrtolerance, and from (init-specs); see [R] Maximize. These options are seldom used.

The following options are available with logit but are not shown in the dialog box:
nocoef specifies that the coefficient table not be displayed. This option is sometimes used by program writers but is of no use interactively.
collinear, coeflegend; see [R] Estimation options.

\section*{Remarks and examples}

Remarks are presented under the following headings:
Basic usage
Model identification

\section*{Basic usage}
logit fits maximum likelihood models with dichotomous dependent (left-hand-side) variables coded as \(0 / 1\) (or, more precisely, coded as 0 and not- 0 ).

For grouped data or data in binomial form, a probit model can be fit using glm with the family(binomial) and link(logit) options.

\section*{\(>\) Example 1}

We have data on the make, weight, and mileage rating of 22 foreign and 52 domestic automobiles. We wish to fit a logit model explaining whether a car is foreign on the basis of its weight and mileage. Here is an overview of our data:
```

. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. keep make mpg weight foreign
. describe
Contains data from https://www.stata-press.com/data/r18/auto.dta
Observations: 74 1978 automobile data
Variables: 4 13 Apr 2022 17:45
(_dta has notes)

| Variable <br> name | Storage <br> type | Display <br> format | Value <br> label | Variable label |
| :--- | :---: | :---: | :--- | :--- |
| make | str18 | $\%-18 \mathrm{~s}$ |  | Make and model |
| mpg | int | $\% 8.0 \mathrm{~g}$ |  | Mileage (mpg) |
| weight | int | $\% 8.0 \mathrm{gc}$ |  | Weight (lbs.) <br> foreign |
| byte | $\% 8.0 \mathrm{~g}$ | origin | Car origin |  |

Sorted by: foreign
Note: Dataset has changed since last saved.
. inspect foreign
foreign: Car origin
Number of observations
M

|  | Total | Integers | Nonintegers |
| :--- | ---: | ---: | ---: |
| Negative | - | - | - |
| Zero | 52 | 52 | - |
| Positive | 22 | 22 | - |
|  | 74 | 74 | - |
| Total | - |  |  |
| Missing | 74 |  |  |

(2 unique values) foreign is labeled and all values are documented in the label.

```

The variable foreign takes on two unique values, 0 and 1 . The value 0 denotes a domestic car, and 1 denotes a foreign car.

The model that we wish to fit is
\[
\operatorname{Pr}(\text { foreign }=1)=F\left(\beta_{0}+\beta_{1} \text { weight }+\beta_{2} \mathrm{mpg}\right)
\]
where \(F(z)=e^{z} /\left(1+e^{z}\right)\) is the cumulative logistic distribution.

To fit this model, we type
```

. logit foreign weight mpg
Iteration 0: Log likelihood = -45.03321
Iteration 1: Log likelihood = -29.238536
Iteration 2: Log likelihood = -27.244139
Iteration 3: Log likelihood = -27.175277
Iteration 4: Log likelihood = -27.175156
Iteration 5: Log likelihood = -27.175156

```
Logistic regression \(\quad\) Number of obs \(=\quad 74\)
Log likelihood = -27.175156
    LR chi2(2) \(=35.72\)
    Prob > chi2 \(=0.0000\)
    Pseudo R2 \(=0.3966\)
\begin{tabular}{llr} 
Number of obs & \(=\) & 74 \\
LR chi2 (2) & \(=\) & 35.72 \\
Prob > chi2 & \(=0.0000\) \\
Pseudo R2 & \(=0.3966\)
\end{tabular}

Pseudo R2 \(=0.3966\)
\begin{tabular}{r|rrccrr}
\hline foreign & Coefficient & Std. err. & z & P>|z| & [95\% conf. interval] \\
\hline weight & -.0039067 & .0010116 & -3.86 & 0.000 & -.0058894 & -.001924 \\
mpg & -.1685869 & .0919175 & -1.83 & 0.067 & -.3487418 & .011568 \\
_cons & 13.70837 & 4.518709 & 3.03 & 0.002 & 4.851859 & 22.56487 \\
\hline
\end{tabular}

We find that heavier cars are less likely to be foreign and that cars yielding better gas mileage are also less likely to be foreign, at least holding the weight of the car constant.

\section*{- Technical note}

Stata interprets a value of 0 as a negative outcome (failure) and treats all other values (except missing) as positive outcomes (successes). Thus if your dependent variable takes on the values 0 and 1 , then 0 is interpreted as failure and 1 as success. If your dependent variable takes on the values 0 , 1 , and 2, then 0 is still interpreted as failure, but both 1 and 2 are treated as successes.

If you prefer a more formal mathematical statement, when you type logit \(y x\), Stata fits the model
\[
\operatorname{Pr}\left(y_{j} \neq 0 \mid \mathbf{x}_{j}\right)=\frac{\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}\right)}{1+\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}\right)}
\]

\section*{Model identification}

The logit command has one more feature, and it is probably the most useful. logit automatically checks the model for identification and, if it is underidentified, drops whatever variables and observations are necessary for estimation to proceed. (logistic, probit, and ivprobit do this as well.)

\section*{> Example 2}

Have you ever fit a logit model where one or more of your independent variables perfectly predicted one or the other outcome?

For instance, consider the following data:
\begin{tabular}{cc} 
Outcome \(y\) & Independent variable \(x\) \\
\hline 0 & 1 \\
0 & 1 \\
0 & 0 \\
1 & 0
\end{tabular}

Say that we wish to predict the outcome on the basis of the independent variable. The outcome is always zero whenever the independent variable is one. In our data, \(\operatorname{Pr}(y=0 \mid x=1)=1\), which means that the logit coefficient on \(x\) must be minus infinity with a corresponding infinite standard error. At this point, you may suspect that we have a problem.

Unfortunately, not all such problems are so easily detected, especially if you have a lot of independent variables in your model. If you have ever had such difficulties, you have experienced one of the more unpleasant aspects of computer optimization. The computer has no idea that it is trying to solve for an infinite coefficient as it begins its iterative process. All it knows is that at each step, making the coefficient a little bigger, or a little smaller, works wonders. It continues on its merry way until either 1) the whole thing comes crashing to the ground when a numerical overflow error occurs or 2) it reaches some predetermined cutoff that stops the process. In the meantime, you have been waiting. The estimates that you finally receive, if you receive any at all, may be nothing more than numerical roundoff.

Stata watches for these sorts of problems, alerts us, fixes them, and properly fits the model.
Let's return to our automobile data. Among the variables we have in the data is one called repair, which takes on three values. A value of 1 indicates that the car has a poor repair record, 2 indicates an average record, and 3 indicates a better-than-average record. Here is a tabulation of our data:
```

. use https://www.stata-press.com/data/r18/repair, clear
(1978 automobile data)
. tabulate foreign repair

| Car origin | Repair |  |  |  |
| :---: | :---: | :---: | :---: | :---: |
|  | 1 | 2 | 3 | Total |
| Domestic | 10 | 27 | 9 | 46 |
| Foreign | 0 | 3 | 9 | 12 |
| Total | 10 | 30 | 18 | 58 |

```

All the cars with poor repair records (repair \(=1\) ) are domestic. If we were to attempt to predict foreign on the basis of the repair records, the predicted probability for the repair \(=1\) category would have to be zero. This in turn means that the logit coefficient must be minus infinity, and that would set most computer programs buzzing.

Let's try Stata on this problem.
```

. logit foreign b3.repair
note: 1.repair != 0 predicts failure perfectly;
1.repair omitted and 10 obs not used.

```
Iteration 0: Log likelihood \(=-26.992087\)
Iteration 1: Log likelihood \(=-22.483187\)
Iteration 2: Log likelihood \(=-22.230498\)
Iteration 3: Log likelihood \(=-22.229139\)
Iteration 4: Log likelihood \(=-22.229138\)

Logistic regression
\begin{tabular}{lrr} 
Number of obs & \(=\) & 48 \\
LR chi2 (1) & \(=\) & 9.53 \\
Prob > chi2 & \(=0.0020\) \\
Pseudo R2 & \(=0.1765\)
\end{tabular}

Log likelihood \(=-22.229138\)
\begin{tabular}{r|rlllll}
\hline foreign & Coefficient & Std. err. & \(z\) & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline repair & 0 & (empty) & & & & \\
1 \\
2 & -2.197225 & .7698003 & -2.85 & 0.004 & -3.706005 & -.6884436 \\
_cons & \(7.94 e-17\) & .4714045 & 0.00 & 1.000 & -.9239359 & .9239359 \\
\hline
\end{tabular}

Remember that all the cars with poor repair records (repair \(=1\) ) are domestic, so the model cannot be fit, or at least it cannot be fit if we restrict ourselves to finite coefficients. Stata noted that fact "note: 1.repair !=0 predicts failure perfectly". This is Stata's mathematically precise way of saying what we said in English. When repair is 1, the car is domestic.

Stata then went on to say "1.repair omitted and 10 obs not used". This is Stata eliminating the problem. First 1.repair had to be removed from the model because it would have an infinite coefficient. Then, the 10 observations that led to the problem had to be eliminated, as well, so as not to bias the remaining coefficients in the model. The 10 observations that are not used are the 10 domestic cars that have poor repair records.

Stata then fit what was left of the model, using the remaining observations. Because no observations remained for cars with poor repair records, Stata reports "(empty)" in the row for repair \(=1\).

\section*{- Technical note}

Stata is pretty smart about catching problems like this. It will catch "one-way causation by a dummy variable", as we demonstrated above.

Stata also watches for "two-way causation", that is, a variable that perfectly determines the outcome, both successes and failures. Here Stata says, "so-and-so predicts outcome perfectly" and stops. Statistics dictates that no model can be fit.

Stata also checks your data for collinear variables; it will say, "so-and-so omitted because of collinearity". No observations need to be eliminated in this case, and model fitting will proceed without the offending variable.

It will also catch a subtle problem that can arise with continuous data. For instance, if we were estimating the chances of surviving the first year after an operation, and if we included in our model age, and if all the persons over 65 died within the year, Stata would say, "age \(>65\) predicts failure perfectly". It would then inform us about the fix-up it takes and fit what can be fit of our model.
logit (and logistic, probit, and ivprobit) will also occasionally display messages such as Note: 4 failures and 0 successes completely determined.

There are two causes for a message like this. The first—and most unlikely-case occurs when a continuous variable (or a combination of a continuous variable with other continuous or dummy variables) is simply a great predictor of the dependent variable. Consider Stata's auto.dta dataset with 6 observations removed.
```

. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. drop if foreign==0 \& gear_ratio > 3.1
(6 observations deleted)
. logit foreign mpg weight gear_ratio, nolog

```
Logistic regression \(\quad\) Number of obs \(=\quad 68\)
LR chi2(3) \(=72.64\)
Prob > chi2 \(=0.0000\)
Pseudo R2 \(=0.8484\)
\begin{tabular}{r|rcccrr}
\hline foreign & Coefficient & Std. err. & \multicolumn{1}{c}{z} & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline mpg & -.4944907 & .2655508 & -1.86 & 0.063 & -1.014961 & .0259792 \\
weight & -.0060919 & .003101 & -1.96 & 0.049 & -.0121698 & -.000014 \\
gear_ratio & 15.70509 & 8.166234 & 1.92 & 0.054 & -.300436 & 31.71061 \\
_cons & -21.39527 & 25.41486 & -0.84 & 0.400 & -71.20747 & 28.41694 \\
\hline
\end{tabular}

Note: 4 failures and 0 successes completely determined.
There are no missing standard errors in the output. If you receive the "completely determined" message and have one or more missing standard errors in your output, see the second case discussed below.

Note gear_ratio's large coefficient. logit thought that the 4 observations with the smallest predicted probabilities were essentially predicted perfectly.
```

. predict p
(option pr assumed; Pr(foreign))
. sort p
. list p in 1/4

```
\begin{tabular}{|r|}
\hline\(p\) \\
\hline \(1.34 \mathrm{e}-10\) \\
\(6.26 \mathrm{e}-09\) \\
\(7.84 \mathrm{e}-09\) \\
\(1.49 \mathrm{e}-08\) \\
\hline
\end{tabular}

If this happens to you, you do not have to do anything. Computationally, the model is sound. The second case discussed below requires careful examination.

The second case occurs when the independent terms are all dummy variables or continuous ones with repeated values (for example, age). Here one or more of the estimated coefficients will have missing standard errors. For example, consider this dataset consisting of 6 observations.
```

. use https://www.stata-press.com/data/r18/logitxmpl, clear

```
. list, separator(0)
\begin{tabular}{|ccc|}
\hline\(y\) & \(x 1\) & \(x 2\) \\
\hline 0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 1 & 0 \\
1 & 1 & 0 \\
0 & 0 & 1 \\
1 & 0 & 1 \\
\hline
\end{tabular}
```

. logit y x1 x2
Iteration 0: Log likelihood = -3.819085
Iteration 1: Log likelihood = -2.9527336
Iteration 2: Log likelihood = -2.8110282
Iteration 3: Log likelihood = -2.7811973
Iteration 4: Log likelihood = -2.7746107
Iteration 5: Log likelihood = -2.7730128
(output omitted)
Iteration 296: Log likelihood = -2.7725887 (not concave)
Iteration 297: Log likelihood = -2.7725887 (not concave)
Iteration 298: Log likelihood = -2.7725887 (not concave)
Iteration 299: Log likelihood = -2.7725887 (not concave)
Iteration 300: Log likelihood = -2.7725887 (not concave)
convergence not achieved

```
Logistic regression \(\quad\) Number of obs \(=\quad 6\)
LR chi2(1) \(=2.09\)
Prob > chi2 \(=0.1480\)
Pseudo R2 \(=0.2740\)
\begin{tabular}{r|rrrrrr}
\hline y & Coefficient & Std. err. & z & P>|z| & [95\% conf. interval] \\
\hline x 1 & 18.3704 & 2 & 9.19 & 0.000 & 14.45047 & 22.29033 \\
x 2 & 18.3704 &. &. &. &. &. \\
_cons & -18.3704 & 1.414214 & -12.99 & 0.000 & -21.14221 & -15.5986 \\
\hline
\end{tabular}

Note: 2 failures and 0 successes completely determined.
convergence not achieved
r(430);
Three things are happening here. First, logit iterates almost forever and then declares nonconvergence. Second, logit can fit the outcome \((\mathrm{y}=0)\) for the covariate pattern \(\mathrm{x} 1=0\) and \(\mathrm{x} 2=0\) (that is, the first two observations) perfectly. This observation is the " 2 failures and 0 successes completely determined". Third, if this observation is excluded, then x 1 , x 2 , and the constant are collinear.

This is the cause of the nonconvergence, the message "completely determined", and the missing standard errors. It happens when you have a covariate pattern (or patterns) with only one outcome and there is collinearity when the observations corresponding to this covariate pattern are excluded.

If this happens to you, confirm the causes. First, identify the covariate pattern with only one outcome. (For your data, replace x 1 and x 2 with the independent variables of your model.)
```

. egen pattern = group(x1 x2)
. quietly logit y x1 x2, iterate(100)
. predict p
(option pr assumed; Pr(y))
. summarize p

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| p | 6 | .3333333 | .2581989 | $1.05 \mathrm{e}-08$ | .5 |

```

If successes were completely determined, that means that there are predicted probabilities that are almost 1 . If failures were completely determined, that means that there are predicted probabilities that are almost 0 . The latter is the case here, so we locate the corresponding value of pattern:
. tabulate pattern if \(p<1 e-7\)
\begin{tabular}{r|rrr}
\begin{tabular}{r} 
group(x1 \\
x2)
\end{tabular} & Freq. & Percent & Cum. \\
\hline 1 & 2 & 100.00 & 100.00 \\
\hline Total & 2 & 100.00 &
\end{tabular}

Once we omit this covariate pattern from the estimation sample, logit can deal with the collinearity:
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline \multicolumn{5}{|l|}{Logistic regression} & \multicolumn{2}{|l|}{\multirow[t]{2}{*}{\begin{tabular}{llr} 
Number of obs & \(=\) & 4 \\
LR chi2 (1) & \(=0.00\) \\
Prob > chi2 & \(=1.0000\) \\
Pseudo R2 & \(=0.0000\)
\end{tabular}}} \\
\hline Log likelihood & \(=-2.7725887\) & & & & & \\
\hline y & Coefficient & Std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. & interval] \\
\hline x 1 & 0 & 2 & 0.00 & 1.000 & -3.919928 & 3.919928 \\
\hline x2 & & (omitted) & & & & \\
\hline _cons & 0 & 1.414214 & 0.00 & 1.000 & -2.771808 & 2.771808 \\
\hline
\end{tabular}

We omit the collinear variable. Then we must decide whether to include or omit the observations with pattern \(=1\). We could include them,
\begin{tabular}{llr}
. logit y x1, nolog & & \\
Logistic regression & Number of obs \(=\) & 6 \\
& LR chi2 \((1)\) & \(=0.37\) \\
& Prob \(>\) chi2 & \(=0.5447\) \\
Log likelihood \(=-3.6356349\) & Pseudo R2 & \(=0.0480\)
\end{tabular}
\begin{tabular}{r|rlccrc}
\hline y & Coefficient & Std. err. & z & P>|z| & [95\% conf. interval] \\
\hline x 1 & 1.098612 & 1.825742 & 0.60 & 0.547 & -2.479776 & 4.677001 \\
_cons & -1.098612 & 1.154701 & -0.95 & 0.341 & -3.361784 & 1.164559 \\
\hline
\end{tabular}
or exclude them,
\begin{tabular}{ll}
. logit y x1 if pattern \(!=1\), nolog & \\
Logistic regression & Number of obs \(=\) \\
& LR chi2 (1) \(=\) \\
& Prob \(>\) chi2 \(=1.0000\) \\
Log likelihood \(=-2.7725887\) & Pseudo R2
\end{tabular}
\begin{tabular}{r|rrrrrr}
\hline y & Coefficient & Std. err. & z & P>|z| & [95\% conf. interval] \\
\hline x 1 & 0 & 2 & 0.00 & 1.000 & -3.919928 & 3.919928 \\
_cons & 0 & 1.414214 & 0.00 & 1.000 & -2.771808 & 2.771808 \\
\hline
\end{tabular}

If the covariate pattern that predicts outcome perfectly is meaningful, you may want to exclude these observations from the model. Here you would report that covariate pattern such and such predicted outcome perfectly and that the best model for the rest of the data is .... But, more likely, the perfect prediction was simply the result of having too many predictors in the model. Then you would omit the extraneous variables from further consideration and report the best model for all the data.

\section*{Stored results}
logit stores the following in e():
Scalars
e(N)
e(N_cds)
e(N_cdf)
\(e(k)\)
e(k_eq)
e(k_eq_model)
e(k_dv)
e(df_m)
e(r2_p)
e(11)
e(ll_0)
e(N_clust)
e(chi2)
e(p)
e(rank)
e(ic)
e(rc)
\(e\) (converged)
Macros
```

e(cmd)
e(cmdline)
e(depvar)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(offset)
e(chi2type)
e(vce)
e(vcetype)
e(opt)
e(which)
e(ml_method)
e(user)

```
number of observations
number of completely determined successes
number of completely determined failures
number of parameters
number of equations in \(e(b)\)
number of equations in overall model test
number of dependent variables
model degrees of freedom
pseudo- \(R^{2}\)
log likelihood
log likelihood, constant-only model
number of clusters
\(\chi^{2}\)
\(p\)-value for model test
rank of e(V)
number of iterations
return code
1 if converged, 0 otherwise
logit
command as typed
name of dependent variable
weight type
weight expression
title in estimation output
name of cluster variable
linear offset variable
Wald or LR; type of model \(\chi^{2}\) test
vcetype specified in vce()
title used to label Std. err.
type of optimization
max or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program
```

e(technique) maximization technique
e(properties) b V
e(estat_cmd) program used to implement estat
e(predict) program used to implement predict
e(marginsok)
e(marginsnotok)
e(asbalanced)
e(asobserved)
Matrices
e(b)
e(Cns)
e(ilog)
e(gradient)
e(mns)
e(rules)
e(V)
e(V_modelbased)
Functions
e(sample) marks estimation sample

```

In addition to the above, the following is stored in r() :

Matrices \(r(\) table \()\)
matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

\section*{Methods and formulas}

Cramer (2003, chap. 9) surveys the prehistory and history of the logit model. The word "logit" was coined by Berkson (1944) and is analogous to the word "probit". For an introduction to probit and logit, see, for example, Aldrich and Nelson (1984), Cameron and Trivedi (2022), Jones (2007), Long (1997), Long and Freese (2014), Pampel (2021), or Powers and Xie (2008).

The likelihood function for logit is
\[
\ln L=\sum_{j \in S} w_{j} \ln F\left(\mathbf{x}_{j} \mathbf{b}\right)+\sum_{j \notin S} w_{j} \ln \left\{1-F\left(\mathbf{x}_{j} \mathbf{b}\right)\right\}
\]
where \(S\) is the set of all observations \(j\), such that \(y_{j} \neq 0, F(z)=e^{z} /\left(1+e^{z}\right)\), and \(w_{j}\) denotes the optional weights. \(\ln L\) is maximized as described in [R] Maximize.

This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce (robust) and vce (cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas. The scores are calculated as \(\mathbf{u}_{j}=\left\{1-F\left(\mathbf{x}_{j} \mathbf{b}\right)\right\} \mathbf{x}_{j}\) for the positive outcomes and \(-F\left(\mathbf{x}_{j} \mathbf{b}\right) \mathbf{x}_{j}\) for the negative outcomes.
logit also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

Joseph Berkson (1899-1982) was born in New York City and studied at the College of the City of New York, Columbia, and Johns Hopkins, earning both an MD and a doctorate in statistics. He then worked at Johns Hopkins before moving to the Mayo Clinic in 1931 as a biostatistician. Among many other contributions, his most influential one drew upon a long-sustained interest in the logistic function, especially his 1944 paper on bioassay, in which he introduced the term "logit". Berkson was a frequent participant in controversy-sometimes humorous, sometimes bitter-on subjects such as the evidence for links between smoking and various diseases and the relative merits of probit and logit methods and of different calculation methods.

\section*{References}

Aldrich, J. H., and F. D. Nelson. 1984. Linear Probability, Logit, and Probit Models. Newbury Park, CA: Sage.
Archer, K. J., and S. A. Lemeshow. 2006. Goodness-of-fit test for a logistic regression model fitted using survey sample data. Stata Journal 6: 97-105.

Berkson, J. 1944. Application of the logistic function to bio-assay. Journal of the American Statistical Association 39: 357-365. https://doi.org/10.2307/2280041.

Buis, M. L. 2010a. Direct and indirect effects in a logit model. Stata Journal 10: 11-29.
——. 2010b. Stata tip 87: Interpretation of interactions in nonlinear models. Stata Journal 10: 305-308.
Cameron, A. C., and P. K. Trivedi. 2022. Microeconometrics Using Stata. 2nd ed. College Station, TX: Stata Press. Corral, P., and M. Terbish. 2015. Generalized maximum entropy estimation of discrete choice models. Stata Journal 15: 512-522.

Cramer, J. S. 2003. Logit Models from Economics and Other Fields. Cambridge: Cambridge University Press.
Drukker, D. M. 2016. Probability differences and odds ratios measure conditional-on-covariate effects and population-parameter effects. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2016/07/26/probability-differences-and-odds-ratios-measure-conditional-on-covariate-effects-and-population-parameter-effects/.

Fernandez-Felix, B. M., E. García-Esquinas, A. Muriel, A. Royuela, and J. Zamora. 2021. Bootstrap internal validation command for predictive logistic regression models. Stata Journal 21: 498-509.

Hilbe, J. M. 2009. Logistic Regression Models. Boca Raton, FL: Chapman and Hall/CRC.
Hosmer, D. W., Jr., S. A. Lemeshow, and R. X. Sturdivant. 2013. Applied Logistic Regression. 3rd ed. Hoboken, NJ: Wiley.

Jones, A. M. 2007. Applied Econometrics for Health Economists: A Practical Guide. 2nd ed. Abingdon, UK: Radcliffe.
Judge, G. G., W. E. Griffiths, R. C. Hill, H. Lütkepohl, and T.-C. Lee. 1985. The Theory and Practice of Econometrics. 2nd ed. New York: Wiley.

Long, J. S. 1997. Regression Models for Categorical and Limited Dependent Variables. Thousand Oaks, CA: Sage.
Long, J. S., and J. Freese. 2014. Regression Models for Categorical Dependent Variables Using Stata. 3rd ed. College Station, TX: Stata Press.

Miranda, A., and S. Rabe-Hesketh. 2006. Maximum likelihood estimation of endogenous switching and sample selection models for binary, ordinal, and count variables. Stata Journal 6: 285-308.

Mitchell, M. N., and X. Chen. 2005. Visualizing main effects and interactions for binary logit models. Stata Journal 5: 64-82.

O'Fallon, W. M. 1998. Berkson, Joseph. In Vol. 1 of Encyclopedia of Biostatistics, ed. P. Armitage and T. Colton, 290-295. Chichester, UK: Wiley.

Orsini, N., R. Bellocco, and P. C. Sjölander. 2013. Doubly robust estimation in generalized linear models. Stata Journal 13: 185-205.

Pampel, F. C. 2021. Logistic Regression: A Primer. 2nd ed. Thousand Oaks, CA: Sage.
Pedace, R. 2013. Econometrics for Dummies. Hoboken, NJ: Wiley.

Pollock, P. H., III, and B. C. Edwards. 2019. A Stata Companion to Political Analysis. 4th ed. Thousand Oaks, CA: CQ Press.

Powers, D. A., and Y. Xie. 2008. Statistical Methods for Categorical Data Analysis. 2nd ed. Bingley, UK: Emerald.
Pregibon, D. 1981. Logistic regression diagnostics. Annals of Statistics 9: 705-724.
https://doi.org/10.1214/aos/1176345513.
Schonlau, M. 2005. Boosted regression (boosting): An introductory tutorial and a Stata plugin. Stata Journal 5: 330-354.
Uberti, L. J. 2022. Interpreting logit models. Stata Journal 22: 60-76.
Xu, J., and J. S. Long. 2005. Confidence intervals for predicted outcomes in regression models for categorical outcomes. Stata Journal 5: 537-559.

\section*{Also see}
[R] logit postestimation - Postestimation tools for logit
[R] brier - Brier score decomposition
[ R ] cloglog - Complementary log-log regression
\([R]\) exlogistic - Exact logistic regression
\([R]\) logistic - Logistic regression, reporting odds ratios
[R] npregress kernel - Nonparametric kernel regression
[R] npregress series - Nonparametric series regression
[R] probit - Probit regression
\([R]\) roc - Receiver operating characteristic (ROC) analysis
[R] ziologit - Zero-inflated ordered logit regression
[BAYES] bayes: logit - Bayesian logistic regression, reporting coefficients
[FMM] fmm: logit - Finite mixtures of logistic regression models
[LASSO] Lasso intro - Introduction to lasso
[ME] melogit — Multilevel mixed-effects logistic regression
[MI] Estimation - Estimation commands for use with mi estimate
[SVY] svy estimation - Estimation commands for survey data
[XT] xtlogit - Fixed-effects, random-effects, and population-averaged logit models
[U] 20 Estimation and postestimation commands

\section*{Title}
logit postestimation - Postestimation tools for logit

Postestimation commands
Remarks and examples
Also see
predict margins
Methods and formulas References

\section*{Postestimation commands}

The following postestimation commands are of special interest after logit:
Command Description
\begin{tabular}{ll} 
estat classification & report various summary statistics, including the classification table \\
estat gof & Pearson or Hosmer-Lemeshow goodness-of-fit test \\
lroc & compute area under ROC curve and graph the curve \\
lsens & graph sensitivity and specificity versus probability cutoff \\
lassogof & calculate goodness-of-fit predictions
\end{tabular}

These commands are not appropriate with svy estimation results.

The following standard postestimation commands are also available:
\begin{tabular}{|c|c|}
\hline Command & Description \\
\hline contrast & contrasts and ANOVA-style joint tests of estimates \\
\hline estat ic & Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) \\
\hline estat summarize & summary statistics for the estimation sample \\
\hline estat vce & variance-covariance matrix of the estimators (VCE) \\
\hline estat (svy) & postestimation statistics for survey data \\
\hline estimates & cataloging estimation results \\
\hline etable & table of estimation results \\
\hline * forecast & dynamic forecasts and simulations \\
\hline * hausman & Hausman's specification test \\
\hline lincom & point estimates, standard errors, testing, and inference for linear combinations of coefficients \\
\hline linktest & link test for model specification \\
\hline * lrtest & likelihood-ratio test \\
\hline margins & marginal means, predictive margins, marginal effects, and average marginal effects \\
\hline marginsplot & graph the results from margins (profile plots, interaction plots, etc.) \\
\hline nlcom & point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients \\
\hline predict & probabilities, influence statistics, residuals, etc. \\
\hline
\end{tabular}
predictnl
pwcompare
suest
test
testnl
point estimates, standard errors, testing, and inference for generalized predictions
pairwise comparisons of estimates
seemingly unrelated estimation
Wald tests of simple and composite linear hypotheses
Wald tests of nonlinear hypotheses
*forecast, hausman, and lrtest are not appropriate with svy estimation results. forecast is also not appropriate with mi estimation results.

\section*{predict}

\section*{Description for predict}
predict creates a new variable containing predictions such as probabilities, linear predictions, standard errors, influence statistics, deviance residuals, leverages, sequential numbers, Pearson residuals, and the equation-level score.

\section*{Menu for predict}

Statistics \(>\) Postestimation

\section*{Syntax for predict}
```

predict [type] newvar [if] [in] [, statistic nooffset rules asif}

```
statistic Description
Main
\(\mathrm{pr} \quad\) probability of a positive outcome; the default
\(\mathrm{xb} \quad\) linear prediction
stdp standard error of the prediction
* dbeta

Pregibon (1981) \(\Delta \widehat{\beta}\) influence statistic
* deviance
deviance residual
* dx2 Hosmer, Lemeshow, and Sturdivant (2013) \(\Delta \chi^{2}\) influence statistic
* \(\underline{\text { ddeviance } \quad H o s m e r, ~ L e m e s h o w, ~ a n d ~ S t u r d i v a n t ~(2013) ~} \Delta D\) influence statistic
*hat
*
Pregibon (1981) leverage
* residuals Pearson residuals; adjusted for number sharing covariate pattern
* rstandard standardized Pearson residuals; adjusted for number sharing covariate pattern score first derivative of the log likelihood with respect to \(\mathbf{x}_{j} \boldsymbol{\beta}\)

Unstarred statistics are available both in and out of sample; type predict ... if e (sample) ... if wanted only for the estimation sample. Starred statistics are calculated only for the estimation sample, even when if e(sample) is not specified.
\(\mathrm{pr}, \mathrm{xb}\), stdp, and score are the only options allowed with svy estimation results.

\section*{Options for predict}
\(\qquad\) Main
pr , the default, calculates the probability of a positive outcome.
xb calculates the linear prediction.
stdp calculates the standard error of the linear prediction.
dbeta calculates the Pregibon (1981) \(\Delta \widehat{\beta}\) influence statistic, a standardized measure of the difference in the coefficient vector that is due to deletion of the observation along with all others that share the same covariate pattern. In Hosmer, Lemeshow, and Sturdivant (2013, 154-155) jargon, this statistic is \(M\)-asymptotic; that is, it is adjusted for the number of observations that share the same covariate pattern.
deviance calculates the deviance residual.
dx2 calculates the Hosmer, Lemeshow, and Sturdivant \((2013,191) \Delta \chi^{2}\) influence statistic, reflecting the decrease in the Pearson \(\chi^{2}\) that is due to deletion of the observation and all others that share the same covariate pattern.
ddeviance calculates the Hosmer, Lemeshow, and Sturdivant (2013, 191) \(\Delta D\) influence statistic, which is the change in the deviance residual that is due to deletion of the observation and all others that share the same covariate pattern.
hat calculates the Pregibon (1981) leverage or the diagonal elements of the hat matrix adjusted for the number of observations that share the same covariate pattern.
number numbers the covariate patterns-observations with the same covariate pattern have the same number. Observations not used in estimation have number set to missing. The first covariate pattern is numbered 1 , the second 2 , and so on.
residuals calculates the Pearson residual as given by Hosmer, Lemeshow, and Sturdivant (2013, 155) and adjusted for the number of observations that share the same covariate pattern.
rstandard calculates the standardized Pearson residual as given by Hosmer, Lemeshow, and Sturdivant \((2013,191)\) and adjusted for the number of observations that share the same covariate pattern.
score calculates the equation-level score, \(\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)\).

Options
nooffset is relevant only if you specified offset (varname) for logit. It modifies the calculations made by predict so that they ignore the offset variable; the linear prediction is treated as \(\mathbf{x}_{j} \mathbf{b}\) rather than as \(\mathbf{x}_{j} \mathbf{b}+\) offset \(_{j}\).
rules requests that Stata use any rules that were used to identify the model when making the prediction. By default, Stata calculates missing for excluded observations.
asif requests that Stata ignore the rules and exclusion criteria and calculate predictions for all observations possible by using the estimated parameter from the model.

\section*{margins}

\section*{Description for margins}
margins estimates margins of response for probabilities and linear predictions.

\section*{Menu for margins}

Statistics \(>\) Postestimation

\section*{Syntax for margins}
```

margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...) ...] [options]

```
statistic Description
\begin{tabular}{ll}
pr & probability of a positive outcome; the default \\
xb & linear prediction \\
stdp & not allowed with margins \\
\(\underline{\text { dbeta }}\) & not allowed with margins \\
\(\underline{\text { deviance }}\) & not allowed with margins \\
\(\underline{\mathrm{dx}} 2\) & not allowed with margins \\
\(\underline{\text { ddeviance }}\) & not allowed with margins \\
\(\underline{\text { hat }}\) & not allowed with margins \\
\(\underline{\text { number }}\) & not allowed with margins \\
\(\underline{\text { residuals }}\) & not allowed with margins \\
\(\underline{\text { rstandard }}\) & not allowed with margins \\
\(\underline{\text { score }}\) & not allowed with margins
\end{tabular}

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see \([R]\) margins.

\section*{Remarks and examples}

Once you have fit a logit model, you can obtain the predicted probabilities by using the predict command for both the estimation sample and other samples; see [U] 20 Estimation and postestimation commands and [R] predict. Here we will make only a few more comments.
predict without arguments calculates the predicted probability of a positive outcome, that is, \(\operatorname{Pr}\left(y_{j}=1\right)=F\left(\mathbf{x}_{j} \mathbf{b}\right)\). With the xb option, predict calculates the linear combination \(\mathbf{x}_{j} \mathbf{b}\), where \(\mathbf{x}_{j}\) are the independent variables in the \(j\) th observation and \(\mathbf{b}\) is the estimated parameter vector. This is sometimes known as the index function because the cumulative distribution function indexed at this value is the probability of a positive outcome.

In both cases, Stata remembers any rules used to identify the model and calculates missing for excluded observations, unless rules or asif is specified. For information about the other statistics available after predict, see \([R]\) logistic postestimation.

\section*{> Example 1: Predicted probabilities}

In example 2 of [R] logit, we fit the logit model logit foreign b3.repair. To obtain predicted probabilities, type
```

. use https://www.stata-press.com/data/r18/repair
(1978 automobile data)
. logit foreign b3.repair
note: 1.repair != 0 predicts failure perfectly
1.repair omitted and 10 obs not used
(output omitted)
. predict p
(option pr assumed; Pr(foreign))
(10 missing values generated)

```
- summarize foreign p
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline foreign & 58 & .2068966 & .4086186 & 0 & 1 \\
\(p\) & 48 & .25 & .1956984 & .1 & .5
\end{tabular}

Stata remembers any rules used to identify the model and sets predictions to missing for any excluded observations. logit omitted the variable 1.repair from our model and excluded 10 observations. Thus when we typed predict p, those same 10 observations were again excluded, and their predictions were set to missing.
predict's rules option uses the rules in the prediction. During estimation, we were told "1.repair \(!=0\) predicts failure perfectly", so the rule is that when 1. repair is not zero, we should predict 0 probability of success or a positive outcome:
```

. predict p2, rules
(option pr assumed; Pr(foreign))
. summarize foreign p p2

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| foreign | 58 | .2068966 | .4086186 | 0 | 1 |
| $p$ | 48 | .25 | .1956984 | .1 | .5 |
| p2 | 58 | .2068966 | .2016268 | 0 | .5 |

```
predict's asif option ignores the rules and exclusion criteria and calculates predictions for all observations possible by using the estimated parameters from the model:
```

. predict p3, asif
(option pr assumed; Pr(foreign))
. summarize foreign p p2 p3

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| foreign | 58 | .2068966 | .4086186 | 0 | 1 |
| p | 48 | .25 | .1956984 | .1 | .5 |
| p2 | 58 | .2068966 | .2016268 | 0 | .5 |
| p3 | 58 | .2931035 | .2016268 | .1 | .5 |

```

Which is right? What predict does by default is the most conservative approach. If many observations had been excluded because of a simple rule, we could be reasonably certain that the rules prediction is correct. The asif prediction is correct only if the exclusion is a fluke, and we would be willing to exclude the variable from the analysis anyway. Then, however, we would refit the model to include the excluded observations.

\section*{> Example 2: Predictive margins}

We can use the command margins, contrast after logit to make comparisons on the probability scale. Let's fit a model predicting low birthweight from characteristics of the mother:
```

. use https://www.stata-press.com/data/r18/lbw, clear
(Hosmer \& Lemeshow data)
. logit low age i.race i.smoke ptl i.ht i.ui
Iteration 0: Log likelihood = -117.336
Iteration 1: Log likelihood = -103.81846
Iteration 2: Log likelihood = -103.40486
Iteration 3: Log likelihood = -103.40384
Iteration 4: Log likelihood = -103.40384
Logistic regression Number of obs = 189

```
LR chi2(7) \(=27.86\)
Prob > chi2 \(=0.0002\)
Pseudo R2 \(=0.1187\)
\begin{tabular}{c|rlrlrr}
\hline low & Coefficient & Std. err. & \multicolumn{1}{c}{\(\mathbf{z}\)} & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline age & -.0403293 & .0357127 & -1.13 & 0.259 & -.1103249 & .0296663 \\
race & & & & & & \\
Black & 1.009436 & .5025122 & 2.01 & 0.045 & .0245302 & 1.994342 \\
Other & 1.001908 & .4248342 & 2.36 & 0.018 & .1692485 & 1.834568 \\
smoke & & & & & & \\
Smoker & .9631876 & .3904357 & 2.47 & 0.014 & .1979477 & 1.728427 \\
ptl & .6288678 & .3399067 & 1.85 & 0.064 & -.0373371 & 1.295073 \\
1.ht & 1.358142 & .6289555 & 2.16 & 0.031 & .125412 & 2.590872 \\
1.ui & .8001832 & .4572306 & 1.75 & 0.080 & -.0959724 & 1.696339 \\
_cons & -1.184127 & .9187461 & -1.29 & 0.197 & -2.984837 & .6165818 \\
\hline
\end{tabular}

The coefficients are log odds-ratios: conditional on the other predictors, smoking during pregnancy is associated with an increase of 0.96 in the log odds of low birthweight. The model is linear in the \(\log\) odds-scale, so the estimate of 0.96 has the same interpretation, whatever the values of the other predictors might be. We could convert 0.96 to an odds ratio by replaying the results with logit, or.

But what if we want to talk about the probability of low birthweight, and not the odds? Then we will need the command margins, contrast. We will use the r. contrast operator to compare each level of smoke with a reference level. (smoke has only two levels, so there will be only one comparison: a comparison of smokers with nonsmokers.)
```

. margins r.smoke, contrast
Contrasts of predictive margins Number of obs = 189
Model VCE: OIM
Expression: Pr(low), predict()

```
\begin{tabular}{r|ccc}
\hline & df & chi2 & P>chi2 \\
\hline smoke & 1 & 6.32 & 0.0119 \\
\hline
\end{tabular}
\begin{tabular}{r|cccr}
\hline & \multicolumn{4}{|c}{\begin{tabular}{c} 
Delta-method \\
std. err.
\end{tabular}} \\
\hline Contrast & [95\% conf. interval] \\
\hline smoke & .1832779 & .0728814 & .0404329 & .3261229 \\
\hline
\end{tabular}

We see that maternal smoking is associated with an \(18.3 \%\) increase in the probability of low birthweight. (We received a contrast in the probability scale because predicted probabilities are the default when margins is used after logit.)

The contrast of \(18.3 \%\) is a difference of margins that are computed by averaging over the predictions for observations in the estimation sample. If the values of the other predictors were different, the contrast for smoke would be different, too. Let's estimate the contrast for 25 -year-old mothers:
```

. margins r.smoke, contrast at(age=25)
Contrasts of predictive margins Number of obs = 189
Model VCE: OIM
Expression: Pr(low), predict()
At: age = 25

```
\begin{tabular}{r|ccc}
\hline & df & chi2 & P>chi2 \\
\hline smoke & 1 & 6.19 & 0.0129 \\
\hline
\end{tabular}
\begin{tabular}{l|cccc}
\hline & \multicolumn{4}{|c}{\begin{tabular}{c} 
Delta-method \\
std. err.
\end{tabular}} \\
\hline Contrast \begin{tabular}{llll} 
smoke
\end{tabular} & .1808089 & .0726777 & .0383632 & .3232547 \\
\hline
\end{tabular}

Specifying a maternal age of 25 changed the contrast to \(18.1 \%\). Our contrast of probabilities changed because the logit model is nonlinear in the probability scale. A contrast of log odds would not have changed.

\section*{Methods and formulas}

See Methods and formulas in [R] logistic postestimation for details on predictions. For all other postestimation commands, see Methods and formulas in the entries for the corresponding commands.

\section*{References}

Hosmer, D. W., Jr., S. A. Lemeshow, and R. X. Sturdivant. 2013. Applied Logistic Regression. 3rd ed. Hoboken, NJ: Wiley.

Newson, R. B. 2013. Attributable and unattributable risks and fractions and other scenario comparisons. Stata Journal 13: 672-698.

Powers, D. A., H. Yoshioka, and M.-S. Yun. 2011. mvdcmp: Multivariate decomposition for nonlinear response models. Stata Journal 11: 556-576.
Pregibon, D. 1981. Logistic regression diagnostics. Annals of Statistics 9: 705-724. https://doi.org/10.1214/aos/1176345513.
Zlotnik, A., and V. Abraira. 2015. A general-purpose nomogram generator for predictive logistic regression models. Stata Journal 15: 537-546.

\section*{Also see}
[R] logit - Logistic regression, reporting coefficients
[R] estat classification - Classification statistics and table
[R] estat gof - Pearson or Hosmer-Lemeshow goodness-of-fit test
\([R]\) lroc - Compute area under ROC curve and graph the curve
\([R]\) Isens - Graph sensitivity and specificity versus probability cutoff
[LASSO] lassogof - Goodness of fit after lasso for prediction
[U] 20 Estimation and postestimation commands

\section*{Title}

Ioneway - Large one-way ANOVA, random effects, and reliability
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
Acknowledgment & References & Also see &
\end{tabular}

\section*{Description}
loneway fits one-way analysis-of-variance (ANOVA) models. loneway relaxes the restriction imposed by oneway that factors must have fewer than 376 levels. The command additionally reports the intraclass correlation, its standard error, and confidence interval; the estimated reliability of the group-averaged mean; the standard deviation of the group effect; and the standard deviation of the within-group effect.

\section*{Quick start}

One-way ANOVA model of \(y\) for factor a loneway y a

Report an exact \(95 \%\) confidence interval for the intraclass correlation
loneway y a, exact
Same as above, but report a \(90 \%\) confidence interval
loneway y a, exact level(90)

\section*{Menu}

Statistics \(>\) Linear models and related \(>\) ANOVA/MANOVA \(>\) Large one-way ANOVA

\section*{Syntax}
loneway response_var group_var [if] [in] [weight] [, options]
options
Main
mean
median
exact
level(\#)

Description
expected value of \(F\) distribution; default is 1
median of \(F\) distribution; default is 1
exact confidence intervals (groups must be equal with no weights)
set confidence level; default is level (95)
by and collect are allowed; see [U] 11.1.10 Prefix commands.
aweights are allowed; see [U] 11.1.6 weight.

\section*{Options}
\(\qquad\) Main
mean specifies that the expected value of the \(F_{k-1, N-k}\) distribution be used as the reference point \(F_{m}\) in the estimation of \(\rho\) instead of the default value of 1.
median specifies that the median of the \(F_{k-1, N-k}\) distribution be used as the reference point \(F_{m}\) in the estimation of \(\rho\) instead of the default value of 1 .
exact requests that exact confidence intervals be computed, as opposed to the default asymptotic confidence intervals. This option is allowed only if the groups are equal in size and weights are not used.
level(\#) specifies the confidence level, as a percentage, for confidence intervals of the coefficients. The default is level (95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals.

\section*{Remarks and examples}
loneway and oneway both fit ANOVA models, but loneway presents different ancillary statistics from oneway:
\begin{tabular}{lcc} 
Feature & oneway & loneway \\
\hline Fit one-way model & x & x \\
\(\quad\) on 375 or fewer levels & x & x \\
\(\quad\) on more than 375 levels & & x \\
Bartlett's test for equal variance & x & \\
Multiple-comparison tests & x & \\
Intraclass correlation and SE & & x \\
Intraclass correlation confidence interval & & x \\
Est. reliability of group-averaged mean & & x \\
Est. SD of group effect & x \\
Est. SD within group & x
\end{tabular}

Remarks are presented under the following headings:
```

The one-way ANOVA model
R
The random-effects ANOVA model
Intraclass correlation
Estimated reliability of the group-averaged score

```

\section*{The one-way ANOVA model}

\section*{> Example 1}
loneway's output looks like that of oneway, except that loneway presents more information at the end. Using our automobile dataset, we have created a (numeric) variable called manufacturer_grp identifying the manufacturer of each car, and within each manufacturer we have retained a maximum of four models, selecting those with the lowest mpg. We can compute the intraclass correlation of mpg for all manufacturers with at least four models as follows:


In addition to the standard one-way ANOVA output, loneway produces the \(R^{2}\), the estimated standard deviation of the group effect, the estimated standard deviation within group, the intragroup correlation, the estimated reliability of the group-averaged mean, and, for unweighted data, the asymptotic standard error and confidence interval for the intragroup correlation.

The \(R^{2}\) is, of course, simply the underlying \(R^{2}\) for a regression of response_var on the levels of group_var, or mpg on the various manufacturers here.

\section*{The random-effects ANOVA model}
loneway assumes that we observe a variable, \(y_{i j}\), measured for \(n_{i}\) elements within \(k\) groups or classes such that
\[
y_{i j}=\mu+\alpha_{i}+\epsilon_{i j}, \quad i=1,2, \ldots, k, \quad j=1,2, \ldots, n_{i}
\]
and \(\alpha_{i}\) and \(\epsilon_{i j}\) are independent zero-mean random variables with variance \(\sigma_{\alpha}^{2}\) and \(\sigma_{\epsilon}^{2}\), respectively. This is the random-effects ANOVA model, also known as the components-of-variance model, in which it is typically assumed that the \(y_{i j}\) are normally distributed.

The interpretation with respect to our example is that the observed value of our response variable, mpg , is created in two steps. First, the \(i\) th manufacturer is chosen, and a value, \(\alpha_{i}\), is determined-the typical mpg for that manufacturer less the overall \(\mathrm{mpg} \mu\). Then a deviation, \(\epsilon_{i j}\), is chosen for the \(j\) th model within this manufacturer. This is how much that particular automobile differs from the typical mpg value for models from this manufacturer.

For our sample of 36 car models, the estimated standard deviations are \(\sigma_{\alpha}=3.8\) and \(\sigma_{\epsilon}=4.6\). Thus, a little more than half of the variation in mpg between cars is attributable to the car model, with the rest attributable to differences between manufacturers. These standard deviations differ from those that would be produced by a (standard) fixed-effects regression in that the regression would require the sum within each manufacturer of the \(\epsilon_{i j}, \epsilon_{i}\). for the \(i\) th manufacturer, to be zero, whereas these estimates merely impose the constraint that the sum is expected to be zero.

\section*{Intraclass correlation}

There are various estimators of the intraclass correlation, such as the pairwise estimator, which is defined as the Pearson product-moment correlation computed over all possible pairs of observations that can be constructed within groups. For a discussion of various estimators, see Donner (1986). loneway computes what is termed the analysis of variance, or ANOVA, estimator. This intraclass correlation is the theoretical upper bound on the variation in response_var that is explainable by group_var, of which \(R^{2}\) is an overestimate because of the serendipity of fitting. This correlation is comparable to an \(R^{2}\) - you do not have to square it.

In our example, the intra-manu correlation, the correlation of mpg within manufacturer, is 0.40 . Because aweights were not used and the default correlation was computed (that is, the mean and median options were not specified), loneway also provided the asymptotic confidence interval and standard error of the intraclass correlation estimate.

\section*{Estimated reliability of the group-averaged score}

The estimated reliability of the group-averaged score or mean has an interpretation similar to that of the intragroup correlation; it is a comparable number if we average response_var by group_var, or mpg by manu in our example. It is the theoretical upper bound of a regression of manufactureraveraged mpg on characteristics of manufacturers. Why would we want to collapse our 36-observation dataset into a 9-observation dataset of manufacturer averages? Because the 36 observations might be a mirage. When General Motors builds cars, do they sometimes put a Pontiac label and sometimes a Chevrolet label on them, so that it appears in our data as if we have two cars when we really have only one, replicated? If that were the case, and if it were the case for many other manufacturers, then we would be forced to admit that we do not have data on 36 cars; we instead have data on nine manufacturer-averaged characteristics.

\section*{Stored results}
loneway stores the following in \(r()\) :

Scalars
r(N)
r(rho)
\(r\) (lb) lower bound of \(95 \%\) CI for rho
r(ub) upper bound of \(95 \%\) CI for rho
\(r\) (rho_t) estimated reliability
\(r\) (se) asymp. SE of intraclass correlation
\(r\) (sd_w) estimated SD within group
r(sd_b) estimated SD of group effect

\section*{Methods and formulas}

The mean squares in the loneway's ANOVA table are computed as
\[
\mathrm{MS}_{\alpha}=\sum_{i} w_{i \cdot}\left(\bar{y}_{i} .-\bar{y}_{. .}\right)^{2} /(k-1)
\]
and
\[
\mathrm{MS}_{\epsilon}=\sum_{i} \sum_{j} w_{i j}\left(y_{i j}-\bar{y}_{i} .\right)^{2} /(N-k)
\]
in which
\[
w_{i .}=\sum_{j} w_{i j} \quad w . .=\sum_{i} w_{i} . \quad \bar{y}_{i .}=\sum_{j} w_{i j} y_{i j} / w_{i .} \quad \text { and } \quad \bar{y} . .=\sum_{i} w_{i \cdot} \cdot \bar{y}_{i .} / w . .
\]

The corresponding expected values of these mean squares are
\[
E\left(\mathrm{MS}_{\alpha}\right)=\sigma_{\epsilon}^{2}+g \sigma_{\alpha}^{2} \quad \text { and } \quad E\left(\mathrm{MS}_{\epsilon}\right)=\sigma_{\epsilon}^{2}
\]
in which
\[
g=\frac{w_{. .}-\sum_{i} w_{i \cdot}^{2} / w_{. .}}{k-1}
\]

In the unweighted case, we get
\[
g=\frac{N-\sum_{i} n_{i}^{2} / N}{k-1}
\]

As expected, \(g=m\) for the case of no weights and equal group sizes in the data, that is, \(n_{i}=m\) for all \(i\). Replacing the expected values with the observed values and solving yields the ANOVA estimates of \(\sigma_{\alpha}^{2}\) and \(\sigma_{\epsilon}^{2}\). Substituting these into the definition of the intraclass correlation
\[
\rho=\frac{\sigma_{\alpha}^{2}}{\sigma_{\alpha}^{2}+\sigma_{\epsilon}^{2}}
\]
yields the ANOVA estimator of the intraclass correlation:
\[
\rho_{A}=\frac{F_{\mathrm{obs}}-1}{F_{\mathrm{obs}}-1+g}
\]
\(F_{\text {obs }}\) is the observed value of the \(F\) statistic from the ANOVA table. For no weights and equal \(n_{i}, \rho_{A}\) \(=\) roh, which is the intragroup correlation defined by Kish (1965). Two slightly different estimators are available through the mean and median options (Gleason 1997). If either of these options is specified, the estimate of \(\rho\) becomes
\[
\rho=\frac{F_{\mathrm{obs}}-F_{m}}{F_{\mathrm{obs}}+(g-1) F_{m}}
\]

For the mean option, \(F_{m}=E\left(F_{k-1, N-K}\right)=(N-k) /(N-k-2)\), that is, the expected value of the ANOVA table's \(F\) statistic. For the median option, \(F_{m}\) is simply the median of the \(F\) statistic. Setting \(F_{m}\) to 1 gives \(\rho_{A}\), so for large samples, these different point estimators are essentially the same. Also, because the intraclass correlation of the random-effects model is by definition nonnegative, for any of the three possible point estimators, \(\rho\) is truncated to zero if \(F_{\text {obs }}\) is less than \(F_{m}\).

For no weighting, interval estimators for \(\rho_{A}\) are computed. If the groups are equal sized (all \(n_{i}\) equal) and the exact option is specified, the following exact (assuming that the \(y_{i j}\) are normally distributed) \(100(1-\alpha) \%\) confidence interval is computed:
\[
\left\{\frac{F_{\mathrm{obs}}-F_{m} F_{u}}{F_{\mathrm{obs}}+(g-1) F_{m} F_{u}}, \frac{F_{\mathrm{obs}}-F_{m} F_{l}}{F_{\mathrm{obs}}+(g-1) F_{m} F_{l}}\right\}
\]
with \(F_{m}=1, F_{l}=F_{\alpha / 2, k-1, N-k}\), and \(F_{u}=F_{1-\alpha / 2, k-1, N-k}, F_{\cdot, k-1, N-k}\) being the cumulative distribution function for the \(F\) distribution with \(k-1\) and \(N-k\) degrees of freedom. If mean or median is specified, \(F_{m}\) is defined as above. If the groups are equal sized and exact is not specified, the following asymptotic \(100(1-\alpha) \%\) confidence interval for \(\rho_{A}\) is computed,
\[
\left[\rho_{A}-z_{\alpha / 2} \sqrt{V\left(\rho_{A}\right)}, \rho_{A}+z_{\alpha / 2} \sqrt{V\left(\rho_{A}\right)}\right]
\]
where \(z_{\alpha / 2}\) is the \(100(1-\alpha / 2)\) percentile of the standard normal distribution and \(\sqrt{V\left(\rho_{A}\right)}\) is the asymptotic standard error of \(\rho\) defined below. This confidence interval is also available for unequal groups. It is not applicable and, therefore, not computed for the estimates of \(\rho\) provided by the mean and median options. Again, because the intraclass coefficient is nonnegative, if the lower bound is negative for either confidence interval, it is truncated to zero. As might be expected, the coverage probability of a truncated interval is higher than its nominal value.

The asymptotic standard error of \(\rho_{A}\), assuming that the \(y_{i j}\) are normally distributed, is also computed when appropriate, namely, for unweighted data and when \(\rho_{A}\) is computed (neither the mean option nor the median option is specified):
\[
V\left(\rho_{A}\right)=\frac{2(1-\rho)^{2}}{g^{2}}(A+B+C)
\]
with
\[
\begin{gathered}
A=\frac{\{1+\rho(g-1)\}^{2}}{N-k} \\
B=\frac{(1-\rho)\{1+\rho(2 g-1)\}}{k-1} \\
C=\frac{\rho^{2}\left\{\sum n_{i}^{2}-2 N^{-1} \sum n_{i}^{3}+N^{-2}\left(\sum n_{i}^{2}\right)^{2}\right\}}{(k-1)^{2}}
\end{gathered}
\]
and \(\rho_{A}\) is substituted for \(\rho\) (Donner 1986).
The estimated reliability of the group-averaged score, known as the Spearman-Brown prediction formula in the psychometric literature (Winer, Brown, and Michels 1991, 1014), is
\[
\rho_{t}=\frac{t \rho}{1+(t-1) \rho}
\]
for group size \(t\). loneway computes \(\rho_{t}\) for \(t=g\).
The estimated standard deviation of the group effect is \(\sigma_{\alpha}=\sqrt{\left(\mathrm{MS}_{\alpha}-\mathrm{MS}_{\epsilon}\right) / g}\). This deviation comes from the assumption that an observation is derived by adding a group effect to a within-group effect.

The estimated standard deviation within group is the square root of the mean square due to error, or \(\sqrt{\mathrm{MS}_{\epsilon}}\).

\section*{Acknowledgment}

We thank John Gleason (retired) of Syracuse University for his contributions to improving loneway.

\section*{References}

Donner, A. 1986. A review of inference procedures for the intraclass correlation coefficient in the one-way random effects model. International Statistical Review 54: 67-82. https://doi.org/10.2307/1403259.

Gleason, J. R. 1997. sg65: Computing intraclass correlations and large ANOVAs. Stata Technical Bulletin 35: 25-31. Reprinted in Stata Technical Bulletin Reprints, vol. 6, pp. 167-176. College Station, TX: Stata Press.
Kish, L. 1965. Survey Sampling. New York: Wiley.
Marchenko, Y. V. 2006. Estimating variance components in Stata. Stata Journal 6: 1-21.
Winer, B. J., D. R. Brown, and K. M. Michels. 1991. Statistical Principles in Experimental Design. 3rd ed. New York: McGraw-Hill.

\section*{Also see}
[R] anova - Analysis of variance and covariance
\([\mathrm{R}]\) icc — Intraclass correlation coefficients
[R] oneway - One-way analysis of variance

\section*{Title}
lowess - Lowess smoothing

Description
Options
References

Quick start
Remarks and examples Also see

Menu
Methods and formulas

Syntax
Acknowledgment

\section*{Description}
lowess carries out a locally weighted regression of yvar on xvar, displays the graph, and optionally saves the smoothed variable.

Warning: lowess is computationally intensive and may therefore take a long time to run on a slow computer. Lowess calculations on 1,000 observations, for instance, require performing 1,000 regressions.

\section*{Quick start}

Locally weighted regression of y 1 on x
lowess y 1 x
Same as above, but with a bandwidth of 0.4
lowess y 1 x , bwidth(.4)
With running-mean smoothing
lowess y 1 x, mean
Without the tricube weighting function
lowess y 1 x , noweight
Generate a new variable v containing the smoothed values of y 1
lowess \(y 1 \mathrm{x}\), generate(v)
Adjust the mean of the smoothed values to equal the mean of the unsmoothed values
lowess \(y 1 \mathrm{x}\), adjust
Lowess smoothing of categorical variable y 2 on x in terms of the \(\log\) of the odds ratio
lowess y2 x, logit

\section*{Menu}

Statistics \(>\) Nonparametric analysis \(>\) Lowess smoothing

\section*{Syntax}
lowess yvar xvar [if] [in] [, options]
\begin{tabular}{|c|c|}
\hline options & Description \\
\hline \multicolumn{2}{|l|}{Main} \\
\hline mean & running-mean smooth; default is running-line least squares \\
\hline noweight & suppress weighted regressions; default is tricube weighting function \\
\hline bwidth(\#) & use \# for the bandwidth; default is bwidth (0.8) \\
\hline logit & transform dependent variable to logits \\
\hline adjust & adjust smoothed mean to equal mean of dependent variable \\
\hline nograph & suppress graph \\
\hline generate(newvar) & create newvar containing smoothed values of yvar \\
\hline \multicolumn{2}{|l|}{Plot} \\
\hline marker_options & change look of markers (color, size, etc.) \\
\hline marker_label_options & add marker labels; change look or position \\
\hline \multicolumn{2}{|l|}{Smoothed line} \\
\hline lineopts(cline_options) & affect rendition of the smoothed line \\
\hline \multicolumn{2}{|l|}{Add plots} \\
\hline addplot (plot) & add other plots to generated graph \\
\hline \multicolumn{2}{|l|}{Y axis, X axis, Titles, Legend, Overall, By} \\
\hline twoway_options & any of the options documented in [G-3] twoway_options \\
\hline
\end{tabular}
yvar and xvar may contain time-series operators; see [U] 11.4.4 Time-series varlists.

\section*{Options}

\footnotetext{
Main
mean specifies running-mean smoothing; the default is running-line least-squares smoothing.
noweight prevents the use of Cleveland's (1979) tricube weighting function; the default is to use the weighting function.
bwidth (\#) specifies the bandwidth. Centered subsets of bwidth() \(\times N\) observations are used for calculating smoothed values for each point in the data except for the end points, where smaller, uncentered subsets are used. The greater the bwidth(), the greater the smoothing. The default is 0.8.
logit transforms the smoothed yvar into logits. Predicted values less than 0.0001 or greater than 0.9999 are set to \(1 / N\) and \(1-1 / N\), respectively, before taking logits.
adjust adjusts the mean of the smoothed yvar to equal the mean of yvar by multiplying by an appropriate factor. This option is useful when smoothing binary ( \(0 / 1\) ) data.
nograph suppresses displaying the graph.
generate(newvar) creates newvar containing the smoothed values of yvar.
}
marker_options affect the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G-3] marker_options.
marker_label_options specify if and how the markers are to be labeled; see [G-3] marker_label_options.
\(\qquad\)
lineopts(cline_options) affects the rendition of the lowess-smoothed line; see [G-3] cline_options.

Add plots
addplot (plot) provides a way to add other plots to the generated graph; see [G-3] addplot_option.
\(\qquad\) Y axis, X axis, Titles, Legend, Overall, By
twoway_options are any of the options documented in [G-3] twoway_options. These include options for titling the graph (see [G-3] title_options), options for saving the graph to disk (see [G-3] saving_option), and the by () option (see [G-3] by_option).

\section*{Remarks and examples}

By default, lowess provides locally weighted scatterplot smoothing. The basic idea is to create a new variable (newvar) that, for each yvar \(y_{i}\), contains the corresponding smoothed value. The smoothed values are obtained by running a regression of \(y\) var on \(x\) var by using only the data \(\left(x_{i}, y_{i}\right)\) and a few of the data near this point. In lowess, the regression is weighted so that the central point \(\left(x_{i}, y_{i}\right)\) gets the highest weight and points that are farther away (based on the distance \(\left.\left|x_{j}-x_{i}\right|\right)\) receive less weight. The estimated regression line is then used to predict the smoothed value \(\widehat{y}_{i}\) for \(y_{i}\) only. The procedure is repeated to obtain the remaining smoothed values, which means that a separate weighted regression is performed for every point in the data.

Lowess is a desirable smoother because of its locality -it tends to follow the data. Polynomial smoothing methods, for instance, are global in that what happens on the extreme left of a scatterplot can affect the fitted values on the extreme right.

\section*{\(>\) Example 1}

The amount of smoothing is affected by bwidth(\#). You are warned to experiment with different values. For instance,
```

. use https://www.stata-press.com/data/r18/lowess1
(Example data for lowess)
. lowess h1 depth

```

Lowess smoother


Now compare that with
```

. lowess h1 depth, bwidth(.4)

```


In the first case, the default bandwidth of 0.8 is used, meaning that \(80 \%\) of the data are used in smoothing each point. In the second case, we explicitly specified a bandwidth of 0.4 . Smaller bandwidths follow the original data more closely.

\section*{\(>\) Example 2}

Two lowess options are especially useful with binary ( \(0 / 1\) ) data: adjust and logit. adjust adjusts the resulting curve (by multiplication) so that the mean of the smoothed values is equal to the mean of the unsmoothed values. logit specifies that the smoothed curve be in terms of the log of the odds ratio:
```

. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)

```
. lowess foreign mpg, ylabel(0 "Domestic" 1 "Foreign") jitter(5) adjust

. lowess foreign mpg, logit yline(0)

\section*{Lowess smoother}

Logit transformed smooth


With binary data, if you do not use the logit option, it is a good idea to specify graph's jitter() option; see [G-2] graph twoway scatter. Because the underlying data (whether the car was manufactured outside the United States here) take on only two values, raw data points are more likely to be on top of each other, thus making it impossible to tell how many points there are. graph's jitter () option adds some noise to the data to shift the points around. This noise affects only the location of points on the graph, not the lowess curve.

When you specify the logit option, the display of the raw data is suppressed.
- Technical note
lowess can be used for more than just lowess smoothing. Lowess can be usefully thought of as a combination of two smoothing concepts: the use of predicted values from regression (rather than means) for imputing a smoothed value and the use of the tricube weighting function (rather than a constant weighting function). lowess allows you to combine these concepts freely. You can use line smoothing without weighting (specify noweight), mean smoothing with tricube weighting (specify mean), or mean smoothing without weighting (specify mean and noweight).

\section*{Methods and formulas}

Let \(y_{i}\) and \(x_{i}\) be the two variables, and assume that the data are ordered so that \(x_{i} \leq x_{i+1}\) for \(i=1, \ldots, N-1\). For each \(y_{i}\), a smoothed value \(y_{i}^{s}\) is calculated.

The subset used in calculating \(y_{i}^{s}\) is indices \(i_{-}=\max (1, i-k)\) through \(i_{+}=\min (i+k, N)\), where \(k=\lfloor(N \times\) bwidth -0.5\() / 2\rfloor\). The weights for each of the observations between \(j=i_{-}, \ldots, i_{+}\) are either 1 (noweight) or the tricube (default),
\[
w_{j}=\left\{1-\left(\frac{\left|x_{j}-x_{i}\right|}{\Delta}\right)^{3}\right\}^{3}
\]
where \(\Delta=1.0001 \max \left(x_{i_{+}}-x_{i}, x_{i}-x_{i_{-}}\right)\). The smoothed value \(y_{i}^{s}\) is then the (weighted) mean or the (weighted) regression prediction at \(x_{i}\).

William Swain Cleveland (1943- ) studied mathematics and statistics at Princeton and Yale. He worked for several years at Bell Labs in New Jersey and now teaches statistics and computer science at Purdue. He has made key contributions in many areas of statistics, including graphics and data visualization, time series, environmental applications, and analysis of Internet traffic data.

\section*{Acknowledgment}
lowess is a modified version of a command originally written by Patrick Royston of the MRC Clinical Trials Unit, London, and coauthor of the Stata Press book Flexible Parametric Survival Analysis Using Stata: Beyond the Cox Model.

\section*{References}

Chambers, J. M., W. S. Cleveland, B. Kleiner, and P. A. Tukey. 1983. Graphical Methods for Data Analysis. Belmont, CA: Wadsworth.
Cleveland, W. S. 1979. Robust locally weighted regression and smoothing scatterplots. Journal of the American Statistical Association 74: 829-836. https://doi.org/10.2307/2286407.
——. 1993. Visualizing Data. Summit, NJ: Hobart.
_-. 1994. The Elements of Graphing Data. Rev. ed. Summit, NJ: Hobart.
Cox, N. J. 2005. Speaking Stata: Smoothing in various directions. Stata Journal 5: 574-593.

Goodall, C. 1990. A survey of smoothing techniques. In Modern Methods of Data Analysis, ed. J. Fox and J. S. Long, 126-176. Newbury Park, CA: Sage.

Lindsey, C., and S. J. Sheather. 2010. Model fit assessment via marginal model plots. Stata Journal 10: 215-225.
Royston, P., and N. J. Cox. 2005. A multivariable scatterplot smoother. Stata Journal 5: 405-412.

\section*{Also see}
[R] lpoly - Kernel-weighted local polynomial smoothing
[R] smooth - Robust nonlinear smoother
[D] ipolate - Linearly interpolate (extrapolate) values

\section*{Title}

Ipoly - Kernel-weighted local polynomial smoothing
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
lpoly performs a kernel-weighted local polynomial regression of yvar on xvar and displays a graph of the smoothed values with (optional) confidence bands.

\section*{Quick start}

Kernel-weighted local polynomial regression of \(y\) on \(x\)
lpoly y x
Same as above, but specify a bandwidth of 2
lpoly y x, bwidth(2)
Same as above, but specify a degree of 1
lpoly y x, bwidth(2) degree(1)
Same as above, but use the alternative Epanechnikov kernel
lpoly y x, bwidth(2) degree(1) kernel(epan2)
Same as above, but create a new variable for the smoothing grid \(g\) and smoothed values \(s\)
lpoly y \(x\), bwidth(2) degree(1) kernel(epan2) generate (g s)
With 95\% confidence bands
lpoly y x, ci
Use twoway to graph multiple local polynomial fits
```

twoway scatter y x || ///
lpoly y x, degree(1) kernel(epan2) || ///
lpoly y x, degree(1) kernel(epan2) bwidth(1) || ///
lpoly y x, degree(1) kernel(epan2) bwidth(7) ||

```

\section*{Menu}

Statistics \(>\) Nonparametric analysis \(>\) Local polynomial smoothing

\section*{Syntax}
lpoly yvar xvar [if] [in] [weight] [, options]
options
Description
Main
kernel (kernel)
bwidth (\#|varname)
degree(\#)
generate ([ newvar \(\left._{x}\right]_{\text {newvar }}^{s}\) )
n(\#)
at (varname)
nograph
noscatter
SE/CI
ci
level(\#)
se(newvar)
pwidth(\#)
var(\#|varname)
Scatterplot
marker_options
marker_label_options
Smoothed line
lineopts(cline_options)
CI plot
ciopts(cline_options)
Add plots
addplot (plot)
Y axis, X axis, Titles, Legend, Overall
twoway_options
specify kernel function; default is kernel (epanechnikov) specify kernel bandwidth
specify degree of the polynomial smooth; default is degree (0)
store smoothing grid in newvar \(r_{x}\) and smoothed points in newvars
obtain the smooth at \# points; default is \(\min (N, 50)\)
obtain the smooth at the values specified by varname suppress graph
suppress scatterplot only
plot confidence bands
set confidence level; default is level (95)
store standard errors in newvar
specify pilot bandwidth for standard error calculation
specify estimates of residual variance
change look of markers (color, size, etc.)
add marker labels; change look or position
affect rendition of the smoothed line
affect rendition of the confidence bands
add other plots to the generated graph
any options other than by () documented in [G-3] twoway_options
kernel
Description
epanechnikov
epan2
biweight
cosine
gaussian
parzen
rectangle
triangle

Epanechnikov kernel function; the default alternative Epanechnikov kernel function biweight kernel function cosine trace kernel function Gaussian kernel function Parzen kernel function rectangle kernel function triangle kernel function
collect is allowed; see [U] 11.1.10 Prefix commands.
fweights and aweights are allowed; see [U] 11.1.6 weight.

\section*{Options}
\(\qquad\) Main
kernel (kernel) specifies the kernel function for use in calculating the weighted local polynomial estimate. The default is kernel (epanechnikov).
bwidth (\#|varname) specifies the half-width of the kernel-the width of the smoothing window around each point. If bwidth() is not specified, a rule-of-thumb (ROT) bandwidth estimator is calculated and used. A local variable bandwidth may be specified in varname, in conjunction with an explicit smoothing grid using the at () option.
degree (\#) specifies the degree of the polynomial to be used in the smoothing. The default is degree ( 0 ), meaning local-mean smoothing.
generate ( \(\left[\right.\) newvar \(\left._{x}\right]\) newvar \(_{s}\) ) stores the smoothing grid in newvar \({ }_{x}\) and the smoothed values in newvar \({ }_{s}\). If at () is not specified, then both newvar \({ }_{x}\) and newvar \({ }_{s}\) must be specified. Otherwise, only newvar \({ }_{s}\) is to be specified.
\(\mathrm{n}(\#)\) specifies the number of points at which the smooth is to be calculated. The default is \(\min (N, 50)\), where \(N\) is the number of observations.
at (varname) specifies a variable that contains the values at which the smooth should be calculated. By default, the smoothing is done on an equally spaced grid, but you can use at () to instead perform the smoothing at the observed \(x\) 's, for example. This option also allows you to more easily obtain smooths for different variables or different subsamples of a variable and then overlay the estimates for comparison.
nograph suppresses drawing the graph of the estimated smooth. This option is often used with the generate() option.
noscatter suppresses superimposing a scatterplot of the observed data over the smooth. This option is useful when the number of resulting points would be so large as to clutter the graph.

\section*{SE/CI}
ci plots confidence bands, using the confidence level specified in level().
level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals.
se(newvar) stores the estimates of the standard errors in newvar. This option requires specifying generate() or at().
pwidth(\#) specifies the pilot bandwidth to be used for standard error computations. The default is chosen to be 1.5 times the value of the ROT bandwidth selector. If you specify pwidth() without specifying se() or ci, then the ci option is assumed.
\(\operatorname{var}(\# \mid\) varname \()\) specifies an estimate of a constant residual variance or a variable containing estimates of the residual variances at each grid point required for standard error computation. By default, the residual variance at each smoothing point is estimated by the normalized weighted residual sum of squares obtained from locally fitting a polynomial of order \(p+2\), where \(p\) is the degree specified in degree(). var (varname) is allowed only if at () is specified. If you specify var () without specifying se() or ci, then the ci option is assumed.

Scatterplot
marker_options affect the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G-3] marker_options.
marker_label_options specify if and how the markers are to be labeled; see [G-3] marker_label_options.

Smoothed line
lineopts (cline_options) affects the rendition of the smoothed line; see [G-3] cline_options.
\(\qquad\) \(\sqrt{\mathrm{Cl} \text { plot }}\)
ciopts (cline_options) affects the rendition of the confidence bands; see [G-3] cline_options.

Add plots
addplot (plot) provides a way to add other plots to the generated graph; see [G-3] addplot_option.

Y axis, X axis, Titles, Legend, Overall
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

\section*{Remarks and examples}

Remarks are presented under the following headings:
Introduction
Local polynomial smoothing
Choice of a bandwidth
Confidence bands

\section*{Introduction}

The last 25 years or so has seen a significant outgrowth in the literature on scatterplot smoothing, otherwise known as univariate nonparametric regression. Of most appeal is the idea of making no assumptions about the functional form for the expected value of a response given a regressor, but instead allowing the data to "speak for themselves". Various methods and estimators fall into the category of nonparametric regression, including local mean smoothing as described independently by Nadaraya (1964) and Watson (1964), the Gasser and Müller (1979) estimator, locally weighted scatterplot smoothing (LOWESS) as described by Cleveland (1979), wavelets (for example, Donoho
[1995]), and splines (Eubank 1999), to name a few. Much of the vast literature focuses on automating the amount of smoothing to be performed and dealing with the bias/variance tradeoff inherent to this type of estimation. For example, for Nadaraya-Watson the amount of smoothing is controlled by choosing a bandwidth.

Smoothing via local polynomials is by no means a new idea but instead one that has been rediscovered in recent years in articles such as Fan (1992). A natural extension of the local mean smoothing of Nadaraya-Watson, local polynomial regression involves fitting the response to a polynomial form of the regressor via locally weighted least squares. Higher-order polynomials have better bias properties than the zero-degree local polynomials of the Nadaraya-Watson estimator; in general, higher-order polynomials do not require bias adjustment at the boundary of the regression space. For a definitive reference on local polynomial smoothing, see Fan and Gijbels (1996).

\section*{Local polynomial smoothing}

Consider a set of scatterplot data \(\left\{\left(x_{1}, y_{1}\right), \ldots,\left(x_{n}, y_{n}\right)\right\}\) from the model
\[
\begin{equation*}
y_{i}=m\left(x_{i}\right)+\sigma\left(x_{i}\right) \epsilon_{i} \tag{1}
\end{equation*}
\]
for some unknown mean and variance functions \(m(\cdot)\) and \(\sigma^{2}(\cdot)\), and symmetric errors \(\epsilon_{i}\) with \(E\left(\epsilon_{i}\right)=0\) and \(\operatorname{Var}\left(\epsilon_{i}\right)=1\). The goal is to estimate \(m\left(x_{0}\right)=E\left[Y \mid X=x_{0}\right]\), making no assumption about the functional form of \(m(\cdot)\).
lpoly estimates \(m\left(x_{0}\right)\) as the constant term (intercept) of a regression, weighted by the kernel function specified in kernel (), of yvar on the polynomial terms (xvar \(\left.-x_{0}\right),\left(x v a r-x_{0}\right)^{2}, \ldots,(x v a r-\) \(\left.x_{0}\right)^{p}\) for each smoothing point \(x_{0}\). The degree of the polynomial, \(p\), is specified in degree(), the amount of smoothing is controlled by the bandwidth specified in bwidth(), and the chosen kernel function is specified in kernel().

\section*{> Example 1}

Consider the motorcycle data as examined (among other places) in Fan and Gijbels (1996). The data consist of 133 observations and measure the acceleration (accel measured in grams [g]) of a dummy's head during impact over time (time measured in milliseconds). For these data, we use lpoly to fit a local cubic polynomial with the default bandwidth (obtained using the ROT method) and the default Epanechnikov kernel.
```

. use https://www.stata-press.com/data/r18/motorcycle
(Motorcycle data from Fan \& Gijbels (1996))
. lpoly accel time, degree(3)

```

- Technical note
lpoly allows specifying in degree() both odd and even orders of the polynomial to be used for the smoothing. However, the odd-order, \(2 k+1\), polynomial approximations are preferable. They have an extra parameter compared with the even-order, \(2 k\), approximations, which leads to a significant bias reduction and there is no increase of variability associated with adding this extra parameter. Using an odd order when estimating the regression function is therefore usually sufficient. For a more thorough discussion, see Fan and Gijbels (1996).

\section*{Choice of a bandwidth}

The choice of a bandwidth is crucial for many smoothing techniques, including local polynomial smoothing. In general, using a large bandwidth gives smooths with a large bias, whereas a small bandwidth may result in highly variable smoothed values. Various techniques exist for optimal bandwidth selection. By default, lpoly uses the ROT method to estimate the bandwidth used for the smoothing; see Methods and formulas for details.

\section*{> Example 2}

Using the motorcycle data, we demonstrate how a local linear polynomial fit changes using different bandwidths.
```

. lpoly accel time, degree(1) kernel(epan2) bwidth(1)
> generate(at smooth1) nograph
. lpoly accel time, degree(1) kernel(epan2) bwidth(7) at(at)
> generate(smooth2) nograph
. label variable smooth1 "Smooth: width = 1"
. label variable smooth2 "Smooth: width = 7"
. lpoly accel time, degree(1) kernel(epan2) at(at) addplot(line smooth* at)
> legend(label(2 "Smooth: width = 3.42 (ROT)") cols(2) pos(6))
> note("kernel = epan2, degree = 1")

```


From this graph, we can see that the local linear polynomial fit with larger bandwidth (width \(=\) 7) corresponds to a smoother line but fails to fit the curvature of the scatterplot data. The smooth obtained using the width equal to one seems to fit most data points, but the corresponding line has several spikes indicating larger variability. The smooth obtained using the ROT bandwidth estimator seems to have a good tradeoff between the fit and variability in this example.

In the above, we also demonstrated how the generate() and addplot() options may be used to produce overlaid plots obtained from lpoly with different options. The nograph option saves time when you need to save only results with generate().

However, to avoid generating variables manually, one can use twoway lpoly instead; see [G-2] graph twoway lpoly for more details.
```

. twoway scatter accel time ||
lpoly accel time, degree(1) kernel(epan2) lpattern(solid) ||
lpoly accel time, degree(1) kernel(epan2) bwidth(1) ||
lpoly accel time, degree(1) kernel(epan2) bwidth(7) ||
legend(label(2 "Smooth: width = 3.42 (ROT)")
label(3 "Smooth: width = 1")
label(4 "Smooth: width = 7") cols(2) pos(6))
title("Local polynomial smooth") note("kernel = epan2, degree = 1")
xtitle("Time (msec)") ytitle("Acceleration (g)")

```

Local polynomial smooth

kernel \(=\) epan2, degree \(=1\)

The ROT estimate is commonly used as an initial guess for the amount of smoothing; this approach may be sufficient when the choice of a bandwidth is less important. In other cases, you can pick your own bandwidth.

When the shape of the regression function has a combination of peaked and flat regions, a variable bandwidth may be preferable over the constant bandwidth to allow for different degrees of smoothness in different regions. The bwidth() option allows you to specify the values of the local variable bandwidths as those stored in a variable in your data.

Similar issues with bias and variability arise when choosing a pilot bandwidth (the pwidth() option) used to compute standard errors of the local polynomial smoother. The default value is chosen to be \(1.5 \times\) ROT. For a review of methods for pilot bandwidth selection, see Fan and Gijbels (1996).

\section*{Confidence bands}

The established asymptotic normality of the local polynomial estimators under certain conditions allows the construction of approximate confidence bands. lpoly offers the ci option to plot these bands.

\section*{> Example 3}

Let us plot the confidence bands for the local polynomial fit from example 1.
```

. lpoly accel time, degree(3) kernel(epan2) ci legend(cols(2) pos(6))

```

Local polynomial smooth


You can obtain graphs with overlaid confidence bands by using twoway lpolyci; see [G-2] graph twoway lpolyci for examples.

Constructing the confidence intervals involves computing standard errors obtained by taking a square root of the estimate of the conditional variance of the local polynomial estimator at each grid point \(x_{0}\). Estimating the conditional variance requires fitting a polynomial of a higher order locally by using a different bandwidth, the pilot bandwidth. The value of the pilot bandwidth may be supplied by using pwidth(). By default, the value of \(1.5 \times\) ROT is used. Also, estimates of the residual variance \(\sigma^{2}\left(x_{0}\right)\) at each grid point, \(x_{0}\), are required to obtain the estimates of the conditional variances. These estimates may be supplied by using the var () option. By default, they are computed using the normalized weighted residual sum of squares from a local polynomial fit of a higher order. See Methods and formulas for details. The standard errors may be saved by using se().

\section*{Stored results}
lpoly stores the following in \(r()\) :
\begin{tabular}{llll} 
Scalars & & \\
\(r(\) degree \()\) & smoothing polynomial degree & \(r(\) bwidth \()\) & \begin{tabular}{l} 
bandwidth of the smooth \\
\(r(\) ngrid \()\)
\end{tabular} \\
\(r(N)\) & number of successful regressions & \(r(\) pwidth \()\) & pilot bandwidth \\
Mample size & & \\
\(r(\) sernel \()\) & name of kernel &
\end{tabular}

\section*{Methods and formulas}

Consider model (1), written in matrix notation,
\[
\mathbf{y}=m(\mathbf{x})+\boldsymbol{\epsilon}
\]
where \(\mathbf{y}\) and \(\mathbf{x}\) are the \(n \times 1\) vectors of scatterplot values, \(\boldsymbol{\epsilon}\) is the \(n \times 1\) vector of errors with zero mean and covariance matrix \(\boldsymbol{\Sigma}=\operatorname{diag}\left\{\sigma\left(x_{i}\right)\right\} \mathbf{I}_{n}\), and \(m()\) and \(\sigma()\) are some unknown functions. Define \(m\left(x_{0}\right)=E\left[Y \mid X=x_{0}\right]\) and \(\sigma^{2}\left(x_{0}\right)=\operatorname{Var}\left[Y \mid X=x_{0}\right]\) to be the conditional mean and conditional variance of random variable \(Y\) (residual variance), respectively, for some realization \(x_{0}\) of random variable \(X\).

The method of local polynomial smoothing is based on the approximation of \(m(x)\) locally by a \(p\) th order polynomial in \(\left(x-x_{0}\right)\) for some \(x\) in the neighborhood of \(x_{0}\). For the scatterplot data \(\left\{\left(x_{1}, y_{1}\right), \ldots,\left(x_{n}, y_{n}\right)\right\}\), the \(p\) th-order local polynomial smooth \(\widehat{m}\left(x_{0}\right)\) is equal to \(\widehat{\beta}_{0}\), an estimate of the intercept of the weighted linear regression,
\[
\begin{equation*}
\widehat{\beta}=\left(\mathbf{X}^{T} \mathbf{W} \mathbf{X}\right)^{-1} \mathbf{X}^{T} \mathbf{W} \mathbf{y} \tag{2}
\end{equation*}
\]
where \(\widehat{\beta}=\left(\widehat{\beta}_{0}, \widehat{\beta}_{1}, \ldots, \widehat{\beta}_{p}\right)^{T}\) is the vector of estimated regression coefficients (with \(\left\{\widehat{\beta}_{j}=\right.\) \(\left.\left.(j!)^{-1} \widehat{m}^{(j)}(x)\right|_{x=x_{0}}, j=0, \ldots, p\right\}\) also representing estimated coefficients from a corresponding Taylor expansion); \(\mathbf{X}=\left\{\left(x_{i}-x_{0}\right)^{j}\right\}_{i, j=1,0}^{n, p}\) is a design matrix; and \(\mathbf{W}=\operatorname{diag}\left\{K_{h}\left(x_{i}-x_{0}\right)\right\}_{n \times n}\) is a weighting matrix with weights \(K_{h}(\cdot)\) defined as \(K_{h}(x)=h^{-1} K(x / h)\), with \(K(\cdot)\) being a kernel function and \(h\) defining a bandwidth. The kernels are defined in Methods and formulas of [R] kdensity.

The default bandwidth is obtained using the ROT method of bandwidth selection. The ROT bandwidth is the plugin estimator of the asymptotically optimal constant bandwidth. This is the bandwidth that minimizes the conditional weighted mean integrated squared error. The ROT plugin bandwidth selector for the smoothing bandwidth \(h\) is defined as follows; assuming constant residual variance \(\sigma^{2}\left(x_{0}\right)=\sigma^{2}\) and odd degree \(p\) :
\[
\begin{equation*}
\widehat{h}=C_{0, p}(K)\left[\frac{\widehat{\sigma}^{2} \int w_{0}(x) d x}{n \int\left\{\widehat{m}^{(p+1)}(x)\right\}^{2} w_{0}(x) f(x) d x}\right]^{1 /(2 p+3)} \tag{3}
\end{equation*}
\]
where \(C_{0, p}(K)\) is a constant, as defined in Fan and Gijbels (1996), that depends on the kernel function \(K(\cdot)\), and the degree of a polynomial \(p\) and \(w_{0}\) is chosen to be an indicator function on the interval \(\left[\min _{\mathbf{x}}+0.05 \times\right.\) range \(_{\mathbf{x}}, \max _{\mathbf{x}}-0.05 \times\) range \(\left._{\mathbf{x}}\right]\) with \(\min _{\mathbf{x}}, \max _{\mathbf{x}}\), and range \({ }_{\mathbf{x}}\) being, respectively, the minimum, maximum, and the range of \(\mathbf{x}\). To obtain the estimates of a constant residual variance, \(\widehat{\sigma}^{2}\), and \((p+1)\) th order derivative of \(m(x)\), denoted as \(\widehat{m}^{(p+1)}(x)\), a polynomial in \(\mathbf{x}\) of order \((p+3)\) is fit globally to \(\mathbf{y} . \widehat{\sigma}^{2}\) is estimated as a standardized residual sum of squares from this fit.

The expression for the asymptotically optimal constant bandwidth used in constructing the ROT bandwidth estimator is derived for the odd-order polynomial approximations. For even-order polynomial fits the expression would depend not only on \(m^{(p+1)}(x)\) but also on \(m^{(p+2)}(x)\) and the design density and its derivative, \(f(x)\) and \(f^{\prime}(x)\). Therefore, the ROT bandwidth selector would require estimation of these additional quantities. Instead, for an even-degree \(p\) of the local polynomial, lpoly uses the value of the ROT estimator (3) computed using degree \(p+1\). As such, for even degrees this is not a plugin estimator of the asymptotically optimal constant bandwidth.

The estimates of the conditional variance of local polynomial estimators are obtained using
\[
\begin{equation*}
\widehat{\operatorname{Var}}\left\{\widehat{m}\left(x_{0}\right) \mid X=x_{0}\right\}=\widehat{\sigma}_{m}^{2}\left(x_{0}\right)=\left(\mathbf{X}^{T} \mathbf{W} \mathbf{X}\right)^{-1}\left(\mathbf{X}^{T} \mathbf{W}^{\mathbf{2}} \mathbf{X}\right)\left(\mathbf{X}^{T} \mathbf{W} \mathbf{X}\right)^{-1} \widehat{\sigma}^{2}\left(x_{0}\right) \tag{4}
\end{equation*}
\]
where \(\widehat{\sigma}^{2}\left(x_{0}\right)\) is estimated by the normalized weighted residual sum of squares from the \((p+2)\) th order polynomial fit using pilot bandwidth \(h^{\star}\).

When the bias is negligible the normal-approximation method yields a \((1-\alpha) \times 100 \%\) confidence interval for \(m\left(x_{0}\right)\),
\[
\left\{\widehat{m}\left(x_{0}\right)-z_{(1-\alpha / 2)} \widehat{\sigma}_{m}\left(x_{0}\right), \widehat{m}\left(x_{0}\right)+z_{(1-\alpha / 2)} \widehat{\sigma}_{m}\left(x_{0}\right)\right\}
\]
where \(z_{(1-\alpha / 2)}\) is the \((1-\alpha / 2)\) th quantile of the standard Gaussian distribution, and \(\widehat{m}\left(x_{0}\right)\) and \(\widehat{\sigma}_{m}\left(x_{0}\right)\) are as defined in (2) and (4), respectively.

\section*{References}

Chetverikov, D., D. Kim, and D. Wilhelm. 2018. Nonparametric instrumental-variable estimation. Stata Journal 18: 937-950.
Cleveland, W. S. 1979. Robust locally weighted regression and smoothing scatterplots. Journal of the American Statistical Association 74: 829-836. https://doi.org/10.2307/2286407.
Cox, N. J. 2005. Speaking Stata: Smoothing in various directions. Stata Journal 5: 574-593.
Donoho, D. L. 1995. Nonlinear solution of linear inverse problems by wavelet-vaguelette decomposition. Applied and Computational Harmonic Analysis 2: 101-126. https://doi.org/10.1006/acha.1995.1008.
Eubank, R. L. 1999. Nonparametric Regression and Spline Smoothing. 2nd ed. New York: Dekker.
Fan, J. 1992. Design-adaptive nonparametric regression. Journal of the American Statistical Association 87: 998-1004. https://doi.org/10.2307/2290637.
Fan, J., and I. Gijbels. 1996. Local Polynomial Modelling and Its Applications. London: Chapman and Hall.
Gasser, T., and H.-G. Müller. 1979. Kernel estimation of regression functions. In Smoothing Techniques for Curve Estimation, Lecture Notes in Mathematics, ed. T. Gasser and M. Rosenblatt, 23-68. New York: Springer.
Gutierrez, R. G., J. M. Linhart, and J. S. Pitblado. 2003. From the help desk: Local polynomial regression and Stata plugins. Stata Journal 3: 412-419.
Nadaraya, E. A. 1964. On estimating regression. Theory of Probability and Its Applications 9: 141-142. https://doi.org/10.1137/1109020.
Sheather, S. J., and M. C. Jones. 1991. A reliable data-based bandwidth selection method for kernel density estimation. Journal of the Royal Statistical Society, Series B 53: 683-690. https://doi.org/10.1111/j.2517-6161.1991.tb01857.x.
Verardi, V., and N. Debarsy. 2012. Robinson's square root of \(N\) consistent semiparametric regression estimator in Stata. Stata Journal 12: 726-735.
Watson, G. S. 1964. Smooth regression analysis. Sankhyā, Series A 26: 359-372.

\section*{Also see}
[R] kdensity - Univariate kernel density estimation
[R] lowess - Lowess smoothing
[R] makespline - Spline generation
[R] npregress intro - Introduction to nonparametric regression
[R] smooth - Robust nonlinear smoother
[G-2] graph twoway lpoly - Local polynomial smooth plots
[G-2] graph twoway lpolyci - Local polynomial smooth plots with CIs

\section*{Title}

Iroc - Compute area under ROC curve and graph the curve
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
lroc graphs the ROC curve and calculates the area under the curve.
lroc requires that the current estimation results be from logistic, logit, probit, or ivprobit; see \([R]\) logistic, \([R]\) logit, \([R]\) probit, or \([R]\) ivprobit.

\section*{Quick start}

Graph and compute area under ROC curve for current estimation results
lroc
Add "My Title" as title of graph
lroc, title(My Title)
Suppress graph
lroc, nograph

\section*{Menu}

Statistics \(>\) Binary outcomes \(>\) Postestimation \(>\) ROC curve after logistic/logit/probit/ivprobit

\section*{Syntax}
lroc \([\) depvar \(][\) if \(][\) in] [weight] [, options]
options Description

Main
all compute area under ROC curve and graph curve for all observations
nograph suppress graph
Advanced
beta (matname) row vector containing model coefficients
Plot
cline_options
marker_options
marker_label_options
Reference line rlopts (cline_options) affect rendition of the reference line

Add plots
addplot (plot) add other plots to the generated graph
Y axis, X axis, Titles, Legend, Overall
twoway_options any options other than by () documented in [G-3] twoway_options
collect is allowed; see [U] 11.1.10 Prefix commands.
fweights are allowed; see [U] 11.1.6 weight.
lroc is not appropriate with svy estimation results.

\section*{Options}
\(\qquad\) Main
all requests that the statistic be computed for all observations in the data, ignoring any if or in restrictions specified by the estimation command.
nograph suppresses graphical output.

Advanced
beta(matname) specifies a row vector containing model coefficients. The columns of the row vector must be labeled with the corresponding names of the independent variables in the data. The dependent variable depvar must be specified immediately after the command name. See Models other than the last fitted model later in this entry.

\section*{Plot}
cline_options, marker_options, and marker_label_options affect the rendition of the ROC curve-the plotted points connected by lines. These options affect the size and color of markers, whether and how the markers are labeled, and whether and how the points are connected; see [G-3] cline_options, [G-3] marker_options, and [G-3] marker_label_options.
rlopts (cline_options) affects the rendition of the reference line; see [G-3] cline_options.
\(\qquad\)
Add plots
addplot (plot) provides a way to add other plots to the generated graph; see [G-3] addplot_option.

Y axis, X axis, Titles, Legend, Overall
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

\section*{Remarks and examples}

Remarks are presented under the following headings:

\author{
Introduction \\ Samples other than the estimation sample \\ Models other than the last fitted model
}

\section*{Introduction}

Stata also has a suite of commands for performing both parametric and nonparametric receiver operating characteristic (ROC) analysis. See \([R]\) roc for an overview of these commands.
lroc graphs the ROC curve - a graph of sensitivity versus one minus specificity as the cutoff \(c\) is varied-and calculates the area under it. Sensitivity is the fraction of observed positive-outcome cases that are correctly classified; specificity is the fraction of observed negative-outcome cases that are correctly classified. When the purpose of the analysis is classification, you must choose a cutoff.

The curve starts at \((0,0)\), corresponding to \(c=1\), and continues to \((1,1)\), corresponding to \(c=0\). A model with no predictive power would be a \(45^{\circ}\) line. The greater the predictive power, the more bowed the curve, and hence the area beneath the curve is often used as a measure of the predictive power. A model with no predictive power has area 0.5 ; a perfect model has area 1.

The ROC curve was first discussed in signal detection theory (Peterson, Birdsall, and Fox 1954) and then was quickly introduced into psychology (Tanner and Swets 1954). It has since been applied in other fields, particularly medicine (for instance, Metz [1978]). For a classic text on ROC techniques, see Green and Swets (1966).
lsens also plots sensitivity and specificity; see [R] Isens.

\section*{> Example 1}

Hardin and Hilbe (2018) examine data from the National Canadian Registry of Cardiovascular Disease (FASTRAK), sponsored by Hoffman-La Roche Canada. They model death within 48 hours based on whether a patient suffers an anterior infarct (heart attack) rather than an inferior infarct using a logistic regression and evaluate the model using an ROC curve. We replicate their analysis here.

Both anterior and inferior refer to sites on the heart where damage occurs. The model is also adjusted for hcabg, whether the subject has had a cardiac bypass surgery (CABG); age, a four-category age-group indicator; and killip, a four-level risk indicator.

We load the data and then estimate the parameters of the logistic regression with logistic. Factor-variable notation is used for each predictor, because they are categorical; see [U] 11.4.3 Factor variables.
\begin{tabular}{ll}
. use https://www.stata-press.com/data/r18/heart \\
(Heart attacks) & \\
- logistic death i.site i.hcabg i.killip i.age & \\
Logistic regression & Number of obs \(=4,483\) \\
& LR chi2 ( 8 ) \(=211.37\) \\
& Prob \(>\) chi2 \(=0.0000\) \\
Log likelihood \(=-636.62553\) & Pseudo R2 \(=0.1424\)
\end{tabular}
\begin{tabular}{c|cccccc}
\hline death & Odds ratio & Std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline site & & & & & & \\
Anterior & 1.901333 & .3185757 & 3.83 & 0.000 & 1.369103 & 2.640464 \\
1.hcabg & 2.105275 & .7430694 & 2.11 & 0.035 & 1.054076 & 4.204801 \\
& & & & & & \\
killip & & & & & & \\
2 & 2.251732 & .4064423 & 4.50 & 0.000 & 1.580786 & 3.207453 \\
3 & 2.172105 & .584427 & 2.88 & 0.004 & 1.281907 & 3.680487 \\
4 & 14.29137 & 5.087654 & 7.47 & 0.000 & 7.112964 & 28.71423 \\
& & & & & & \\
age & & & & & & \\
\(60-69\) & 1.63726 & .5078582 & 1.59 & 0.112 & .8914261 & 3.007115 \\
\(70-79\) & 4.532029 & 1.206534 & 5.68 & 0.000 & 2.689568 & 7.636647 \\
>=80 & 8.893222 & 2.41752 & 8.04 & 0.000 & 5.219991 & 15.15125 \\
& & & & & & .0038529 \\
_cons & .0063961 & .0016541 & -19.54 & 0.000 & .010618 \\
\hline
\end{tabular}

Note: _cons estimates baseline odds.
The odds ratios for a unit change in each covariate are reported by logistic. At fixed values of the other covariates, patients who enter Canadian hospitals with an anterior infarct have nearly twice the odds of death within 48 hours than those with an inferior infarct. Those who have had a previous CABG have approximately twice the risk of death of those who have not. Those with higher Killip risks and those who are older are also at greater risk of death.

We use lroc to draw the ROC curve for the model. The area under the curve of approximately 0.8 indicates acceptable discrimination for the model.
. lroc
Logistic model for death
Number of observations \(=4483\)
Area under ROC curve \(=0.7965\)


Area under ROC curve \(=0.7965\)

\section*{Samples other than the estimation sample}
lroc can be used with samples other than the estimation sample. By default, lroc remembers the estimation sample used with the last logistic, logit, probit, or ivprobit command. To override this, simply use an if or in restriction to select another set of observations, or specify the all option to force the command to use all the observations in the dataset.

See example 3 in [R] estat gof for an example of using lroc with a sample other than the estimation sample.

\section*{Models other than the last fitted model}

By default, lroc uses the last model fit by logistic, logit, probit, or ivprobit. You may also directly specify the model to lroc by inputting a vector of coefficients with the beta() option and passing the name of the dependent variable depvar to lroc.

\section*{> Example 2}

Suppose that someone publishes the following logistic model of low birthweight:
\(\operatorname{Pr}(\) low \(=1)=F(-0.02\) age -0.01 lwt +1.3 black +1.1 smoke \(+0.5 \mathrm{ptl}+1.8 \mathrm{ht}+0.8 \mathrm{ui}+0.5)\)
where \(F\) is the cumulative logistic distribution. These coefficients are not odds ratios; they are the equivalent of what logit produces.

We can see whether this model fits our data. First, we enter the coefficients as a row vector and label its columns with the names of the independent variables plus _cons for the constant (see \([\mathrm{P}]\) matrix define and \([\mathrm{P}]\) matrix rownames).
```

. use https://www.stata-press.com/data/r18/lbw3, clear
(Hosmer \& Lemeshow data)
. matrix input b = (-.02, -.01, 1.3, 1.1, .5, 1.8, .8, .5)
. matrix colnames b = age lwt black smoke ptl ht ui _cons

```

Here we use lroc to examine the predictive ability of the model:
```

. lroc low, beta(b) nograph
Logistic model for low
Number of observations = 189
Area under ROC curve = 0.7275

```

The area under the curve indicates that this model does have some predictive power. We can obtain a graph of sensitivity and specificity as a function of the cutoff probability by typing
```

. lsens low, beta(b)

```


See [R] Isens.

\section*{Stored results}
lroc stores the following in \(r()\) :
Scalars
\(r(N) \quad\) number of observations
\(r\) (area) area under the ROC curve

\section*{Methods and formulas}

The ROC curve is a graph of sensitivity against ( \(1-\) specificity). This is guaranteed to be a monotone nondecreasing function because the number of correctly predicted successes increases and the number of correctly predicted failures decreases as the classification cutoff \(c\) decreases.

The area under the ROC curve is the area on the bottom of this graph and is determined by integrating the curve. The vertices of the curve are determined by sorting the data according to the predicted index, and the integral is computed using the trapezoidal rule.

\section*{References}

Green, D. M., and J. A. Swets. 1966. Signal Detection Theory and Psychophysics. New York: Wiley.
Hardin, J. W., and J. M. Hilbe. 2018. Generalized Linear Models and Extensions. 4th ed. College Station, TX: Stata Press.
Hosmer, D. W., Jr., S. A. Lemeshow, and R. X. Sturdivant. 2013. Applied Logistic Regression. 3rd ed. Hoboken, NJ: Wiley.

Metz, C. E. 1978. Basic principles of ROC analysis. Seminars in Nuclear Medicine 8: 283-298. https://doi.org/10.1016/s0001-2998(78)80014-2.
Peterson, W. W., T. G. Birdsall, and W. C. Fox. 1954. The theory of signal detectability. Transactions IRE Professional Group on Information Theory PGIT-4: 171-212. https://doi.org/10.1109/TIT.1954.1057460.

Tanner, W. P., Jr., and J. A. Swets. 1954. A decision-making theory of visual detection. Psychological Review 61: 401-409. https://doi.org/10.1037/h0058700.

\section*{Also see}
[R] logistic - Logistic regression, reporting odds ratios
[R] logit - Logistic regression, reporting coefficients
[R] probit - Probit regression
[R] ivprobit - Probit model with continuous endogenous covariates
[R] Isens - Graph sensitivity and specificity versus probability cutoff
\([R]\) estat classification - Classification statistics and table
[R] estat gof - Pearson or Hosmer-Lemeshow goodness-of-fit test
\([R]\) roc - Receiver operating characteristic (ROC) analysis
[U] 20 Estimation and postestimation commands

\section*{Title}

Irtest — Likelihood-ratio test after estimation
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
lrtest performs a likelihood-ratio test of the null hypothesis that the parameter vector of a statistical model satisfies some smooth constraint. To conduct the test, both the unrestricted and the restricted models must be fit using the maximum likelihood method (or some equivalent method), and the results of at least one must be stored using estimates store.
lrtest also supports composite models. In a composite model, we assume that the log likelihood and dimension (number of free parameters) of the full model are obtained as the sum of the log-likelihood values and dimensions of the constituting models.

\section*{Quick start}

Likelihood-ratio test that the coefficients for x 2 and x 3 are equal to 0
```

logit y x1 x2 x3
estimates store full
logit y x1 if e(sample)
estimates store restricted
lrtest full restricted

```

Display additional information, including AIC and BIC
lrtest full restricted, stats
Likelihood-ratio test that the coefficients for x 1 and x 3 are equal
constraint \(1 \mathrm{x} 1=\mathrm{x} 3\)
logit \(y\) x1 x2 x3, constraints(1)
estimates store constrained
lrtest full constrained
Compare stored estimates full with the last model run
lrtest full

\section*{Menu}

Statistics \(>\) Postestimation

\section*{Syntax}
lrtest modelspec \({ }_{1}\) [ modelspec \(_{2}\) ] [, options]
modelspec \(_{1}\) and modelspec \(c_{2}\) specify the restricted and unrestricted model in any order. modelspec\# is name \(|\).\(| (namelist)\)
name is the name under which estimation results were stored using estimates store (see [R] estimates store), and "." refers to the last estimation results, whether or not these were already stored. If modelspec \(_{2}\) is not specified, the last estimation result is used; this is equivalent to specifying modelspec \({ }_{2}\) as ".".
If namelist is specified for a composite model, modelspec \({ }_{1}\) and modelspec \(_{2}\) cannot have names in common; for example, lrtest (A B C) (CDE) is not allowed because both model specifications include C.
options
Description
stats display statistical information about the two models
dir display descriptive information about the two models
\(\mathrm{df}(\#) \quad\) override the automatic degrees-of-freedom calculation; seldom used
force force testing even when apparently invalid
collect is allowed; see [U] 11.1.10 Prefix commands.

\section*{Options}
stats displays statistical information about the unrestricted and restricted models, including the information indices of Akaike and Schwarz.
dir displays descriptive information about the unrestricted and restricted models; see estimates dir in [R] estimates store.
\(\mathrm{df}(\#)\) is seldom specified; it overrides the automatic degrees-of-freedom calculation.
force forces the likelihood-ratio test calculations to take place in situations where lrtest would normally refuse to do so and issue an error. Such situations arise when one or more assumptions of the test are violated, for example, if the models were fit with vce(robust), vce(cluster clustvar), or pweights; when the dependent variables in the two models differ; when the null log likelihoods differ; when the samples differ; or when the estimation commands differ. If you use the force option, there is no guarantee as to the validity or interpretability of the resulting test.

\section*{Remarks and examples}

The standard way to use lrtest is to do the following:
1. Fit either the restricted model or the unrestricted model by using one of Stata's estimation commands and then store the results using estimates store name.
2. Fit the alternative model (the unrestricted or restricted model) and then type 'lrtest name .'. lrtest determines for itself which of the two models is the restricted model by comparing the degrees of freedom.
Often, you may want to store the alternative model with estimates store name \({ }_{2}\), for instance, if you plan additional tests against models yet to be fit. The likelihood-ratio test is then obtained as lrtest name name 2 .

Remarks are presented under the following headings:

\author{
Nested models \\ Composite models
}

\section*{Nested models}
lrtest may be used with any estimation command that reports a log likelihood, including heckman, logit, poisson, stcox, and streg. You must check that one of the model specifications implies a statistical model that is nested within the model implied by the other specification. Usually, this means that both models are fit with the same estimation command (for example, both are fit by logit, with the same dependent variables) and that the set of covariates of one model is a subset of the covariates of the other model. Second, lrtest is valid only for models that are fit by maximum likelihood or by some equivalent method, so it does not apply to models that were fit with probability weights or clusters. Specifying the vce (robust) option similarly would indicate that you are worried about the valid specification of the model, so you would not use lrtest. Third, lrtest assumes that under the null hypothesis, the test statistic is (approximately) distributed as \(\chi^{2}\). This assumption is not true for likelihood-ratio tests of "boundary conditions", such as tests for the presence of overdispersion or random effects (Gutierrez, Carter, and Drukker 2001).

\section*{> Example 1}

We have data on infants born with low birthweights along with the characteristics of the mother (Hosmer, Lemeshow, and Sturdivant 2013; see also [R] logistic). We fit the following model:
\begin{tabular}{ll}
. use https://www.stata-press.com/data/r18/lbw \\
(Hosmer \& Lemeshow data) & \\
- logistic low age lwt i.race smoke ptl ht ui & \\
Logistic regression & Number of obs \(=\)\begin{tabular}{r}
189 \\
\\
\\
Log likelihood \(=-100.724\)
\end{tabular}\(\quad\) LR chi2 (8) \(=33.22\) \\
\end{tabular}
\begin{tabular}{r|rrcccr}
\hline low & Odds ratio & Std. err. & z & P>|z| & [95\% conf. interval] \\
age & .9732636 & .0354759 & -0.74 & 0.457 & .9061578 & 1.045339 \\
lwt & .9849634 & .0068217 & -2.19 & 0.029 & .9716834 & .9984249 \\
race & & & & & & \\
Black & 3.534767 & 1.860737 & 2.40 & 0.016 & 1.259736 & 9.918406 \\
Other & 2.368079 & 1.039949 & 1.96 & 0.050 & 1.001356 & 5.600207 \\
& & & & & & \\
smoke & 2.517698 & 1.00916 & 2.30 & 0.021 & 1.147676 & 5.523162 \\
ptl & 1.719161 & .5952579 & 1.56 & 0.118 & .8721455 & 3.388787 \\
ht & 6.249602 & 4.322408 & 2.65 & 0.008 & 1.611152 & 24.24199 \\
ui & 2.1351 & .9808153 & 1.65 & 0.099 & .8677528 & 5.2534 \\
_cons & 1.586014 & 1.910496 & 0.38 & 0.702 & .1496092 & 16.8134 \\
\hline
\end{tabular}

\footnotetext{
Note: _cons estimates baseline odds.
}

We now wish to test the constraint that the coefficients on age, 1 wt , ptl , and ht are all zero or, equivalently here, that the odds ratios are all 1 . One solution is to type
```

. test age lwt ptl ht
( 1) [low] age = 0
( 2) [low]lwt = 0
( 3) [low]ptl = 0
(4) [low]ht = 0
chi2( 4) = 12.38
Prob > chi2 = 0.0147

```

This test is based on the inverse of the information matrix and is therefore based on a quadratic approximation to the likelihood function; see \([\mathrm{R}]\) test. A more precise test would be to refit the model, applying the proposed constraints, and then calculate the likelihood-ratio test.

We first save the current model:
```

. estimates store full

```

We then fit the constrained model, which here is the model omitting age, lwt, ptl, and ht:
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline \multicolumn{5}{|l|}{Logistic regression} & \begin{tabular}{l}
Number of obs LR chi2(4) \\
Prob > chi2 \\
Pseudo R2
\end{tabular} & \[
\begin{array}{lr}
= & 189 \\
= & 18.80 \\
= & 0.0009 \\
= & 0.0801
\end{array}
\] \\
\hline low & Odds ratio & Std. err. & z & \(P>|z|\) & [95\% conf. & interval] \\
\hline \multicolumn{7}{|l|}{race} \\
\hline Black & 3.052746 & 1.498087 & 2.27 & 0.023 & 1. 166747 & 7.987382 \\
\hline Other & 2.922593 & 1.189229 & 2.64 & 0.008 & 1.316457 & 6.488285 \\
\hline smoke & 2.945742 & 1.101838 & 2.89 & 0.004 & 1.415167 & 6.131715 \\
\hline ui & 2.419131 & 1.047359 & 2.04 & 0.041 & 1.035459 & 5.651788 \\
\hline _cons & . 1402209 & . 0512295 & -5.38 & 0.000 & . 0685216 & . 2869447 \\
\hline
\end{tabular}

Note: _cons estimates baseline odds.
That done, lrtest compares this model with the model we previously stored:
```

. lrtest full .
Likelihood-ratio test
Assumption: . nested within full
LR chi2(4) = 14.42
Prob > chi2 = 0.0061

```

Let's compare results. test reported that age, lwt, ptl, and ht were jointly significant at the \(1.5 \%\) level; lrtest reports that they are significant at the \(0.6 \%\) level. Given the quadratic approximation made by test, we could argue that lrtest's results are more accurate.
lrtest explicates the assumption that, from a comparison of the degrees of freedom, it has assessed that the last fit model (.) is nested within the model stored as full. In other words, full is the unconstrained model and . is the constrained model.

The names in "(Assumption: . nested in full)" are actually links. Click on a name, and the results for that model are replayed.

Aside: The nestreg command provides a simple syntax for performing likelihood-ratio tests for nested model specifications; see \([R]\) nestreg. In the previous example, we fit a full logistic model, used estimates store to store the full model, fit a constrained logistic model, and used lrtest to report a likelihood-ratio test between two models. To do this with one call to nestreg, use the lrtable option.

\section*{- Technical note}
lrtest determines the degrees of freedom of a model as the rank of the (co)variance matrix \(e(V)\). There are two issues here. First, the numerical determination of the rank of a matrix is a subtle problem that can, for instance, be affected by the scaling of the variables in the model. The rank of a matrix depends on the number of (independent) linear combinations of coefficients that sum exactly to zero. In the world of numerical mathematics, it is hard to tell whether a very small number is really nonzero or is a real zero that happens to be slightly off because of roundoff error from the finite precision with which computers make floating-point calculations. Whether a small number is being classified as one or the other, typically on the basis of a threshold, affects the determined degrees of freedom. Although Stata generally makes sensible choices, it is bound to make mistakes occasionally. The moral of this story is to make sure that the calculated degrees of freedom is as you expect before interpreting the results.

\section*{Technical note}

A second issue involves regress and related commands such as anova. Mainly for historical reasons, regress does not treat the residual variance, \(\sigma^{2}\), the same way that it treats the regression coefficients. Type estat vce after regress, and you will see the regression coefficients, not \(\widehat{\sigma}^{2}\). Most estimation commands for models with ancillary parameters (for example, streg and heckman) treat all parameters as equals. There is nothing technically wrong with regress here; we are usually focused on the regression coefficients, and their estimators are uncorrelated with \(\widehat{\sigma}^{2}\). But, formally, \(\sigma^{2}\) adds a degree of freedom to the model, which does not matter if you are comparing two regression models by a likelihood-ratio test. This test depends on the difference in the degrees of freedom, and hence being "off by 1 " in each does not matter. But, if you are comparing a regression model with a larger model-for example, a heteroskedastic regression model fit by arch-the automatic determination of the degrees of freedom is incorrect, and you must specify the \(\mathrm{df}(\#)\) option.
> Example 2
Returning to the low-birthweight data in example 1, we now wish to test that the coefficient on 2 .race (black) is equal to that on 3.race (other). The base model is still stored under the name full, so we need only fit the constrained model and perform the test. With \(z\) as the index of the logit model, the base model is
\[
z=\beta_{0}+\beta_{1} \text { age }+\beta_{2} \text { lwt }+\beta_{3} 2 . \text { race }+\beta_{4} 3 . \text { race }+\cdots
\]

If \(\beta_{3}=\beta_{4}\), this can be written as
\[
z=\beta_{0}+\beta_{1} \text { age }+\beta_{2} \text { lwt }+\beta_{3}(2 . \text { race }+3 . \text { race })+\cdots
\]

We can fit the constrained model as follows:


Note: _cons estimates baseline odds.
Comparing this model with our original model, we obtain
```

. lrtest full .
Likelihood-ratio test
Assumption: . nested within full
LR chi2(1) = 0.55
Prob > chi2 = 0.4577

```

By comparison, typing test 2.race=3.race after fitting our base model results in a significance level of 0.4572 . Alternatively, we can first store the restricted model, here using the name equal. Next, lrtest is invoked specifying the names of the restricted and unrestricted models (we do not care about the order). This time, we also add the option stats requesting a table of model statistics, including the model selection indices AIC and BIC.
```

. estimates store equal
. lrtest equal full, stats
Likelihood-ratio test
Assumption: equal nested within full
LR chi2(1) = 0.55
Prob > chi2 = 0.4577
Akaike's information criterion and Bayesian information criterion

```
\begin{tabular}{r|rrrrrr}
\hline Model & N & ll(null) & ll(model) & df & AIC & BIC \\
\hline equal & 189 & & .0 & -100.9997 & 8 & 217.9994 \\
full & 189 & -117.336 & -100.724 & 9 & 219.448 & 248.6237 \\
\hline
\end{tabular}

Note: BIC uses \(N=\) number of observations. See [R] IC note.

\section*{Composite models}
lrtest supports composite models; that is, models that can be fit by fitting a series of simpler models or by fitting models on subsets of the data. Theoretically, a composite model is one in which the likelihood function, \(L(\theta)\), of the parameter vector, \(\theta\), can be written as the product
\[
L(\theta)=L_{1}\left(\theta_{1}\right) \times L_{2}\left(\theta_{2}\right) \times \cdots \times L_{k}\left(\theta_{k}\right)
\]
of likelihood terms with \(\theta=\left(\theta_{1}, \ldots, \theta_{k}\right)\) a partitioning of the full parameter vector. In such a case, the full-model likelihood \(L(\theta)\) is maximized by maximizing the likelihood terms \(L_{j}\left(\theta_{j}\right)\) in turn. Obviously, \(\log L(\widehat{\theta})=\sum_{j=1}^{k} \log L_{j}\left(\widehat{\theta}_{j}\right)\). The degrees of freedom for the composite model is obtained as the sum of the degrees of freedom of the constituting models.

\section*{> Example 3}

As an example of the application of composite models, we consider a test of the hypothesis that the coefficients of a statistical model do not differ between different portions ("regimes") of the covariate space. Economists call a test for such a hypothesis a Chow test.

We continue the analysis of the data on children of low birthweight by using logistic regression modeling and study whether the regression coefficients are the same among the three races: white, black, and other. A likelihood-ratio Chow test can be obtained by fitting the logistic regression model for each of the races and then comparing the combined results with those of the model previously stored as full. Because the full model included dummies for the three races, this version of the Chow test allows the intercept of the logistic regression model to vary between the regimes (races).
\begin{tabular}{|c|c|c|}
\hline \multirow[t]{3}{*}{Logistic regression} & Number of obs & \(=96\) \\
\hline & LR chi2 (6) & 13.86 \\
\hline & Prob > chi2 & 0.0312 \\
\hline Log likelihood = -45.927061 & Pseudo R2 & = 0.1311 \\
\hline
\end{tabular}
\begin{tabular}{r|rrrrrr}
\hline low & Odds ratio & Std. err. & \(\mathbf{z}\) & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline age & .9869674 & .0527757 & -0.25 & 0.806 & .8887649 & 1.096021 \\
lwt & .9900874 & .0106101 & -0.93 & 0.353 & .9695089 & 1.011103 \\
smoke & 4.208697 & 2.680133 & 2.26 & 0.024 & 1.20808 & 14.66222 \\
ptl & 1.592145 & .7474264 & 0.99 & 0.322 & .6344379 & 3.995544 \\
ht & 2.900166 & 3.193537 & 0.97 & 0.334 & .3350554 & 25.1032 \\
ui & 1.229523 & .9474768 & 0.27 & 0.789 & .2715165 & 5.567715 \\
_cons & .4891008 & .993785 & -0.35 & 0.725 & .0091175 & 26.23746 \\
\hline
\end{tabular}

Note: _cons estimates baseline odds.
estimates store white
. logistic low age lwt smoke ptl ht ui if 2.race, nolog
Logistic regression
Number of obs \(=26\)
LR chi2(6) \(=10.12\)
Prob > chi2 \(=0.1198\)
Log likelihood = -12.654157
Pseudo R2 \(=0.2856\)
\begin{tabular}{r|rrcccr}
\hline low & Odds ratio & Std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & {\([95 \%\) conf. interval] } \\
\hline age & .8735313 & .1377846 & -0.86 & 0.391 & .6412332 & 1.189983 \\
lwt & .9747736 & .016689 & -1.49 & 0.136 & .9426065 & 1.008038 \\
smoke & 16.50373 & 24.37044 & 1.90 & 0.058 & .9133647 & 298.2083 \\
ptl & 4.866916 & 9.33151 & 0.83 & 0.409 & .1135573 & 208.5895 \\
ht & 85.05605 & 214.6382 & 1.76 & 0.078 & .6049308 & 11959.27 \\
ui & 67.61338 & 133.3313 & 2.14 & 0.033 & 1.417399 & 3225.322 \\
_cons & 48.7249 & 169.9216 & 1.11 & 0.265 & .0523961 & 45310.94 \\
\hline
\end{tabular}

Note: _cons estimates baseline odds.
. estimates store black
. logistic low age lwt smoke ptl ht ui if 3.race, nolog
Logistic regression \(\quad\) Number of obs \(=\quad 67\)
LR chi2(6) \(=14.06\)
Prob > chi2 \(=0.0289\)
Log likelihood = -37. 228444
Pseudo R2 \(=0.1589\)
\begin{tabular}{r|rrcccr}
\hline low & Odds ratio & Std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline age & .9263905 & .0665386 & -1.06 & 0.287 & .8047407 & 1.06643 \\
lwt & .9724499 & .015762 & -1.72 & 0.085 & .9420424 & 1.003839 \\
smoke & .7979034 & .6340585 & -0.28 & 0.776 & .1680885 & 3.787586 \\
ptl & 2.845675 & 1.777944 & 1.67 & 0.094 & .8363053 & 9.682908 \\
ht & 7.767503 & 10.00537 & 1.59 & 0.112 & .6220764 & 96.98826 \\
ui & 2.925006 & 2.046473 & 1.53 & 0.125 & .7423107 & 11.52571 \\
_cons & 49.09444 & 113.9165 & 1.68 & 0.093 & .5199275 & 4635.769 \\
\hline
\end{tabular}

Note: _cons estimates baseline odds.
. estimates store other
We are now ready to perform the likelihood-ratio Chow test:
```

. lrtest (full) (white black other), stats
Likelihood-ratio test
Assumption: full nested within (white, black, other)
LR chi2(12) = 9.83
Prob > chi2 = 0.6310
Akaike's information criterion and Bayesian information criterion

```
\begin{tabular}{r|rrrrrr}
\hline Model & N & ll(null) & ll(model) & df & AIC & BIC \\
\hline full & 189 & -117.336 & -100.724 & 9 & 219.448 & 248.6237 \\
white & 96 & -52.85752 & -45.92706 & 7 & 105.8541 & 123.8046 \\
black & 26 & -17.71291 & -12.65416 & 7 & 39.30831 & 48.11499 \\
other & 67 & -44.26039 & -37.22844 & 7 & 88.45689 & 103.8897 \\
\hline
\end{tabular}

Note: BIC uses \(N=\) number of observations. See [R] IC note.
We cannot reject the hypothesis that the logistic regression model applies to each of the races at any reasonable significance level. By specifying the stats option, we can verify the degrees of freedom of the test: \(12=7+7+7-9\). We can obtain the same test by fitting an expanded model with interactions between all covariates and race.
. logistic low race\#\#c. (age lwt smoke ptl ht ui)
Logistic regression
Number of obs \(=189\)
LR chi2(20) \(=43.05\)
Prob > chi2 \(=0.0020\)
Log likelihood \(=-95.809661\)
Pseudo R2 \(=0.1835\)
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline low & Odds ratio & Std. err. & z & \(P>|z|\) & \multicolumn{2}{|l|}{[95\% conf. interval]} \\
\hline race & & & & & & \\
\hline Black & 99.62137 & 402.0829 & 1.14 & 0.254 & . 0365434 & 271578.9 \\
\hline Other & 100.3769 & 309.586 & 1.49 & 0.135 & . 2378638 & 42358.38 \\
\hline age & . 9869674 & . 0527757 & -0.25 & 0.806 & . 8887649 & 1.096021 \\
\hline lwt & . 9900874 & . 0106101 & -0.93 & 0.353 & . 9695089 & 1.011103 \\
\hline smoke & 4.208697 & 2.680133 & 2.26 & 0.024 & 1.20808 & 14.66222 \\
\hline ptl & 1.592145 & . 7474264 & 0.99 & 0.322 & . 6344379 & 3.995544 \\
\hline ht & 2.900166 & 3.193537 & 0.97 & 0.334 & . 3350554 & 25.1032 \\
\hline ui & 1.229523 & . 9474768 & 0.27 & 0.789 & . 2715165 & 5.567715 \\
\hline \multicolumn{7}{|l|}{race\#c.age} \\
\hline Black & . 885066 & . 1474079 & -0.73 & 0.464 & . 638569 & 1.226714 \\
\hline Other & . 9386232 & . 0840486 & -0.71 & 0.479 & . 7875366 & 1.118695 \\
\hline \multicolumn{7}{|l|}{race\#c.lwt} \\
\hline Black & . 9845329 & . 0198857 & -0.77 & 0.440 & . 9463191 & 1.02429 \\
\hline Other & . 9821859 & . 0190847 & -0.93 & 0.355 & . 9454839 & 1.020313 \\
\hline \multicolumn{7}{|l|}{race\#c.smoke} \\
\hline Black & 3.921338 & 6.305992 & 0.85 & 0.395 & . 167725 & 91.67917 \\
\hline Other & . 1895844 & . 1930601 & -1.63 & 0.102 & . 025763 & 1.395113 \\
\hline \multicolumn{7}{|l|}{race\#c.ptl} \\
\hline Black & 3.05683 & 6.034089 & 0.57 & 0.571 & . 0638301 & 146.3918 \\
\hline Other & 1.787322 & 1.396789 & 0.74 & 0.457 & . 3863582 & 8.268285 \\
\hline \multicolumn{7}{|l|}{race\#c.ht} \\
\hline Black & 29.328 & 80.7482 & 1.23 & 0.220 & . 1329492 & 6469.623 \\
\hline Other & 2.678295 & 4.538712 & 0.58 & 0.561 & . 0966916 & 74.18702 \\
\hline \multicolumn{7}{|l|}{race\#c.ui} \\
\hline Black & 54.99155 & 116.4274 & 1.89 & 0.058 & . 8672471 & 3486.977 \\
\hline Other & 2.378976 & 2.476124 & 0.83 & 0.405 & . 309335 & 18.29579 \\
\hline _cons & . 4891008 & . 993785 & -0.35 & 0.725 & . 0091175 & 26.23746 \\
\hline
\end{tabular}

Note: _cons estimates baseline odds.
. lrtest full .
Likelihood-ratio test
Assumption: full nested within .
LR chi2(12) \(=9.83\)
Prob \(>\) chi2 \(=0.6310\)
Applying lrtest for the full model against the model with all interactions yields the same test statistic and \(p\)-value as for the full model against the composite model for the three regimes. Here the specification of the model with interactions was convenient, and logistic had no problem computing the estimates for the expanded model. In models with more complicated likelihoods, such as Heckman's selection model (see [R] heckman) or complicated survival-time models (see [ST] streg), fitting the models with all interactions may be numerically demanding and may be much more time consuming than fitting a series of models separately for each regime.

Given the model with all interactions, we could also test the hypothesis of no differences among the regions (races) by a Wald version of the Chow test by using the testparm command; see \([R]\) test.
```

. testparm race\#c.(age lwt smoke ptl ht ui)
( 1) [low]2.race\#c.age = 0
( 2) [low]3.race\#c.age = 0
( 3) [low]2.race\#c.lwt = 0
(4) [low]3.race\#c.lwt = 0
( 5) [low]2.race\#c.smoke = 0
( 6) [low]3.race\#c.smoke = 0
( 7) [low]2.race\#c.ptl = 0
( 8) [low]3.race\#c.ptl = 0
( 9) [low]2.race\#c.ht = 0
(10) [low]3.race\#c.ht = 0
(11) [low]2.race\#c.ui = 0
(12) [low]3.race\#c.ui = 0
chi2( 12) = 8.24
Prob > chi2 = 0.7663

```

We conclude that, here, the Wald version of the Chow test is similar to the likelihood-ratio version of the Chow test.

\section*{Stored results}
lrtest stores the following in \(r()\) :
Scalars
\(\mathrm{r}(\mathrm{p}) \quad p\)-value for likelihood-ratio test
\(r(d f) \quad\) degrees of freedom
r(chi2) LR test statistic
Programmers wishing their estimation commands to be compatible with lrtest should note that lrtest requires that the following results be returned:
\begin{tabular}{ll}
\(e(c m d)\) & name of estimation command \\
\(e(l l)\) & log likelihood \\
\(e(V)\) & variance-covariance matrix of the estimators \\
\(e(N)\) & number of observations
\end{tabular}
lrtest also verifies that e(N), e(ll_0), and e(depvar) are consistent between two noncomposite models.

\section*{Methods and formulas}

Let \(L_{0}\) and \(L_{1}\) be the log-likelihood values associated with the full and constrained models, respectively. The test statistic of the likelihood-ratio test is LR \(=-2\left(L_{1}-L_{0}\right)\). If the constrained model is true, LR is approximately \(\chi^{2}\) distributed with \(d_{0}-d_{1}\) degrees of freedom, where \(d_{0}\) and \(d_{1}\) are the model degrees of freedom associated with the full and constrained models, respectively (Greene 2018, 554-555).
lrtest determines the degrees of freedom of a model as the rank of \(e(V)\), computed as the number of nonzero diagonal elements of invsym (e(V)).

\section*{References}

Greene, W. H. 2018. Econometric Analysis. 8th ed. New York: Pearson.
Gutierrez, R. G., S. L. Carter, and D. M. Drukker. 2001. sg160: On boundary-value likelihood-ratio tests. Stata Technical Bulletin 60: 15-18. Reprinted in Stata Technical Bulletin Reprints, vol. 10, pp. 269-273. College Station, TX: Stata Press.

Hosmer, D. W., Jr., S. A. Lemeshow, and R. X. Sturdivant. 2013. Applied Logistic Regression. 3rd ed. Hoboken, NJ: Wiley.

Raciborski, R. 2015. Spotlight on irt. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2015/07/31/spotlight-on-irt/.
Tauchmann, H. 2023. lgrgtest: Lagrange multiplier test after constrained maximum-likelihood estimation. Stata Journal 23: 386-401.

\section*{Also see}
[R] test - Test linear hypotheses after estimation
[R] testnl - Test nonlinear hypotheses after estimation
\([\mathrm{R}]\) nestreg - Nested model statistics

\section*{Title}

Isens - Graph sensitivity and specificity versus probability cutoff
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
Reference & Also see & &
\end{tabular}

\section*{Description}
lsens graphs sensitivity and specificity versus probability cutoff and optionally creates new variables containing these data.
lsens requires that the current estimation results be from logistic, logit, probit, or ivprobit; see \([R]\) logistic, \([R]\) logit, \([R]\) probit, or \([R]\) ivprobit.

\section*{Quick start}

Graph sensitivity and specificity versus a probability cutoff using current estimation results lsens

Generate variables v1, v2, and v3 to contain probability cutoffs, sensitivity, and specificity lsens, genprob(v1) gensens(v2) genspec(v3)

Add "My Title" to graph
lsens, genprob(v1) gensens(v2) genspec(v3) title(My Title)

\section*{Menu}

Statistics \(>\) Binary outcomes \(>\) Postestimation \(>\) Sensitivity/specificity plot

\section*{Syntax}
lsens [depvar] [if] [in] [weight] [, options]
options
Main
all
genprob (varname)
gensens (varname)
genspec (varname)
replace
nograph
Advanced
beta (matname)
Plot
connect_options
Add plots
addplot (plot) add other plots to the generated graph
Y axis, X axis, Titles, Legend, Overall
twoway_options any options other than by () documented in [G-3] twoway_options
collect is allowed; see [U] 11.1.10 Prefix commands.
fweights are allowed; see [U] 11.1.6 weight.
lsens is not appropriate with svy estimation results.

\section*{Options}
\(\qquad\) Main
all requests that the statistic be computed for all observations in the data, ignoring any if or in restrictions specified by the estimation command.
genprob (varname), gensens (varname), and genspec (varname) specify the names of new variables created to contain, respectively, the probability cutoffs and the corresponding sensitivity and specificity.
replace requests that existing variables specified for genprob(), gensens(), or genspec() be overwritten.
nograph suppresses graphical output.

Advanced
beta (matname) specifies a row vector containing model coefficients. The columns of the row vector must be labeled with the corresponding names of the independent variables in the data. The dependent variable depvar must be specified immediately after the command name. See Models other than the last fitted model later in this entry.
connect_options affect the rendition of the plotted points connected by lines; see connect_options in [G-2] graph twoway scatter.

Add plots
addplot (plot) provides a way to add other plots to the generated graph. See [G-3] addplot_option.

Y axis, X axis, Titles, Legend, Overall
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

\section*{Remarks and examples}

Remarks are presented under the following headings:

\section*{Introduction}

Models other than the last fitted model

\section*{Introduction}
lsens plots sensitivity and specificity; it plots both sensitivity and specificity versus probability cutoff \(c\). The graph is equivalent to what you would get from estat classification (see \([R]\) estat classification) if you varied the cutoff probability \(c\) from 0 to 1 .

\section*{\(>\) Example 1}

We illustrate lsens after logistic; see [R] logistic.
```

. use https://www.stata-press.com/data/r18/lbw
(Hosmer \& Lemeshow data)
. logistic low age i.race smoke ui
(output omitted)
. lsens

```

lsens optionally creates new variables containing the probability cutoff, sensitivity, and specificity.
```

. lsens, genprob(p) gensens(sens) genspec(spec) nograph

```

The variables created will have \(M+2\) distinct nonmissing values: one for each of the \(M\) covariate patterns, one for \(c=0\), and another for \(c=1\). Values are recorded for \(p=0\), for each of the observed predicted probabilities, and for \(p=1\). The total number of observations required to do this can be fewer than \(\_N\), the same as \(\_N\), or \(\_N+1\), or \(\_N+2\). If more observations are added, they are added at the end of the dataset and the values of the original variables are set to missing in the added observations. How the values added align with existing observations is irrelevant.

\section*{- Technical note}
logistic, logit, probit, or ivprobit and lsens keep track of the estimation sample. If you type, for instance, logistic ... if \(x==1\), then when you type lsens, the statistics will be calculated on the \(x==1\) subsample of the data automatically.

You should specify if or in with lsens only when you wish to produce graphs and calculate statistics for a set of observations other than the estimation sample.

If the logistic model was fit with fweights, lsens properly accounts for the weights in its calculations. You do not have to specify the weights when you run lsens. Weights should be specified with lsens only when you wish to use a different set of weights.

\section*{Models other than the last fitted model}

By default, lsens uses the last model fit. You may also directly specify the model to lsens by inputting a vector of coefficients with the beta() option and passing the name of the dependent variable depvar to lsens.
\(>\) Example 2
Suppose that someone publishes the following logistic model of low birthweight:
\[
\operatorname{Pr}(\mathrm{low}=1)=F(-0.02 \text { age }-0.01 \text { lwt }+1.3 \text { black }+1.1 \text { smoke }+0.5 \mathrm{ptl}+1.8 \mathrm{ht}+0.8 \mathrm{ui}+0.5)
\]
where \(F\) is the cumulative logistic distribution. These coefficients are not odds ratios; they are the equivalent of what logit produces.

We can see whether this model fits our data. First, we enter the coefficients as a row vector and label its columns with the names of the independent variables plus _cons for the constant (see \([\mathrm{P}]\) matrix define and \([\mathrm{P}]\) matrix rownames).
```

. use https://www.stata-press.com/data/r18/lbw3, clear
(Hosmer \& Lemeshow data)
. matrix input b = (-0.02, -.01, 1.3, 1.1, .5, 1.8, .8, .5)
. matrix colnames b = age lwt black smoke ptl ht ui _cons

```

We can use lroc (see [R] lroc) to examine the predictive ability of the model:
```

. lroc low, beta(b) nograph
Logistic model for low
Number of observations = 189
Area under ROC curve = 0.7275

```

The area under the curve indicates that this model does have some predictive power. We can obtain a graph of sensitivity and specificity as a function of the cutoff probability by typing
```

lsens low, beta(b)

```


\section*{Stored results}
lsens stores the following in \(r()\) :
Scalars
\(r(N) \quad\) number of observations

\section*{Methods and formulas}

Let \(j\) index observations and \(c\) be the cutoff probability. Let \(p_{j}\) be the predicted probability of a positive outcome and \(y_{j}\) be the actual outcome, which we will treat as 0 or 1 , although Stata treats it as 0 and non- 0 , excluding missing observations.

A prediction is classified as positive if \(p_{j} \geq c\) and otherwise is classified as negative. The classification is correct if it is positive and \(y_{j}=1\) or if it is negative and \(y_{j}=0\).

Sensitivity is the fraction of \(y_{j}=1\) observations that are correctly classified. Specificity is the percentage of \(y_{j}=0\) observations that are correctly classified.

\section*{Reference}

Hosmer, D. W., Jr., S. A. Lemeshow, and R. X. Sturdivant. 2013. Applied Logistic Regression. 3rd ed. Hoboken, NJ: Wiley.

\section*{Also see}
[R] logistic - Logistic regression, reporting odds ratios
[R] logit - Logistic regression, reporting coefficients
[R] probit - Probit regression
\([\mathrm{R}]\) ivprobit - Probit model with continuous endogenous covariates
[R] lroc - Compute area under ROC curve and graph the curve
[R] estat classification - Classification statistics and table
[R] estat gof - Pearson or Hosmer-Lemeshow goodness-of-fit test
\([R]\) roc - Receiver operating characteristic (ROC) analysis
[U] 20 Estimation and postestimation commands

\section*{Title}

Iv - Letter-value displays
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
lv shows a letter-value display (Tukey 1977, 44-49; Hoaglin 1983) for each variable in varlist. If no variables are specified, letter-value displays are shown for each numeric variable in the data.

\section*{Quick start}

Letter-value display for all numeric variables in the dataset
lv
Letter-value display for v 1
lv v1
Also generate new variables _mid, _spread, _psigma, and _z2 containing midsummaries, spreads, pseudosigmas, and \(z^{2}\) values
lv v1, generate
Letter-value displays for v1 separately for each value of v2
bysort v2: lv v1

\section*{Menu}

Statistics \(>\) Summaries, tables, and tests \(>\) Distributional plots and tests \(>\) Letter-value display

\section*{Syntax}
lv [varlist] [if] [in] [, generate tail(\#)]
by and collect are allowed; see [U] 11.1.10 Prefix commands.

\section*{Options}
\(\qquad\) Main
generate adds four new variables to the data: _mid, containing the midsummaries; _spread, containing the spreads; _psigma, containing the pseudosigmas; and \(\quad z 2\), containing the squared values from a standard normal distribution corresponding to the particular letter value. If the variables _mid, _spread, _psigma, and _z2 already exist, their contents are replaced. At most, only the first 11 observations of each variable are used; the remaining observations contain missing. If varlist specifies more than one variable, the newly created variables contain results for the last variable specified. The generate option may not be used with the by prefix.
tail(\#) indicates the inverse of the tail density through which letter values are to be displayed: 2 corresponds to the median (meaning half in each tail), 4 to the fourths (roughly the 25th and 75th percentiles), 8 to the eighths, and so on. \# may be specified as \(4,8,16,32,64,128,256,512\), or 1,024 and defaults to a value of \# that has corresponding depth just greater than 1 . The default is taken as 1,024 if the calculation results in a number larger than 1,024 . Given the intelligent default, this option is rarely specified.

\section*{Remarks and examples}

Letter-value displays are a collection of observations drawn systematically from the data, focusing especially on the tails rather than the middle of the distribution. The displays are called letter-value displays because letters have been (almost arbitrarily) assigned to tail densities:
\begin{tabular}{llll} 
Letter & Tail area & Letter & Tail area \\
\hline M & \(1 / 2\) & B & \(1 / 64\) \\
F & \(1 / 4\) & A & \(1 / 128\) \\
E & \(1 / 8\) & Z & \(1 / 256\) \\
D & \(1 / 16\) & Y & \(1 / 512\) \\
C & \(1 / 32\) & X & \(1 / 1024\)
\end{tabular}

\section*{> Example 1}

We have data on the mileage ratings of 74 automobiles. To obtain a letter-value display, we type


The decimal points can be made to line up and thus the output made more readable by specifying a display format for the variable; see [U] 12.5 Formats: Controlling how data are displayed.
\begin{tabular}{|c|c|c|c|c|c|}
\hline \# 74 & \multicolumn{3}{|c|}{Mileage (mpg)} & & \\
\hline M \(\quad 37.5\) & & 20.00 & & spread & pseudosigma \\
\hline F 19 & 18.00 & 21.50 & 25.00 & 7.00 & 5.22 \\
\hline E 10 & 15.00 & 21.50 & 28.00 & 13.00 & 5.77 \\
\hline D 5.5 & 14.00 & 22.25 & 30.50 & 16.50 & 5.58 \\
\hline C 3 & 14.00 & 24.50 & 35.00 & 21.00 & 5.83 \\
\hline B 2 & 12.00 & 23.50 & 35.00 & 23.00 & 5.73 \\
\hline A 1.5 & 12.00 & 25.00 & 38.00 & 26.00 & 6.04 \\
\hline \multirow[t]{2}{*}{1} & 12.00 & 26.50 & 41.00 & 29.00 & 6.17 \\
\hline & & & & \# below & \# above \\
\hline inner fence & 7.50 & & 35.50 & 0 & 1 \\
\hline outer fence & -3.00 & & 46.00 & 0 & 0 \\
\hline
\end{tabular}

At the top, the number of observations is indicated as 74. The first line shows the statistics associated with M, the letter value that puts half the density in each tail, or the median. The median has depth 37.5 (that is, in the ordered data, M is 37.5 observations in from the extremes) and has value 20 . The next line shows the statistics associated with F or the fourths. The fourths have depth 19 (that is, in the ordered data, the lower fourth is observation 19, and the upper fourth is observation \(74-19+1\) ), and the values of the lower and upper fourths are 18 and 25 . The number in the middle is the point halfway between the fourths - called a midsummary. If the distribution were perfectly symmetric, the midsummary would equal the median. The spread is the difference between the lower and upper summaries \((25-18=7)\). For fourths, half the data lie within a \(7-m p g\) band. The pseudosigma is a calculation of the standard deviation using only the lower and upper summaries and assuming that the variable is normally distributed. If the data really were normally distributed, all the pseudosigmas would be roughly equal.

After the letter values, the line labeled with depth 1 reports the minimum and maximum values. Here the halfway point between the extremes is 26.5 , which is greater than the median, indicating that 41 is more extreme than 12, at least relative to the median. And with each letter value, the midsummaries are increasing - our data are skewed. The pseudosigmas are also increasing, indicating
that the data are spreading out relative to a normal distribution, although, given the evident skewness, this elongation may be an artifact of the skewness.

At the end is an attempt to identify outliers, although the points so identified are merely outside some predetermined cutoff. Points outside the inner fence are called outside values or mild outliers. Points outside the outer fence are called severe outliers. The inner fence is defined as (3/2)IQR and the outer fence as 3 IQR above and below the F summaries, where the interquartile range ( IQR ) is the spread of the fourths.

\section*{Technical note}

The form of the letter-value display has varied slightly with different authors. lv displays appear as described by Hoaglin (1983) but as modified by Emerson and Stoto (1983), where they included the midpoint of each of the spreads. This format was later adopted by Hoaglin (1985). If the distribution is symmetric, the midpoints will all be roughly equal. On the other hand, if the midpoints vary systematically, the distribution is skewed.

The pseudosigmas are obtained from the lower and upper summaries for each letter value. For each letter value, they are the standard deviation a normal distribution would have if its spread for the given letter value were to equal the observed spread. If the pseudosigmas are all roughly equal, the data are said to have neutral elongation. If the pseudosigmas increase systematically, the data are said to be more elongated than a normal, that is, have thicker tails. If the pseudosigmas decrease systematically, the data are said to be less elongated than a normal, that is, have thinner tails.

Interpretation of the number of mild and severe outliers is more problematic. The following discussion is drawn from Hamilton (1991):

Obviously, the presence of any such outliers does not rule out that the data have been drawn from a normal distribution; in large datasets, there will most certainly be observations outside (3/2)IQR and 3IQR. Severe outliers, however, make up about two per million ( \(0.0002 \%\) ) of a normal population. In samples, they lie far enough out to have substantial effects on means, standard deviations, and other classical statistics. The \(0.0002 \%\), however, should be interpreted carefully; outliers appear more often in small samples than one might expect from population proportions because of sampling variation in estimated quartiles. Monte Carlo simulation by Hoaglin, Iglewicz, and Tukey (1986) obtained these results on the percentages and numbers of outliers in random samples from a normal population:
\begin{tabular}{rcccc} 
& \multicolumn{2}{c}{ percentage } & \multicolumn{2}{c}{ number } \\
n & any outliers & severe & any outliers & severe \\
\hline 10 & 2.83 & .362 & .283 & .0362 \\
20 & 1.66 & .074 & .332 & .0148 \\
50 & 1.15 & .011 & .575 & .0055 \\
100 & .95 & .002 & .95 & .002 \\
200 & .79 & .001 & 1.58 & .002 \\
300 & .75 & .001 & 2.25 & .003 \\
\(\infty\) & .70 & .0002 & \(\infty\) & \(\infty\)
\end{tabular}

Thus, the presence of any severe outliers in samples of less than 300 is sufficient to reject normality. Hoaglin, Iglewicz, and Tukey (1981) suggested the approximation \(0.00698+0.4 / n\) for the fraction of mild outliers in a sample of size \(n\) or, equivalently, \(0.00698 n+0.4\) for the number of outliers.

\section*{> Example 2}

The generate option adds the _mid, _spread, _psigma, and _z2 variables to our data, making possible many of the diagnostic graphs suggested by Hoaglin (1985).
```

. lv mpg, generate
(output omitted)
. list _mid _spread _psigma _z2 in 1/12

```
\begin{tabular}{|c|c|c|c|c|}
\hline & _mid & _spread & _psigma & _z2 \\
\hline 1. & 20 & . & . & . \\
\hline 2. & 21.5 & 7 & 5.216359 & . 4501955 \\
\hline 3. & 21.5 & 13 & 5.771728 & 1.26828 \\
\hline 4. & 22.25 & 16.5 & 5.576303 & 2.188846 \\
\hline 5. & 24.5 & 21 & 5.831039 & 3.24255 \\
\hline 6. & 23.5 & 23 & 5.732448 & 4.024532 \\
\hline 7. & 25 & 26 & 6.040635 & 4.631499 \\
\hline 8. & . & . & . & . \\
\hline 9. & - & - & - & \\
\hline 10. & - & - & . & \\
\hline 11. & 26.5 & 29 & 6.16562 & 5.53073 \\
\hline 12. & . & - & . & \\
\hline
\end{tabular}

Observations 12 through the end are missing for these new variables. The definition of the observations is always the same. The first observation contains the M summary; the second, the F; the third, the E; and so on. Observation 11 always contains the summary for depth 1 . Observations \(8-10\)-corresponding to letter values Z , Y , and X - contain missing because these statistics were not calculated. We have only 74 observations, and their depth would be 1.

Hoaglin (1985) suggests graphing the midsummary against \(z^{2}\). If the distribution is not skewed, the points in the resulting graph will be along a horizontal line:
```

. scatter _mid _z2

```


The graph clearly indicates the skewness of the distribution. We might also graph _psigma against _z2 to examine elongation.

\section*{Stored results}
\(l v\) stores the following in \(r()\) :
Scalars
\begin{tabular}{|c|c|c|c|}
\hline r (N) & number of observations & \(r\) (u_C) & upper 32nd \\
\hline \(r\) (min) & minimum & \(r\) (l_B) & lower 64th \\
\hline \(r\) (max) & maximum & \(r\) (u_B) & upper 64th \\
\hline \(r\) (median) & median & \(r\left(1 \_A\right)\) & lower 128th \\
\hline \(r\) (l_F) & lower 4th & r (u_A) & upper 128th \\
\hline r (u_F) & upper 4th & r (l_Z) & lower 256th \\
\hline \(r\) (l_E) & lower 8th & r (u_Z) & upper 256th \\
\hline r (u_E) & upper 8th & r(l_Y) & lower 512th \\
\hline r (l_D) & lower 16th & r (u_Y) & upper 512th \\
\hline r (u_D) & upper 16th & r(l_X) & lower 1024th \\
\hline r (l_C) & lower 32nd & r (u_X) & upper 1024th \\
\hline
\end{tabular}

The lower/upper 8ths, 16ths, ..., 1024ths will be defined only if there are sufficient data.

\section*{Methods and formulas}

Let \(N\) be the number of (nonmissing) observations on \(x\), and let \(x_{(i)}\) refer to the ordered data when \(i\) is an integer. Define \(x_{(i+0.5)}=\left(x_{(i)}+x_{(i+1)}\right) / 2\); the median is defined as \(x_{\{(N+1) / 2\}}\).

Define \(x_{[d]}\) as the pair of numbers \(x_{(d)}\) and \(x_{(N+1-d)}\), where \(d\) is called the depth. Thus, \(x_{[1]}\) refers to the minimum and maximum of the data. Define \(m=(N+1) / 2\) as the depth of the median, \(f=(\lfloor m\rfloor+1) / 2\) as the depth of the fourths, \(e=(\lfloor f\rfloor+1) / 2\) as the depth of the eighths, and so on. Depths are reported on the far left of the letter-value display. The corresponding fourths of the data are \(x_{[f]}\), the eighths are \(x_{[e]}\), and so on. These values are reported inside the display. The middle value is defined as the corresponding midpoint of \(x_{[\cdot]}\). The spreads are defined as the difference in \(x_{[\cdot]}\).

The corresponding point \(z_{i}\) on a standard normal distribution is obtained as (Hoaglin 1985, 456-457)
\[
z_{i}= \begin{cases}F^{-1}\left\{\left(d_{i}-1 / 3\right) /(N+1 / 3)\right\} & \text { if } d_{i}>1 \\ F^{-1}\{0.695 /(N+0.390)\} & \text { otherwise }\end{cases}
\]
where \(d_{i}\) is the depth of the letter value. The corresponding pseudosigma is obtained as the ratio of the spread to \(-2 z_{i}\) (Hoaglin 1985, 431).

Define \(\left(F_{l}, F_{u}\right)=x_{[f]}\). The inner fence has cutoffs \(F_{l}-\frac{3}{2}\left(F_{u}-F_{l}\right)\) and \(F_{u}+\frac{3}{2}\left(F_{u}-F_{l}\right)\). The outer fence has cutoffs \(F_{l}-3\left(F_{u}-F_{l}\right)\) and \(F_{u}+3\left(F_{u}-F_{l}\right)\).

The inner-fence values reported by lv are almost equal to those used by graph, box to identify outside points. The only difference is that graph uses a slightly different definition of fourths, namely, the 25 th and 75 th percentiles as defined by summarize; see \([R]\) summarize.

\section*{References}

Cox, N. J. 2016. Speaking Stata: Letter values as selected quantiles. Stata Journal 16: 1058-1071.
Emerson, J. D., and M. A. Stoto. 1983. Transforming data. In Understanding Robust and Exploratory Data Analysis, ed. D. C. Hoaglin, C. F. Mosteller, and J. W. Tukey, 97-128. New York: Wiley.
Fox, J. 1990. Describing univariate distributions. In Modern Methods of Data Analysis, ed. J. Fox and J. S. Long, 58-125. Newbury Park, CA: Sage.

Hamilton, L. C. 1991. sed4: Resistant normality check and outlier identification. Stata Technical Bulletin 3: 15-18. Reprinted in Stata Technical Bulletin Reprints, vol. 1, pp. 86-90. College Station, TX: Stata Press.
Hoaglin, D. C. 1983. Letter values: A set of selected order statistics. In Understanding Robust and Exploratory Data Analysis, ed. D. C. Hoaglin, C. F. Mosteller, and J. W. Tukey, 33-57. New York: Wiley.
__. 1985. Using quantiles to study shape. In Exploring Data Tables, Trends, and Shapes, ed. D. C. Hoaglin, C. F. Mosteller, and J. W. Tukey, 417-460. New York: Wiley.
Hoaglin, D. C., B. Iglewicz, and J. W. Tukey. 1981. Small-sample performance of a resistant rule for outlier detection. In 1980 Proceedings of the Statistical Computing Section. Washington, DC: American Statistical Association.
_-. 1986. Performance of some resistant rules for outlier labeling. Journal of the American Statistical Association 81: 991-999. https://doi.org/10.2307/2289073.
Tukey, J. W. 1977. Exploratory Data Analysis. Reading, MA: Addison-Wesley.

\section*{Also see}
[R] Diagnostic plots - Distributional diagnostic plots
[R] stem - Stem-and-leaf displays
[R] summarize - Summary statistics

\section*{Title}
makespline - Spline generation
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
makespline generates a set of variables that form B-spline, piecewise polynomial spline, and restricted cubic spline basis functions from a list of existing variables. B-spline and piecewise polynomial spline bases may be first, second, or third order, with knots at percentiles of the data or uniformly spaced over the range of the variables. Restricted cubic splines, also known as natural splines, may only be of third order.

\section*{Quick start}

Generate variables from x 1 and x 2 that form third-order B-spline basis functions, with one knot at the median of each variable
makespline bspline x1 x2
Same as above, but with three knots at the 25 th, 50 th, and 75 th percentiles
makespline bspline x 1 x 2 , knots(3)
Same as above, but use second-order B-splines
makespline bspline x1 x2, knots(3) order(2)
Generate variables that form a linear spline for x 1 with knots at 10 and 20
makespline piecewise x1, knotslist(10 20) order(1)
Same as above, but do not rescale x 1 before creating spline variables
makespline piecewise x1, knotslist(10 20) order(1) norescalevars
Generate variables that form third-order piecewise polynomial splines for x 1 and x 2 , with knots at their 25 th, 50 th, and 75 th percentiles
makespline piecewise x1 x2, knots(3)
Same as above, but with three knots at evenly spaced points over the range of \(x 1\) and of \(x 2\) makespline piecewise x 1 x 2 , knots(3) uniformknots

Specify values of knots in matrix K to generate variables forming restricted cubic splines makespline rcs \(x 1\) x2, knotsmat(K)

Generate variables that form a linear spline for x 1 without rescaling the values of x 1 makespline linear x1

\section*{Menu}

Data \(>\) Create or change data \(>\) Other variable-creation commands \(>\) Spline generation

\section*{Syntax}
makespline basis varlist \([\) if \(][\) in \(]\) [weight \(][\), options \(]\)
basis Description
\begin{tabular}{ll} 
bspline & B-spline \\
piecewise & piecewise polynomial spline \\
rcs & restricted cubic spline \\
linear & linear spline-piecewise basis of order 1 without rescaling variables
\end{tabular}
options Description

Main
bsepsilon(\#)
local
harrell
order(\#)
knots(\#)
knotslist(knotvals)
knotsmat (matname)
distinct(\#)
replace
norescalevars
uniformknots
float
basis(stub|newvarlist)
rescale(stub|newvarlist)
specify the distance (\#) from the variable's boundary for B-spline knot placement; default is bsepsilon(0.01)
generate first-order polynomial spline variables centered around adjacent knots
place knots according to percentiles in Harrell (2001); only for rcs basis
use a spline basis of order \#; default is order (3)
use a spline basis function with \# knots
use knots specified in knotvals
use knots in matrix matname
set minimum number of distinct values required for variables used to construct splines to \#; default is distinct(10)
replace existing variables having the same names as the new basis and rescaled variables, if they exist
do not rescale variables before generating spline basis
place knots at evenly spaced points over the range of each variable; default is placement at percentiles
set type for generated variables to float instead of double
store elements of spline basis function using stub or newvarlist store rescaled values of variables using stub or newvarlist
collect is allowed; see [U] 11.1.10 Prefix commands.
fweights, aweights, and iweights are allowed; see [U] 11.1.6 weight.

\section*{Options}
\(\qquad\)
bsepsilon(\#) specifies the distance from the boundary of the variable where B-spline knots may be placed. The default is bsepsilon(0.01).
local specifies that a basis function for a first-order piecewise polynomial be generated with variables centered around adjacent knots. When splines are generated for only one variable and used in estimation, the regression coefficients measure slopes for the intervals defined by knots.
harrell specifies that knots be placed according to the percentiles recommended in Harrell (2001, 23). This option may be used only with basis rcs and when specifying 3 to 7 knots.
order(\#) specifies that a spline of order \# be used as the basis. order() may be 1, 2, or 3 for basis bspline and basis piecewise. order() may only be 3 for basis rcs. For basis linear or when the local option is specified, order() may only be 1 . The default is order (3), cubic splines.
knots(\#) specifies that a spline or B-spline basis function with \# interior knots be used. The number of knots must be an integer greater than or equal to 1 . The maximum number of knots is either 4,096 or two-thirds of the sample size, whichever is smaller. Also, the number of knots must be less than the number of distinct values in the variable used to generate the basis function. The default is knots(1) if exact knot values are not specified using knotslist() or knotsmat(). For basis rcs, the default is knots(3), and the number of knots must be 3 or greater.
knotslist (knotvals) specifies in knotvals the values of knots to be used for each variable. The knot values must be specified in the order of varlist, and a backslash ( \(\backslash\) ) must be used to separate knots for different variables. For example, if splines are generated for x1 and x2, the knots may be specified as knotslist (20 \(4060 \backslash 510\) 15) .
knotsmat (matname) specifies that, in matname, the knots for each variable be the values in each row. The number of knots should be the same for each variable, and there must be as many rows as there are variables. If rows of matname are not labeled with varnames, then rows are assumed to be in the order of varlist.
distinct(\#) specifies the minimum number of distinct values required for the variables used to construct the basis functions. Intuitively, using discrete variables for continuous interpolation is difficult to justify. \# specifies the number of distinct values necessary for a variable to be considered continuous. The default is distinct (10).
replace specifies that the variables generated to form the basis function be replaced. If basis (stub|newvarlist) or rescale (stub|newvarlist) are specified, the variables named with stub, or those listed in newvarlist, are replaced. Otherwise, variables with the default names are replaced.
norescalevars specifies that the original values of the variables in varlist be used to generate the basis function. By default, variables are first rescaled to \([0,1]\). norescalevars may not be used with basis bspline or basis linear.
uniformknots specifies that knots be placed at evenly spaced points over the range of each variable. The default is placement at percentiles of each of the specified variables.
float specifies that variables be generated as floats. Because of numerical precision and stability, the default is double.
basis (stub|newvarlist) specifies that the elements of the basis function be generated with the specified names.
When stub is specified, this prefix is used to generate enumerated variables for each element of the basis function.

When newvarlist is specified, variables with these names are generated for the elements of the basis function.
rescale (stub \(\mid\) newvarlist) specifies that the rescaled variables used to generate the basis function be stored with the specified names. This option applies only to basis piecewise and basis rcs.
When stub is specified, this prefix is used to generate enumerated variables for the rescaled variables.

When newvarlist is specified, variables with these names are generated for the rescaled variables.

\section*{Remarks and examples}
makespline generates new variables that form B-splines, piecewise polynomial splines, and restricted cubic splines from existing variables. Splines allow for different low-order polynomials in different regions of the original variables, and they approximate a smooth function by continuously connecting these low-order polynomials. Knots define the boundaries of the regions.

The standard piecewise polynomial variables created by makespline piecewise allow the functions to be linear, quadratic, or cubic in each region. makespline linear provides a convenient method for creating linear splines from the original variables, without rescaling. This is useful when you wish to directly interpret regression coefficients in the metric of the original variables. The terms in the standard piecewise polynomial spline function can be highly collinear and may be numerically unstable when used in estimation. B-splines, which can be created by makespline bspline, avoid this problem by creating orthogonal spline terms. For an introduction to piecewise polynomial splines and B-splines, see Piecewise polynomial splines and \(B\)-splines in [ R ] npregress intro. makespline rcs creates restricted cubic splines, also known as natural splines, in which the function is linear before the first knot, cubic between adjacent knots, and linear again after the last knot. This can improve performance in the tails over the standard cubic spline.

In addition to selecting the type of spline, makespline allows you to specify the location of knots-the locations where the function changes. You can specify the number of knots you wish to allow, and makespline will place the knots based on percentiles of the data or uniformly spaced across the range of values in the data. Alternatively, you can specify the exact values at which you wish the knots to be placed.

Regardless of the type of spline, we can refer to our newly created variables as a spline basis function. A basis is a collection of terms that can approximate a smooth function arbitrarily well. A basis function, such as one of the spline functions created by makespline, is a subset of the basis terms that can be used to approximate the mean function.

The basis function variables generated by makespline are useful for nonparametric and semiparametric estimation. For instance, makespline can be used when we want to fit models such as
\[
\begin{equation*}
\mathbf{y}=\mathbf{x}_{1} \beta+g\left(\mathbf{x}_{2}, \mathbf{x}_{3}, \ldots, \mathbf{x}_{k}\right)+\epsilon \tag{1}
\end{equation*}
\]

In the expression above, the outcome \(\mathbf{y}\), the covariates \(\mathbf{x}_{1}, \ldots, \mathbf{x}_{k}\), and the unobservable \(\epsilon\) are \(n \times 1\) vectors of covariates. The function \(g(\cdot)\) is unknown and \(\mathbf{x}_{1}\) enters the model linearly. These types of models are commonly used when we are interested in estimating the effect of \(\mathbf{x}_{1}\) on the mean of \(\mathbf{y}\). We are agnostic about the functional form in which the controls, \(\mathbf{x}_{2}, \ldots, \mathbf{x}_{k}\), enter the model, but to get a precise estimate of the effect of \(\mathbf{x}_{1}\), we need a reliable approximation of \(g(\cdot)\). We may use makespline to generate the basis functions that best approximate \(g(\cdot)\) and then use the basis functions to fit the model in (1).

For instance, we can generate basis functions with basis as the stub name:
```

makespline bspline x2-x5, basis(basis)

```

This would generate a third-order B-spline basis function for each of the variables in \(\mathrm{x} 2-\mathrm{x} 5\), with knots at the medians of \(x 2-x 5\). Each of the basis functions would consist of five variables; see Methods and formulas in [ R ] npregress series for details.

Once we have these basis functions, we can fit the model in (1) by typing
```

regress y x1 c.(basis*)\#\#c.(basis*)

```
where c.(basis*)\#\#c.(basis*) specifies that the terms in the basis functions be included in the model on their own as well as interacted with each of the other terms.

Above, we are assuming that we constructed a good approximation of the unknown function \(g(\cdot)\). We could go further and select from among these spline basis terms by using a technique such as lasso for prediction, described in [LASSO] lasso, or, if we are interested in inferences on estimated effects, a technique such as the partialing-out or double-selection lasso method, described in [LASSO] Lasso inference intro.

Let's say we are interested in getting a reliable estimate of the effect of \(x 1\) on the mean of the outcome. We would type
```

poregress y x1, controls(c.(basis*)\#\#c.(basis*))

```

The method used by the above command is partialing-out lasso, which selects from the elements of the basis function to provide an optimal approximation of \(g(\cdot)\) while accounting for the implied model selection error. The result is an estimate of the effect of \(x 1\) on the outcome with reliable standard errors.

Of course, the model does not have to be like the one presented in (1). It could be
\[
\mathbf{y}=g\left(\mathbf{x}_{1}, \mathbf{x}_{2}, \ldots, \mathbf{x}_{k}\right)+\epsilon
\]
or
\[
\mathbf{y}=g\left(\mathbf{x}_{1}\right)+g\left(\mathbf{x}_{2}\right)+\cdots+g\left(\mathbf{x}_{k}\right)+\epsilon
\]
or we might instead be interested in using the basis functions for visualization.

\section*{> Example 1: Generating and naming B-spline basis functions}

Below, we generate a third-order B-spline basis function with one knot placed at the median. The basis function is constructed from the variable price.
```

. sysuse auto

```
(1978 automobile data)
. makespline bspline price
The basis function consists of these five variables:
\begin{tabular}{ccccc}
\begin{tabular}{c}
. describe _* \\
Variable \\
name
\end{tabular} & \begin{tabular}{r} 
Storage \\
type
\end{tabular} & \begin{tabular}{c} 
Display \\
format
\end{tabular} & \begin{tabular}{c} 
Value \\
label
\end{tabular} & Variable label \\
\hline _bsp_1_1 & double \(\% 10.0 \mathrm{~g}\) & & B-spline basis term 1 for price \\
_bsp_1_2 & double \(\% 10.0 \mathrm{~g}\) & & B-spline basis term 2 for price \\
_bsp_1_3 & double \(\% 10.0 \mathrm{~g}\) & & B-spline basis term 3 for price \\
_bsp_1_4 & double \(\% 10.0 \mathrm{~g}\) & & B-spline basis term 4 for price \\
_bsp_1_5 & double \(\% 10.0 \mathrm{~g}\) & & B-spline basis term 5 for price
\end{tabular}

The default naming convention is to give the elements of the basis function a name that starts with _bsp and two subscripts. The first subscript enumerates the basis functions, and the second subscript enumerates the elements within the basis function. For example, if we created basis functions for two variables, we would obtain the following:
```

. makespline bspline price mpg, replace
. describe _*

| Variable <br> name | Storage <br> type | Display <br> format | Value <br> label | Variable label |
| :---: | :---: | :---: | :---: | :--- |
| _bsp_1_1 | double $\% 10.0 \mathrm{~g}$ |  | B-spline basis term 1 for price |  |
| _bsp_1_2 | double $\% 10.0 \mathrm{~g}$ |  | B-spline basis term 2 for price |  |
| _bsp_1_3 | double $\% 10.0 \mathrm{~g}$ |  | B-spline basis term 3 for price |  |
| _bsp_1_4 | double $\% 10.0 \mathrm{~g}$ |  | B-spline basis term 4 for price |  |
| _bsp_1_5 | double $\% 10.0 \mathrm{~g}$ |  | B-spline basis term 5 for price |  |
| _bsp_2_1 | double $\% 10.0 \mathrm{~g}$ |  | B-spline basis term 1 for mpg |  |
| _bsp_2_2 | double $\% 10.0 \mathrm{~g}$ |  | B-spline basis term 2 for mpg |  |
| _bsp_2_3 | double $\% 10.0 \mathrm{~g}$ |  | B-spline basis term 3 for mpg |  |
| _bsp_2_4 | double $\% 10.0 \mathrm{~g}$ |  | B-spline basis term 4 for mpg |  |
| _bsp_2_5 | double $\% 10.0 \mathrm{~g}$ |  | B-spline basis term 5 for mpg |  |

```

If we want to change the stub name _bsp to autobasis, we could use the basis() option as follows:


Alternatively, we could provide names for each of the variables that form a basis function. For example,
\begin{tabular}{|c|c|c|c|c|}
\hline Variable name & Storage type & Display format & \begin{tabular}{l}
Value \\
label
\end{tabular} & Variable labe \\
\hline mpg1 & double & \%10.0g & & B-spline basi \\
\hline mpg2 & double & \%10.0g & & B-spline basi \\
\hline mpg3 & double & \%10.0g & & B-spline basi \\
\hline mpg4 & double & \%10.0g & & B-spline basi \\
\hline mpg5 & double & \%10.0g & & B-spline basi \\
\hline
\end{tabular}
> Example 2: Generating and naming piecewise polynomial spline basis functions
Below, we generate a third-order piecewise polynomial spline with one knot at the median and show the variables we generated:
```

. makespline piecewise mpg
. describe *_sp*

| Variable <br> name | Storage <br> type | Display <br> format | Value <br> label | Variable label |
| :--- | ---: | :--- | :--- | :--- |
| _rs_sp_1 | double $\% 10.0 \mathrm{~g}$ |  | mpg rescaled to [0,1] <br> _sp_1_1 | double $\% 10.0 \mathrm{~g}$ |$\quad$| Piecewise polynomial basis term 1 |
| :---: |
| for mpg |

```

The only syntactical difference is that, after makespline, we specify piecewise instead of bspline to be the basis. makespline generates two variables in this case. They are the elements that are necessary to construct a basis function.

The default naming convention is to give the elements of the spline function a name that starts with _sp and has two subscripts. The first subscript enumerates the piecewise polynomial spline for a given variable, and the second subscript denotes the knot number. The rescaled variable starts with _rs_sp followed by a subscript denoting the element in the variable list.

Again, we may use a stub to modify the names that precede the subscripts, or we may specify a name for each new variable. Below, we also specified the names for the rescaled variables:
\begin{tabular}{|c|c|c|c|c|}
\hline Variable name & Storage type & Display format & \begin{tabular}{l}
Value \\
label
\end{tabular} & Variable label \\
\hline mpgsp & double & \(\% 10.0 \mathrm{~g}\) & & Piecewise polynomial basis term 1 for mpg \\
\hline mpgrs & double & \%10.0g & & mpg rescaled to [ 0,1\(]\) \\
\hline pricesp & double & \% 10.0 g & & Piecewise polynomial basis term 1 for price \\
\hline pricers & double & \% 10.0 g & & price rescaled to [0,1] \\
\hline
\end{tabular}

The logic behind the variables generated is that they consist of all the elements needed to approximate the unknown function \(g(\cdot)\) of the specified variables nonparametrically. In this case, a third-order piecewise polynomial spline approximation of \(g(\cdot)\) consists of the levels, square, and cube of mpgsp, mpgrs, pricesp, and pricers. Specifically, to include the fully interacted basis functions in a model, we would need to include the term below in our specification:
```

c.(c.mpgrs\#\#c.mpgrs\#\#c.mpgrs mpgsp)\#\#c.(c.pricers\#\#c.pricers\#\#c.pricers pricesp)

```
makespline simplifies this task by returning a local macro with the terms needed to fit \(g(\cdot)\). The local macro has the name \(r\) (regressors). In this case, it expands to the following:
```

. display "'r(regressors)'"
c.(c.mpgrs\#\#c.mpgrs\#\#c.mpgrs mpgsp)\#\#c.(c.pricers\#\#c.pricers\#\#c.pricers pricesp)

```

Note that when you generate basis functions for more than one variable, as we did above, \(r\) (regressors) fully interacts these basis functions. These fully interacted basis functions can be included when fitting a model by adding ' \(r\) (regressors)' to your list of covariates.

\section*{> Example 3: Using makespline in semiparametric estimation}

As we mentioned previously, basis functions are particularly useful for approximating unknown functions. For example, say we want to obtain the average marginal effect of \(x 1\) on the conditional mean of the continuous outcome \(y\). We have two controls, \(x 2\) and \(x 3\), but it is unclear whether they enter the model linearly or with another functional form.

To approximate the unknown function of x 2 and x 3 , we construct two B -spline basis functions with eight knots each. We use the simulated dataset and then the makespline bspline command:
```

. use https://www.stata-press.com/data/r18/splines, clear
(Simulated data)
. makespline bspline x2 x3, knots(8)

```

This yields basis functions with 12 elements. Once you fully interact the two basis functions, you get 168 regressors. Using all of them to approximate the unknown function would not be a sound idea. Thus, we will use poregress to perform partialing-out lasso linear regression. This estimator will select from the 168 covariates to provide a good approximation to the unknown function and at the same time provide a reliable estimate of the marginal effect of interest.

Rather than interact the basis terms manually, we can simply refer to the macro r(regressors), which contains the full interaction of the basis functions:


We obtain an average marginal effect of 2.95 .
A researcher does not know the true value of the effect; however, we do. These are simulated data. The model is given by
\[
\mathbf{y}=3 \mathbf{x}_{1}+3 \sin \left\{3\left(\mathbf{x}_{2}-\mathbf{x}_{3}\right)\right\}+\epsilon
\]

The unknown function of \(\mathbf{x}_{2}\) and \(\mathbf{x}_{3}\) is complex, yet we obtained a precise estimate of the average marginal effect.

\section*{> Example 4: Using makespline for estimation and graphing}

It is common to use linear splines to create a graph after estimation. The knots of a regressor define a piecewise polynomial that can be visualized conditional on the values of other covariates.

Below, we study the effect of mileage in miles per gallon ( mpg ) on car prices (price). We regress price on mpg, three linear polynomial basis terms defined by knots at the quartiles of mpg, and a dummy variable, foreign (1 if cars are foreign).

We first generate the variables that form the polynomial basis and then fit the regression.
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline \multicolumn{6}{|l|}{\begin{tabular}{l}
. makespline linear mpg, knots(3) basis(mpg) \\
. regress price mpg mpg_* i.foreign
\end{tabular}} & \\
\hline Source & SS & df & MS & Number of obs & = & 74 \\
\hline & & & & F (5, 68) & & 13.49 \\
\hline Model & 316201619 & 5 & 63240323.8 & Prob > F & = & 0.0000 \\
\hline Residual & 318863777 & 68 & 4689173.19 & R -squared & & 0.4979 \\
\hline & & & & Adj R-squared & & 0.4610 \\
\hline Total & 635065396 & 73 & 8699525.97 & Root MSE & = & 2165.4 \\
\hline
\end{tabular}
\begin{tabular}{r|rlrrrr}
\hline price & Coefficient & Std. err. & t & \(\mathrm{P}>|\mathrm{t}|\) & [95\% conf. interval] \\
\hline mpg & -1330.299 & 213.0425 & -6.24 & 0.000 & -1755.419 & -905.1798 \\
mpg_1_1 & 1698.953 & 622.5153 & 2.73 & 0.008 & 456.7432 & 2941.163 \\
mpg_1_2 & -622.9298 & 651.5686 & -0.96 & 0.342 & -1923.115 & 677.2551 \\
mpg_1_3 & 139.4188 & 277.0762 & 0.50 & 0.616 & -413.4783 & 692.3158 \\
& & & & & & \\
foreign & & & & & & \\
Foreign & 1676.381 & 609.4723 & 2.75 & 0.008 & 460.1983 & 2892.564 \\
_cons & 28796.47 & 3449.408 & 8.35 & 0.000 & 21913.28 & 35679.65 \\
\hline
\end{tabular}

The regression line for our model is given by the following command:
```

generate double xb = _b[_cons] + _b[1.foreign]*foreign + ///
mpg*_b[mpg] + (mpg>18)*(mpg-18)*_b[mpg_1_1] + ///
(mpg>20)*(mpg-20)*_b[mpg_1_2] + ///
(mpg>25)*(mpg-25)*_b[mpg_1_3]

```

The effect of mpg changes at the knots. If mpg is less than or equal to 18 , it is \(\_\mathrm{b}[\mathrm{mpg}]\); if it is greater than 18 but less than or equal to 20 , it is ( \(\_\)b \([\mathrm{mpg}]+\ldots \mathrm{b}\left[\mathrm{mpg} \_1 \_1\right]\) ) ; if it is greater than 20 but less than or equal to 25 , it is ( \(\_\mathrm{b}[\mathrm{mpg}]+\ldots \mathrm{b}\left[\mathrm{mpg}-1 \_1\right]+\ldots \mathrm{b}\left[\mathrm{mpg}-1 \_2\right]\) ) ; and if it is greater than 25 , it is ( \(\quad \mathrm{b}[\mathrm{mpg}]+\ldots \mathrm{b}\left[\mathrm{mpg} \_1 \_1\right]+\ldots \mathrm{b}\left[\mathrm{mpg}-1 \_2\right]+\ldots \mathrm{b}\left[\mathrm{mpg}-1 \_3\right]\) ).

We can plot regression lines for foreign and domestic cars. We first generate the predictions for foreign and domestic cars.
```

. generate xb_domestic = _b[_cons] + mpg*_b[mpg]
> + (mpg>18)*(mpg-18)*_b[mpg_1_1]
> + (mpg>20)*(mpg-20)*_b[mpg_1_2]
> + (mpg>25)*(mpg-25)*_b[mpg_1_3]
. generate xb_foreign = _b[_cons] + _b[1.foreign] + mpg*_b[mpg]
+ (mpg>18)*(mpg-18)*_b[mpg_1_1]
+(mpg>20)*(mpg-20)*_b[mpg_1_2]
+ (mpg>25)*(mpg-25)*_b[mpg_1_3]

```

Then we plot both regression lines referencing the placement of the knots with vertical lines. In the graph, we also include the values of the dependent variable. We can inspect graphically how the effect of mpg differs across the regions defined by the knots.
```

twoway line xb_domestic mpg,
lcolor(blue) lpattern(dash) sort ||
line xb_foreign mpg,
lcolor(red) lpatter(dash_dot) sort ||
scatter price mpg if foreign==0, mcolor(blue%30) |
scatter price mpg if foreign==1, mcolor(red%30)
xline(18 20 25)
title(Fitted values for domestic and foreign cars)
subtitle(Spline regression with knots at quartiles of mpg)

```

Fitted values for domestic and foreign cars


\section*{Stored results}
makespline stores the following in \(r()\) :
Scalars
\(r\) (N_knots) number of knots
\(r\) (local) \(\quad 1\) if local was specified, 0 otherwise
r(bsepsilon)
distance from variable's boundary for B-spline knot placement
Macros
r(basis)
r(regressors)
r(basisnames\#)
\(r\) (wtype)
\(r\) (wexp)
Matrices
\(r\) (minmax) minimum and maximum of all variables
\(r\) (knots) matrix of knots

\section*{Methods and formulas}

See Methods and formulas in \([\mathrm{R}]\) npregress series for piecewise polynomial spline and B-spline computation.

When the local option is specified, let \(V_{i}, i=1, \ldots, n\), be the variables to be created; \(k_{i}\), \(i=1, \ldots, n-1\), be the corresponding knots; and \(\mathcal{V}\) be the original variable rescaled to be in \([0,1]\). Then
\[
\begin{aligned}
V_{1} & =\min \left(\mathcal{V}, k_{1}\right) \\
V_{i} & =\max \left\{\min \left(\mathcal{V}, k_{i}\right), k_{i-1}\right\}-k_{i-1} \quad i=2, \ldots, n-1 \\
V_{n} & =\max \left(\mathcal{V}, k_{n-1}\right)-k_{n-1}
\end{aligned}
\]

When the res basis is specified, let \(k_{i}, i=1, \ldots, n\), be the knot values; \(V_{i}, i=1, \ldots, n-1\), be the variables to be created; and \(\mathcal{V}\) be the original variable rescaled to be in \([0,1]\). Then
\[
\begin{aligned}
& V_{1}= \mathcal{V} \\
& V_{i+1}= \frac{\left(\mathcal{V}-k_{i}\right)_{+}^{3}-\left(k_{n}-k_{n-1}\right)^{-1}\left\{\left(\mathcal{V}-k_{n-1}\right)_{+}^{3}\left(k_{n}-k_{i}\right)-\left(\mathcal{V}-k_{n}\right)_{+}^{3}\left(k_{n-1}-k_{i}\right)\right\}}{\left(k_{n}-k_{1}\right)^{2}} \\
& \quad i=1, \ldots, n-2
\end{aligned}
\]
where
\[
(u)_{+}=\left\{\begin{array}{l}
u, \text { if } u>0 \\
0, \text { if } u \leq 0
\end{array}\right.
\]

When the harrell option is specified, the knots are placed using the percentiles recommended in Harrell \((2001,23)\). These percentiles are based on the chosen number of knots as follows:
\begin{tabular}{c|lllllll}
\(\begin{array}{c}\text { No. } \\
\text { of knots }\end{array}\) & \multicolumn{7}{|c}{ Percentiles }
\end{tabular}\(]\)\begin{tabular}{cllllll} 
\\
\hline 3 & 10 & 50 & 90 & & & \\
4 & 5 & 35 & 65 & 95 & & \\
5 & 5 & 27.5 & 50 & 72.5 & 95 & \\
6 & 5 & 23 & 41 & 59 & 77 & 95 \\
7 & 2.5 & 18.33 & 34.17 & 50 & 65.83 & 81.67 \\
\hline
\end{tabular}

\section*{References}

Chetverikov, D., D. Kim, and D. Wilhelm. 2018. Nonparametric instrumental-variable estimation. Stata Journal 18: 937-950.
de Boor, C. 2001. A Practical Guide to Splines. Rev. ed. New York: Springer.
Eubank, R. L. 1999. Nonparametric Regression and Spline Smoothing. 2nd ed. New York: Dekker.
Hansen, B. E. 2009. University of Wisconsin-Madison, ECON 718, NonParametric Econometrics, Spring 2009, course notes. Last visited on 2019/01/15. https://www.ssc.wisc.edu/~bhansen/718/718.htm.
__. 2018. Econometrics. https://www.ssc.wisc.edu/~bhansen/econometrics/Econometrics.pdf.
Harrell, F. E., Jr. 2001. Regression Modeling Strategies: With Applications to Linear Models, Logistic Regression, and Survival Analysis. New York: Springer.
Li, Q., and J. S. Racine. 2007. Nonparametric Econometrics: Theory and Practice. Princeton, NJ: Princeton University Press.

Schoenberg, I. J., ed. 1969. Approximations with Special Emphasis on Spline Functions. New York: Academic Press.
Schumaker, L. L. 2007. Spline Functions: Basic Theory. 3rd ed. Cambridge: Cambridge University Press.

\section*{Also see}
[R] npregress series - Nonparametric series regression
\([\mathrm{R}]\) npregress series postestimation - Postestimation tools for npregress series
[R] npregress intro - Introduction to nonparametric regression
[R] kdensity - Univariate kernel density estimation
[R] Ipoly - Kernel-weighted local polynomial smoothing

\section*{Title}
margins - Marginal means, predictive margins, and marginal effects

Description
Options
References

Quick start
Remarks and examples
Also see

Menu
Stored results

Syntax
Methods and formulas

\section*{Description}

Margins are statistics calculated from predictions of a previously fit model at fixed values of some covariates and averaging or otherwise integrating over the remaining covariates.

The margins command estimates margins of responses for specified values of covariates and presents the results as a table.

Capabilities include estimated marginal means, least-squares means, average and conditional marginal and partial effects (which may be reported as derivatives or as elasticities), average and conditional adjusted predictions, and predictive margins.

\section*{Quick start}

Estimated marginal means (least-squares means)
Estimated marginal mean of \(y\) for each level of a after anova y a\#\#b margins a, asbalanced

Estimated marginal mean of \(y\) for each level of the interaction of \(a\) and \(b\) after anova \(y\) a\#\#b\#\#c margins a\#b, asbalanced

Estimated marginal means of \(\mathrm{y} 1, \mathrm{y} 2\), and y 3 for each level of a after manova y1 y2 y3 \(=\mathrm{a} \# \# \mathrm{~b}\) margins a, asbalanced

\section*{Adjusted means and adjusted predictions}

Adjusted mean of y for each level of a when x is at its mean after regress y i.a x margins a, atmeans

Same as above, but set x to 10 rather than to its mean
margins a, at ( \(\mathrm{x}=10\) )
Same as above, and also report adjusted means for \(\mathrm{x}=20, \mathrm{x}=30\), and \(\mathrm{x}=40\)
margins \(a\), at \((x=(10(10) 40))\)
Adjusted predicted probability of \(y=1\) for each level of a when \(x\) is at its mean after
logit y a\#\#c.x
margins a, atmeans
Adjusted predicted probability for each level of the interaction of a and b , holding x at 25 , after logit y a\#\#b\#\#c.x
margins \(a \# b\), at ( \(x=25\) )
Adjusted prediction for each level of a when \(\mathrm{x}=25\) and \(\mathrm{b}=1\) margins \(a\), at \((x=25 b=1)\)

Predictive margins and potential-outcome means
Overall predictive margin, the average predicted probability of \(\mathrm{y}=1\), after logit y a\#\#b x1 x2 margins

Predictive margins (potential-outcome means) for each level of a margins a

Predictive margins for a , when x 1 is set to \(10,20,30\), and 40
margins a, at (x1=(10(10)40))
Predictive margins for levels of the interaction of \(a\) and \(b\) margins a\#b

Predictive margins for a for all combinations of \(\mathrm{x} 1=10,20,30\) and \(\mathrm{x} 2=50,100,150\)
margins a, at \((x 1=(10(10) 30) x 2=(50(50) 150))\)
Predictive margins for a , first for \(\mathrm{x} 1=10,20,30\) with x 2 at its observed values, then for \(\mathrm{x} 2=50\), 100,150 with x1 at its observed values
```

    margins a, at(x1=(10(10)30)) at(x2=(50(50)150))
    ```

Predictive margins for a after svy:logit y a\#\#b x1 x2
margins a, vce(unconditional)
Average predicted probabilities of \(y=1, y=2, \ldots\) after mlogit \(y\) x1 x2 i.a margins

Predictive margins for each level of a for each outcome of \(y\)
margins a

\section*{Average marginal effects and average partial effects}

Average marginal effect of x 1 on the predicted probability of \(\mathrm{y}=1\) after probit \(\mathrm{y} \mathrm{c} . \mathrm{x} 1 \# \# \mathrm{c} . \mathrm{x} 2 \# \# \mathrm{a}\) with continuous \(x 1\) and \(x 2\) and binary a
```

margins, dydx(x1)

```

Average marginal effect (average partial effect) of binary a margins, dydx(a)

Average marginal effect of \(x 1\) when \(x 2\) is set to \(10,20,30\), and 40
margins, dydx(x1) at(x2=(10(10)40))
Average marginal effect of x 1 when a is set to 0 and then to 1
margins a, dydx(x1)
Average marginal effect of each variable in the model
margins, dydx(*)
Average marginal effect of all variables on the truncated expected value of \(y, e(0,\).\() , after\) tobit y x1 x2 x3, ll(0)
```

margins, dydx(*) predict(e(0,.))

```

Same as above, and report marginal effects for censored expected value of \(y\), ystar \((0,\).\() , and for\) the linear prediction, xb
margins, dydx(*) predict(e(0,.)) predict(ystar(0,.)) predict(xb)

\section*{Conditional marginal effects and conditional partial effects}

Marginal effect of \(x 1\) on the predicted probability of \(y=1\), setting all variables to their means, after probit y c.x1\#\#c.x2\#\#a with continuous x1 and x2 and binary a margins, dydx(x1) atmeans

Marginal effect (partial effect) of a when all variables are set to their means margins, dydx(a) atmeans

Marginal effect of \(x 1\) when \(a=0\) and \(x 1\) and \(x 2\) are set to their means margins, \(d y d x(x 1)\) at (a=0 (mean) \(x 1 \mathrm{x} 2\) )

Same as above
margins, dydx(x1) at (a=0) atmeans
Marginal effect of x 1 when for all possible combinations of \(\mathrm{a}=0,1, \mathrm{x} 1=50,100\), and \(\mathrm{x} 2=10,20,30,40\)
margins \(a, \operatorname{dydx}(\mathrm{x} 1)\) at \((\mathrm{x} 1=(50100) \mathrm{x} 2=(10(10) 40)\) )
Marginal effects of \(x 1, x 2\), and a with all variables set to their means
margins, \(d y d x(*)\) atmeans

\section*{Menu}

Statistics \(>\) Postestimation

\section*{Syntax}
\[
\text { margins }[\text { marginlist }][\text { if }][\text { in }][\text { weight }][\text {, response_options options }]
\]
where marginlist is a list of factor variables or interactions that appear in the current estimation results. The variables may be typed with or without the i. prefix, and you may use any factor-variable syntax:
```

. margins i.sex i.group i.sex\#i.group
. margins sex group sex\#i.group
. margins sex\#\#group

```
response_options Description

Main
predict(pred_opt) expression(pnl_exp) dydx (varlist) eyex (varlist)
dyex (varlist)
eydx (varlist)
continuous
estimate margins for predict, pred_opt
estimate margins for pnl_exp
estimate marginal effect of variables in varlist
estimate elasticities of variables in varlist
estimate semielasticity— \(d(\widehat{y}) / d(\ln x)\)
estimate semielasticity- \(d(\ln \widehat{y}) / d(x)\)
treat factor-level indicators as continuous

\section*{Main}
grand

At
at (atspec)
atmeans
asbalanced
if/in/over
over (varlist)
subpop (subspec)
Within
within(varlist)
Contrast
contrast_options
Pairwise comparisons
pwcompare_options
SE
vce(delta)
vce(unconditional)
nose
Advanced
noweights
noesample
emptycells (empspec)
estimtolerance(tol)
noestimcheck
force
chainrule
nochainrule
Reporting
level (\#)
mcompare(method)
noatlegend
post
display_options
df (\#)
add the overall margin; default if no marginlist
estimate margins at specified values of covariates estimate margins at the means of covariates treat all factor variables as balanced
estimate margins at unique values of varlist
estimate margins for subpopulation
estimate margins at unique values of the nesting factors in varlist
any options documented in \([R]\) margins, contrast
any options documented in [R] margins, pwcompare
estimate SEs using delta method; the default estimate SEs allowing for sampling of covariates do not estimate SEs
ignore weights specified in estimation
do not restrict margins to the estimation sample treatment of empty cells for balanced factors specify numerical tolerance used to determine estimable functions; default is estimtolerance (1e-5)
suppress estimability checks
estimate margins despite potential problems
use the chain rule when computing derivatives
do not use the chain rule
set confidence level; default is level (95)
adjust for multiple comparisons; default is mcompare (noadjust)
suppress legend of fixed covariate values
post margins and their VCE as estimation results
control columns and column formats, row spacing, line width and factor-variable labeling
use \(t\) distribution with \# degrees of freedom for computing \(p\)-values and confidence intervals
noadjust
bonferroni [adjustall] sidak [adjustall]
scheffe
do not adjust for multiple comparisons; the default
Bonferroni's method; adjust across all terms
Šidák's method; adjust across all terms
Scheffé's method

Time-series operators are allowed if they were used in the estimation.
See at () under Options for a description of atspec.
collect is allowed; see [U] 11.1.10 Prefix commands.
fweights, aweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
\(\mathrm{df}(\#)\) does not appear in the dialog box.

\section*{Options}

Warning: The option descriptions are brief and use jargon. Skip to Remarks and examples if you are reading about margins for the first time.

predict (pred_opt) and expression(pnl_exp) are mutually exclusive; they specify the response. If neither is specified, the response will be the default prediction that would be produced by predict after the underlying estimation command. Some estimation commands, such as mlogit, document a different default prediction for margins than for predict.
predict (pred_opt) specifies the option(s) to be specified with the predict command to produce the variable that will be used as the response. After estimation by logistic, you could specify predict( xb ) to obtain linear predictions rather than the predict command's default, the probabilities.
Multiple predict() options can be specified to compute margins of multiple predictions simultaneously.
expression(pnl_exp) specifies the response as an expression. See [R] predictnl for a full description of pnl_exp. After estimation by logistic, you might specify expression ( \(\exp (\) predict \((x b))\) ) to use relative odds rather than probabilities as the response. For examples, see Example 12: Margins of a specified expression.
dydx (varlist), eyex (varlist), dyex (varlist), and eydx (varlist) request that margins report derivatives of the response with respect to varlist rather than on the response itself. eyex(), dyex(), and eydx() report derivatives as elasticities; see Expressing derivatives as elasticities.
continuous is relevant only when one of \(d y d x()\) or eydx () is also specified. It specifies that the levels of factor variables be treated as continuous; see Derivatives versus discrete differences. This option is implied if there is a single-level factor variable specified in dydx() or eydx().
grand specifies that the overall margin be reported. grand is assumed when marginlist is empty.

At
at (atspec) specifies values for covariates to be treated as fixed.
at (age \(=20\) ) fixes covariate age to the value specified. at () may be used to fix continuous or factor covariates.
at (age \(=20\) sex=1) simultaneously fixes covariates age and sex at the values specified.
at (age \(=(20304050)\) ) fixes age first at 20 , then at \(30, \ldots\) margins produces separate results for each specified value.
at \((\operatorname{age}=(20(10) 50))\) does the same as at \((\operatorname{age}=(20304050))\); that is, you may specify a numlist.
at ((mean) age (median) distance) fixes the covariates at the summary statistics specified. at ( (p25) _all) fixes all covariates at their 25th percentile values. See Syntax of at() for the full list of summary-statistic modifiers.
at ((mean) _all (median) \(\times \mathrm{x} 2=1.2 \mathrm{z}=\left(\begin{array}{ll}1 & 2\end{array}\right)\) ) is processed from general to specific, with settings for named covariates overriding general settings specified via _all. Thus, all covariates are fixed at their means except for \(x\) (fixed at its median), x2 (fixed at 1.2), and \(z\) (fixed first at 1 , then at 2 , and finally at 3 ).
at ((means) _all (asobserved) x2) is a convenient way to set all covariates except \(x 2\) to the mean.

Multiple at () options can be specified, and each will produce a different set of margins.
See Syntax of at() for more information.
atmeans specifies that covariates be fixed at their means and is shorthand for at ((mean) _all). atmeans differs from at ((mean) _all) in that atmeans will affect subsequent at () options. For instance,
```

. margins ..., atmeans at((p25) x) at((p75) x)

```
produces two sets of margins with both sets evaluated at the means of all covariates except x . asbalanced is shorthand for at ((asbalanced) _factor) and specifies that factor covariates be evaluated as though there were an equal number of observations in each level; see Obtaining margins as though the data were balanced. asbalanced differs from at ((asbalanced) _factor) in that asbalanced will affect subsequent at () options in the same way as atmeans does.
if/in/over
over (varlist) specifies that separate sets of margins be estimated for the groups defined by varlist.
The variables in varlist must contain nonnegative integer (or missing) values. The variables need not be covariates in your model. When over() is combined with the vce(unconditional) option, each group is treated as a subpopulation; see [SVY] Subpopulation estimation.
\(\operatorname{subpop}([\) varname ] [if]) is intended for use with the vce (unconditional) option. It specifies that margins be estimated for the single subpopulation identified by the indicator variable or by the if expression or by both. Zero or missing indicates that the observation be excluded; nonzero or nonmissing, that it be included. See [SVY] Subpopulation estimation for why subpop() is preferred to if expressions and in ranges when also using vce(unconditional). If subpop() is used without vce(unconditional), it is treated merely as an additional if qualifier.

\section*{Within}
within (varlist) allows for nested designs. varlist contains the nesting variable(s) over which margins are to be estimated. See Obtaining margins with nested designs. As with over (varlist), when within(varlist) is combined with vce(unconditional), each level of the variables in varlist is treated as a subpopulation.

Pairwise comparisons
pwcompare_options are any of the options documented in \([\mathrm{R}]\) margins, pwcompare.
vce(delta) and vce(unconditional) specify how the VCE and, correspondingly, standard errors are calculated.
vce (delta) is the default. The delta method is applied to the formula for the response and the VCE of the estimation command. This method assumes that values of the covariates used to calculate the response are given or, if all covariates are not fixed using at (), that the data are given.
vce(unconditional) specifies that the covariates that are not fixed be treated in a way that accounts for their having been sampled. The VCE is estimated using the linearization method. This method allows for heteroskedasticity or other violations of distributional assumptions and allows for correlation among the observations in the same manner as vce (robust) and vce (cluster ...), which may have been specified with the estimation command. This method also accounts for complex survey designs if the data are svyset. See Obtaining margins with survey data and representative samples. When you use complex survey data, this method requires that the linearized variance estimation method be used for the model. See [SVY] svy postestimation for an example of margins with replication-based methods.
nose suppresses calculation of the VCE and standard errors. See Requirements for model specification for an example of the use of this option.

\section*{Advanced}
noweights specifies that any weights specified on the previous estimation command be ignored by margins. By default, margins uses the weights specified on the estimator to average responses and to compute summary statistics. If weights are specified on the margins command, they override previously specified weights, making it unnecessary to specify noweights. The noweights option is not allowed after svy: estimation when the vce (unconditional) option is specified.

For multilevel models, such as meglm, the default behavior is to construct a single weight value for each observation by multiplying the corresponding multilevel weights within the given observation.
noesample specifies that margins not restrict its computations to the estimation sample used by the previous estimation command. See Example 15: Margins evaluated out of sample.
With the default delta-method VCE, noesample margins may be estimated on samples other than the estimation sample; such results are valid under the assumption that the data used are treated as being given.

You can specify noesample and vce(unconditional) together, but if you do, you should be sure that the data in memory correspond to the original e (sample). To show that you understand that, you must also specify the force option. Be aware that making the vce(unconditional) calculation on a sample different from the estimation sample would be equivalent to estimating the coefficients on one set of data and computing the scores used by the linearization on another set; see [P] _robust.
emptycells(strict) and emptycells(reweight) are relevant only when the asbalanced option is also specified. emptycells() specifies how empty cells are handled in interactions involving factor variables that are being treated as balanced; see Obtaining margins as though the data were balanced.
emptycells(strict) is the default; it specifies that margins involving empty cells be treated as not estimable.
emptycells(reweight) specifies that the effects of the observed cells be increased to accommodate any missing cells. This makes the margin estimable but changes its interpretation. emptycells(reweight) is implied when the within() option is specified.
estimtolerance (tol) specifies the numerical tolerance used to determine estimable functions. The default is estimtolerance ( \(1 \mathrm{e}-5\) ).

A linear combination of the model coefficients \(z\) is found to be not estimable if
\[
\text { mreldif }(z, z \times H)>t o l
\]
where \(H\) is defined in Methods and formulas.
noestimcheck specifies that margins not check for estimability. By default, the requested margins are checked and those found not estimable are reported as such. Nonestimability is usually caused by empty cells. If noestimcheck is specified, estimates are computed in the usual way and reported even though the resulting estimates are manipulable, which is to say they can differ across equivalent models having different parameterizations. See Estimability of margins.
force instructs margins to proceed in some situations where it would otherwise issue an error message because of apparent violations of assumptions. Do not be casual about specifying force. You need to understand and fully evaluate the statistical issues. For an example of the use of force, see Using margins after the estimates use command.
chainrule and nochainrule specify whether margins uses the chain rule when numerically computing derivatives. You need not specify these options when using margins after any official Stata estimator, margins will choose the appropriate method automatically.
Specify nochainrule after estimation by a community-contributed command. We recommend using nochainrule, even though chainrule is usually safe and is always faster. nochainrule is safer because it makes no assumptions about how the parameters and covariates join to form the response.
nochainrule is implied when the expression() option is specified.
level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals. mcompare (method) specifies the method for computing \(p\)-values and confidence intervals that account for multiple comparisons within a factor-variable term.
Most methods adjust the comparisonwise error rate, \(\alpha_{c}\), to achieve a prespecified experimentwise error rate, \(\alpha_{e}\).
mcompare(noadjust) is the default; it specifies no adjustment.
\[
\alpha_{c}=\alpha_{e}
\]
mcompare(bonferroni) adjusts the comparisonwise error rate based on the upper limit of the Bonferroni inequality
\[
\alpha_{e} \leq m \alpha_{c}
\]
where \(m\) is the number of comparisons within the term.
The adjusted comparisonwise error rate is
\[
\alpha_{c}=\alpha_{e} / m
\]
mcompare (sidak) adjusts the comparisonwise error rate based on the upper limit of the probability inequality
\[
\alpha_{e} \leq 1-\left(1-\alpha_{c}\right)^{m}
\]
where \(m\) is the number of comparisons within the term.
The adjusted comparisonwise error rate is
\[
\alpha_{c}=1-\left(1-\alpha_{e}\right)^{1 / m}
\]

This adjustment is exact when the \(m\) comparisons are independent.
mcompare (scheffe) controls the experimentwise error rate using the \(F\) or \(\chi^{2}\) distribution with degrees of freedom equal to the rank of the term.
mcompare (method adjustall) specifies that the multiple-comparison adjustments count all comparisons across all terms rather than performing multiple comparisons term by term. This leads to more conservative adjustments when multiple variables or terms are specified in marginslist. This option is compatible only with the bonferroni and sidak methods.
noatlegend specifies that the legend showing the fixed values of covariates be suppressed.
post causes margins to behave like a Stata estimation (e-class) command. margins posts the vector of estimated margins along with the estimated variance-covariance matrix to e(), so you can treat the estimated margins just as you would results from any other estimation command. For example, you could use test to perform simultaneous tests of hypotheses on the margins, or you could use lincom to create linear combinations. See Example 10: Testing margins-contrasts of margins.
display_options: noci, nopvalues, vsquish, nofvlabel, fvwrap(\#), fvwrapon(style),
cformat (\% frit), pf ormat (\%fmt), sformat (\%fmt), and nolstretch.
noci suppresses confidence intervals from being reported in the coefficient table.
nopvalues suppresses \(p\)-values and their test statistics from being reported in the coefficient table.
vsquish specifies that the blank space separating factor-variable terms or time-series-operated variables from other variables in the model be suppressed.
nofvlabel displays factor-variable level values rather than attached value labels. This option overrides the fvlabel setting; see \([R]\) set showbaselevels.
fvwrap(\#) allows long value labels to wrap the first \# lines in the coefficient table. This option overrides the fvwrap setting; see [R] set showbaselevels.
fvwrapon(style) specifies whether value labels that wrap will break at word boundaries or break based on available space.
fvwrapon(word), the default, specifies that value labels break at word boundaries.
fvwrapon(width) specifies that value labels break based on available space.
This option overrides the fvwrapon setting; see [R] set showbaselevels.
cformat (\% fint) specifies how to format margins, standard errors, and confidence limits in the table of estimated margins.
pformat (\% fmt) specifies how to format \(p\)-values in the table of estimated margins.
sformat (\%fmt) specifies how to format test statistics in the table of estimated margins.
nolstretch specifies that the width of the table of estimated margins not be automatically widened to accommodate longer variable names. The default, lstretch, is to automatically widen the table of estimated margins up to the width of the Results window. Specifying lstretch or nolstretch overrides the setting given by set lstretch. If set lstretch has not been set, the default is lstretch. nolstretch is not shown in the dialog box.

The following option is available with margins but is not shown in the dialog box:
\(\mathrm{df}(\#)\) specifies that the \(t\) distribution with \# degrees of freedom be used for computing \(p\)-values and confidence intervals. The default typically is to use the standard normal distribution. However, if the estimation command computes the residual degrees of freedom (e(df_r)) and predict (xb) is specified with margins, the default is to use the \(t\) distribution with e(df_r) degrees of freedom.

\section*{Remarks and examples}

Remarks are presented under the following headings:
Introduction
Obtaining margins of responses
Example 1: A simple case after regress
Example 2: A simple case after logistic
Example 3: Average response versus response at average
Example 4: Multiple margins from one command
Example 5: Margins with interaction terms
Example 6: Margins with continuous variables
Example 7: Margins of continuous variables
Example 8: Margins of interactions
Example 9: Decomposing margins
Example 10: Testing margins-contrasts of margins
Example 11: Margins of a specified prediction
Example 12: Margins of a specified expression
Example 13: Margins with multiple outcomes (responses)
Example 14: Margins with multiple equations
Example 15: Margins evaluated out of sample
Obtaining margins of derivatives of responses (a.k.a. marginal effects)
Use at() freely, especially with continuous variables
Expressing derivatives as elasticities
Derivatives versus discrete differences
Example 16: Average marginal effect (partial effects)
Example 17: Average marginal effect of all covariates
Example 18: Evaluating marginal effects over the response surface
Obtaining margins with survey data and representative samples
Example 19: Inferences for populations, margins of response
Example 20: Inferences for populations, marginal effects
Example 21: Inferences for populations with svyset data
Standardizing margins
Obtaining margins as though the data were balanced
Balancing using asbalanced
Balancing by standardization
Balancing nonlinear responses
Treating a subset of covariates as balanced
Using fvset design
Balancing in the presence of empty cells
Obtaining margins with nested designs
Introduction to nested designs
Margins with nested designs as though the data were balanced
Coding of nested designs
Special topics
Requirements for model specification
Estimability of margins
Manipulability of tests
Using margins after the estimates use command
Syntax of at()
Estimation commands that may be used with margins
Video examples
Glossary

\section*{Introduction}
margins is a postestimation command, a command for use after you have fit a model using an estimation command such as regress or logistic, or using almost any other estimation command.
margins estimates and reports margins of responses and margins of derivatives of responses, also known as marginal effects. A margin is a statistic based on a fitted model in which some of or all the covariates are fixed. Marginal effects are changes in the response for change in a covariate, which can be reported as a derivative, elasticity, or semielasticity.

For a brief overview of margins, see Williams (2012).

\section*{Obtaining margins of responses}

What we call margins of responses are also known as predictive margins, adjusted predictions, and recycled predictions. When applied to balanced data, margins of responses are also called estimated marginal means and least-squares means.

A margin is a statistic based on a fitted model calculated over a dataset in which some of or all the covariates are fixed at values different from what they really are. For instance, after a linear regression fit on males and females, the marginal mean (margin of mean) for males is the predicted mean of the dependent variable, where every observation is treated as if it represents a male; thus, those observations that in fact do represent males are included, as well as those observations that represent females. The marginal mean for females would be similarly obtained by treating all observations as if they represented females.

In making the calculation, sex is treated as male or female everywhere it appears in the model. The model might be
```

. regress y age bp i.sex sex\#c.age sex\#c.bp

```
and then, in making the marginal calculation of the mean for males and females, margins not only accounts for the direct effect of i.sex but also for the indirect effects of sex\#c.age and sex\#c.bp.

The response being margined can be any statistic produced by \([\mathrm{R}]\) predict, or any expression of those statistics.

Standard errors are obtained by the delta method, at least by default. The delta method assumes that the values at which the covariates are evaluated to obtain the marginal responses are fixed. When your sample represents a population, whether you are using svy or not (see [SVY] svy), you can specify margins' vce(unconditional) option and margins will produce standard errors that account for the sampling variability of the covariates. Some researchers reserve the term predictive margins to describe this.

The best way to understand margins is to see some examples. You can run the following examples yourself if you type
```

. use https://www.stata-press.com/data/r18/margex
(Artificial data for margins)

```

\section*{Example 1: A simple case after regress}
```

. regress y i.sex i.group
(output omitted)
. margins sex
Predictive margins Number of obs = 3,000
Model VCE: OLS
Expression: Linear prediction, predict()

```
\begin{tabular}{c|rrrrrr}
\hline & \multicolumn{4}{|c}{\begin{tabular}{c} 
Delta-method \\
Margin \\
std. err.
\end{tabular}} & t & \(\mathrm{P}>|\mathrm{t}|\) \\
\hline sex & & & & & [95\% conf. interval] \\
Male & 60.56034 & .5781782 & 104.74 & 0.000 & 59.42668 & 61.69401 \\
Female & 78.88236 & .5772578 & 136.65 & 0.000 & 77.7505 & 80.01422 \\
\hline
\end{tabular}

The numbers reported in the Margin column are average values of the linear prediction of \(y\), as noted above the output table after Expression:. Based on a linear regression of y on sex and group, 60.6 would be the average value of the linear prediction of \(y\) if everyone in the data were treated as if they were male, and 78.9 would be the average value if everyone were treated as if they were female.

\section*{Example 2: A simple case after logistic}
margins may be used after almost any estimation command.
```

. logistic outcome i.sex i.group
(output omitted)
. margins sex
Predictive margins Number of obs = 3,000
Model VCE: OIM
Expression: Pr(outcome), predict()

```
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline & \multicolumn{2}{|l|}{\begin{tabular}{l}
Delta-method \\
Margin std. err.
\end{tabular}} & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% con & interval] \\
\hline sex & & & & & & \\
\hline Male & . 1286796 & . 0111424 & 11.55 & 0.000 & . 106841 & . 1505182 \\
\hline Female & . 1905087 & . 0089719 & 21.23 & 0.000 & . 1729241 & . 2080933 \\
\hline
\end{tabular}

The numbers reported in the Margin column are average predicted probabilities. Based on a logistic regression of outcome on sex and group, 0.13 would be the average probability of outcome if everyone in the data were treated as if they were male, and 0.19 would be the average probability if everyone were treated as if they were female.
margins reports average values after regress and average probabilities after logistic. By default, margins makes tables of whatever it is that predict (see [R] predict) predicts by default. Alternatively, margins can make tables of anything that predict can produce if you use margins' predict() option; see Example 11: Margins of a specified prediction.

\section*{Example 3: Average response versus response at average}

In example 2, margins reported average probabilities of outcome for sex \(=0\) and \(\operatorname{sex}=1\). If we instead wanted the predicted probabilities evaluated at the mean of the covariates, we would specify margins' atmeans option. We previously typed
```

. logistic outcome i.sex i.group
(output omitted)
margins sex
(output omitted)

```
and now we type
```

. margins sex, atmeans
Adjusted predictions Number of obs = 3,000
Model VCE: OIM
Expression: Pr(outcome), predict()
At: 0.sex = . }4993333\mathrm{ (mean)
1.sex = . }5006667\mathrm{ (mean)
1.group = . 3996667 (mean)
2.group = . }3726667\mathrm{ (mean)
3.group = . 2276667 (mean)

```
\begin{tabular}{c|cccccc}
\hline & \multicolumn{4}{|c}{ Delta-method } \\
& Margin \begin{tabular}{c} 
std. err.
\end{tabular} & \(z\) & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline sex & & & & & & \\
Male & .0966105 & .0089561 & 10.79 & 0.000 & .0790569 & .1141641 \\
Female & .1508362 & .0118064 & 12.78 & 0.000 & .127696 & .1739764 \\
\hline
\end{tabular}

The prediction at the average of the covariates is different from the average of the predictions. The first is the expected probability of a person with average characteristics, a person who, in another problem, might be \(3 / 4\) married and have 1.2 children. The second is the average of the probability among actual persons in the data.

When you specify atmeans or any other at option, margins reports the values used for the covariates in the legend above the table. margins lists the values for all the covariates, including values it may not use, in the results that follow. In this example, margins reported means for sex even though those means were not used. They were not used because we asked for the margins of sex, so sex was fixed first at 0 and then at 1 .

If you wish to suppress this legend, specify the nolegend option.

\section*{Example 4: Multiple margins from one command}

More than one margin can be reported by just one margins command. You can type
```

. margins sex group

```
and doing that is equivalent in terms of the output to typing
- margins sex
. margins group
When multiple margins are requested on the same command, each is estimated separately. There is, however, a difference when you also specify margins' post option. Then, the variance-covariance matrix for all margins requested is posted, and that is what allows you to test, for example, equality of margins. Testing equality of margins is covered in Example 10: Testing margins-contrasts of margins.

In any case, below we request margins for sex and for group.
```

. margins sex group
Predictive margins Number of obs = 3,000
Model VCE: OIM
Expression: Pr(outcome), predict()

```
\begin{tabular}{c|rrrrrr}
\hline & \multicolumn{4}{|c}{\begin{tabular}{c} 
Delta-method \\
Margin \\
std. err.
\end{tabular}} & z & \(\mathrm{P}>|\mathrm{z}|\) \\
sex & & & & & [95\% conf. interval]
\end{tabular}

\section*{Example 5: Margins with interaction terms}

The estimation command on which margins bases its calculations may contain interaction terms, such as an interaction of sex and group:
```

. logistic outcome i.sex i.group sex\#group
(output omitted)
. margins sex group
Predictive margins Number of obs = 3,000
Model VCE: OIM
Expression: Pr(outcome), predict()

```
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline & \multicolumn{3}{|c|}{Delta-method} & \(P>|z|\) & \multicolumn{2}{|l|}{[95\% conf. interval]} \\
\hline sex & & & & & & \\
\hline Male & . 1561738 & . 0132774 & 11.76 & 0.000 & . 1301506 & . 182197 \\
\hline Female & . 1983749 & . 0101546 & 19.54 & 0.000 & . 1784723 & . 2182776 \\
\hline group & & & & & & \\
\hline 1 & . 3211001 & . 0176403 & 18.20 & 0.000 & . 2865257 & . 3556744 \\
\hline 2 & . 1152127 & . 0099854 & 11.54 & 0.000 & . 0956417 & . 1347838 \\
\hline 3 & . 0265018 & . 0109802 & 2.41 & 0.016 & . 0049811 & . 0480226 \\
\hline
\end{tabular}

We fit the model by typing logistic outcome i.sex i.group sex\#group, but the meaning would have been the same had we typed logistic outcome sex\#\#group.

As mentioned in example 4, the results for sex and the results for group are calculated independently, and we would have obtained the same results had we typed margins sex followed by margins group.

The margin for male \((\operatorname{sex}=0)\) is 0.16 . The probability 0.16 is the average probability if everyone in the data were treated as if sex \(=0\), including sex \(=0\) in the main effect and sex \(=0\) in the interaction of sex with group.

Had we specified margins sex, atmeans, we would have obtained not average probabilities but the probabilities evaluated at the average. Rather than obtaining 0.16 , we would have obtained 0.10
for sex \(=0\). The 0.10 is calculated by taking the fitted model, plugging in sex \(=0\) everywhere, and plugging in the average value of the group indicator variables everywhere they are used. That is, rather than treating the group indicators as being \((1,0,0),(0,1,0)\), or \((0,0,1)\) depending on observation, the group indicators are treated as being \((0.40,0.37,0.23)\), which are the average values of group \(=1\), group \(=2\), and group \(=3\).

\section*{Example 6: Margins with continuous variables}

To the above example, we will add the continuous covariate age to the model and then rerun margins sex group.
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline \multicolumn{5}{|l|}{\begin{tabular}{l}
Predictive margins \\
Model VCE: OIM
\end{tabular}} & \multicolumn{2}{|l|}{Number of obs \(=3,000\)} \\
\hline \multicolumn{7}{|l|}{Expression: Pr(outcome), predict()} \\
\hline & \multicolumn{3}{|c|}{Delta-method} & \(P>|z|\) & \multicolumn{2}{|l|}{[95\% conf. interval]} \\
\hline sex & & & & & & \\
\hline Male & . 1600644 & . 0125653 & 12.74 & 0.000 & . 1354368 & . 184692 \\
\hline Female & . 1966902 & . 0100043 & 19.66 & 0.000 & . 1770821 & . 2162983 \\
\hline \multicolumn{7}{|l|}{group} \\
\hline 1 & . 2251302 & . 0123233 & 18.27 & 0.000 & . 200977 & . 2492834 \\
\hline 2 & . 150603 & . 0116505 & 12.93 & 0.000 & . 1277685 & . 1734376 \\
\hline 3 & . 0736157 & . 0337256 & 2.18 & 0.029 & . 0075147 & . 1397167 \\
\hline
\end{tabular}

Compared with the results presented in example 5, results for sex change little, but results for groups 1 and 3 change markedly. The tables differ because now we are adjusting for the continuous covariate age, as well as for sex and group.

We will continue examining interactions in example 8. Because we have added a continuous variable, let's take a detour to explain how to obtain margins for continuous variables and to explain their interpretation.

\section*{Example 7: Margins of continuous variables}

Continuing with our example of
```

. logistic outcome i.sex i.group sex\#group age

```
let's examine the continuous covariate age.

You are not allowed to type margins age; doing that will produce an error:
```

. margins age
factor age not found in list of covariates
r(322);

```

The message "age not found in list of covariates" is margins' way of saying, "Yes, age might be in the model, but if it is, it is not included as a factor variable; it is in as a continuous variable." Sometimes, Stata is overly terse. margins might also say that because age is continuous there are an infinite number of values at which it could evaluate the margins. At what value(s) should age be fixed? margins requires more guidance with continuous covariates. We can provide that guidance by using the at () option and typing
```

. margins, at(age=40)

```

To understand why that yields the desired result, let us tell you that if you were to type
```

. margins

```
margins would report the overall margin-the margin that holds nothing constant. Because our model is logistic, the average value of the predicted probabilities would be reported. The at () option fixes one or more covariates to the value(s) specified and can be used with both factor and continuous variables. Thus, if you typed margins, at (age=40), then margins would average over the data the responses for everybody, setting age \(=40\). Here is what happens when you type that:
```

. margins, at(age=40)
Predictive margins Number of obs = 3,000
Model VCE: OIM
Expression: Pr(outcome), predict()
At: age = 40

|  | Delta-method |  |  |  |  |  |
| :---: | ---: | :---: | :---: | :---: | :---: | :---: |
|  | Margin | std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| _cons | .1133603 | .0070731 | 16.03 | 0.000 | .0994972 | .1272234 |

```

Reported is the margin for age \(=40\), adjusted for the other covariates in our model.
If we wanted to obtain the margins for age \(30,35,40,45\), and 50 , we could type . margins, at (age=(30 354045 50) )
or, equivalently, . margins, at (age=(30(5)50))

\section*{Example 8: Margins of interactions}

Our model is
. logistic outcome i.sex i.group sex\#group age

We can obtain the margins of all possible combinations of the levels of sex and the levels of group by typing
```

. margins sex\#group
Predictive margins Number of obs = 3,000
Model VCE: OIM
Expression: Pr(outcome), predict()

```
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline & \multicolumn{2}{|r|}{Delta-method rgin std. err.} & z & \(P>|z|\) & [95\% con & interval] \\
\hline sex\#group & & & & & & \\
\hline Male\#1 & . 2379605 & . 0237178 & 10.03 & 0.000 & . 1914745 & . 2844465 \\
\hline Male\#2 & . 0658294 & . 0105278 & 6.25 & 0.000 & . 0451953 & . 0864636 \\
\hline Male\#3 & . 0538001 & . 0136561 & 3.94 & 0.000 & . 0270347 & . 0805656 \\
\hline Female\#1 & . 2158632 & . 0112968 & 19.11 & 0.000 & . 1937218 & . 2380045 \\
\hline Female\#2 & . 2054406 & . 0183486 & 11.20 & 0.000 & . 1694781 & . 2414032 \\
\hline Female\#3 & . 085448 & . 0533914 & 1.60 & 0.110 & -. 0191973 & . 1900932 \\
\hline
\end{tabular}

The first line in the table reports the marginal probability for sex \(=0\) (male) and group \(=1\). That is, it reports the estimated probability if everyone in the data were treated as if they were sex \(=0\) and group \(=1\).

Also reported are all the other combinations of sex and group.
By the way, we could have typed margins sex\#group even if our fitted model did not include sex\#group. Estimation is one thing, and asking questions about the nature of the estimates is another. margins does, however, require that i.sex and i.group appear somewhere in the model, because fixing a value outside the model would just produce the grand margin, and you can separately ask for that if you want it by typing margins without arguments.

\section*{Example 9: Decomposing margins}

We have the model
. logistic outcome i.sex i.group sex\#group age
In example 6, we typed margins sex and obtained 0.160 for males and 0.197 for females. We are going to decompose each of those numbers. Let us explain:
1. The margin for males, 0.160 , treats everyone as if they were male, and that amounts to simultaneously

1a. treating males as males and
1 b . treating females as males.
2. The margin for females, 0.197 , treats everyone as if they were female, and that amounts to simultaneously

2a. treating males as females and
\(2 b\). treating females as females.
The margins 1 a and 1 b are the decomposition of 1 , and the margins 2 a and 2 b are the decomposition of 2 .

We could obtain 1a and 2a by typing
because the qualifier if sex=\(=0\) would restrict margins to running on only the males. Similarly, we could obtain 1 b and 2 b by typing
```

. margins if sex==1, at(sex=(0 1))

```

We run these examples below:
```

. margins if sex==0, at(sex=(0 1))
Predictive margins Number of obs = 1,498
Model VCE: OIM
Expression: Pr(outcome), predict()
1._at: sex = 0
2._at: sex = 1

```
\begin{tabular}{c|cccccc}
\hline & \multicolumn{3}{c}{\begin{tabular}{c} 
Delta-method \\
std. err.
\end{tabular}} & \(\mathbf{z}\) & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline Margin & & & & & & \\
1 & .0794393 & .0062147 & 12.78 & 0.000 & .0672586 & .0916199 \\
2 & .1335584 & .0127351 & 10.49 & 0.000 & .1085981 & .1585187 \\
\hline
\end{tabular}
. margins if sex==1, at (sex=(0 1))
Predictive margins \(\quad\) Number of obs \(=1,502\)
Model VCE: OIM
Expression: \(\operatorname{Pr}\) (outcome), predict()
1._at: sex \(=0\)
2._at: sex \(=1\)
\begin{tabular}{c|cccccc}
\hline & \multicolumn{3}{|c}{ Delta-method } \\
& Margin \begin{tabular}{c} 
std. err.
\end{tabular} & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline _at & & & & & & \\
1 & .2404749 & .0199709 & 12.04 & 0.000 & .2013326 & .2796171 \\
2 & .2596538 & .0104756 & 24.79 & 0.000 & .2391219 & .2801857 \\
\hline
\end{tabular}

Putting together the results from example 6 and the results above, we have
\begin{tabular}{cl} 
Margin treating individuals as themselves & 0.170 \\
Margin treating individuals as male & 0.160 \\
Margin treating male as male & 0.079 \\
Margin treating female as male & 0.240 \\
Margin treating individuals as female & 0.197 \\
Margin treating male as female & 0.134 \\
Margin treating female as female & 0.260
\end{tabular}

\section*{Example 10: Testing margins-contrasts of margins}

Continuing with the previous example, it would be interesting to test the equality of 2 b and 1 b , to test whether the average probability of a positive outcome for females treated as females is equal to that for females treated as males. That test would be different from testing the overall significance of sex in our model. The test performed on our model would be a test of whether the probability of a positive outcome differs between males and females when they have equal values of the other covariates. The test of equality of margins is a test of whether the average probabilities differ given the different pattern of values of the other covariates that the two sexes have in our data.

We can also perform such tests by treating the results from margins as estimation results. There are three steps required to perform tests on margins. First, you must arrange it so that all the margins of interest are reported by just one margins command. Second, you must specify margins' post option. Third, you perform the test with the test command.

Such tests and comparisons can be readily performed by contrasting margins; see [R] margins, contrast. Also see Contrasts of margins—effects (discrete marginal effects) in [R] marginsplot.

In the previous example, we used two commands to obtain our results, namely,
. margins if sex==0, at (sex=(01))
. margins if sex==1, at (sex=(01))
We could, however, have obtained the same results by typing just one command:
```

. margins, over(sex) at(sex=(0 1))

```

Performing margins, over (sex) first restricts the sample to sex==0 and then restricts it to sex==1, and that is equivalent to the two different if conditions that we specified before.

To test whether females treated as females is equal to females treated as males, we will need to type
```

. margins, over(sex) at(sex=(0 1)) post
. test _b[2._at\#1.sex] = _b[1._at\#1.sex]

```

We admit that the second command may seem to have come out of nowhere. When we specify post on the margins command, margins behaves as if it were an estimation command, which means that 1) it posts its estimates and full VCE to e(), 2) it gains the ability to replay results just as any estimation command can, and 3 ) it gains access to the standard postestimation commands. Item 3 explains why we could use test. We learned that we wanted to test _b [2._at\#1.sex] and _b[1._at\#1.sex] by replaying the estimation results, but this time with the standard estimation command coeflegend option. So, what we typed was
```

. margins, over(sex) at(sex=(0 1)) post
. margins, coeflegend
. test _b[2._at\#1.sex] = _b[1._at\#1.sex]

```

We will let you try margins, coeflegend for yourself. The results of running the other two commands are
```

. margins, over(sex) at(sex=(0 1)) post
Predictive margins Number of obs = 3,000
Model VCE: OIM
Expression: Pr(outcome), predict()
Over: sex
1._at: 0.sex
sex = 0
1.sex
sex = 0
2._at: 0.sex
sex = 1
1.sex
sex = 1

```
\begin{tabular}{c|cccccc}
\hline & \multicolumn{3}{|c}{ Delta-method } \\
& Margin & std.err. & \(z\) & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline _at\#sex & & & & & & \\
1\#Male & .0794393 & .0062147 & 12.78 & 0.000 & .0672586 & .0916199 \\
1\#Female & .2404749 & .0199709 & 12.04 & 0.000 & .2013326 & .2796171 \\
2\#Male & .1335584 & .0127351 & 10.49 & 0.000 & .1085981 & .1585187 \\
2\#Female & .2596538 & .0104756 & 24.79 & 0.000 & .2391219 & .2801857 \\
\hline
\end{tabular}
```

. test _b[2._at\#1.sex] = _b[1._at\#1.sex]
( 1) - 1bn._at\#1.sex + 2._at\#1.sex = 0
chi2( 1) = 0.72
Prob > chi2 = 0.3951

```

We can perform the same test in one command using contrasts of margins:
```

. logistic outcome i.sex i.group sex\#group age
(output omitted)
. margins, over(sex) at (sex=(0 1)) contrast(atcontrast(r._at) wald)
Contrasts of predictive margins Number of obs $=3,000$
Model VCE: OIM
Expression: $\operatorname{Pr}$ (outcome), predict()
Over: sex
1._at: 0.sex
sex $=0$
1. sex
sex $=0$
2._at: 0.sex
sex $=1$
1.sex
sex $=1$

|  | df | chi2 | P>chi2 |
| :---: | :---: | :---: | :---: |
|  |  |  |  |
| (2 vs 1) Malesex |  |  |  |
| $(2$ vs 1) Female | 1 | 14.59 | 0.0001 |
| Joint | 2 | 16.13 | 0.0003 |

```
\begin{tabular}{c|ccrr}
\hline & \multicolumn{4}{|c}{\begin{tabular}{c} 
Delta-method \\
std. err.
\end{tabular}} \\
\hline Contrast & [95\% conf. interval] \\
\hline (2 vs 1) Male & .0541192 & .0141706 & .0263453 & .081893 \\
\((2 \mathrm{vs} 1)\) Female & .0191789 & .0225516 & -.0250215 & .0633793 \\
\hline
\end{tabular}

We refit our logistic model because its estimation results were replaced when we posted our margins. The syntax to perform the contrast we want is admittedly not obvious. Contrasting (testing) across at () groups is more difficult than contrasting across the margins themselves or across over() groups, because we have no natural place for the contrast operators (r., in our case). We also explicitly requested Wald tests of the contrasts, which are not provided by default. Nevertheless, the \(\chi^{2}\) statistic and its \(p\)-value for (2 vs 1 ) for female matches the results of our test command. We also obtain the test of whether the response of males treated as males is equal to the response of males treated as females.

For a gentler introduction to contrasts of margins, see \([\mathrm{R}]\) margins, contrast.

\section*{Example 11: Margins of a specified prediction}

We will fit the model
. use https://www.stata-press.com/data/r18/margex
. tobit ycn i.sex i.group sex\#group age, ul(90)
and we will tell the following story about the variables: We run a peach orchard where we allow people to pick their own peaches. A person receives one empty basket in exchange for \(\$ 20\), along with the right to enter the orchard. There is no official limit on how many peaches a person can pick, but only 90 peaches will fit into a basket. The dependent variable in the above tobit model, ycn, is the number of peaches picked. We use tobit, a special case of censored-normal regression, because ycn is censored at 90 .

After fitting this model, if we typed
```

. margins sex

```
we would obtain the margins for males and for females of the uncensored number of peaches picked. We would obtain that because predict after tobit produces the uncensored number by default. To obtain the censored prediction, we would have to specify predict's ystar(.,90) option. If we want the margins based on that response, we type
```

. margins sex, predict(ystar(.,90))

```

The results of typing that are
```

. tobit ycn i.sex i.group sex\#group age, ul(90)
(output omitted)
. margins sex, predict(ystar(.,90))
Predictive margins Number of obs = 3,000
Model VCE: OIM
Expression: E(ycn*|ycn<90), predict(ystar(.,90))

```
\begin{tabular}{c|rrrrrr}
\hline & \multicolumn{4}{|c}{ Delta-method } \\
& Margin \begin{tabular}{c} 
std. err.
\end{tabular} & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline sex & & & & & & \\
Male & 62.21804 & .5996952 & 103.75 & 0.000 & 61.04266 & 63.39342 \\
Female & 78.34272 & .4555278 & 171.98 & 0.000 & 77.4499 & 79.23553 \\
\hline
\end{tabular}

In our previous examples, sex \(=1\) has designated females, so evidently the females visiting our orchard are better at filling baskets than the men.

\section*{Example 12: Margins of a specified expression}

Continuing with our peach orchard example and the previously fit model
. use https://www.stata-press.com/data/r18/margex
. tobit ycn i.sex i.group sex\#group age, ul(90)
let's examine how well our baskets are working for us. What is the proportion of the number of peaches actually picked to the number that would have been picked were the baskets larger? As mentioned in example 11, predict, ystar (.,90) produces the expected number picked given the limit of basket size. predict, xb would predict the expected number without a limit. We want the ratio of those two predictions. That ratio will measure as a proportion how well the baskets work. Thus, we could type
```

. margins sex, expression(predict(ystar(.,90))/predict(xb))

```

That would give us the proportion for everyone treated as male and everyone treated as female, but what we want to know is how well baskets work for true males and true females, so we will type
```

. margins, over(sex) expression(predict(ystar(0,90))/predict(xb))
Predictive margins Number of obs = 3,000
Model VCE: OIM
Expression: predict(ystar(0,90))/predict(xb)
Over: sex

```
\begin{tabular}{c|cccccc}
\hline & \multicolumn{3}{|c}{ Delta-method } \\
& Margin \begin{tabular}{c} 
std. err.
\end{tabular} & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline sex & & & & & & \\
Male & .9811785 & .0013037 & 752.60 & 0.000 & .9786233 & .9837338 \\
Female & .9419962 & .0026175 & 359.88 & 0.000 & .9368659 & .9471265 \\
\hline
\end{tabular}

By the way, we could count the number of peaches saved by the limited basket size during the period of data collection by typing
```

. count
3,000
. margins, expression(3000*(predict(xb)-predict(ystar(.,90))))
(output omitted)

```

The number of peaches saved turns out to be 9,183 .

\section*{Example 13: Margins with multiple outcomes (responses)}

Estimation commands such as mlogit and mprobit (see [R] mlogit and [R] mprobit) calculate multiple responses, and those multiple responses are reflected in the options available with predict after estimation. Obtaining margins for such estimators is thus the same as obtaining margins of a specified prediction, which was demonstrated in example 11. The solution is to include the predict_opt that selects the desired response in margins' predict (predict_opt) option.

If we fit the multinomial logistic model
```

. mlogit group i.sex age

```
then to obtain the margins for the probability that group \(=1\), we would type
```

. margins sex, predict(outcome(1))

```
and to obtain the margins for the probability that group \(=3\), we would type
- margins sex, predict(outcome(3))

To obtain the margins for each of these outcomes simultaneously, type
```

. margins sex, predict(outcome(1)) predict(outcome(3))

```

We learned about the outcome(1) and outcome (3) options by looking in [ R\(]\) mlogit postestimation. For an example using margins with a multiple-outcome estimator, see example 4 in \([R]\) mlogit postestimation.

\section*{Example 14: Margins with multiple equations}

Estimation commands such as mvreg, manova, sureg, and reg3 (see [MV] mvreg, [MV] manova, \([R]\) sureg, and \([R]\) reg3) fit multiple equations. Obtaining margins for such estimators is the same as obtaining margins with multiple outcomes (see example 13), which in turn is the same as obtaining margins of a specified prediction (see example 11). You place the relevant option from the estimator's predict command into margins' predict (predict_opt) option.

If we fit the seemingly unrelated regression model
```

. sureg (y = i.sex age) (distance = i.sex i.group)

```
we can obtain the marginal means of \(y\) for males and females by typing
```

. margins sex, predict(equation(y))

```
and we can obtain the marginal means of distance by typing
```

. margins sex, predict(equation(distance))

```

We could obtain the difference between the margins of \(y\) and distance by typing
```

. margins sex, expression(predict(equation(y)) -
> predict(equation(distance)))

```

More examples can be found in [MV] manova and [MV] manova postestimation.

\section*{Example 15: Margins evaluated out of sample}

You can fit your model on one dataset and use margins on another if you specify margins' noesample option. Remember that margins reports estimated average responses, and, unless you lock all the covariates at fixed values by using the at () option, the remaining variables are allowed to vary as they are observed to vary in the data. That is indeed the point of using margins. The fitted model provides the basis for adjusting for the remaining variables, and the data provide their values. The predictions produced by margins are of interest assuming the data used by margins are in some sense interesting or representative. In some cases, you might need to fit your model on one set of data and perform margins on another.

In example 11, we fit the model
```

. tobit ycn i.sex i.group sex\#group age, ul(90)

```
and we told a story about our peach orchard in which we charged people \(\$ 20\) to collect a basket of peaches, where baskets could hold at most 90 peaches. Let us now tell you that we believe the data on which we estimated those margins were unrepresentative, or at least, we have a more representative sample stored in another .dta file. That dataset includes the demographics of our customers but does not include counts of peaches picked. It is a lot of work counting those peaches.

Thus, we will fit our model just as we did previously using the detailed data, but we will bring the other, more representative dataset into memory before issuing the margins sex, predict (ystar(.,90)) command, and we will add noesample to it.
```

. use https://www.stata-press.com/data/r18/margex
(Artificial data for margins)
. tobit ycn i.sex i.group sex\#group age, ul(90)
(output omitted)
. use https://www.stata-press.com/data/r18/peach
. margins sex, predict(ystar(.,90)) noesample
Predictive margins Number of obs = 2,727
Model VCE: OIM
Expression: E(ycn*|ycn<90), predict(ystar(.,90))

```
\begin{tabular}{c|rrrrrr}
\hline & \multicolumn{4}{|c}{ Delta-method } \\
& Margin \begin{tabular}{c} 
std. err.
\end{tabular} & \(z\) & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline sex & & & & & & \\
0 & 56.79774 & 1.003731 & 56.59 & 0.000 & 54.83046 & 58.76502 \\
1 & 75.02146 & .6437446 & 116.54 & 0.000 & 73.75974 & 76.28317 \\
\hline
\end{tabular}

In example 12, we produced an estimate of the number of peaches saved by the limited-size baskets. We can update that estimate using the new demographic data by typing
```

. count
2,727
. margins, exp(2727*(predict(xb)-predict(ystar(.,90)))) noesample
(output omitted)

```

By running the above, we find that the updated number of peaches saved is 6,408 .

\section*{Obtaining margins of derivatives of responses (a.k.a. marginal effects)}

Derivatives of responses are themselves responses, so everything said above in Obtaining margins of responses is equally true of derivatives of responses, and every example above could be repeated here substituting the derivative of the response for the response.

Derivatives are of interest because they are an informative way of summarizing fitted results. The change in a response for a change in the covariate is easy to understand and to explain. In simple models, one hardly needs margins to assist in obtaining such margins. Consider the simple linear regression
\[
y=\beta_{0}+\beta_{1} \times \operatorname{sex}+\beta_{2} \times \text { age }+\epsilon
\]

The derivatives of the responses are
\[
\begin{aligned}
& d y / d(\text { sex })=\beta_{1} \\
& d y / d(\text { age })=\beta_{2}
\end{aligned}
\]
margins computes these derivatives using \(\widehat{\beta}_{1}\) and \(\widehat{\beta}_{2}\), the coefficients from the linear predictor of \(y\). How does the linear predictor \(\widehat{y}\) change between males and females? It changes by \(\widehat{\beta}_{1}\). How does \(\widehat{y}\) change with age? It changes by \(\widehat{\beta}_{2}\) per year.

If you make the model a little more complicated, however, the need for margins arises. Consider the model
\[
y=\beta_{0}+\beta_{1} \times \operatorname{sex}+\beta_{2} \times \text { age }+\beta_{3} \times \text { age }^{2}+\epsilon
\]

Now, the derivative with respect to age is
\[
d y / d(\text { age })=\beta_{2}+2 \times \beta_{3} \times \text { age }
\]

The change in \(y\) for a change in age itself changes with age, and so to better understand the fitted results, you might want to make a table of the change in \(y\) for a change in age for age \(=30\), age \(=40\), and age \(=50\). margins can do that.

Consider an even more complicated model, such as
\[
\begin{align*}
y=\beta_{0} & +\beta_{1} \times \mathrm{sex}+\beta_{2} \times \text { age }+\beta_{3} \times \mathrm{age}^{2}+\beta_{4} \times \mathrm{bp}+\beta_{5} \times \mathrm{sex} \times \mathrm{bp}+\beta_{6} \times \mathrm{tmt} \\
& +\beta_{7} \times \mathrm{tmt} \times \text { age }+\beta_{8} \times \mathrm{tmt} \times \mathrm{age}^{2}+\epsilon \tag{1}
\end{align*}
\]

The derivatives are
\[
\begin{aligned}
d y / d(\operatorname{sex}) & =\beta_{1}+\beta_{5} \times \mathrm{bp} \\
d y / d(\mathrm{age}) & =\beta_{2}+2 \times \beta_{3} \times \text { age }+\beta_{7} \times \mathrm{tmt}+2 \times \beta_{8} \times \mathrm{tmt} \times \text { age } \\
d y / d(\mathrm{bp}) & =\beta_{4}+\beta_{5} \times \mathrm{sex} \\
d y / d(\mathrm{tmt}) & =\beta_{6}+\beta_{7} \times \text { age }+\beta_{8} \times \operatorname{age}^{2}
\end{aligned}
\]

At this point, margins becomes indispensable.

\section*{Use at() freely, especially with continuous variables}

An option one tends to use frequently with derivatives of responses is at(). Such use is often to better understand or to communicate how the response varies, or, in technical jargon, to explore the nature of the response surface.

For instance, the effect \(d y / d(\mathrm{tmt})\) in (1) is equal to \(\beta_{6}+\beta_{7} \times\) age \(+\beta_{8} \times \mathrm{age}^{2}\), and so simply to understand how treatment varies with age, we may want to fix age at various values. We might type
```

. margins, dydx(tmt) at(age=(30 40 50))

```

\section*{Expressing derivatives as elasticities}

You specify the dydx (varname) option on the margins command to use \(d y / d(\) varname) as the response variable. If you want that derivative expressed as an elasticity, you can specify eyex (varname), eydx (varname), or dyex (varname). You substitute e for d where you want an elasticity. In the discussion below, when we talk about \(y\), we are referring to the linear predictions from the model.

The formulas are
\[
\begin{aligned}
\operatorname{dydx}() & =d y / d x \\
\operatorname{eyex}() & =d y / d x \times(x / y) \\
\operatorname{eydx}() & =d y / d x \times(1 / y)=d \ln (y) / d x \\
\operatorname{dyex}() & =d y / d x \times(x)=d y / d \ln (x)
\end{aligned}
\]
and the interpretations are
\begin{tabular}{lllll} 
dydx (): & & change in \(y\) for a & & change in \(x\) \\
eyex (): & proportional & change in \(y\) for a & proportional & \begin{tabular}{l} 
change in \(x\) \\
eydx ():
\end{tabular} \\
proportional & change in \(y\) for a & & change in \(x\) \\
dyex(): & & change in \(y\) for a & proportional & change in \(x\)
\end{tabular}

As margins always does with response functions, calculations are made at the observational level and are then averaged. Let's assume that in observation \(5, d y / d x=0.5, y=15\), and \(x=30\); then
\[
\begin{array}{r}
\operatorname{dydx}()=0.5 \\
\operatorname{eyex}()=1.0 \\
\operatorname{eydx}()=0.03 \\
\operatorname{dyex}()=15.0
\end{array}
\]

Many social scientists would informally explain the meaning of eyex () = as " \(y\) increases \(100 \%\) when \(x\) increases \(100 \%\) " or as " \(y\) doubles when \(x\) doubles", although neither statement is literally true. eyex (), eydx(), and dyex() are rates evaluated at a point, just as dydx() is a rate, and all such interpretations are valid only for small (infinitesimal) changes in \(x\). It is true that eyex() \(=1\) means \(y\) increases with \(x\) at a rate such that, if the rate were constant, \(y\) would double if \(x\) doubled. This issue of casual interpretation is no different from casually interpreting dydx () as if it represents the response to a unit change. It is not necessarily true that \(\operatorname{dydx}()=0.5\) means that " \(y\) increases by 0.5 if \(x\) increases by 1 ". It is true that " \(y\) increases with \(x\) at a rate such that, if the rate were constant, \(y\) would increase by 0.5 if \(x\) increased by 1 ".
dydx(), eyex(), eydx(), and dyex() may be used with continuous \(x\) variables. dydx() and eydx () may also be used with factor variables. For eydx (), effects are based on \(d \ln (y) / d x\) for continuous covariates and on differences of \(\ln (y)\) for discrete covariates. Estimates based on logarithms have better computational properties and allow us to think of semielasticities as coefficients in a regression with log dependent variables. As we discuss next, when we are analyzing marginal effects of categorical variables, we think in terms of discrete differences instead of derivatives.

\section*{Derivatives versus discrete differences}

In (1),
\[
\begin{aligned}
y=\beta_{0} & +\beta_{1} \times \mathrm{sex}+\beta_{2} \times \text { age }+\beta_{3} \times \mathrm{age}^{2}+\beta_{4} \times \mathrm{bp}+\beta_{5} \times \mathrm{sex} \times \mathrm{bp}+\beta_{6} \times \mathrm{tmt} \\
& +\beta_{7} \times \mathrm{tmt} \times \mathrm{age}+\beta_{8} \times \mathrm{tmt} \times \mathrm{age}^{2}+\epsilon
\end{aligned}
\]

Let us call your attention to the derivatives of \(y\) with respect to age and sex:
\[
\begin{gather*}
d y / d(\text { age })=\beta_{2}+2 \times \beta_{3} \times \text { age }+\beta_{7} \times \mathrm{tmt}+2 \times \beta_{8} \times \mathrm{tmt} \times \text { age }  \tag{2}\\
d y / d(\text { sex })=\beta_{1}+\beta_{5} \times \mathrm{bp} \tag{3}
\end{gather*}
\]
age is presumably a continuous variable, and (2) is precisely how margins calculates its derivatives when you type margins, dydx (age). sex, however, is presumably a factor variable, and margins does not necessarily make the calculation using (3) were you to type margins, dydx (sex). We will explain, but let us first clarify what we mean by a continuous and a factor variable. Say that you fit (1) by typing
```

. regress y i.sex age c.age\#c.age i.bp bp\#sex
> i.tmt tmt\#c.age tmt\#c.age\#c.age

```

It is important that sex entered the model as a factor variable. It would not do to type regress y sex ... because then sex would be a continuous variable, or at least it would be a continuous variable from Stata's point of view. The model estimates would be the same, but margins' understanding of those estimates would be a little different. With the model fit using i.sex, margins understands that either sex is 0 or sex is 1 . With the model fit using sex, margins thinks sex is continuous and, for instance, sex \(=1.5\) is a possibility.
margins calculates \(\operatorname{dydx}()\) differently for continuous and for factor variables. For continuous variables, margins calculates \(d y / d x\) using the fitted values from regress. For factor variables, margins calculates the discrete first difference from the base category. To obtain that for sex, write down the model and then subtract from it the model evaluated at the base category for sex, which is \(s e x=0\). If you do that, you will get the same formula as we obtained for the derivative, namely,
\[
\text { discrete difference }\{(\operatorname{sex}=1)-(\operatorname{sex}=0)\}=\beta_{1}+\beta_{5} \times \mathrm{bp}
\]

We obtain the same formula because our model is linear regression. Outside of linear regression, and outside of linear response functions generally, the discrete difference is not equal to the derivative. The discrete difference is not equal to the derivative for logistic regression, probit, etc. The discrete difference calculation is generally viewed as better for factor variables than the derivative calculation because the discrete difference is what would actually be observed.

If you want the derivative calculation for your factor variables, specify the continuous option on the margins command.

\section*{Example 16: Average marginal effect (partial effects)}

Concerning the title of this example, the way we use the term marginal effect, the effects of factor variables are calculated using discrete first differences. If you wanted the continuous calculation, you would specify margins' continuous option in what follows.
```

. use https://www.stata-press.com/data/r18/margex
(Artificial data for margins)
. logistic outcome treatment\#\#group age c.age\#c.age treatment\#c.age
(output omitted)
. margins, dydx(treatment)
Average marginal effects Number of obs = 3,000
Model VCE: OIM
Expression: Pr(outcome), predict()
dy/dx wrt: 1.treatment

```
\begin{tabular}{l|rccccc}
\hline & dy/dx \begin{tabular}{c} 
Delta-method \\
std. err.
\end{tabular} & \(z\) & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline 1.treatment & .0385625 & .0162848 & 2.37 & 0.018 & .0066449 & .0704801 \\
\hline
\end{tabular}

Note: dy/dx for factor levels is the discrete change from the base level.
The average marginal effect of treatment on the probability of a positive outcome is 0.039 .

\section*{Example 17: Average marginal effect of all covariates}

We will continue with the model
. logistic outcome treatment\#\#group age c.age\#c.age treatment\#c.age
if we wanted the average marginal effects for all covariates, we would type margins, \(d y d x(*)\) or margins, dydx (_all); they mean the same thing. This is probably the most common way margins, dydx() is used.
. margins, dydx(*)
Average marginal effects \(\quad\) Number of obs \(=3,000\)
Model VCE: OIM
Expression: Pr(outcome), predict()
dy/dx wrt: 1.treatment 2.group 3.group age
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline & \multicolumn{2}{|l|}{Delta-method dy/dx std. err.} & z & \(P>|z|\) & [95\% conf & interval] \\
\hline 1.treatment & . 0385625 & . 0162848 & 2.37 & 0.018 & . 0066449 & . 0704801 \\
\hline group & & & & & & \\
\hline 2 & -. 0776906 & . 0181584 & -4.28 & 0.000 & -. 1132805 & -. 0421007 \\
\hline 3 & -. 1505652 & . 0400882 & -3.76 & 0.000 & -. 2291366 & -. 0719937 \\
\hline age & . 0095868 & . 0007796 & 12.30 & 0.000 & . 0080589 & . 0111148 \\
\hline
\end{tabular}

Note: dy/dx for factor levels is the discrete change from the base level.

\section*{Example 18: Evaluating marginal effects over the response surface}

Continuing with the model
```

. logistic outcome treatment\#\#group age c.age\#c.age treatment\#c.age

```

What follows maps out the entire response surface of our fitted model. We report the marginal effect of treatment evaluated at age \(=20,30, \ldots, 60\), by each level of group.
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline \multicolumn{5}{|l|}{Conditional marginal effects Model VCE: OIM} & \multicolumn{2}{|l|}{Number of obs \(=3,000\)} \\
\hline \multicolumn{7}{|l|}{```
Expression: Pr(outcome), predict()
dy/dx wrt: 1.treatment
1._at: age = 20
2._at: age = 30
3._at: age = 40
4._at: age = 50
5._at: age = 60
```} \\
\hline & dy/dx & lta-meth std. err & z & \(P>|z|\) & [95\% con & interval] \\
\hline 0. treatment & (base out & & & & & \\
\hline \multicolumn{7}{|l|}{1.treatment _at\#group} \\
\hline 11 & -. 0208409 & . 0152862 & -1.36 & 0.173 & -. 0508013 & . 0091196 \\
\hline 12 & . 009324 & . 0059896 & 1.56 & 0.120 & -. 0024155 & . 0210635 \\
\hline 13 & . 0006558 & . 0048682 & 0.13 & 0.893 & -. 0088856 & . 0101972 \\
\hline 21 & -. 0436964 & . 0279271 & -1.56 & 0.118 & -. 0984325 & . 0110397 \\
\hline 22 & . 0382959 & . 0120405 & 3.18 & 0.001 & . 014697 & . 0618949 \\
\hline 23 & . 0064564 & . 0166581 & 0.39 & 0.698 & -. 0261929 & . 0391057 \\
\hline 31 & -. 055676 & . 0363191 & -1.53 & 0.125 & -. 1268601 & . 015508 \\
\hline 32 & . 1152235 & . 0209858 & 5.49 & 0.000 & . 074092 & . 156355 \\
\hline 33 & . 0284808 & . 0471293 & 0.60 & 0.546 & -. 0638908 & . 1208524 \\
\hline 41 & -. 027101 & . 0395501 & -0.69 & 0.493 & -. 1046177 & . 0504158 \\
\hline 42 & . 2447682 & . 0362623 & 6.75 & 0.000 & . 1736954 & . 315841 \\
\hline 43 & . 0824401 & . 1025028 & 0.80 & 0.421 & -. 1184616 & . 2833418 \\
\hline 51 & . 0292732 & . 0587751 & 0.50 & 0.618 & -. 0859239 & . 1444703 \\
\hline 52 & . 3757777 & . 0578106 & 6.50 & 0.000 & . 2624709 & . 4890844 \\
\hline 53 & . 1688268 & . 1642191 & 1.03 & 0.304 & -. 1530368 & . 4906904 \\
\hline
\end{tabular}

Note: dy/dx for factor levels is the discrete change from the base level.

\section*{Obtaining margins with survey data and representative samples}

The standard errors and confidence intervals produced by margins are based by default on the delta method applied to the VCE of the current estimates. Delta-method standard errors treat the covariates at which the response is evaluated as given or fixed. Such standard errors are appropriate if you specify at () to fix the covariates, and they are appropriate when you are making inferences about groups exactly like your sample whether you specify at () or not.

On the other hand, if you have a representative sample of the population or if you have complex survey data and if you want to make inferences about the underlying population, you need to account for the variation in the covariates that would arise in repeated sampling. You do that using vce (unconditional), which invokes a different standard error calculation based on Korn and Graubard (1999). Syntactically, there are three cases. They all involve specifying the vce (unconditional) option on the margins command:
1. You have a representative random sample, and you have not svyset your data.

When you fit the model, you need to specify the vce (robust) or vce (cluster clustvar) option. When you issue the margins command, you need to specify the vce (unconditional) option.
2. You have a weighted sample, and you have not svyset your data.

You need to specify [pw=weight] when you fit the model and, of course, specify the vce (unconditional) option on the margins command. You do not need to specify the weights on the margins command because margins will obtain them from the estimation results.
3. You have svyset your data, whether it be a simple random sample or something more complex including weights, strata, sampling units, or poststratification, and you are using the linearized variance estimator.
You need to use the svy prefix when you fit the model. You need to specify vce (unconditional) when you issue the margins command. You do not need to respecify the weights.

Even though the data are svyset, and even though the estimation was svy estimation, margins does not default to vce(unconditional). It does not default to vce (unconditional) because there are valid reasons to want the data-specific, vce (delta) standard-error estimates. Whether you specify vce (unconditional) or not, margins uses the weights, so you do not need to respecify them even if you are using vce (unconditional).
vce(unconditional) is allowed only after estimation with vce(robust), vce(cluster ...), or the svy prefix with the linearized variance estimator. If the VCE of the current estimates was specified as clustered, so will be the VCE estimates of margins. If the estimates were from a survey estimation, the survey settings in the dataset will be used by margins.

When you use vce(unconditional), never specify if exp or in range on the margins command; instead, specify the subpop (if exp) option. You do that for the usual reasons; see [SVY] Subpopulation estimation. If you specify over (varlist) to examine subgroups, the subgroups will automatically be treated as subpopulations.

If you are using a replication-based variance estimator, you may want to use this method to estimate the variance of your margins; see [SVY] svy postestimation.

\section*{Example 19: Inferences for populations, margins of response}

In example 6, we fit the model
. logistic outcome i.sex i.group sex\#group age
and we obtained margins by sex and margins by group,
```

. margins sex group

```

If our data were randomly drawn from the population of interest and we wanted to account for this, we would have typed
```

. logistic outcome i.sex i.group sex\#group age, vce(robust)
. margins sex group, vce(unconditional)

```

We do that below:
```

. logistic outcome i.sex i.group sex\#group age, vce(robust)
(output omitted)
. margins sex group, vce(unconditional)
Predictive margins Number of obs = 3,000
Expression: Pr(outcome), predict()

```
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline & Margin & \begin{tabular}{l}
condition \\
std. err
\end{tabular} & z & \(P>|z|\) & \multicolumn{2}{|l|}{[95\% conf. interval]} \\
\hline sex & & & & & & \\
\hline Male & . 1600644 & . 0131685 & 12.16 & 0.000 & . 1342546 & . 1858743 \\
\hline Female & . 1966902 & . 0104563 & 18.81 & 0.000 & . 1761963 & . 2171841 \\
\hline group & & & & & & \\
\hline 1 & . 2251302 & . 0127069 & 17.72 & 0.000 & . 200225 & . 2500354 \\
\hline 2 & . 150603 & . 0118399 & 12.72 & 0.000 & . 1273972 & . 1738088 \\
\hline 3 & . 0736157 & . 0343188 & 2.15 & 0.032 & . 0063522 & . 1408793 \\
\hline
\end{tabular}

The estimated margins are the same as they were in example 6 , but the standard errors and confidence intervals differ, although not by much. Given that we have 3,000 observations in our randomly drawn sample, we should expect this.

\section*{Example 20: Inferences for populations, marginal effects}

In example 17, we fit a logistic model and then obtained the average marginal effects for all covariates by typing
```

. logistic outcome treatment\#\#group age c.age\#c.age treatment\#c.age
. margins, dydx(*)

```

To repeat that and also obtain standard errors for our population, we would type
```

. logistic outcome treatment\#\#group age c.age\#c.age treatment\#c.age,
> vce(robust)
. margins, dydx(*) vce(unconditional)

```

The results are
```

. logistic outcome treatment\#\#group age c.age\#c.age treatment\#c.age, vce(robust)
(output omitted)
. margins, dydx(*) vce(unconditional)
Average marginal effects Number of obs = 3,000
Expression: Pr(outcome), predict()
dy/dx wrt: 1.treatment 2.group 3.group age

```
\begin{tabular}{r|rrccccc}
\hline & \multicolumn{4}{|c}{ Unconditional } \\
& dy/dx & std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline 1. treatment & .0385625 & .0163872 & 2.35 & 0.019 & .0064442 & .0706808 \\
group & & & & & & \\
2 & -.0776906 & .0179573 & -4.33 & 0.000 & -.1128863 & -.0424949 \\
3 & -.1505652 & .0411842 & -3.66 & 0.000 & -.2312848 & -.0698456 \\
age & .0095868 & .0007814 & 12.27 & 0.000 & .0080553 & .0111183 \\
\hline
\end{tabular}

Note: dy/dx for factor levels is the discrete change from the base level.

\section*{Example 21: Inferences for populations with svyset data}

See example 3 in [SVY] svy postestimation.

\section*{Standardizing margins}

A standardized margin is the margin calculated on data different from the data used to fit the model. Typically, the word standardized is reserved for situations in which the alternate population is a reference population, which may be real or artificial, and which is treated as fixed.

Say that you work for a hospital and have fit a model of mortality on the demographic characteristics of the hospital's patients. At this stage, were you to type
. margins
you would obtain the mortality rate for your hospital. You have another dataset, hstandard.dta, that contains demographic characteristics of patients across all hospitals along with the population of each hospital recorded in the pop variable. You could obtain the expected mortality rate at your hospital if your patients matched the characteristics of the standard population by typing
```

. use https://www.stata-press.com/data/r18/hstandard, clear
. margins [fw=pop], noesample

```

You specified noesample because the margin is being calculated on data other than the data used to fit the model. You specified [ \(\mathrm{fw}=\mathrm{pop}\) ] because the reference dataset you are using included population counts, as many reference datasets do.

\section*{Obtaining margins as though the data were balanced}

Here we discuss what are commonly called estimated marginal means or least-squares means. These are margins assuming that all levels of factor variables are equally likely or, equivalently, that the design is balanced. The seminal reference on these margins is Searle, Speed, and Milliken (1980).

In designed experiments, observations are often allocated in a balanced way so that the variances can be easily compared and decomposed. At the Acme Portable Widget Company, they are experimenting with a new machine. The machine has three temperature settings and two pressure settings; a combination of settings will be optimal on any particular day, determined by the weather. At start-up, one runs a quick test and chooses the optimal setting for the day. Across different days, each setting will be used about equally, says the manufacturer.

In experiments with the machine, 10 widgets were collected for stress testing at each of the settings over a six-week period. We wish to know the average stress-test value that can be expected from these machines over a long period.

\section*{Balancing using asbalanced}

The data were intended to be balanced, but unfortunately, the stress test sometimes destroys samples before the stress can be measured. Thus, even though the experiment was designed to be balanced, the data are not balanced. You specify the asbalanced option to estimate the margins as if the data were balanced. So that you can compare the asbalanced results with the observed results, we will also include margins without the asbalanced option in what follows:
```

. use https://www.stata-press.com/data/r18/acmemanuf
. regress y pressure\#\#temp
(output omitted)
. margins
Predictive margins Number of obs = 49
Model VCE: OLS
Expression: Linear prediction, predict()

```
\begin{tabular}{l|rrrrrr}
\hline & \multicolumn{4}{|c}{ Delta-method } \\
& Margin & std. err. & t & \(\mathrm{P}>|\mathrm{t}|\) & [95\% conf. interval] \\
\hline _cons & 109.9214 & 1.422629 & 77.27 & 0.000 & 107.0524 & 112.7904 \\
\hline
\end{tabular}
. margins, asbalanced
Adjusted predictions \(\quad\) Number of obs \(=49\)
Model VCE: OLS
Expression: Linear prediction, predict()
At: pressure (asbalanced)
    temp (asbalanced)
\begin{tabular}{r|rrrrrr}
\hline & \multicolumn{3}{|c}{ Delta-method } \\
& Margin & std. err. & t & \(\mathrm{P}>|\mathrm{t}|\) & [95\% conf. interval] \\
\hline _cons & 115.3758 & 1.530199 & 75.40 & 0.000 & 112.2899 & 118.4618 \\
\hline
\end{tabular}

\section*{- Technical note}

Concerning how asbalanced calculations are performed, if a factor variable has \(l\) levels, then each level's coefficient contributes to the response weighted by \(1 / l\). If two factors, \(a\) and \(b\), interact, then each coefficient associated with their interaction is weighted by \(1 /\left(l_{a} \times l_{b}\right)\).

If a balanced factor interacts with a continuous variable, then each coefficient in the interaction is applied to the value of the continuous variable, and the results are weighted equally. So, if the factor being interacted has \(l_{a}\) levels, the effect of each coefficient on the value of the continuous covariate is weighted by \(1 / l_{a}\).

\section*{Balancing by standardization}

To better understand the balanced results, we can perform the balancing ourselves by using the standardizing method shown in Standardizing margins. To do that, we will input a balanced dataset and then type margins, noesample.
```

. use https://www.stata-press.com/data/r18/acmemanuf
. regress y pressure\#\#temp
(output omitted)
. drop _all
. input pressure temp
pressure temp
1. 1 1
2. 1 }
3. 1 3
4. 2 1
5. 2 2
6. 2 3
7. end
. margins, noesample
Predictive margins Number of obs = 6
Model VCE: OLS
Expression: Linear prediction, predict()

|  | MarginDelta-method <br> std. err. | t | $\mathrm{P}>\|\mathrm{t\mid}\|$ | [95\% conf. interval] |  |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| _cons | 115.3758 | 1.530199 | 75.40 | 0.000 | 112.2899 | 118.4618 |

```

We obtain the same results as previously.

\section*{Balancing nonlinear responses}

If our testing had produced a binary outcome, say, acceptable/unacceptable, rather than a continuous variable, we would type
```

. use https://www.stata-press.com/data/r18/acmemanuf, clear
. logistic acceptable pressure\#\#temp
. margins, asbalanced

```

The result of doing that would be 0.680 . If we omitted the asbalanced option, the result would have been 0.667 . The two results are so similar because acmemanuf.dta is nearly balanced.

Even though the asbalanced option can be used on both linear and nonlinear responses, such as probabilities, there is an issue of which you should be aware. The most widely used formulas for balancing responses apply the balancing to the linear prediction, average that as if it were balanced, and then apply the nonlinear transform. That is the calculation that produced 0.680 .

An alternative would be to apply the standardization method. That amounts to making the linear predictions observation by observation, applying the nonlinear transform to each, and then averaging the nonlinear result as if it were balanced. You could do that by typing
```

. use https://www.stata-press.com/data/r18/acmemanuf, clear
. logistic acceptable pressure\#\#temp
. clear
. input pressure temp
(see above for entered data)
. margins, noesample

```

The result from the standardization procedure would be 0.672 . These two ways of averaging nonlinear responses are discussed in detail in Lane and Nelder (1982) within the context of general linear models.

Concerning the method used by the asbalanced option, if your data start balanced and you have a nonlinear response, you will get different results with and without the asbalanced option!

\section*{Treating a subset of covariates as balanced}

So far, we have treated all the covariates as if they were balanced. margins will allow you to treat a subset of the covariates as balanced, too. For instance, you might be performing an experiment in which you are randomly allocating patients to a treatment arm and so want to balance on arm, but you do not want to balance the other characteristics because you want mean effects for the experiment's population.

In this example, we will imagine that the outcome of the experiment is continuous. We type
```

. use https://www.stata-press.com/data/r18/margex, clear

```
. regress y arm\#\#sex sex\#\#agegroup
. margins, at((asbalanced) arm)

If we wanted results balanced on agegroup as well, we could type . margins, at((asbalanced) arm agegroup)

If we wanted results balanced on all three covariates, we could type . margins, at((asbalanced) arm agegroup sex)
or we could type
. margins, at((asbalanced) _factor)
or we could type . margins, asbalanced

\section*{Using fvset design}

As a convenience feature, equivalent to
```

. regress y arm\#\#sex sex\#\#agegroup
. margins, at((asbalanced) arm sex)

```
is
. fvset design asbalanced arm sex
. regress y arm\#\#sex sex\#\#agegroup
- margins

The advantage of the latter is that you have to set the variables as balanced only once. This is useful when balancing is a design characteristic of certain variables and you wish to avoid accidentally treating them as unbalanced.

If you save your data after fvsetting, the settings will be remembered in future sessions. If you want to clear the setting(s), type
```

. fvset clear varlist

```

See [R] fvset.

\section*{Balancing in the presence of empty cells}

The issue of empty cells is not exclusively an issue of balancing, but there are special considerations when balancing. Empty cells are discussed generally in Estimability of margins.

An empty cell is an interaction of levels of two or more factor variables for which you have no data. Usually, margins involving empty cells cannot be estimated. When balancing, there is an alternate definition of the margin that allows the margin to be estimated. margins makes the alternate calculation when you specify the emptycells (reweight) option. By default, margins uses the emptycells(strict) option.

If you have empty cells in your data and you request margins involving the empty cells, those margins will be marked as not estimable even if you specify the asbalanced option.
```

. use https://www.stata-press.com/data/r18/estimability, clear
(margins estimability)
. regress y sex\#\#group
(output omitted)
. margins sex, asbalanced
Adjusted predictions Number of obs = 69
Model VCE: OLS
Expression: Linear prediction, predict()
At: sex (asbalanced)
group (asbalanced)

|  | MarginDelta-method <br> std. err.$\quad \mathrm{t}$ | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: | :---: |
| sex <br> Male <br> Female | 21.91389 <br> .$1.119295 \quad 19.58$ <br> (not estimable) | 0.000 | 19.67572 | 24.15206 |

```

This example is discussed in Estimability of margins, although without the asbalanced option. What is said there is equally relevant to the asbalanced case. For reasons explained there, the margin for sex \(=1\) (female) cannot be estimated.

The margin for sex \(=1\) can be estimated in the asbalanced case if you are willing to make an assumption. Remember that margins makes the balanced calculation by summing the responses associated with the levels and then dividing by the number of levels. If you specify emptycells(reweight), margins sums what is available and divides by the number available. Thus, you are assuming that, whatever the responses in the empty cells, those responses are such that they would not change the overall mean of what is observed.

The results of specifying emptycells(reweight) are
```

. margins sex, asbalanced emptycells(reweight)
Adjusted predictions Number of obs = 69
Model VCE: OLS
Expression: Linear prediction, predict()
Empty cells: reweight
At: sex (asbalanced)
group (asbalanced)

```
\begin{tabular}{c|rrrrrrr}
\hline & \multicolumn{4}{|c}{\begin{tabular}{c} 
Delta-method \\
Margin
\end{tabular}} & Ptd.err. & t & \(\mathrm{P}>|\mathrm{t}|\) \\
[95\% conf. interval] \\
\hline sex & & & & & & \\
Male & 21.91389 & 1.119295 & 19.58 & 0.000 & 19.67572 & 24.15206 \\
Female & 24.85185 & 1.232304 & 20.17 & 0.000 & 22.38771 & 27.316 \\
\hline
\end{tabular}

\section*{Obtaining margins with nested designs}

\section*{Introduction to nested designs}

Factors whose meaning depends on other factors are called nested factors, and the factors on which their meaning depends are called the nesting factors. For instance, assume that we have a sample of patients and each patient is assigned to one doctor. Then, patient is nested within doctor. Let the identifiers of the first 5 observations of our data be
\begin{tabular}{lll} 
Doctor & Patient & Name \\
\hline 1 & 1 & Fred \\
1 & 2 & Mary \\
1 & 3 & Bob \\
2 & 1 & Karen \\
2 & 2 & Hank \\
\hline
\end{tabular}

The first patient on one doctor's list has nothing whatsoever to do with the first patient on another doctor's list. The meaning of patient \(=1\) is defined only when the value of doctor is supplied.

Nested factors enter into models as interactions of nesting and nested; the nested factor does not appear by itself. We might estimate a model such as
```

. regress y ... i.doctor doctor\#patient ...

```

You do not include i.patient because the coding for patient has no meaning except within doctor. Patient 1 is Fred for doctor 1 and Karen for doctor 2, etc.
margins provides an option to help account for the structure of nested models. The within (varlist) option specifies that margins estimate and report a set of margins for the value combinations of varlist. We might type
```

. margins, within(doctor)

```

Margin calculations are performed first for doctor \(=1\), then for doctor \(=2\), and so on.
Sometimes you need to specify within(), and other times you do not. Let's consider the particular model
```

. regress y i.doctor doctor\#patient i.sex sex\#doctor\#patient

```

The guidelines are the following:
1. You may compute overall margins by typing margins.
2. You may compute overall margins within levels of a nesting factor by typing margins, within(doctor).
3. You may compute margins of a nested factor within levels of its nesting factor by typing margins patient, within(doctor).
4. You may compute margins of factors in your model, as long as the factor does not nest other factors and is not nested within other factors, by typing margins sex.
5. You may not compute margins of a nesting factor, such as margins doctor, because they are not estimable.

For examples using within(), see [R] anova.

\section*{Margins with nested designs as though the data were balanced}

To obtain margins with nested designs as though the data were balanced, the guidelines are the same as above except that 1) you add the asbalanced option and 2) whenever you do not specify within(), you specify emptycells(reweight). The updated guidelines are
1. You may compute overall margins by typing margins, asbalanced emptycells(reweight).
2. You may compute overall margins within levels of a nesting factor by typing margins, asbalanced within(doctor).
3. You may compute margins of a nested factor within levels of its nesting factor by typing margins patient, asbalanced within(doctor).
4. You may compute margins of factors in your model, as long as the factor does not nest other factors and is not nested within other factors, by typing margins sex, asbalanced emptycells(reweight).
5. You may not compute margins of a nesting factor, such as margins doctor, because they are not estimable.

Just as explained in Using fvset design, rather than specifying the asbalanced option, you may set the balancing characteristic on the factor variables once and for all by using the command fvset design asbalanced varlist.

\section*{- Technical note}

Specifying either emptycells (reweight) or within(varlist) causes margins to rebalance over all empty cells in your model. If you have interactions in your model that are not involved in the nesting, margins will lose its ability to detect estimability.

\section*{- Technical note}

Careful readers will note that the description of within(varlist) matches closely the description of over (varlist). The concept of nesting is similar to the concept of subpopulations. within() differs from over () in that it gracefully handles the missing cells when margins are computed as balanced.

\section*{Coding of nested designs}

In the Introduction to this section, we showed a coding of the nested variable patient, where the coding started over with each doctor:
\begin{tabular}{lll} 
Doctor & Patient & Name \\
\hline 1 & 1 & Fred \\
1 & 2 & Mary \\
1 & 3 & Bob \\
2 & 1 & Karen \\
2 & 2 & Hank \\
\hline
\end{tabular}

That coding style is not required. The data could just as well have been coded
\begin{tabular}{lll} 
Doctor & Patient & Name \\
\hline 1 & 1 & Fred \\
1 & 2 & Mary \\
1 & 3 & Bob \\
2 & 4 & Karen \\
2 & 5 & Hank \\
\hline
\end{tabular}
or even
\begin{tabular}{lll} 
Doctor & Patient & Name \\
\hline 1 & 1037239 & Fred \\
1 & 2223942 & Mary \\
1 & 0611393 & Bob \\
2 & 4433329 & Karen \\
2 & 6110271 & Hank \\
\hline
\end{tabular}

Actually, either of the above two alternatives is better than the first one because margins will be better able to give you feedback about estimability should you make a mistake following the guidelines. On the other hand, both of these alternatives require more memory at the estimation step. If you run short of memory, you will need to recode your patient ID to the first coding style, which you could do by typing
```

. sort doctor patient

```
. by doctor: generate newpatient = _n

Alternatively, you can set emptycells drop and continue to use your patient ID variable just as it is coded. If you do this, we recommend that you remember to type set emptycells keep when you are finished; margins is better able to determine estimability that way. If you regularly work with large nested models, you can set emptycells keep, permanently so that the setting persists across sessions. See \([R]\) set emptycells.

\section*{Special topics}

\section*{Requirements for model specification}

The results that margins reports are based on the most recently fit model or, in Stata jargon, the most recently issued estimation command. Here we discuss 1) mechanical requirements for how you specify that estimation command, 2) workarounds to use when those restrictions prove impossible, and 3) requirements for margins' predict (pred_opt) option to work.

Concerning 1, when you specify the estimation command, covariates that are logically factor variables must be Stata factor variables, and that includes indicator variables, binary variables, and dummies. It will not do to type
```

. regress y ... sex ...

```
even if sex is a \(0 / 1\) variable. You must type
```

. regress y ... i.sex ...

```

If you violate this rule, you will not get incorrect results, but you will discover that you will be unable to obtain margins on sex:
```

. margins sex
factor sex not found in list of covariates
r(111);

```

It is also important that if the same continuous variable appears in your model more than once, differently transformed, those transforms be performed via Stata's factor-variable notation. It will not do to type
```

. generate age2 = age^2
. regress y ... age age2 ...

```

You must type
```

. regress y ... age c.age\#c.age ...

```

You must do that because margins needs to know everywhere that variable appears in the model if it is to be able to set covariates to fixed values.

Concerning 2, sometimes the transformations you desire may not be achievable using the factorvariable notation; in those situations, there is a workaround. Let's assume you wish to estimate
```

. generate age1_5 = age^1.5
. regress y ... age age1_5 ...

```

There is no factor-variable notation for including age and age \({ }^{1.5}\) in a model, so obviously you are going to obtain the estimates by typing just what we have shown. In what follows, it would be okay if there are interactions of age and age1_5 with other variables specified by the factor-variable notation, so the model could just as well be
```

. regress y ... age age1_5 sex\#c.age sex\#c.age1_5 ...

```

Let's assume you have fit one of these two models. On any subsequent margins command where you leave age free to vary, there will be no issue. You can type
```

. margins sex

```
and results will be correct. Issues arise when you attempt to fix age at predetermined values. The following would produce incorrect results:
```

. margins sex, at(age=20)

```

The results would be incorrect because they leave age1_5 free to vary, and, logically, fixing age implies that age1_5 should also be fixed. Because we were unable to state the relationship between age and age1_5 using the factor-variable notation, margins does not know to fix age1_5 at \(20^{1.5}\) when it fixes age at 20 . To get the correct results, you must fix the value of age1_5 yourself:
```

. margins sex, at(age=20 age1_5=89.442719)

```

That command produces correct results. In the command, 89.442719 is \(20^{1.5}\).
In summary, when there is a functional relationship between covariates of your model and that functional relationship is not communicated to margins via the factor-variable notation, then it becomes your responsibility to ensure that all variables that are functionally related are set to the appropriate fixed values when any one of them is set to a fixed value.

Concerning 3, we wish to amend our claim that you can calculate margins for anything that predict will produce. We need to add a qualifier. Let us show you an example where the statement is not true. After regress, predict will predict something it calls \(\mathrm{pr}(a, b)\), which is the probability \(a \leq y \leq b\). Yet if we attempted to use \(\operatorname{pr}()\) with margins after estimation by regress, we would obtain
```

. margins sex, predict(pr (10,20))
prediction is a function of possibly stochastic quantities other than e(b)
r(498);

```

What we should have stated was that you can calculate margins for anything that predict will produce for which all the estimated quantities used in its calculation appear in e(V), the estimated VCE. \(\operatorname{pr}()\) is a function of \(\boldsymbol{\beta}\), the estimated coefficients, and of \(s^{2}\), the estimated variance of the residual. regress does not post the variance of the residual variance (sic) in e(V), or even estimate it, and therefore, predict \((\operatorname{pr}(10,20)\) ) cannot be specified with margins after estimation by regress.

It is unlikely that you will ever encounter these kinds of problems because there are so few predictions where the components are not posted to \(e(V)\). If you do encounter the problem, the solution may be to specify nose to suppress the standard error calculation. If the problem is not with computing the margin, but with computing its standard error, margins will report the result:
```

. margins sex, predict(pr (10,20)) nose
(output appears with SEs, tests, and CIs left blank)

```

\section*{Technical note}

Programmers: If you run into this after running an estimation command that you have written, be aware that as of Stata 11, you are supposed to set in e(marginsok) the list of options allowed with predict that are okay to use with margins. When that list is not set, margins looks for violations of its assumptions and, if it finds any, refuses to proceed.

\section*{Estimability of margins}

Sometimes margins will report that a margin cannot be estimated:
```

. use https://www.stata-press.com/data/r18/estimability, clear
(margins estimability)
. regress y sex\#\#group
(output omitted)
. margins sex
Predictive margins Number of obs = 69
Model VCE: OLS
Expression: Linear prediction, predict()

```
\begin{tabular}{c|ccccc}
\hline & Margin \begin{tabular}{c} 
Delta-method \\
std. err.
\end{tabular} & t & \(\mathrm{P}>|\mathrm{t}|\) & [95\% conf. interval] \\
\hline \begin{tabular}{c} 
sex \\
Male \\
Female
\end{tabular} & 21 \begin{tabular}{ccccc}
.8500245 & 24.71 & 0.000 & 19.30027 & 22.69973 \\
\hline
\end{tabular} &. & & & \\
\hline
\end{tabular}

In the above output, the margin for sex \(=0\) (male) is estimated, but the margin for sex \(=1\) (female) is not estimable. This occurs because of empty cells. An empty cell is an interaction of levels of two or more factor variables for which you have no data. In the example, the lack of estimability arises because we have two empty cells:
\begin{tabular}{l|rrrrr}
. table sex group, nototals \\
\hline & 1 & 2 & 3 & 4 & 5 \\
\hline Sex & & & & \\
Male & 2 & 9 & 27 & 8 & 2 \\
Female & 9 & 9 & 3 & & \\
\hline
\end{tabular}

To calculate the marginal mean response for \(\operatorname{sex}=1\), we have no responses to average over for group \(=4\) and group \(=5\). We obviously could calculate that mean for the observations that really are sex \(=1\), but remember, the marginal calculation for sex \(=1\) treats everyone as if female, and we will thus have 8 and 2 observations for which we have no basis for estimating the response.

There is no solution for this problem unless you are willing to treat the data as if it were balanced and adjust your definition of a margin; see Balancing in the presence of empty cells.

\section*{Manipulability of tests}

Manipulability is a problem that arises with some tests, and in particular, arises with Wald tests. Tests of margins are based on Wald tests, hence our interest. This is a generic issue and not specific to the margins command.

Let's understand the problem. Consider performing a test of whether some statistic \(\phi\) is 0 . Whatever the outcome of that test, it would be desirable if the outcome were the same were we to test whether the \(\operatorname{sqrt}(\phi)\) were 0 , or whether \(\phi^{2}\) were 0 , or whether any other monotonic transform of \(\phi\) were 0 (for \(\phi^{2}\), we were considering only the positive half of the number line). If a test does not have that property, it is manipulable.

Wald tests are manipulable, and that means the tests produced by margins are manipulable. You can see this for yourself by typing
. use https://www.stata-press.com/data/r18/margex, clear
. replace \(\mathrm{y}=\mathrm{y}-65\)
. regress y sex\#\#group
. margins, \(\mathrm{df}(\).
. margins, expression(predict(xb)~2)
To compare the results from the two margins commands, we added the \(\mathrm{df}(\).\() option to the\) first one, forcing it to report a \(z\) statistic even though a \(t\) statistic would have been appropriate in this case. We would prefer if the test against zero produced by margins, \(\mathrm{df}(\).\() was equal to the\) test produced by margins, expression (predict (xb) ~2). But alas, they produce different results. The first produces \(z=12.93\), and the second produces \(z=12.57\).

The difference is not much in our example, but behind the scenes, we worked to make it small. We subtracted 65 from y so that the experiment would be for a case where it might be reasonable that you would be testing against 0 . One does not typically test whether the mean income in the United States is zero or whether the mean blood pressure of live patients is zero. Had we left y as it was originally, we would have obtained \(z=190\) and \(z=96\). We did not want to show that comparison to you first because the mean of y is so far from 0 that you probably would never be testing it. The corresponding difference in \(\phi\) is tiny.

Regardless of the example, it is important that you base your tests in the metric where the likelihood surface is most quadratic. For further discussion on manipulability, see Manipulability in [R] predictnl.

This manipulability is not limited to Wald tests after estimation; you can also see the manipulability of results produced by linear regression just by applying nonlinear transforms to a covariate (Phillips and Park 1988; Gould 1996).

\section*{Using margins after the estimates use command}

Assume you fit and used estimates save (see [R] estimates save) to save the estimation results:
```

. regress y sex\#\#group age c.age*c.age if site==1
....
. estimates save mymodel
file mymodel.ster saved

```

Later, perhaps in a different Stata session, you reload the estimation results by typing
```

. estimates use mymodel

```

You plan to use margins with the reloaded results. You must remember that margins bases its results not only on the current estimation results but also on the current data in memory. Before you can use margins, you must reload the dataset on which you fit the model or, if you wish to produce standardized margins, some other dataset.
```

. use mydata, clear
(data for fitting models)

```

If the dataset you loaded contained the data for standardization, you can stop reading; you know that to produce standardized margins, you need to specify the noesample option.

We reloaded the original data and want to produce margins for the estimation sample. In addition to the data, margins requires that e (sample) be set, as margins will remind us:
```

. margins sex
e(sample) does not identify the estimation sample
r(322);

```

The best solution is to use estimates esample to rebuild e(sample):
```

. estimates esample: y sex group age if site==1

```

If we knew we had no missing values in y and the covariates, we could type
```

. estimates esample: if site==1

```

Either way, margins would now work:
```

. margins sex
(usual output appears)

```

There is an alternative. We do not recommend it, but we admit that we have used it. Rather than rebuilding e(sample), you can use margins' noesample option to tell margins to skip using e (sample). You could then specify the appropriate if statement (if necessary) to identify the estimation sample:
```

. estimates use mymodel
. use mydata, clear
(data for fitting models)
. margins sex if !missing(y, sex, group age) \& site==1, noesample
(usual output appears)

```

In the above, we are not really running on a sample different from the estimation sample; we are merely using noesample to fool margins, and then we are specifying on the margins command the conditions equivalent to re-create e(sample).

If we wish to obtain vce (unconditional) results, however, noesample will be insufficient. We must also specify the force option,
```

. margins sex if !missing(y, sex, group age) \& site==1,
> vce(unconditional) noesample force
(usual output appears)

```

Regardless of the approach you choose-resetting e(sample) or specifying noesample and possibly force-make sure you are right. In the vce (delta) case, you want to be right to ensure that you obtain the results you want. In the vce (unconditional) case, you need to be right because otherwise results will be statistically invalid.

\section*{Syntax of at()}

In at (atspec), atspec may contain one or more of the following specifications:
```

varlist
(stat) varlist
varname = \#
varname $=$ (numlist $)$
varname $=$ generate $(\exp )$

```
where
1. varnames must be covariates in the previously fit model (estimation command).
2. Variable names (whether in varname or varlist) may be continuous variables, factor variables, or specific level variables, such as age, group, or 3.group.
3. varlist may also be one of three standard lists:
a. _all (all covariates),
b. - factor (all factor-variable covariates), or
c. -continuous (all continuous covariates).
4. stat can be any of the following:
\begin{tabular}{|c|c|c|}
\hline stat & Description & Variables allowed \\
\hline asobserved & at observed values in the sample (default) & all \\
\hline mean & means (default for varlist) & all \\
\hline median & medians & continuous \\
\hline p1 & 1st percentile & continuous \\
\hline p2 & 2nd percentile & continuous \\
\hline & 3rd-49th percentiles & continuous \\
\hline p50 & 50th percentile (same as median) & continuous \\
\hline & 51st-97th percentiles & continuous \\
\hline p98 & 98th percentile & continuous \\
\hline p99 & 99th percentile & continuous \\
\hline min & minimums & continuous \\
\hline max & maximums & continuous \\
\hline zero & fixed at zero & continuous \\
\hline base & base level & factors \\
\hline asbalanced & all levels equally probable and sum to 1 & factors \\
\hline
\end{tabular}

Any stat except zero, base, and asbalanced may be prefixed with an o to get the overall statistic-the sample over all over() groups. For example, omean, omedian, and op25. Overall statistics differ from their correspondingly named statistics only when the over() or within() option is specified. When no stat is specified, mean is assumed. If stat is not followed by a varlist, stat is ignored.
atspec can involve multiple settings for one covariate as well as settings for multiple covariates. The following rules are applied when more than one covariate or value is included:
1. When more than one covariate is referenced in atspec but each covariate is set to only one value, all settings are applied in combination. For example, at ( \(x 1=5 \times 2=0\) ) results in margins being estimated under one scenario, with \(x 1\) set to 5 , \(x 2\) set to 0 , and all other covariates set to their observed values (the default).
2. When multiple values are specified for a covariate, the covariate will be set to each of the values in turn. For example, at \((x 1=5 \mathrm{x} 1=10)\) or, equivalently, at \((\mathrm{x} 1=(510)\) ) specifies that x 1 be set first to 5 and then to 10 .
3. When multiple values are specified for more than one covariate, all possible combinations of settings are applied in turn. For example, at ( \(\mathrm{x} 1=\binom{5}{10} \mathrm{x} 2=\left(\begin{array}{ll}0 & 1\end{array}\right)\) ) results in margins being estimated under four scenarios: \((\mathrm{x} 1=5 \mathrm{x} 2=0)\), \((\mathrm{x} 1=5 \mathrm{x} 2=1)\), \((\mathrm{x} 1=10\) \(\mathrm{x} 2=0)\), and \((\mathrm{x} 1=10 \mathrm{x} 2=1)\).
4. Settings may be specified for groups of covariates using three general varlists__all, _factor, and _continuous. When atspec includes both specifications with general varlists and specifications with named covariates, the specifications for named covariates take precedence over general ones. For example, at (x1=10 (means) _all) sets \(x 1\) to 10 while setting all other covariates to their means.
5. Only one (stat) varlist specification can be applied to a covariate. If more than one is specified, the rightmost specification is respected. For example, at ((means) x1 x2 (medians) x1 x 2 ) sets both x 1 and x 2 to their medians.
6. When both a (stat) specification and another specification are included for a named covariate, the other specification takes precedence. For example, at (x1=5 (means) x1) sets x1 to 5 .
In addition, at () can be repeated. When multiple at () options are specified, atspecs are processed sequentially. For instance, at \((x 1=5)\) at \((x 2=0)\) results in margins being estimated under two scenarios. The first sets \(x 1\) to 5 and all other covariates, including \(x 2\), to their observed values. The second sets \(x 2\) to 0 and all other covariates to their observed values. Note that this is different from the single at \((x 1=5 \times 2=0)\) specification, which sets x 1 and x 2 to the specified values simultaneously.

\section*{Estimation commands that may be used with margins}
margins may be used after most estimation commands.
margins cannot be used after estimation commands that do not produce full variance matrices, such as exlogistic and expoisson (see [R] exlogistic and [R] expoisson).
margins is all about covariates and cannot be used after estimation commands that do not post the covariates, which eliminates gmm (see [R]gmm).
margins cannot be used after some estimation commands that have an odd data organization, such as nlogit. However, margins has been specially adapted to work with many choice model estimation commands; see [CM] margins.

\section*{Video examples}

Introduction to margins, part 1: Categorical variables
Introduction to margins, part 2: Continuous variables
Introduction to margins, part 3: Interactions

\section*{Glossary}
adjusted mean. A margin when the response is the linear predictor from linear regression, ANOVA, etc. For some authors, adjusting also implies adjusting for unbalanced data. See Obtaining margins of responses and see Obtaining margins as though the data were balanced.
average marginal effect. See marginal effect and average marginal effect.
average partial effect. See partial effect and average partial effect.
conditional margin. A margin when the response is evaluated at fixed values of all the covariates. If any covariates are left to vary, the margin is called a predictive margin.
effect. The effect of \(x\) is the derivative of the response with respect to covariate \(x\), or it is the difference in responses caused by a discrete change in \(x\). Also see marginal effect.

The effect of \(x\) measures the change in the response for a change in \(x\). Derivatives or differences might be reported as elasticities. If \(x\) is continuous, the effect is measured continuously. If \(x\) is a factor, the effect is measured with respect to each level of the factor and may be calculated as a discrete difference or as a continuous change, as measured by the derivative. margins calculates the discrete difference by default and calculates the derivative if the continuous option is specified.
elasticity and semielasticity. The elasticity of \(y\) with respect to \(x\) is \(d(\ln y) / d(\ln x)=(x / y) \times(d y / d x)\), which is approximately equal to the proportional change in \(y\) for a proportional change in \(x\).
The semielasticity of \(y\) with respect to \(x\) is either 1\() d y / d(\ln x)=x \times(d y / d x)\) or 2\() d(\ln y) / d x=\) \((1 / y) \times(d y / d x)\), which is approximately 1\()\) the change in \(y\) for a proportional change in \(x\) or 2) the proportional change in \(y\) for a change in \(x\).
margins uses model predictions or user-specified statistics in expression() in place of \(y\) above.
empty cell. An interaction of levels of two or more factor variables for which you have no data. For instance, you have sex interacted with group in your model, and in your data there are no females in group 1. Empty cells affect which margins can be estimated; see Estimability of margins.
estimability. Estimability concerns whether a margin can be uniquely estimated (identified); see Estimability of margins.
estimated marginal mean. This is one of the few terms that has the same meaning across authors. An estimated marginal mean is a margin assuming the levels of each factor covariate are equally likely (balanced), including interaction terms. This is obtained using margins' asbalanced option. In addition, there is an alternate definition of estimated marginal mean in which margins involving empty cells are redefined so that they become estimable. This is invoked by margins' emptycells (reweight) option. See Balancing in the presence of empty cells.
least-squares mean. Synonym for estimated marginal mean.
margin. A statistic calculated from predictions or other statistics of a previously fit model at fixed values of some covariates and averaging or otherwise integrating over the remaining covariates. The prediction or other statistic on which the margin is based is called the response.
If all the covariates are fixed, then the margin is called a conditional margin. If any covariates are left to vary, the margin is called a predictive margin.

In this documentation, we divide margins on the basis of whether the statistic is a response or a derivative of a response; see Obtaining margins of responses and Obtaining margins of derivatives of responses.
marginal effect and average marginal effect. The marginal effect of \(x\) is the margin of the effect of \(x\). The term is popular with social scientists, and because of that, you might think the word marginal in marginal effect means derivative because of terms like marginal cost and marginal revenue. Marginal used in that way, however, refers to the derivative of revenue and the derivative of cost; it refers to the numerator, whereas marginal effect refers to the denominator. Moreover, effect is already a derivative or difference.
Some researchers interpret marginal in marginal effect to mean instantaneous, and thus a marginal effect is the instantaneous derivative rather than the discrete first difference, corresponding to margins' continuous option. Researchers who use marginal in this way refer to the discrete difference calculation of an effect as a partial effect.
Other researchers define marginal effect to be the margin when all covariates are held fixed and the average marginal effect when some covariates are not fixed.
out-of-sample prediction. Predictions made in one dataset using the results from a model fit on another. Sample here refers to the sample on which the model was fit, and out-of-sample refers to the dataset on which the predictions are made.
partial effect and average partial effect. Some authors restrict the term marginal effect to mean derivatives and use the term partial effect to denote discrete differences; see marginal effect and average marginal effect.
population marginal mean. The theoretical (true) value that is estimated by estimated marginal mean. We avoid this term because it can be confused with the concept of a population in survey statistics, with which the population marginal mean has no connection.
posting results, posting margins. A Stata concept having to do with storing the results from the margins command in \(e()\) so that those results can be used as if they were estimation results, thus allowing the subsequent use of postestimation commands, such as test, testnl, lincom, and nlcom (see \([R]\) test, \([R]\) testnl, \([R]\) lincom, and \([R]\) nlcom). This is achieved by specifying margins' post option. See Example 10: Testing margins-contrasts of margins.
predictive margin. A margin in which all the covariates are not fixed. When all covariates are fixed, it is called a conditional margin.
recycled prediction. A synonym for predictive margin.
response. A prediction or other statistic derived from combining the parameter estimates of a fitted model with data or specified values on covariates. Derivatives of responses are themselves responses. Responses are what we take margins of.
standardized margin. The margin calculated on data different from the data used to fit the model. The term standardized is usually reserved for situations in which the alternate population is a reference population, which may be real or artificial, and which is treated as fixed.
subpopulation. A subset of your sample that represents a subset of the population, such as the males in a sample of people. In survey contexts when it is desired to account for sampling of the covariates, standard errors for marginal statistics and effects need to account for both the population and the subpopulation. This is accomplished by specifying the vce (unconditional) option and one of the subpop() or over () options. In fact, the above is allowed even when your data are not svyset because vce(unconditional) implies that the sample represents a population.

\section*{Stored results}
margins stores the following in \(r()\) :
Scalars
```

r(N)
r(N_sub)
r(N_clust)
r(N_psu)
r(N_strata)
r(df_r)
r(N_poststrata)
r(k_predict)
r(k_margins)
r(k_by)
r(k_at)
r(level)

```

Macros
```

r(cmd)
r(cmdline)
r(est_cmd)
r(est_cmdline)
r(title)
r(subpop)
r(model_vce)
r(model_vcetype)
r(vce)
r(vcetype)
r(clustvar)
r(margins)
r(predict\#_opts)
r(predict\#_label)
r(expression)
r(xvars)
r(derivatives)
r(over)
r(within)
r(by)
r(by\#)
r(atstats\#)
r(emptycells)
r(mcmethod)
r(mcadjustall)

```

Matrices
```

r(b)
r(V)
r(Jacobian)
r(_N)
r(at)
r(chainrule)
r(error)

```
r(table)
number of observations
subpopulation observations
number of clusters
number of sampled PSUs, survey data only
number of strata, survey data only
variance degrees of freedom, survey data only
number of post strata, survey data only
number of predict() options
number of terms in marginlist
number of subpopulations
number of at() options
confidence level of confidence intervals
margins
command as typed
\(e(\mathrm{cmd})\) from original estimation results
\(e\) (cmdline) from original estimation results
title in output
subspec from subpop()
vcetype from estimation command
Std. err. title from estimation command
vcetype specified in vce()
title used to label Std. err.
name of cluster variable
marginlist
the \#th predict() option
label from the \#th predict() option
response expression
varlist from dydx(), dyex(), eydx(), or eyex()
"", "dy/dx", "dy/ex", "ey/dx", "ey/ex"
varlist from over()
varlist from within()
union of \(r\) (over) and \(r\) (within) lists
interaction notation identifying the \#th subpopulation
the \#th at() specification
empspec from emptycells()
method from mcompare()
adjustall or empty
estimates
variance-covariance matrix of the estimates
Jacobian matrix
sample size corresponding to each margin estimate
matrix of values from the at() options
chain rule information from the fitted model
margin estimability codes;
0 means estimable,
8 means not estimable
matrix containing the margins with their standard errors, test statistics, \(p\)-values, and confidence intervals
margins with the post option also stores the following in e():
Scalars
```

e(N)
e(N_sub)
e(N_clust)
e(N_psu)
e(N_strata)
e(df_r)
e(N_poststrata)
e(k_predict)
e(k_margins)
e(k_by)
e(k_at)

```

Macros
    e(cmd)
    e(cmdline)
    e(est_cmd)
    e(est_cmdline)
    e(wtype)
    e(wexp)
    e(title)
    e(subpop)
    e(model_vce)
    e(model_vcetype)
    e(vce)
    e(vcetype)
    e(clustvar)
    e(properties)
    e(margins)
    e(asbalanced)
    e(asobserved)
    e(predict\#_opts)
    e(predict\#_label)
    e(expression)
    e(xvars)
    e(derivatives)
    e(over)
    e(within)
    e(by)
    e(by\#)
    e(atstats\#)
    e(emptycells)
Matrices
    e(b)
    e(V)
    e(Jacobian)
    e(_N)
    e(error)
    e(at)
    e(chainrule)

Functions
e(sample)
number of observations subpopulation observations number of clusters
number of sampled PSUs, survey data only
number of strata, survey data only
variance degrees of freedom, survey data only
number of post strata, survey data only
number of predict() options
number of terms in marginlist
number of subpopulations
number of at () options

\section*{margins}
command as typed
\(e(c m d)\) from original estimation results
\(e\) (cmdline) from original estimation results
weight type
weight expression
title in estimation output
subspec from subpop()
vcetype from estimation command
Std. err. title from estimation command
vcetype specified in vce()
title used to label Std. err.
name of cluster variable
\(\mathrm{b} V\), or just b if nose is specified
marginlist
factor variables fvset as asbalanced
factor variables fvset as asobserved
the \#th predict() option
label from the \#th predict() option
prediction expression
varlist from dydx(), dyex(), eydx(), or eyex()
"", "dy/dx", "dy/ex", "ey/dx", "ey/ex"
varlist from over()
varlist from within()
union of \(r\) (over) and \(r\) (within) lists
interaction notation identifying the \#th subpopulation
the \#th at() specification
empspec from emptycells()
estimates
variance-covariance matrix of the estimates
Jacobian matrix
sample size corresponding to each margin estimate
error code corresponding to \(e(b)\)
matrix of values from the at () options
chain rule information from the fitted model
marks estimation sample

\section*{Methods and formulas}

Margins are statistics calculated from predictions of a previously fit model at fixed values of some covariates and averaging or otherwise integrating over the remaining covariates. There are many names for the different statistics that margins can compute: estimates marginal means (see Searle,

Speed, and Milliken [1980]), predictive margins (see Graubard and Korn [2004]), marginal effects (see Greene [2018]), and average marginal/partial effects (see Wooldridge [2010] and Bartus [2005]).

Methods and formulas are presented under the following headings:

\author{
Notation \\ Marginal effects \\ Fixing covariates and balancing factors \\ Estimable functions \\ Standard errors conditional on the covariates \\ Unconditional standard errors
}

\section*{Notation}

Let \(\boldsymbol{\theta}\) be the vector of parameters in the current model fit, let \(\mathbf{z}\) be a vector of covariate values, and let \(f(\mathbf{z}, \boldsymbol{\theta})\) be a scalar-valued function returning the value of the predictions of interest. The following table illustrates the parameters and default prediction for several of Stata's estimation commands.
\begin{tabular}{llll} 
Command & \(\boldsymbol{\theta}\) & \(\mathbf{z}\) & \(f(\mathbf{z}, \boldsymbol{\theta})\) \\
\hline regress & \(\boldsymbol{\beta}\) & \(\mathbf{x}\) & \(\mathbf{x} \boldsymbol{\beta}\) \\
cloglog & \(\boldsymbol{\beta}\) & \(\mathbf{x}\) & \(1-e^{-e^{x \beta}}\) \\
logit & \(\boldsymbol{\beta}\) & \(\mathbf{x}\) & \(1 /\left(1+e^{-x \beta}\right)\) \\
poisson & \(\boldsymbol{\beta}\) & \(\mathbf{x}\) & \(e^{x \beta}\) \\
probit & \(\boldsymbol{\beta}\) & \(\mathbf{x}\) & \(\Phi(\mathbf{x} \boldsymbol{\beta})\) \\
biprobit & \(\boldsymbol{\beta}_{1}, \boldsymbol{\beta}_{2}, \rho\) & \(\mathbf{x}\) \\
mlogit & \(\boldsymbol{\beta}_{1}, \boldsymbol{\beta}_{2}, \ldots, \boldsymbol{x}_{k}\) & \(\mathbf{x}\) & \(\Phi_{2}\left(\mathbf{x}_{1} \boldsymbol{\beta}_{1}, \mathbf{x}_{2} \boldsymbol{\beta}_{2}, \rho\right)\) \\
nbreg & \(\boldsymbol{\beta}, \ln \alpha\) & \(\mathbf{x}\) & \(e^{-x \beta_{1}} /\left(\sum_{i} e^{-x \beta_{i}}\right)\) \\
\end{tabular}
\(\Phi()\) and \(\Phi_{2}()\) are cumulative distribution functions: \(\Phi()\) for the standard normal distribution and \(\Phi_{2}()\) for the standard bivariate normal distribution.
margins computes estimates of
\[
p(\boldsymbol{\theta})=\frac{1}{M_{S_{\mathrm{p}}}} \sum_{j=1}^{M} \delta_{j}\left(S_{\mathrm{p}}\right) f\left(\mathbf{z}_{j}, \boldsymbol{\theta}\right)
\]
where \(\delta_{j}\left(S_{\mathrm{p}}\right)\) identifies elements within the subpopulation \(S_{\mathrm{p}}\) (for the prediction of interest),
\[
\delta_{j}\left(S_{\mathrm{p}}\right)= \begin{cases}1, & j \in S_{\mathrm{p}} \\ 0, & j \notin S_{\mathrm{p}}\end{cases}
\]
\(M_{S_{\mathrm{p}}}\) is the subpopulation size,
\[
M_{S_{\mathrm{p}}}=\sum_{j=1}^{M} \delta_{j}\left(S_{\mathrm{p}}\right)
\]
and \(M\) is the population size.

Let \(\widehat{\boldsymbol{\theta}}\) be the vector of parameter estimates. Then, margins estimates \(p(\boldsymbol{\theta})\) via
\[
\widehat{p}=\frac{1}{w} \sum_{j=1}^{N} \delta_{j}\left(S_{\mathrm{p}}\right) w_{j} f\left(\mathbf{z}_{j}, \widehat{\boldsymbol{\theta}}\right)
\]
where
\[
w .=\sum_{j=1}^{N} \delta_{j}\left(S_{\mathrm{p}}\right) w_{j}
\]
\(\delta_{j}\left(S_{\mathrm{p}}\right)\) indicates whether observation \(j\) is in subpopulation \(S_{\mathrm{p}}, w_{j}\) is the weight for the \(j\) th observation, and \(N\) is the sample size.

\section*{Marginal effects}
margins also computes marginal and partial effects. For the marginal effect of continuous covariate \(x\), margins computes
\[
\widehat{p}=\frac{1}{w} \sum_{j=1}^{N} \delta_{j}\left(S_{\mathrm{p}}\right) w_{j} h\left(\mathbf{z}_{j}, \widehat{\boldsymbol{\theta}}\right)
\]
where
\[
h(\mathbf{z}, \boldsymbol{\theta})=\frac{\partial f(\mathbf{z}, \boldsymbol{\theta})}{\partial x}
\]

The marginal effect for level \(k\) of factor variable \(A\) is the simple contrast (also known as difference) comparing its margin with the margin at the base level.
\[
h(\mathbf{z}, \boldsymbol{\theta})=f(\mathbf{z}, \boldsymbol{\theta} \mid A=k)-f(\mathbf{z}, \boldsymbol{\theta} \mid A=\text { base })
\]

Elasticities and semielasticities given by eyex() and dyex() are only available for continuous covariates. Elasticities are computed using
\[
h(\mathbf{z}, \boldsymbol{\theta})=\frac{\partial \ln \{f(\mathbf{z}, \boldsymbol{\theta})\}}{\partial \ln (x)}
\]
and semielasticities, dyex(), using
\[
h(\mathbf{z}, \boldsymbol{\theta})=\frac{\partial f(\mathbf{z}, \boldsymbol{\theta})}{\partial \ln (x)}
\]

Semielasticities of the form eydx() for continuous covariates are computed using
\[
h(\mathbf{z}, \boldsymbol{\theta})=\frac{\partial \ln \{f(\mathbf{z}, \boldsymbol{\theta})\}}{\partial x}
\]
and for discrete covariates using
\[
h(\mathbf{z}, \boldsymbol{\theta})=\ln \{f(\mathbf{z}, \boldsymbol{\theta} \mid A=k)\}-\ln \{f(\mathbf{z}, \boldsymbol{\theta} \mid A=\text { base })\}
\]

\section*{Fixing covariates and balancing factors}
margins controls the values in each \(\mathbf{z}\) vector through the marginlist, the at () option, the atmeans option, and the asbalanced and emptycells() options. Suppose z is composed of the elements from the equation specification

\section*{A\#\#B x}
where A is a factor variable with \(a\) levels, B is a factor variable with \(b\) levels, and x is a continuous covariate. To simplify the notation for this discussion, assume the levels of A and B start with 1 and are contiguous. Then
\[
\mathbf{z}=\left(A_{1}, \ldots, A_{a}, B_{1}, \ldots, B_{b}, A_{1} B_{1}, A_{1} B_{2}, \ldots, A_{a} B_{b}, x, 1\right)
\]
where \(A_{i}, B_{j}\), and \(A_{i} B_{j}\) represent the indicator values for the factor variables A and B and the interaction A\#B.

When factor A is in the marginlist, margins replaces A with \(i\) and then computes the mean of the subsequent prediction, for \(i=1, \ldots, a\). When the interaction term A\#B is in the marginlist, margins replaces A with \(i\) and B with \(j\), and then computes the mean of the subsequent prediction, for all combinations of \(i=1, \ldots, a\) and \(j=1, \ldots, b\).

The at () option sets model covariates to fixed values. For example, at ( \(x=15\) ) causes margins to temporarily set \(x\) to 15 for each observation in the dataset before computing any predictions. Similarly, at ((median) x) causes margins to temporarily set \(x\) to the median of x using the current dataset.

When factor variable A is specified as asbalanced, margins sets each \(A_{i}\) to \(1 / a\). Thus, each \(\mathbf{z}\) vector will look like
\[
\mathbf{z}=\left(1 / a, \ldots, 1 / a, B_{1}, \ldots, B_{b}, B_{1} / a, B_{2} / a, \ldots, B_{b} / a, x, 1\right)
\]

If \(B\) is also specified as asbalanced, then each \(B_{j}\) is set to \(1 / b\), and each \(\mathbf{z}\) vector will look like
\[
\mathbf{z}=(1 / a, \ldots, 1 / a, 1 / b, \ldots, 1 / b, 1 / a b, 1 / a b, \ldots, 1 / a b, x, 1)
\]

If emptycells (reweight) is also specified, then margins uses a different balancing weight for each element of \(\mathbf{z}\), depending on how many empty cells the element is associated with. Let \(\delta_{i j}\) indicate that the \(i j\) th cell of A\#B was observed in the estimation sample.
\[
\delta_{i j}= \begin{cases}0, & \mathrm{~A}=i \text { and } \mathrm{B}=j \text { was an empty cell } \\ 1, & \text { otherwise }\end{cases}
\]

For the grand margin, the affected elements of \(\mathbf{z}\) and their corresponding balancing weights are
\[
\begin{aligned}
& A_{i}=\frac{\sum_{j} \delta_{i j}}{\sum_{k} \sum_{j} \delta_{k j}} \\
& B_{j}=\frac{\sum_{i} \delta_{i j}}{\sum_{i} \sum_{k} \delta_{i k}}
\end{aligned}
\]
\[
A_{i} B_{j}=\frac{\delta_{i j}}{\sum_{k} \sum_{l} \delta_{k l}}
\]

For the \(j\) th margin of B , the affected elements of \(\mathbf{z}\) and their corresponding balancing weights are
\[
\begin{gathered}
A_{i}=\frac{\delta_{i j}}{\sum_{k} \delta_{k j}} \\
B_{l}= \begin{cases}1, & \text { if } l=j \text { and not all } \delta_{i j} \text { are zero } \\
0, & \text { otherwise }\end{cases} \\
A_{i} B_{l}=\frac{\delta_{i l}}{\sum_{k} \delta_{k l}} B_{l}
\end{gathered}
\]

\section*{Estimable functions}

The fundamental idea behind estimable functions is clearly defined in the statistical literature for linear models; see Searle and Gruber (2017). Assume that we are working with the following linear model:
\[
\mathbf{y}=\mathbf{X b}+\mathbf{e}
\]
where \(\mathbf{y}\) is an \(N \times 1\) vector of responses, \(\mathbf{X}\) is an \(N \times p\) matrix of covariate values, \(\mathbf{b}\) is a \(p \times 1\) vector of coefficients, and \(\mathbf{e}\) is a vector of random errors. Assuming a constant variance for the random errors, the normal equations for the least-squares estimator, \(\widehat{\mathbf{b}}\), are
\[
\mathbf{X}^{\prime} \mathbf{X} \widehat{\mathbf{b}}=\mathbf{X}^{\prime} \mathbf{y}
\]

When \(\mathbf{X}\) is not of full column rank, we will need a generalized inverse (g-inverse) of \(\mathbf{X}^{\prime} \mathbf{X}\) to solve for \(\widehat{\mathbf{b}}\). Let \(\mathbf{G}\) be a g-inverse of \(\mathbf{X}^{\prime} \mathbf{X}\).

Searle and Gruber (2017) defines a linear function of the parameters as estimable if it is identically equal to some linear function of the expected values of the \(\mathbf{y}\) vector. Let \(\mathbf{H}=\mathbf{G X}^{\prime} \mathbf{X}\). Then this definition simplifies to the following rule:
\[
\mathbf{z b} \text { is estimable if } \mathbf{z}=\mathbf{z H}
\]
margins generalizes this to nonlinear functions by assuming the prediction function \(f(\mathbf{z}, \boldsymbol{\theta})\) is a function of one or more of the linear predictions from the equations in the model that \(\boldsymbol{\theta}\) represents.
\[
f(\mathbf{z}, \boldsymbol{\theta})=h\left(\mathbf{z}_{1} \boldsymbol{\beta}_{1}, \mathbf{z}_{2} \boldsymbol{\beta}_{2}, \ldots, \mathbf{z}_{k} \boldsymbol{\beta}_{k}\right)
\]
\(\mathbf{z}_{i} \boldsymbol{\beta}_{i}\) is considered estimable if \(\mathbf{z}_{i}=\mathbf{z}_{i} \mathbf{H}_{i}\), where \(\mathbf{H}_{i}=\mathbf{G}_{i} \mathbf{X}_{i}^{\prime} \mathbf{X}_{i}, \mathbf{G}_{i}\) is a \(g\)-inverse for \(\mathbf{X}_{i}^{\prime} \mathbf{X}_{i}\), and \(\mathbf{X}_{i}\) is the matrix of covariates from the \(i\) th equation of the fitted model. margins considers \(p(\boldsymbol{\theta})\) to be estimable if every \(\mathbf{z}_{i} \boldsymbol{\beta}_{i}\) is estimable.

\section*{Standard errors conditional on the covariates}

By default, margins uses the delta method to estimate the variance of \(\widehat{p}\).
\[
\widehat{\operatorname{Var}}(\widehat{p} \mid \mathbf{z})=\mathbf{v}^{\prime} \mathbf{V} \mathbf{v}
\]
where \(\mathbf{V}\) is a variance estimate for \(\widehat{\boldsymbol{\theta}}\) and
\[
\mathbf{v}=\left.\frac{\partial \widehat{p}}{\partial \boldsymbol{\theta}}\right|_{\theta=\widehat{\theta}}
\]

This variance estimate is conditional on the \(\mathbf{z}\) vectors used to compute the marginalized predictions.

\section*{Unconditional standard errors}
margins with the vce (unconditional) option uses linearization to estimate the unconditional variance of \(\widehat{\boldsymbol{\theta}}\). Linearization uses the variance estimator for the total of a score variable for \(\widehat{p}\) as an approximate estimator for \(\operatorname{Var}(\widehat{p})\); see [SVY] Variance estimation. margins requires that the model was fit using some form of linearized variance estimator and that predict, scores computes the appropriate score values for the linearized variance estimator.

The score for \(\widehat{p}\) from the \(j\) th observation is given by
\[
s_{j}=\frac{\partial \widehat{p}}{\partial w_{j}}=-\frac{\delta_{j}\left(S_{\mathrm{p}}\right)}{w .} \widehat{p}+\frac{\delta_{j}\left(S_{\mathrm{p}}\right)}{w .} f\left(\mathbf{z}_{j}, \widehat{\boldsymbol{\theta}}\right)+\frac{1}{w .} \sum_{i=1}^{N} \delta_{i}\left(S_{\mathrm{p}}\right) w_{i} \frac{\partial f\left(\mathbf{z}_{i}, \widehat{\boldsymbol{\theta}}\right)}{\partial w_{j}}
\]

The remaining partial derivative can be decomposed using the chain rule.
\[
\frac{\partial f\left(\mathbf{z}_{i}, \widehat{\boldsymbol{\theta}}\right)}{\partial w_{j}}=\left(\left.\frac{\partial f\left(\mathbf{z}_{i}, \boldsymbol{\theta}\right)}{\partial \boldsymbol{\theta}}\right|_{\theta=\widehat{\theta}}\right)\left(\frac{\partial \widehat{\boldsymbol{\theta}}}{\partial w_{j}}\right)^{\prime}
\]

This is the inner product of two vectors, the second of which is not a function of the \(i\) index. Thus, the score is
\[
s_{j}=-\frac{\delta_{j}\left(S_{\mathrm{p}}\right)}{w .} \widehat{p}+\frac{\delta_{j}\left(S_{\mathrm{p}}\right)}{w .} f\left(\mathbf{z}_{j}, \widehat{\boldsymbol{\theta}}\right)+\left(\left.\frac{\partial \widehat{p}}{\partial \boldsymbol{\theta}}\right|_{\theta=\widehat{\theta}}\right)\left(\frac{\partial \widehat{\boldsymbol{\theta}}}{\partial w_{j}}\right)^{\prime}
\]

If \(\widehat{\boldsymbol{\theta}}\) was derived from a system of equations (such as in linear regression or maximum likelihood estimation), then \(\widehat{\boldsymbol{\theta}}\) is the solution to
\[
\mathbf{G}(\boldsymbol{\theta})=\sum_{j=1}^{N} \delta_{j}\left(S_{\mathrm{m}}\right) w_{j} \mathbf{g}\left(\boldsymbol{\theta}, \mathbf{y}_{j}, \mathbf{x}_{j}\right)=\mathbf{0}
\]
where \(S_{\mathrm{m}}\) identifies the subpopulation used to fit the model, \(\mathbf{g}()\) is the model's gradient function, and \(\mathbf{y}_{j}\) and \(\mathbf{x}_{j}\) are the values of the dependent and independent variables for the \(j\) th observation. We can use linearization to derive a first-order approximation for \(\partial \widehat{\boldsymbol{\theta}} / \partial w_{j}\).
\[
\mathbf{G}(\widehat{\boldsymbol{\theta}}) \approx \mathbf{G}\left(\boldsymbol{\theta}_{0}\right)+\left.\frac{\partial \mathbf{G}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right|_{\theta=\theta_{0}}\left(\widehat{\boldsymbol{\theta}}-\boldsymbol{\theta}_{0}\right)
\]

Let \(\mathbf{H}\) be the Hessian matrix
\[
\mathbf{H}=\left.\frac{\partial \mathbf{G}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right|_{\theta=\theta_{0}}
\]

Then
\[
\widehat{\boldsymbol{\theta}} \approx \boldsymbol{\theta}_{0}+(-\mathbf{H})^{-1} \mathbf{G}\left(\boldsymbol{\theta}_{0}\right)
\]
and
\[
\left.\frac{\partial \widehat{\boldsymbol{\theta}}}{\partial w_{j}} \approx(-\mathbf{H})^{-1} \frac{\partial \mathbf{G}(\boldsymbol{\theta})}{\partial w_{j}}\right|_{\theta=\widehat{\boldsymbol{\theta}}}=(-\mathbf{H})^{-1} \delta_{j}\left(S_{\mathrm{m}}\right) \mathbf{g}\left(\widehat{\boldsymbol{\theta}}, \mathbf{y}_{j}, \mathbf{x}_{j}\right)
\]

The computed value of the score for \(\widehat{p}\) for the \(j\) th observation is
\[
s_{j}=\mathbf{v}^{\prime} \mathbf{u}_{j}
\]
where
\[
\mathbf{v}=\left[\begin{array}{c}
-\frac{\widehat{p}}{w .} \\
\frac{1}{w .} \\
\frac{\partial \widehat{p}}{\partial \widehat{\boldsymbol{\theta}}}(-\mathbf{H})^{-1}
\end{array}\right]
\]
and
\[
\mathbf{u}_{j}=\left[\begin{array}{c}
\delta_{j}\left(S_{\mathrm{p}}\right) \\
\delta_{j}\left(S_{\mathrm{p}}\right) f\left(\mathbf{z}_{j}, \widehat{\boldsymbol{\theta}}\right) \\
\delta_{j}\left(S_{\mathrm{m}}\right) \mathbf{g}\left(\widehat{\boldsymbol{\theta}}, \mathbf{y}_{j}, \mathbf{x}_{j}\right)
\end{array}\right]
\]

Thus, the variance estimate for \(\widehat{p}\) is
\[
\widehat{\operatorname{Var}}(\widehat{p})=\mathbf{v}^{\prime} \widehat{\operatorname{Var}}(\widehat{\mathbf{U}}) \mathbf{v}
\]
where
\[
\widehat{\mathbf{U}}=\sum_{j=1}^{N} w_{j} \mathbf{u}_{j}
\]
margins uses the model-based variance estimates for \((-\mathbf{H})^{-1}\) and the scores from predict for \(\mathbf{g}\left(\widehat{\boldsymbol{\theta}}, \mathbf{y}_{j}, \mathbf{x}_{j}\right)\).

\section*{References}

Bartus, T. 2005. Estimation of marginal effects using margeff. Stata Journal 5: 309-329.
Baum, C. F. 2010. Stata tip 88: Efficiently evaluating elasticities with the margins command. Stata Journal 10: 309-312.

Bruun, N. H. 2019. Visualizing effect modification on contrasts. Stata Journal 19: 566-580.
Buis, M. L. 2010. Stata tip 87: Interpretation of interactions in nonlinear models. Stata Journal 10: 305-308.
Chang, I.-M., R. Gelman, and M. Pagano. 1982. Corrected group prognostic curves and summary statistics. Journal of Chronic Diseases 35: 669-674. https://doi.org/10.1016/0021-9681(82)90019-4.

Cummings, P. 2011. Estimating adjusted risk ratios for matched and unmatched data: An update. Stata Journal 11: 290-298.

Drukker, D. M. 2016a. Doctors versus policy analysts: Estimating the effect of interest. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2016/07/19/doctors-versus-policy-analysts-estimating-the-effect-of-interest/.

2016b. Probability differences and odds ratios measure conditional-on-covariate effects and population-parameter effects. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2016/07/26/probability-differences-and-odds-ratios-measure-conditional-on-covariate-effects-and-population-parameter-effects/.

Falcaro, M., R. B. Newson, and P. D. Sasieni. 2022. Stata tip 146: Using margins after a Poisson regression model to estimate the number of events prevented by an intervention. Stata Journal 22: 224-230.

Gould, W. W. 1996. cre43: Wald test of nonlinear hypotheses after model estimation. Stata Technical Bulletin 29: 2-4. Reprinted in Stata Technical Bulletin Reprints, vol. 5, pp. 15-18. College Station, TX: Stata Press.

Graubard, B. I., and E. L. Korn. 2004. Predictive margins with survey data. Biometrics 55: 652-659. https://doi.org/10.1111/j.0006-341X.1999.00652.x.

Greene, W. H. 2018. Econometric Analysis. 8th ed. New York: Pearson.
Korn, E. L., and B. I. Graubard. 1999. Analysis of Health Surveys. New York: Wiley.
Lane, P. W., and J. A. Nelder. 1982. Analysis of covariance and standardization as instances of prediction. Biometrics 38: 613-621. https://doi.org/10.2307/2530043.

Lindsey, C. 2015a. Using mlexp to estimate endogenous treatment effects in a heteroskedastic probit model. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2015/12/10/using-mlexp-to-estimate-endogenous-treatment-effects-in-a-heteroskedastic-probit-model/.

2015b. Using mlexp to estimate endogenous treatment effects in a probit model. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2015/11/05/using-mlexp-to-estimate-endogenous-treatment-effects-in-a-probitmodel/.
_-. 2016. Estimating covariate effects after gmm. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2016/10/04/estimating-covariate-effects-after-gmm/.

Lindsey, C., and E. Pinzon. 2016. Multiple equation models: Estimation and marginal effects using gsem. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2016/06/07/multiple-equation-models-estimation-and-marginal-effects-using-gsem/.
MacDonald, K. 2018. Exploring results of nonparametric regression models. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2018/06/18/exploring-results-of-nonparametric-regression-models/.

Mitchell, M. N. 2015. Stata for the Behavioral Sciences. College Station, TX: Stata Press.
——. 2021. Interpreting and Visualizing Regression Models Using Stata. 2nd ed. College Station, TX: Stata Press.
Newson, R. B. 2013. Attributable and unattributable risks and fractions and other scenario comparisons. Stata Journal 13: 672-698.

Phillips, P. C. B., and J. Y. Park. 1988. On the formulation of Wald tests of nonlinear restrictions. Econometrica 56: 1065-1083. https://doi.org/10.2307/1911359.

Pinzon, E. 2016a. Effects of nonlinear models with interactions of discrete and continuous variables: Estimating, graphing, and interpreting. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2016/07/12/effects-for-nonlinear-models-with-interactions-of-discrete-and-continuous-variables-estimating-graphing-and-interpreting/.
——. 2016b. probit or logit: Ladies and gentlemen, pick your weapon. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2016/01/07/probit-or-logit-ladies-and-gentlemen-pick-your-weapon/.

2016c. regress, probit, or logit? The Stata Blog: Not Elsewhere Classified.
https://blog.stata.com/2016/01/14/regress-probit-or-logit/.
Radean, M. 2023. ginteff: A generalized command for computing interaction effects. Stata Journal 23: 301-335.
Rios-Avila, F. 2021. Estimation of marginal effects for models with alternative variable transformations. Stata Journal 21: 81-96.

Searle, S. R. 1997. Linear Models for Unbalanced Data. New York: Wiley.
Searle, S. R., and M. H. J. Gruber. 2017. Linear Models. 2nd ed. Hoboken, NJ: Wiley.
Searle, S. R., F. M. Speed, and G. A. Milliken. 1980. Population marginal means in the linear model: An alternative to least squares means. American Statistician 34: 216-221. https://doi.org/10.2307/2684063.

Terza, J. V. 2017a. Two-stage residual inclusion estimation: A practitioners guide to Stata implementation. Stata Journal 17: 916-938.
——. 2017b. Causal effect estimation and inference using Stata. Stata Journal 17: 939-961.
Williams, R. 2012. Using the margins command to estimate and interpret adjusted predictions and marginal effects. Stata Journal 12: 308-331.
Wooldridge, J. M. 2010. Econometric Analysis of Cross Section and Panel Data. 2nd ed. Cambridge, MA: MIT Press.

\section*{Also see}
[R] contrast - Contrasts and linear hypothesis tests after estimation
[R] margins, contrast - Contrasts of margins
[R] margins, pwcompare - Pairwise comparisons of margins
[R] margins postestimation - Postestimation tools for margins
\([R]\) marginsplot - Graph results from margins (profile plots, etc.)
[R] lincom - Linear combinations of parameters
[R] nlcom - Nonlinear combinations of parameters
\([R]\) predict - Obtain predictions, residuals, etc., after estimation
\([R]\) predictnl - Obtain nonlinear predictions, standard errors, etc., after estimation
[U] 20 Estimation and postestimation commands

\section*{Title}
margins postestimation - Postestimation tools for margins

Postestimation commands Remarks and examples Also see

\section*{Postestimation commands}

The following standard postestimation command is available after margins:
Command Description
marginsplot graph the results from margins-profile plots, interaction plots, etc.
For information on marginsplot, see \([R]\) marginsplot.

The following standard postestimation commands are available after margins, post:
\begin{tabular}{ll} 
Command & Description \\
\hline contrast & contrasts and ANOVA-style joint tests of estimates \\
estat summarize & summary statistics for the estimation sample \\
estat vce & variance-covariance matrix of the estimators (VCE) \\
estimates & cataloging estimation results \\
etable & \begin{tabular}{l} 
table of estimation results \\
point estimates, standard errors, testing, and inference for linear combinations of \\
lincom
\end{tabular} \\
nlcom & \begin{tabular}{l} 
point estimates, standard errors, testing, and inference for nonlinear combinations \\
of coefficients
\end{tabular} \\
pwcompare & \begin{tabular}{l} 
pairwise comparisons of estimates
\end{tabular} \\
test & Wald tests of simple and composite linear hypotheses \\
testnl & Wald tests of nonlinear hypotheses
\end{tabular}

\section*{Remarks and examples}

Continuing with the example from Example 8: Margins of interactions in [R] margins, we use the dataset and refit the logistic model of outcome:
```

. use https://www.stata-press.com/data/r18/margex
(Artificial data for margins)
. logistic outcome sex\#\#group age
(output omitted)

```

We then estimate the margins for males and females and post the margins as estimation results with a full VCE.
```

. margins sex, post
Predictive margins Number of obs = 3,000
Model VCE: OIM
Expression: Pr(outcome), predict()

```
\begin{tabular}{c|rccccc}
\hline & \multicolumn{4}{|c}{\begin{tabular}{c} 
Delta-method \\
std. err.
\end{tabular}} & \(\mathbf{z}\) & \(\mathrm{P}>|\mathrm{z}|\) \\
\hline Margin & [95\% conf. interval] \\
\hline sex & & & & & & \\
Male & .1600644 & .0125653 & 12.74 & 0.000 & .1354368 & .184692 \\
Female & .1966902 & .0100043 & 19.66 & 0.000 & .1770821 & .2162983 \\
\hline
\end{tabular}

We can now use nlcom (see [R] nlcom) to estimate a risk ratio of females to males using the average probabilities for females and males posted by margins:
```

. nlcom (risk_ratio: _b[1.sex] / _b[0.sex])
risk_ratio: _b[1.sex] / _b[0.sex]

```
\begin{tabular}{l|rccccc}
\hline & Coefficient & Std. err. & \(z\) & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline risk_ratio & 1.228819 & .1149538 & 10.69 & 0.000 & 1.003514 & 1.454124 \\
\hline
\end{tabular}

We could similarly estimate the average risk difference between females and males:
```

. nlcom (risk_diff: _b[1.sex] - _b[0.sex])
risk_diff: _b[1.sex] - _b[0.sex]

```
\begin{tabular}{c|cccccr}
\hline & Coefficient & Std. err. & z & P>|z| & [95\% conf. interval] \\
\hline risk_diff & .0366258 & .0160632 & 2.28 & 0.023 & .0051425 & .068109 \\
\hline
\end{tabular}

\section*{Also see}
[R] margins - Marginal means, predictive margins, and marginal effects
[R] marginsplot - Graph results from margins (profile plots, etc.)
[U] 20 Estimation and postestimation commands

\section*{Title}
margins, contrast - Contrasts of margins
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Suboptions & Remarks and examples & Stored results & Methods and formulas \\
Reference & Also see & &
\end{tabular}

\section*{Description}
margins with the contrast option or with contrast operators performs contrasts of margins. This extends the capabilities of contrast to any of the nonlinear responses, predictive margins, or other margins that can be estimated by margins.

\section*{Quick start}

Reference category contrasts of the predictive margins for a after logit y a\#\#b x1 margins r.a

Contrasts of predictive margins for a with the previous level margins ar.a

Test the equality of predictive margins for a
margins a, contrast
Reference category contrasts of predictive margins for \(\mathrm{x} 1=10\) and \(\mathrm{x} 1=20\) with \(\mathrm{x} 1=0\) margins, at(x1=(0 10 20)) contrast(atcontrast(r))

Average partial effect of increasing x1 by 100 for each observation after probit y x1 x2 margins, at((asobserved) _all) at(x1=generate(x1+100)) ///
contrast (atcontrast(r))

\section*{Menu}

Statistics \(>\) Postestimation

\section*{Syntax}
```

margins [marginlist] [if] [in] [weight] [, contrast margins_options]
margins [marginlist] [if] [in] [weight] [, contrast(suboptions) margins_options]

```
where marginlist is a list of factor variables or interactions that appear in the current estimation results. The variables may be typed with or without contrast operators, and you may use any factor-variable syntax:
```

. margins sex\#\#group, contrast
. margins sex\#\#g.group, contrast
. margins sex@group, contrast

```

See the operators (op.) table in [R] contrast for the list of contrast operators. Contrast operators may also be specified on the variables in margins's over() and within() options to perform contrasts across the levels of those variables.
suboptions
Description

Contrast
overall
lincom
atcontrast (op[.-at])
predictjoint
atjoint
overjoint
withinjoint
marginswithin
cieffects
pveffects
effects
nowald
noatlevels
nosvyadjust
predict (op [._predict]) apply the op. contrast operator to the groups defined by multiple predict() options
add a joint hypothesis test for all specified contrasts
treat user-defined contrasts as linear combinations
apply the \(o p\). contrast operator to the groups defined by at ()
test jointly across all groups defined by multiple predict() options
test jointly across all groups defined by at ()
test jointly across all levels of the unoperated over () variables test jointly across all levels of the unoperated within() variables perform contrasts within the levels of the unoperated terms in marginlist
show effects table with confidence intervals
show effects table with \(p\)-values
show effects table with confidence intervals and \(p\)-values
suppress table of Wald tests
report only the overall Wald test for terms that use the within @ or nested \| operator
compute unadjusted Wald tests for survey results
collect is allowed; see [U] 11.1.10 Prefix commands.
fweights, aweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.

\section*{Suboptions}

\footnotetext{
Contrast
overall specifies that a joint hypothesis test over all terms be performed.
}
lincom specifies that user-defined contrasts be treated as linear combinations. The default is to require that all user-defined contrasts sum to zero. (Summing to zero is part of the definition of a contrast.)
predict (op[._predict]) specifies that the op. contrast operator be applied to the groups defined by multiple specifications of margins's predict() option. The default behavior, by comparison, is to perform tests and contrasts within these groups.
atcontrast ( \(o p\left[. \_a t\right]\) ) specifies that the \(o p\). contrast operator be applied to the groups defined by the at () option(s). The default behavior, by comparison, is to perform tests and contrasts within the groups defined by the at () option(s).

See example 6 in Remarks and examples.
predictjoint specifies that joint tests be performed across all groups defined by multiple specifications of margins's predict() option. The default behavior, by comparison, is to perform contrasts and tests within each group.
atjoint specifies that joint tests be performed across all groups defined by the at() option. The default behavior, by comparison, is to perform contrasts and tests within each group.
See example 5 in Remarks and examples.
overjoint specifies how unoperated variables in the over () option are treated.
Each variable in the over () option may be specified either with or without a contrast operator. For contrast-operated variables, the specified contrast comparisons are always performed.
overjoint specifies that joint tests be performed across all levels of the unoperated variables. The default behavior, by comparison, is to perform contrasts and tests within each combination of levels of the unoperated variables.

See example 3 in Remarks and examples.
withinjoint specifies how unoperated variables in the within() option are treated.
Each variable in the within() option may be specified either with or without a contrast operator. For contrast-operated variables, the specified contrast comparisons are always performed.
withinjoint specifies that joint tests be performed across all levels of the unoperated variables. The default behavior, by comparison, is to perform contrasts and tests within each combination of levels of the unoperated variables.
marginswithin specifies how unoperated variables in marginlist are treated.
Each variable in marginlist may be specified either with or without a contrast operator. For contrast-operated variables, the specified contrast comparisons are always performed.
marginswithin specifies that contrasts and tests be performed within each combination of levels of the unoperated variables. The default behavior, by comparison, is to perform joint tests across all levels of the unoperated variables.

See example 4 in Remarks and examples.
cieffects specifies that a table containing a confidence interval for each individual contrast be reported.
pveffects specifies that a table containing a \(p\)-value for each individual contrast be reported.
effects specifies that a single table containing a confidence interval and \(p\)-value for each individual contrast be reported.
nowald suppresses the table of Wald tests.
noatlevels indicates that only the overall Wald test be reported for each term containing within or nested (@ or I) operators.
nosvyadjust is for use with svy estimation commands. It specifies that the Wald test be carried out without the default adjustment for the design degrees of freedom. That is to say the test is carried out as \(W / k \sim F(k, d)\) rather than as \((d-k+1) W /(k d) \sim F(k, d-k+1)\), where \(k\) is the dimension of the test and \(d\) is the total number of sampled PSUs minus the total number of strata.

\section*{Remarks and examples}

Remarks are presented under the following headings:

> Contrasts of margins
> Contrasts and the over() option
> \(\quad\) The overjoint suboption
> The marginswithin suboption
> Contrasts and the at() option
> Estimating treatment effects with margins
> Conclusion

\section*{Contrasts of margins}
> Example 1
Estimating contrasts of margins is as easy as adding a contrast operator to the variable name. Let's review Example 2: A simple case after logistic of [R] margins. Variable sex is coded 0 for males and 1 for females.
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline \multicolumn{7}{|l|}{```
(Artificial data for margins)
. logistic outcome i.sex i.group
    (output omitted)
. margins sex
```} \\
\hline \multicolumn{7}{|l|}{\multirow[t]{2}{*}{}} \\
\hline & & & & & & \\
\hline & \multicolumn{2}{|l|}{Delta-metho Margin std. err.} & z & \(P>|z|\) & \multicolumn{2}{|l|}{[95\% conf. interval]} \\
\hline sex & \multicolumn{6}{|l|}{\multirow[t]{3}{*}{\[
\begin{array}{rrrrrr}
.1286796 & .0111424 & 11.55 & 0.000 & .106841 & .1505182 \\
.1905087 & .0089719 & 21.23 & 0.000 & .1729241 & .2080933
\end{array}
\]}} \\
\hline Male & & & & & & \\
\hline Female & & & & & & \\
\hline
\end{tabular}

The first margin, 0.13 , is the average probability of a positive outcome, treating everyone as if they were male. The second margin, 0.19 , is the average probability of a positive outcome, treating everyone as if they were female. We can compare females with males by rerunning margins and adding a contrast operator:


The r. prefix for sex is the reference-category contrast operator-see [R] contrast. (The default reference category is zero, the lowest value of sex.) Contrast operators in a marginlist work just as they do in the termlist of a contrast command.

The contrast estimate of 0.06 says that unconditional on group, females on average are about \(6 \%\) more likely than males to have a positive outcome. The \(\chi^{2}\) statistic of 16.61 shows that the contrast is significantly different from zero.

You may be surprised that we did not need to include the contrast option to estimate our contrast. If we had included the option, our output would not have changed:
```

. margins r.sex, contrast
Contrasts of predictive margins Number of obs = 3,000
Model VCE: OIM
Expression: Pr(outcome), predict()

```
\begin{tabular}{r|ccc}
\hline & df & chi2 & P>chi2 \\
\hline sex & 1 & 16.61 & 0.0000 \\
\hline
\end{tabular}
\begin{tabular}{c|cccc}
\hline & \multicolumn{4}{|c}{\begin{tabular}{c} 
Delta-method \\
std. err.
\end{tabular}} \\
\hline [95\% conf. interval] \\
\hline (Female vs Male) & .0618291 & .0151719 & .0320927 & .0915656 \\
\hline
\end{tabular}

The contrast option is useful mostly for its suboptions, which control the output and how contrasts are estimated in more complicated situations. But contrast may be specified on its own (without contrast operators or suboptions) if we do not need estimates or confidence intervals:
```

. margins sex group, contrast
Contrasts of predictive margins Number of obs = 3,000
Model VCE: OIM
Expression: Pr(outcome), predict()

```
\begin{tabular}{r|ccc}
\hline & df & chi2 & P>chi2 \\
\hline sex & 1 & 16.61 & 0.0000 \\
group & 2 & 225.76 & 0.0000 \\
\hline
\end{tabular}

Each \(\chi^{2}\) statistic is a joint test of constituent contrasts. The test for group has two degrees of freedom because group has three levels.

\section*{Contrasts and the over() option}

\section*{> Example 2}

It is common to estimate margins at combinations of factor levels, and margins, contrast includes several suboptions for contrasting such margins. Let's fit a model with two categorical predictors and their interaction:
. logistic outcome agegroup\#\#group
\(\begin{array}{ll}\text { Logistic regression } & \text { Number of obs }=3,000 \\ & \text { LR chi2 }(8) \quad=520.64 \\ \text { Log likelihood }=-1105.7504 & \text { Prob }>\text { chi2 }=0.0000 \\ & \text { Pseudo R2 }=0.1906\end{array}\)
\begin{tabular}{c|rccccc}
\hline outcome & Odds ratio & Std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline agegroup & & & & & & \\
\(30-39\) & 3.54191 & 2.226951 & 2.01 & 0.044 & 1.032882 & 12.14576 \\
\(40+\) & 16.23351 & 9.611879 & 4.71 & 0.000 & 5.086452 & 51.80955 \\
& & & & & & \\
group & & & & & & \\
2 & .834507 & .5663738 & -0.27 & 0.790 & .2206611 & 3.15598 \\
3 & .2146739 & .1772903 & -1.86 & 0.062 & .042541 & 1.083306 \\
& & & & & & \\
agegroup\# & & & & & & \\
group & & & & & & \\
30-39\#2 & .4426927 & .3358505 & -1.07 & 0.283 & .1000772 & 1.958257 \\
30-39\#3 & 1.16088 & 1.103521 & 0.16 & 0.875 & .1801538 & 7.480508 \\
\(40+\# 2\) & .440672 & .3049393 & -1.18 & 0.236 & .1135259 & 1.71055 \\
\(40+\# 3\) & .4407892 & .4034666 & -0.89 & 0.371 & .0732998 & 2.650693 \\
& & & & & & \\
_cons & .0379747 & .0223371 & -5.56 & 0.000 & .0119897 & .1202762 \\
\hline
\end{tabular}

Note: _cons estimates baseline odds.
Each of agegroup and group has three levels. To compare each age group with the reference category on the probability scale, we can again use margins with the \(r\). contrast operator.
. margins r.agegroup
Contrasts of predictive margins Number of obs \(=3,000\)
Model VCE: OIM
Expression: Pr(outcome), predict()
\begin{tabular}{r|rrr}
\hline & df & chi2 & P>chi2 \\
\hline agegroup & & & \\
\((30-39\) vs 20-29) & 1 & 10.04 & 0.0015 \\
\((40+\) vs 20-29) & 1 & 224.44 & 0.0000 \\
Joint & 2 & 238.21 & 0.0000 \\
\hline
\end{tabular}
\begin{tabular}{c|cccc}
\hline & \multicolumn{4}{|c}{\begin{tabular}{c} 
Delta-method \\
std. err.
\end{tabular}} \\
[95\% conf. interval] \\
\hline agegroup & & & & \\
\((30-39\) vs 20-29) & .044498 & .0140448 & .0169706 & .0720253 \\
\((40+\) vs 20-29) & .2059281 & .0137455 & .1789874 & .2328688 \\
\hline
\end{tabular}

Our model includes an interaction, though, so it would be nice to estimate the contrasts separately for each value of group. We need the over () option:
```

. margins r.agegroup, over(group)
Contrasts of predictive margins Number of obs = 3,000
Model VCE: OIM
Expression: Pr(outcome), predict()
Over: group

```
\begin{tabular}{r|rrr}
\hline & df & chi2 & P>chi2 \\
\hline agegroup@group & & & \\
\((30-39\) vs 20-29) 1 & 1 & 6.94 & 0.0084 \\
\((30-39\) vs 20-29) 2 & 1 & 1.18 & 0.2783 \\
\((40+\) vs 20-29) 3 20-29) 1 & 1 & 3.10 & 0.0783 \\
\((40+\) vs 20-29) 2 & 1 & 173.42 & 0.0000 \\
\((40+\) vs 20-29) 3 & 1 & 57.77 & 0.0000 \\
Joint & 1 & 5.12 & 0.0236 \\
& 6 & 266.84 & 0.0000 \\
\hline
\end{tabular}
\begin{tabular}{r|rrrr}
\hline & Contrast \begin{tabular}{c} 
Delta-method \\
std. err.
\end{tabular} & [95\% conf. interval] \\
\hline agegroup@group & & & & \\
(30-39 vs 20-29) 1 & .0819713 & .0311208 & .0209757 & .142967 \\
\((30-39\) vs 20-29) 2 & .0166206 & .0153309 & -.0134275 & .0466686 \\
\((30-39\) vs 20-29) 3 & .0243462 & .0138291 & -.0027584 & .0514507 \\
\((40+\) vs 20-29) 1 & .3447797 & .0261811 & .2934658 & .3960937 \\
\((40+\) vs 20-29) 2 & .1540882 & .0202722 & .1143554 & .193821 \\
\((40+\) vs 20-29) 3 & .0470319 & .0207774 & .006309 & .0877548 \\
\hline
\end{tabular}

The effect of agegroup appears to be greatest for the first level of group.
Including a variable in the over () option is not equivalent to including the variable in the main marginlist. The variables in the marginlist are manipulated in the analysis, so that we can measure, for example, the effect of being in age group 3 and not age group 1. (The manipulation could be mimicked by running replace and then predict, but the manipulations actually performed by margins do not
change the data in memory.) The variables in the over() option are not so manipulated-the values of the over () variables are left as they were observed, and the marginlist variables are manipulated separately for each observed over () group.

\section*{The overjoint suboption}

\section*{Example 3}

Each variable in an over () option may be specified with or without contrast operators. Our option over (group) did not include a contrast operator, so margins estimated the contrasts separately for each level of group. If we had instead specified over (r.group), we would have received differences of the contrasts:
\begin{tabular}{|c|c|c|c|c|}
\hline \multicolumn{2}{|l|}{\multirow[t]{2}{*}{\begin{tabular}{l}
Contrasts of predictive margins \\
Model VCE: OIM \\
Expression: Pr(outcome), predict() \\
Over: group
\end{tabular}}} & \multicolumn{3}{|r|}{Number of obs \(=3,000\)} \\
\hline & & & & \\
\hline & & df chi2 & \(2 \quad \mathrm{P}>\mathrm{chi2}\) & \\
\hline \[
\] & & \begin{tabular}{lr}
1 & 3.55 \\
1 & 33.17 \\
1 & 2.86 \\
1 & 79.36 \\
4 & 83.88
\end{tabular} & \[
\begin{aligned}
& 0.0596 \\
& 0.0000 \\
& 0.0906 \\
& 0.0000 \\
& 0.0000
\end{aligned}
\] & \\
\hline & \multicolumn{4}{|c|}{Delta-method} \\
\hline \multicolumn{5}{|l|}{group\#agegroup} \\
\hline (2 vs 1) (30-39 vs 20-29) & -. 0653508 & . 0346921 & -. 133346 & . 0026445 \\
\hline (2 vs 1) (40+ vs 20-29) & -. 1906915 & . 0331121 & -. 25559 & -. 1257931 \\
\hline (3 vs 1) (30-39 vs 20-29) & -. 0576252 & . 0340551 & -. 1243719 & . 0091216 \\
\hline (3 vs 1) (40+ vs 20-29) & -. 2977479 & . 0334237 & -. 3632572 & -. 2322386 \\
\hline
\end{tabular}

The contrasts are double differences: the estimate of -0.19 , for example, says that the difference in the probability of success between age group 3 and age group 1 is smaller in group 2 than in group 1 . We can jointly test pairs of the double differences with the overjoint suboption:
\begin{tabular}{|c|c|c|c|}
\hline Contrasts of predictive margin Model VCE: OIM & & & \multirow[t]{2}{*}{Number of} \\
\hline \multicolumn{3}{|l|}{\begin{tabular}{l}
Expression: Pr(outcome), predict() \\
Over: group
\end{tabular}} & \\
\hline & df & chi2 & P>chi2 \\
\hline \begin{tabular}{l}
group\#agegroup \\
(joint) (30-39 vs 20-29)
\end{tabular} & 2 & 3.62 & 0.1641 \\
\hline (joint) (40+ vs 20-29) & 2 & 79.45 & 0.0000 \\
\hline Joint & 4 & 83.88 & 0.0000 \\
\hline
\end{tabular}

The contrast (overjoint) option overrides the default behavior of over() and requests joint tests over the levels of the unoperated variable group. The \(\chi^{2}\) statistic of 3.62 tests that the first and third contrasts from the previous table are jointly zero. The \(\chi^{2}\) statistic of 79.45 jointly tests the other pair of contrasts.

\section*{The marginswithin suboption}

\section*{Example 4}

Another suboption that may usefully be combined with over() is marginswithin. marginswithin requests that contrasts be performed within the levels of unoperated variables in the main marginlist, instead of performing them jointly across the levels. marginswithin affects only unoperated variables because contrast operators take precedence over suboptions.

Let's first look at the default behavior, which occurs when marginswithin is not specified:
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline \multicolumn{5}{|l|}{\multirow[t]{2}{*}{\begin{tabular}{l}
Contrasts of predictive margins \\
Model VCE: OIM \\
Expression: Pr(outcome), predict() \\
Over: group
\end{tabular}}} & \multicolumn{2}{|l|}{\multirow[t]{4}{*}{Number of obs \(=3,000\)}} \\
\hline & & & & & & \\
\hline & & df & chi2 & \(\mathrm{P}>\mathrm{chi2}\) & & \\
\hline group\#agegroup
\((2\) vs 1)
(joint)
\((3\) vs 1) & & \[
\begin{aligned}
& 2 \\
& 2 \\
& 4
\end{aligned}
\] & \[
\begin{aligned}
& 33.94 \\
& 83.38 \\
& 83.88
\end{aligned}
\] & \[
\begin{aligned}
& 0.0000 \\
& 0.0000 \\
& 0.0000
\end{aligned}
\] & & \\
\hline & \multicolumn{2}{|l|}{Contrast \begin{tabular}{c} 
Delta-method \\
std. err.
\end{tabular}} & z & \(P>|z|\) & \multicolumn{2}{|l|}{[95\% conf. interval]} \\
\hline \multirow[t]{6}{*}{} & \multicolumn{6}{|l|}{\multirow[b]{6}{*}{\(\begin{array}{llllll}-.0653508 & .0346921 & -1.88 & 0.060 & -.133346 & .0026445 \\ -.1906915 & .0331121 & -5.76 & 0.000 & -.25559 & -.1257931 \\ -.0576252 & .0340551 & -1.69 & 0.091 & -.1243719 & .0091216 \\ & & & & & \\ -.2977479 & .0334237 & -8.91 & 0.000 & -.3632572 & -.2322386\end{array}\)}} \\
\hline & & & & & & \\
\hline & & & & & & \\
\hline & & & & & & \\
\hline & & & & & & \\
\hline & & & & & & \\
\hline
\end{tabular}

Here agegroup in the main marginlist is an unoperated variable, so margins by default performs joint tests across the levels of agegroup: the \(\chi^{2}\) statistic of 33.94 , for example, jointly tests whether the first two contrast estimates in the lower table differ significantly from zero.

When we specify marginswithin, the contrasts will instead be performed within the levels of agegroup:
```

. margins agegroup, over(r.group) contrast(marginswithin effects)
Contrasts of predictive margins Number of obs = 3,000
Model VCE: OIM
Expression: Pr(outcome), predict()
Over: group

```
\begin{tabular}{c|crc}
\hline & df & chi2 & P>chi2 \\
\hline group@agegroup & & & \\
\((2\) vs 1) 20-29 & 1 & 0.06 & 0.7991 \\
\((2\) vs 1) 30-39 & 1 & 7.55 & 0.0060 \\
\((2\) vs 1\() 40+\) \\
\((3\) vs 1) 20-29 & 1 & 68.39 & 0.0000 \\
\((3\) vs 1) 30-39 & 1 & 1.80 & 0.1798 \\
(3 vs 1) 40+ & 1 & 10.47 & 0.0012 \\
Joint & 1 & 159.89 & 0.0000 \\
\hline
\end{tabular}


The joint tests in the top table have been replaced by one-degree-of-freedom tests, one for each combination of the two reference comparisons and three levels of agegroup. The reference-category contrasts for group have been performed within levels of agegroup.

\section*{Contrasts and the at() option}

\section*{Example 5}

The at () option of margins is used to set predictors to particular values. When at() is used, contrasts are by default performed within each at () level:
```

. margins r.agegroup, at (group=(1/3))
Contrasts of adjusted predictions Number of obs = 3,000
Expression: Pr(outcome), predict()
1._at: group = 1
2._at: group = 2
3._at: group = 3

```
\begin{tabular}{r|rrr}
\hline & df & chi2 & P>chi2 \\
\hline agegroup@_at & & & \\
\((30-39\) vs 20-29) 1 & 1 & 6.94 & 0.0084 \\
\((30-39\) vs 20-29) 2 & 1 & 1.18 & 0.2783 \\
\((40+\) vs 20-29) 3 & 1 & 3.10 & 0.0783 \\
\((40+\) vs 20-29) 1 & 1 & 173.42 & 0.0000 \\
\((40+\) vs 20-29) 2 & 1 & 57.77 & 0.0000 \\
& 1 & 5.12 & 0.0236 \\
Joint & 6 & 266.84 & 0.0000 \\
\hline
\end{tabular}
\begin{tabular}{r|rrrr}
\hline & \multicolumn{4}{|c}{\begin{tabular}{c} 
Delta-method \\
std. err.
\end{tabular}} \\
\hline Contrast & [95\% conf. interval] \\
\hline agegroup@_at & & & & \\
\((30-39\) vs 20-29) 1 & .0819713 & .0311208 & .0209757 & .142967 \\
\((30-39\) vs 20-29) 2 & .0166206 & .0153309 & -.0134275 & .0466686 \\
\((40+\) vs 20-29) 3 & .0243462 & .0138291 & -.0027584 & .0514507 \\
\((40+\) vs 20-29) 1 & .3447797 & .0261811 & .2934658 & .3960937 \\
\((40+\) vs \(20-29)\) & 2 & .1540882 & .0202722 & .1143554 \\
\hline
\end{tabular}

Our option at (group=(1/3)) manipulates the values of group and is therefore not equivalent to over (group). We see that the reference-category contrasts for agegroup have been performed within each at () level. For a similar example that uses the ._at operator instead of the at () option, see Contrasts of at() groups-discrete effects in [R] marginsplot.

The default within behavior of at () may be changed to joint behavior with the atjoint suboption:
```

. margins r.agegroup, at(group=(1/3)) contrast(atjoint)
Contrasts of adjusted predictions Number of obs = 3,000
Model VCE: OIM
Expression: Pr(outcome), predict()
1._at: group = 1
2._at: group = 2
3._at: group = 3

```
\begin{tabular}{r|rrr}
\hline & df & chi2 & P>chi2 \\
\hline (joint) (30-39 \begin{tabular}{r} 
at\#agegroup \\
(joint) \((40+\) vs 20-29)
\end{tabular} & & & \\
& Joint & 2 & 3.62 \\
\hline
\end{tabular}

Now, the tests are performed jointly over the levels of group, the at () variable. The atjoint suboption is the analogue for at() of the overjoint suboption from example 3.
> Example 6
What if we would like to apply a contrast operator, like r., to the at () levels? It is not possible to specify the operator inside the at() option. Instead, we need a new suboption, atcontrast():
```

. margins r.agegroup, at(group=(1/3)) contrast(atcontrast(r))
Contrasts of adjusted predictions Number of obs = 3,000
Model VCE: OIM
Expression: Pr(outcome), predict()
1._at: group = 1
2._at: group = 2
3._at: group = 3

```
\begin{tabular}{|c|c|c|c|}
\hline & df & chi2 & \(\mathrm{P}>\mathrm{chi2}\) \\
\hline _at\#agegroup & & & \\
\hline (2 vs 1) (30-39 vs 20-29) & 1 & 3.55 & 0.0596 \\
\hline (2 vs 1) (40+ vs 20-29) & 1 & 33.17 & 0.0000 \\
\hline (3 vs 1) (30-39 vs 20-29) & 1 & 2.86 & 0.0906 \\
\hline (3 vs 1) (40+ vs 20-29) & 1 & 79.36 & 0.0000 \\
\hline Joint & 4 & 83.88 & 0.0000 \\
\hline
\end{tabular}
\begin{tabular}{|c|c|c|c|c|}
\hline & \multicolumn{2}{|l|}{Delta-method Contrast std. err.} & \multicolumn{2}{|l|}{[95\% conf. interval]} \\
\hline _at\#agegroup & & & & \\
\hline (2 vs 1) (30-39 vs 20-29) & -. 0653508 & . 0346921 & -. 133346 & . 0026445 \\
\hline (2 vs 1) (40+ vs 20-29) & -. 1906915 & . 0331121 & -. 25559 & -. 1257931 \\
\hline (3 vs 1) (30-39 vs 20-29) & -. 0576252 & . 0340551 & -. 1243719 & . 0091216 \\
\hline (3 vs 1) (40+ vs 20-29) & -. 2977479 & . 0334237 & -. 3632572 & -. 2322386 \\
\hline
\end{tabular}

When we specify contrast(atcontrast(r)), margins will apply the r. reference-category operator to the levels of group, the variable specified inside at (). The default reference category is 1, the lowest level of group.

\section*{Estimating treatment effects with margins}
margins with the contrast option can also be used to estimate treatment effects in certain cases. A treatment effect represents the change in an outcome variable that is attributable to a particular event, controlling for all other factors that could affect the outcome. For example, we might want to know how a person's wage changes as a result of being in a union. Here the outcome variable is the person's wage, and the "event" is membership in a union. The treatment effect measures the difference in a person's wage as a result of being or not being in a union once we control for the person's educational background, level of experience, industry, and other factors.

In fact, Stata has an entire manual dedicated to estimators designed specifically for estimating treatment effects; see the Stata Causal Inference and Treatment-Effects Estimation Reference Manual. Here we show how margins can be used to estimate treatment effects using the regression-adjustment estimator when the conditional independence assumption is met; see [CAUSAL] teffects intro. Regression adjustment simply means that we are going to use a regression model to predict the outcome variable, controlling for treatment status and other characteristics. The conditional independence assumption implies that we have enough variables in our dataset so that once we control for them in our regression model, the outcomes one would obtain with and without treatment are independent of how treatment status is determined.

\section*{> Example 7: Regression adjustment with a binary treatment variable}
nlsw88. dta contains women's wages (wage) in dollars per hour, a binary variable indicating their union status (union), years of experience (ttl_exp), and a variable, grade, indicating the number of years of schooling completed. We want to know how being in a union (the treatment) affects women's wages. Traditionally, a wage equation of the form
\[
\ln \text { wage }_{i}=\beta_{0}+\beta_{1} \text { union }_{i}+\beta_{2} \text { grade }_{i}+\beta_{3} \text { ttl_exp }+\beta_{4} \text { ttl_exp }{ }^{2}+\epsilon_{i}
\]
would be fit. However, there are two shortcomings that we will improve upon. First, to avoid the problem of predicting the level of a log-transformed dependent variable, we will use poisson with the vce (robust) option to fit an exponential regression model; see Wooldridge (2010, sec. 18.2) for background on this approach. Second, the previous equation implies that factors other than union status have the same impact on wages for both union and nonunion workers. Regression-adjustment estimators allow all the variables to have different impacts depending on the level of the treatment variable, and we can accomplish that here using factor-variable notation. In Stata, we fit our model by typing
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline \multicolumn{7}{|l|}{\begin{tabular}{l}
(NLSW, 1988 extract) \\
. poisson wage i.union\#\#(c.grade c.ttl_exp\#\#c.ttl_exp), vce(robust) \\
note: noncount dependent variable encountered; results correspond to an exponential-mean model rather than a poisson model.
\end{tabular}} \\
\hline \multicolumn{7}{|l|}{\begin{tabular}{l}
Iteration 1: Log pseudolikelihood \(=-4770.7693\) \\
Iteration 2: Log pseudolikelihood = -4770.7693
\end{tabular}} \\
\hline \multicolumn{5}{|l|}{Poisson regression} & Number of ob Wald chi2(7) Prob > chi2 & \[
\begin{aligned}
& =1,876 \\
& =1047.11 \\
& =0.0000
\end{aligned}
\] \\
\hline \multicolumn{5}{|l|}{Log pseudolikelihood \(=-4770.7693\)} & Pseudo R2 & \(=0.1195\) \\
\hline wage & Coefficient & Robust std. err. & z & \(P>|z|\) & [95\% conf & interval] \\
\hline \multicolumn{7}{|l|}{union} \\
\hline Union & . 8638376 & . 168233 & 5.13 & 0.000 & . 534107 & 1.193568 \\
\hline grade & . 0895252 & . 0056874 & 15.74 & 0.000 & . 0783782 & . 1006722 \\
\hline ttl_exp & . 0805737 & . 0114534 & 7.03 & 0.000 & . 0581255 & . 103022 \\
\hline \multicolumn{7}{|l|}{c.ttl_exp\#} \\
\hline \multicolumn{7}{|l|}{\begin{tabular}{l}
union\# \\
c.grade
\end{tabular}} \\
\hline Union & -. 0310298 & . 0088259 & -3.52 & 0.000 & -. 0483282 & -. 0137314 \\
\hline \multicolumn{7}{|l|}{\[
\begin{gathered}
\text { union\# } \\
\text { c.ttl_exp }
\end{gathered}
\]} \\
\hline Union & -. 0404226 & . 0230113 & -1.76 & 0.079 & -. 085524 & . 0046788 \\
\hline \multicolumn{7}{|l|}{\multirow[t]{2}{*}{\[
\begin{array}{r}
\text { union\# } \\
\text { c.ttl_exp\# } \\
\text { c.ttl_exp }
\end{array}
\]}} \\
\hline & & & & & & \\
\hline Union & . 0011808 & . 0008428 & 1.40 & 0.161 & -. 0004711 & . 0028327 \\
\hline _cons & . 017488 & . 0893602 & 0.20 & 0.845 & -. 1576547 & . 1926308 \\
\hline
\end{tabular}

To see how union status affects wages, we can use margins:
```

. margins r.union, vce(unconditional)
Contrasts of predictive margins Number of obs = 1,876
Expression: Predicted number of events, predict()

```
\begin{tabular}{r|ccc}
\hline & df & chi2 & \(\mathrm{P}>\) chi2 \\
\hline union & 1 & 26.22 & 0.0000 \\
\hline
\end{tabular}
\begin{tabular}{l|llll}
\hline & \multicolumn{4}{|c}{\begin{tabular}{c} 
Unconditional \\
std. err.
\end{tabular}} \\
\hline [95\% conf. interval] \\
\hline union & Contrast & Union vs Nonunion) & 1.004119 & .1960944
\end{tabular}

The estimated contrast 1.004 indicates that on average, belonging to a union causes a woman's wage to be slightly more than a dollar higher than if she were not in the union. This estimated contrast is called the average treatment effect (ATE). Conceptually, we predicted the wage of each woman in the estimation sample assuming she was in a union and obtained the sample mean. We then predicted each woman's wage assuming she was not in a union and obtained that sample mean. The difference between these two sample means represents the ATE.

We obtain essentially the same results by using teffects ra:


The point estimates of the ATE are identical to those we obtained using margins, though the standard errors differ slightly from those reported by margins. The standard errors from the two estimators are, however, asymptotically equivalent, meaning they would coincide with a sufficiently large dataset. The last statistic in this output table indicates the untreated potential-outcome mean (untreated POM), which is the mean predicted wage assuming each woman did not belong to a union.

If we specify the pomeans option with teffects ra, we can obtain both the treated and the untreated POMs, which represent the predicted mean wages assuming all women were or were not in the union:


Notice that the difference between these two POMs equals 1.004119 , which is the ATE we obtained earlier.

In some applications, the average treatment effect of the treated (ATET) is more germane than the ATE. For example, if the untreated subjects in the sample could not possibly receive treatment (perhaps because a medical condition precludes their taking an experimental drug), then considering the counterfactual outcome had those subjects taken the drug may not be relevant. In these cases, the ATET is a better statistic because it measures the effect of the treatment only for those subjects who actually did receive treatment. Like the ATE, the ATET involves computing predicted outcomes for each treatment level, obtaining the sample means, and computing the difference between those two means. Unlike the ATE, however, we use only observations corresponding to treated subjects.

\section*{> Example 8: Regression adjustment with a binary treatment variable (continued)}

Here we calculate the ATET of union membership, first using margins. Because teffects ra overwrote our estimation results, we first quietly refit our poisson model. We then call margins to obtain the ATET:
```

. quietly poisson wage i.union\#\#(c.grade c.ttl_exp\#\#c.ttl_exp), vce(robust)
. margins r.union, subpop(union) vce(unconditional)
Contrasts of predictive margins Number of obs = 1,876
Subpop. no. obs = 460

```
Expression: Predicted number of events, predict()
\begin{tabular}{r|ccc}
\hline & df & chi2 & P>chi2 \\
\hline union & 1 & 18.86 & 0.0000 \\
\hline
\end{tabular}
\begin{tabular}{l|cccc}
\hline & \multicolumn{4}{|c}{\begin{tabular}{c} 
Unconditional \\
std. err.
\end{tabular}} \\
\hline [95\% conf. interval] \\
\hline union & .901419 & .2075863 & .4945574 & 1.308281 \\
\hline
\end{tabular}

The key here was specifying the subpop(union) option to restrict margin's computations to those women who are union members. The results indicate that being in the union causes the union members' wages to be about 90 cents higher than they would otherwise be.

To replicate these results using teffects ra, we include the atet option to obtain ATETs:
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline \begin{tabular}{l}
Iteration 0 : \\
Iteration 1:
\end{tabular} & EE criterion EE criterion & \[
\begin{aligned}
& =2.611 \mathrm{e} \\
& =9.347 \mathrm{e}
\end{aligned}
\] & & & & \\
\hline \multicolumn{4}{|l|}{Treatment-effects estimation} & Numbe & & 1,876 \\
\hline \multicolumn{7}{|l|}{Estimator : regression adjustment} \\
\hline \multicolumn{7}{|l|}{Outcome model : Poisson} \\
\hline \multicolumn{7}{|l|}{Treatment model: none} \\
\hline wage & Coefficient & Robust std. err & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% con & interval] \\
\hline \multicolumn{7}{|l|}{ATET} \\
\hline union (Union & & & & & & \\
\hline vs & & & & & & \\
\hline Nonunion) & . 901419 & . 2075309 & 4.34 & 0.000 & . 4946658 & 1.308172 \\
\hline \multicolumn{7}{|l|}{POmean} \\
\hline union & & & & & & \\
\hline Nonunion & 7.776417 & . 162121 & 47.97 & 0.000 & 7.458665 & 8.094168 \\
\hline
\end{tabular}

We obtain the same point estimate of the effect of union status as with margins. As before, the standard errors differ slightly between the two estimators, but they are asymptotically equivalent. The output also indicates that among the women who are in a union, their average wage would be \(\$ 7.78\) if they were not in a union.

\section*{- Technical note}

One advantage of the ATET over the ATE is that the ATET can be consistently estimated with slightly weaker assumptions than are required to consistently estimate the ATE. See Comparing the ATE and ATET in Remarks and examples of [CAUSAL] teffects intro advanced.

Both margins and teffects can estimate treatment effects using regression adjustment, so which should you use? In addition to regression adjustment, the teffects command implements other estimators of treatment effects; some of these estimators possess desirable robustness properties that we cannot replicate using margins. Moreover, all the teffects estimators use a common syntax and automatically present the estimated treatment effects, whereas we must first fit our own regression model and then call margins to obtain the treatment effects.

On the other hand, particularly with the at () option, margins gives us more flexibility in specifying our scenarios. The teffects commands allow us to measure the effect of a single binary or multinomial treatment, but we can have margins compute the effects of arbitrary interventions, as we illustrate in the next example.

\section*{> Example 9: Interventions involving multiple variables}

Suppose we want to see how women's wages would be affected if we could increase each woman's education level by one year. That is, we want to measure the treatment effect of an additional year of schooling. We assume that if a woman attains another year of schooling, she cannot simultaneously work. Thus, an additional year of education implies her total work experience must decrease by a year. The flexible at () option of margins allows us to manipulate both variables at once:
```

. quietly poisson wage i.union\#\#(c.grade c.ttl_exp\#\#c.ttl_exp), vce(robust)
. margins, at((asobserved) _all)
> at(grade=generate(grade+1) ttl_exp=generate(ttl_exp-1))
> contrast(atcontrast(r._at))
Contrasts of predictive margins Number of obs = 1,876
Model VCE: Robust
Expression: Predicted number of events, predict()
1._at: (asobserved)
2._at: grade = grade+1
ttl_exp = ttl_exp-1

|  | df | chi2 | P>chi2 |
| ---: | :---: | :---: | :---: |
| _at | 1 | 58.53 | 0.0000 |


|  | Delta-method Contrast std. err. |  | [95\% con | interval] |
| :---: | :---: | :---: | :---: | :---: |
| $\left(2 \mathrm{vs} \overline{1}^{-\mathrm{at}}\right.$ | . 3390392 | . 0443161 | . 2521813 | . 4258971 |

```

The first at () option instructs margins to obtain predicted wages for all women in the sample using their existing values for grade and ttl_exp and to record the mean of those predictions. The second at () option instructs margins to obtain the mean predicted wage under the counterfactual scenario where each woman's education level is increased by one year and total work experience is simultaneously decreased by one year. The contrast() option instructs margins to compute the difference between the two means. The output indicates that increasing education by one year, which will necessarily decrease work experience by the same amount, will cause the average wage to increase by about 34 cents per hour, a statistically significant amount.

\section*{Conclusion}
margins, contrast is a powerful command, and its abundance of suboptions may seem daunting. The suboptions are in the service of only three goals, however. There are three things that margins, contrast can do with a factor variable or a set of at () definitions:
1. Perform contrasts across the levels of the factor or set (as in example 1).
2. Perform a joint test across the levels of the factor or set (as in example 5).
3. Perform other tests and contrasts within each level of the factor or set (as in example 4).

The default behavior for variables specified inside at(), over(), and within() is to perform contrasts within groups; the default behavior for variables in the marginlist is to perform joint tests across groups.

\section*{Stored results}
margins, contrast stores the following additional results in \(r()\) :
Scalars
    \(r\) (k_terms) number of terms participating in contrasts

Macros
margins
overall or empty

Matrices
Macros
    r(overall)
```

```
    r(cmd)
```

    r(cmd)
    r(cmd2)
    r(cmd2)
        r(L)
    r(chi2)
        r(p)
        r(df)
    ```
contrast
(L)
matrix of contrasts applied to the margins
vector of \(\chi^{2}\) statistics
vector of \(p\)-values corresponding to \(r\) (chi2)
vector of degrees of freedom corresponding to \(r(p)\)
margins, contrast with the post option also stores the following additional results in e():
Scalars
e(k_terms) number of terms participating in contrasts
Macros
e(cmd)
e(cmd2)
contrast
margins
e(overall) overall or empty
Matrices
e(L) matrix of contrasts applied to the margins
e(chi2)
vector of \(\chi^{2}\) statistics
e(p)
vector of \(p\)-values corresponding to e(chi2)
e(df)
vector of degrees of freedom corresponding to \(e(p)\)

\section*{Methods and formulas}

See Methods and formulas in \([\mathrm{R}]\) margins and Methods and formulas in \([\mathrm{R}]\) contrast.

\section*{Reference}

Wooldridge, J. M. 2010. Econometric Analysis of Cross Section and Panel Data. 2nd ed. Cambridge, MA: MIT Press.

\section*{Also see}
[R] contrast - Contrasts and linear hypothesis tests after estimation
[ \(R\) ] lincom - Linear combinations of parameters
[R] margins - Marginal means, predictive margins, and marginal effects
[R] margins postestimation - Postestimation tools for margins
[R] margins, pwcompare - Pairwise comparisons of margins
[R] pwcompare - Pairwise comparisons

\section*{Title}
margins, pwcompare - Pairwise comparisons of margins
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Suboptions & Remarks and examples & Stored results & Methods and formulas \\
Also see & & &
\end{tabular}

\section*{Description}
margins with the pwcompare option performs pairwise comparisons of margins. margins, pwcompare extends the capabilities of pwcompare to any of the nonlinear responses, predictive margins, or other margins that can be estimated by margins.

\section*{Quick start}

All pairwise comparisons of the predictive margins for a after logit y a\#\#b x1 margins a, pwcompare

Same as above, but report both tests and confidence intervals for differences in predictive margins margins a, pwcompare(effects)

Same as above, but adjust \(p\)-values and confidence intervals for multiple comparisons using Bonferroni's method
margins a, pwcompare(effects) mcompare(bonferroni)
Report predictive margins for the levels of a, and group those that are not significantly different margins a, pwcompare(groups)

All pairwise comparisons of the predictive margins for combinations of levels of \(a\) and \(b\) margins a\#b, pwcompare

\section*{Menu}

Statistics \(>\) Postestimation

\section*{Syntax}
```

margins [marginlist] [if] [in] [weight] [, pwcompare margins_options]
margins [marginlist] [if] [in] [weight] [, pwcompare(suboptions) margins_options]

```
where marginlist is a list of factor variables or interactions that appear in the current estimation results. The variables may be typed with or without the i. prefix, and you may use any factor-variable syntax:
```

. margins i.sex i.group i.sex\#i.group, pwcompare

```
. margins sex group sex\#i.group, pwcompare
. margins sex\#\#group, pwcompare
suboptions Description
Pairwise comparisons
cieffects
pveffects
effects
cimargins
groups
sort
show effects table with confidence intervals; the default show effects table with \(p\)-values show effects table with confidence intervals and \(p\)-values show table of margins and confidence intervals show table of margins and group codes sort the margins or contrasts in each term
collect is allowed; see [U] 11.1.10 Prefix commands.
fweights, aweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.

\section*{Suboptions}
cieffects specifies that a table of the pairwise comparisons with their standard errors and confidence intervals be reported. This is the default.
pveffects specifies that a table of the pairwise comparisons with their standard errors, test statistics, and \(p\)-values be reported.
effects specifies that a table of the pairwise comparisons with their standard errors, test statistics, \(p\)-values, and confidence intervals be reported.
cimargins specifies that a table of the margins with their standard errors and confidence intervals be reported.
groups specifies that a table of the margins with their standard errors and group codes be reported. Margins with the same letter in the group code are not significantly different at the specified significance level.
sort specifies that the reported tables be sorted on the margins or contrasts in each term.

\section*{Remarks and examples}

You should be familiar with the concepts and syntax of both margins and pwcompare before using the pwcompare option of margins. These remarks build on those in \([R]\) margins and \([R]\) pwcompare.
margins can perform pairwise comparisons of any of the margins that it estimates.
We begin by fitting a logistic regression model using the NHANES II dataset, ignoring the complex survey nature of the data. Our dependent variable is highbp, an indicator for whether a person has high blood pressure. We fit an interacted model including two factor variables representing the region of the country as well as the continuous covariate bmi.
```

. use https://www.stata-press.com/data/r18/nhanes2
. logistic highbp region\#\#c.bmi
(output omitted)

```

By default, margins will compute the predictive margins of the probability of a positive outcome for each of the terms in marginlist after logistic regression. We will margin on region so that margins will estimate the average predicted probabilities of having high blood pressure conditional on being in each of the four regions and unconditional on BMI. We can specify the pwcompare option to obtain all possible pairwise comparisons of these predictive margins:
```

. margins region, pwcompare
Pairwise comparisons of predictive margins Number of obs = 10,351
Model VCE: OIM
Expression: Pr(highbp), predict()

```
\begin{tabular}{c|rrrr}
\hline & \multicolumn{3}{|c}{\begin{tabular}{c} 
Delta-method \\
std. err.
\end{tabular}} & \begin{tabular}{r} 
Unadjusted \\
[95\% conf. interval]
\end{tabular} \\
\hline region & & & & \\
MW vs NE & -.0377194 & .0133571 & -.0638987 & -.01154 \\
S vs NE & -.0156843 & .0133986 & -.041945 & .0105764 \\
W vs NE & -.006873 & .0136595 & -.0336451 & .019899 \\
S vs MW & .0220351 & .0124564 & -.0023789 & .0464492 \\
W vs MW & .0308463 & .0127366 & .0058831 & .0558096 \\
W vs S & .0088112 & .0127801 & -.0162373 & .0338598 \\
\hline
\end{tabular}

This table gives each of the pairwise differences with confidence intervals. We can see that the confidence interval in the row labeled MW vs NE does not include 0 . At the \(5 \%\) level, the predictive margins for the first and second regions, the Northeast and the Midwest, are significantly different. The same is true of the second and fourth regions, the Midwest and the West. With many pairwise comparisons, output in this format can be difficult to sort through. We can organize it by adding the group suboption:
```

. margins region, pwcompare(group)
Pairwise comparisons of predictive margins Number of obs = 10,351
Model VCE: OIM
Expression: Pr(highbp), predict()
Model VCE: OIM
Expression: $\operatorname{Pr}($ highbp $), ~ p r e d i c t()$

```
\begin{tabular}{c|ccr}
\hline & \multicolumn{3}{|c}{\begin{tabular}{c} 
Delta-method Unadjusted \\
std. err.
\end{tabular}} \\
\hline Margin & groups \\
\hline region & & & \\
NE & .4388358 & .010069 & B \\
MW & .4011164 & .0087764 & A \\
S & .4231516 & .0088395 & AB \\
W & .4319628 & .0092301 & B \\
\hline
\end{tabular}
```

Note: Margins sharing a letter in the group label are not significantly different at the $5 \%$ level.
are not significantly different at the 5%
level.

```

The group output includes the predictive margins for each region and letters denoting margins that are not significantly different from one another. In this case, the Northeast (NE), South (S), and West (W) regions have the letter B in the "Unadjusted Groups" column. The letter B indicates that the average predicted probability for the Northeast region is not significantly different from the average predicted probabilities for the South and West regions at the \(5 \%\) significance level. The Midwest (MW) region does not share a letter with the Northeast region nor the West region, which indicates that the average predicted probability for the Midwest region is significantly different for each of the other two regions at our \(5 \%\) level.

We can also include the mcompare(bonferroni) option to perform tests using Bonferroni's method to account for making multiple comparisons.
```

. margins region, pwcompare(group) mcompare(bonferroni)
Pairwise comparisons of predictive margins Number of obs = 10,351
Model VCE: OIM
Expression: Pr(highbp), predict()

```
\begin{tabular}{r|r}
\hline & \(\begin{array}{r}\text { Number of } \\
\text { comparisons }\end{array}\) \\
\hline region & 6 \\
\hline
\end{tabular}
\begin{tabular}{c|ccc}
\hline & \multicolumn{3}{|c}{\begin{tabular}{c} 
Delta-method Bonferroni \\
std. err.
\end{tabular}} \\
\hline Margin & groups \\
\hline region & & & \\
NE & .4388358 & .010069 & B \\
MW & .4011164 & .0087764 & A \\
S & .4231516 & .0088395 & AB \\
W & .4319628 & .0092301 & AB \\
\hline
\end{tabular}

Note: Margins sharing a letter in the group label are not significantly different at the 5\% level.

We now see the letter A on the row corresponding to the West region. At the \(5 \%\) level and with Bonferroni's adjustment, the predictive margins for the probability in the Midwest and West regions are not significantly different.

\section*{Stored results}
margins, pwcompare stores the following additional results in \(r()\) :
Scalars
\(r\) (k_terms) number of terms participating in pairwise comparisons
Macros
```

r(cmd)
r(cmd2)
r(group\#)
r(mcmethod_vs)
r(mctitle_vs)
r(mcadjustall_vs)

```

Matrices
```

        r(b)
    r(V)
    ```
    \(r\left(b \_v s\right)\)
    r(V_vs)
    \(r\) (error_vs)
    r(table_vs)
    r(L)
pwcompare
margins
group code for the \#th margin in \(r(b)\)
method from mcompare()
title for method from mcompare()
adjustall or empty
margin estimates
variance-covariance matrix of the margin estimates
margin difference estimates
variance-covariance matrix of the margin difference estimates
margin difference estimability codes;
0 means estimable,
8 means not estimable
matrix containing the margin differences with their standard errors, test statistics,
\(p\)-values, and confidence intervals
matrix that produces the margin differences
margins, pwcompare with the post option also stores the following additional results in e():
Scalars
e(k_terms) number of terms participating in pairwise comparisons
Macros
e(cmd)
pwcompare
e(cmd2)
margins
Matrices
e(b)
e(V)
e(b_vs)
margin estimates
variance-covariance matrix of the margin estimates
e(V_vs)
margin difference estimates
e(error_vs)
variance-covariance matrix of the margin difference estimates
margin difference estimability codes;
0 means estimable,
8 means not estimable
e(L) matrix that produces the margin differences

\section*{Methods and formulas}

See Methods and formulas in \([R]\) margins and Methods and formulas in \([R]\) pwcompare.

\footnotetext{
Also see
[R] contrast - Contrasts and linear hypothesis tests after estimation
\([\mathrm{R}]\) margins - Marginal means, predictive margins, and marginal effects
[R] margins, contrast - Contrasts of margins
[R] margins postestimation - Postestimation tools for margins
[R] pwcompare - Pairwise comparisons
}

\section*{Title}
marginsplot - Graph results from margins (profile plots, etc.)
\begin{tabular}{lll} 
Description & Menu & Syntax \\
Options & Remarks and examples & Addendum: Advanced uses of dimlist \\
Acknowledgments & References & Also see
\end{tabular}

\section*{Description}
marginsplot graphs the results of the immediately preceding margins command; see \([R]\) margins. Common names for some of the graphs that marginsplot can produce are profile plots and interaction plots.
marginsplot is also available after estat lcprob and estat lcmean; see [SEM] estat lcprob, [SEM] estat lcmean, [FMM] estat lcprob, and [FMM] estat lemean.

\section*{Menu}

Statistics \(>\) Postestimation

\section*{Syntax}
marginsplot [, options]
options
Main
xdimension(dimlist [, dimopts]) use dimlist to define \(x\) axis plotdimension(dimlist [, dimopts]) bydimension(dimlist [, dimopts]) graphdimension(dimlist [, dimopts]) horizontal noci
derivlabels
name(name |stub [, replace])
Description create plots for groups in dimlist create subgraphs for groups in dimlist create graphs for groups in dimlist swap \(x\) and \(y\) axes do not plot confidence intervals name of graph, or stub if multiple graphs use labels attached to marginal-effects variables

Labels
allxlabels
nolabels
allsimplelabels
nosimplelabels
separator (string)
noseparator
place ticks and labels on the \(x\) axis for each value label groups with their values, not their labels forgo variable name and equal signs in all labels include variable name and equal signs in all labels separator for labels when multiple variables are specified in a dimension
do not use a separator
```

Plot
plotopts(plot_options)
plot\#opts(plot_options)
recast(plottype)
Cl plot
ciopts(rcap_options)
ci\#opts(rcap_options)
recastci(plottype)
mcompare(method)
level(\#)

```
Pairwise
    unique
    csort
Add plots
    addplot (plot)
Y axis, X axis, Titles, Legend, Overall, By
    twoway_options
    byopts (byopts)
affect rendition of all margin plots affect rendition of \#th margin plot plot margins using plottype
affect rendition of all confidence interval plots affect rendition of \#th confidence interval plot plot confidence intervals using plottype adjust for multiple comparisons set confidence level
plot only unique pairwise comparisons
sort comparison categories first
add other plots to the graph
any options documented in [G-3] twoway_options how subgraphs are combined, labeled, etc.
where dimlist may be any of the dimensions across which margins were computed in the immediately preceding margins command; see [R] margins. That is to say, dimlist may be any variable used in the margins command, including variables specified in the at(), over(), and within() options. More advanced specifications of dimlist are covered in Addendum: Advanced uses of dimlist.
dimopts Description
\begin{tabular}{ll} 
labels(lablist) & \begin{tabular}{l} 
list of quoted strings to label each level of the dimension \\
list of enumerated labels
\end{tabular} \\
elabels(elablist) & \begin{tabular}{l} 
label groups with their values, not their labels \\
nolabels
\end{tabular} \\
forgo variable name and equal signs in all labels \\
allsimplelabels & \begin{tabular}{l} 
include variable name and equal signs in all labels \\
nosimplelabels
\end{tabular} \\
\begin{tabular}{l} 
separator for labels when multiple variables are specified \\
in the dimension
\end{tabular} \\
noseparator & do not use a separator
\end{tabular}
where lablist is defined as
"label" ["label" [...]]
elablist is defined as
\# "label" [\# "label" [...] ]
and the \#s are the indices of the levels of the dimension-1 is the first level, 2 is the second level, and so on.

Description
marker_options
marker_label_options
cline_options
change look of markers (color, size, etc.) add marker labels; change look or position change look of the line
method
noadjust
bonferroni [adjustall]
sidak [adjustall]
scheffe

Description
do not adjust for multiple comparisons
Bonferroni's method; adjust across all terms
Šidák's method; adjust across all terms
Scheffé's method

\section*{Options}
\(\qquad\) Main
xdimension(), plotdimension(), bydimension(), and graphdimension() specify the variables from the preceding margins command whose group levels will be used for the graph's \(x\) axis, plots, by () subgraphs, and graphs.
marginsplot chooses default dimensions based on the margins command. In most cases, the first variable appearing in an at () option and evaluated over more than one value is used for the \(x\) axis. If no at () variable meets this condition, the first variable in the marginlist is usually used for the \(x\) axis and the remaining variables determine the plotted lines or markers. Pairwise comparisons and graphs of marginal effects (derivatives) have different defaults. In all cases, you may override the defaults and explicitly control which variables are used on each dimension of the graph by using these dimension options.
Each of these options supports suboptions that control the labeling of the dimension—axis labels for xdimension(), plot labels for plotdimension(), subgraph titles for bydimension(), and graph titles for graphdimension() titles.
For examples using the dimension options, see Controlling the graph's dimensions.
xdimension (dimlist [, dimopts]) specifies the variables for the \(x\) axis in dimlist and controls the content of those labels with dimopts.
plotdimension(dimlist [, dimopts]) specifies in dimlist the variables whose group levels determine the plots and optionally specifies in dimopts the content of the plots' labels.
bydimension(dimlist [, dimopts]) specifies in dimlist the variables whose group levels determine the by () subgraphs and optionally specifies in dimopts the content of the subgraphs' titles. For an example using by (), see Three-way interactions.
graphdimension(dimlist [, dimopts]) specifies in dimlist the variables whose group levels determine the graphs and optionally specifies in dimopts the content of the graphs' titles.
horizontal reverses the default \(x\) and \(y\) axes. By default, the \(y\) axis represents the estimates of the margins and the \(x\) axis represents one or more factors or continuous covariates. Specifying horizontal swaps the axes so that the \(x\) axis represents the estimates of the margins. This option can be useful if the labels on the factor or continuous covariates are long.

The horizontal option is discussed in Horizontal is sometimes better.
noci removes plots of the pointwise confidence intervals. The default is to plot the confidence intervals.
derivlabels specifies that variable labels attached to marginal-effects variables be used in place of the variable names in titles and legends. Marginal-effects variables are the ones specified in margins's option dydx(), dyex(), eydx(), or eyex().
name (name \(\mid\) stub [, replace]) specifies the name of the graph or graphs. If the graphdimension() option is specified, or if the default action is to produce multiple graphs, then the argument of name () is taken to be stub and graphs named stub1, stub2, ... are created.
The replace suboption causes existing graphs with the specified name or names to be replaced. If name() is not specified, default names are used and the graphs may be replaced by subsequent marginsplot or other graphing commands.

\section*{Labels}

With the exception of allxlabels, all of these options may be specified either directly as options or as dimopts within options xdimension(), plotdimension(), bydimension(), and graphdimension(). When specified in one of the dimension options, only the labels for that dimension are affected. When specified outside the dimension options, all labels on all dimensions are affected. Specifications within the dimension options take precedence.
allxlabels specifies that tick marks and labels be placed on the \(x\) axis for each value of the \(x\)-dimension variables. By default, if there are more than 25 ticks, default graph axis labeling rules are applied. Labeling may also be specified using the standard graph twoway \(x\)-axis label rules and options-xlabel(); see [G-3] axis_label_options.
nolabels specifies that value labels not be used to construct graph labels and titles for the group levels in the dimension. By default, if a variable in a dimension has value labels, those labels are used to construct labels and titles for axis ticks, plots, subgraphs, and graphs.
Graphs of contrasts and pairwise comparisons are an exception to this rule and are always labeled with values rather than value labels.
allsimplelabels and nosimplelabels control whether graphs' labels and titles include just the values of the variables or include variable names and equal signs. The default is to use just the value label for variables that have value labels and to use variable names and equal signs for variables that do not have value labels. An example of the former is "Female" and the latter is "country=2".
Sometimes, value labels are universally descriptive, and sometimes they have meaning only when considered in relation to their variable. For example, "Male" and "Female" are typically universal, regardless of the variable from which they are taken. "High" and "Low" may not have meaning unless you know they are in relation to a specific measure, say, blood-pressure level. The allsimplelabels and nosimplelabels options let you override the default labeling.
allsimplelabels specifies that all titles and labels use just the value or value label of the variable.
nosimplelabels specifies that all titles and labels include varname \(=\) before the value or value label of the variable.
separator (string) and noseparator control the separator between label sections when more than one variable is used to specify a dimension. The default separator is a comma followed by a space, but no separator may be requested with noseparator or the default may be changed to any string with separator ().
For example, if plotdimension(a b) is specified, the plot labels in our graph legend might be " \(a=1, b=1 ", " a=1, b=2 ", \ldots\) Specifying separator (: ) would create labels " \(a=1: b=1 "\), \(" a=1: b=2 ", \ldots\)
plotopts (plot_options) affects the rendition of all margin plots. The plot_options can affect the size and color of markers, whether and how the markers are labeled, and whether and how the points are connected; see [G-3] marker_options, [G-3] marker_label_options, and [G-3] cline_options.
These settings may be overridden for specific plots by using the plot\#opts() option.
plot\#opts (plot_options) affects the rendition of the \#th margin plot. The plot_options can affect the size and color of markers, whether and how the markers are labeled, and whether and how the points are connected; see [G-3] marker_options, [G-3] marker_label_options, and [G-3] cline_options.
recast (plottype) specifies that margins be plotted using plottype. plottype may be scatter, line, connected, bar, area, spike, dropline, or dot; see [G-2] graph twoway. When recast () is specified, the plot-rendition options appropriate to the specified plottype may be used in lieu of plot_options. For details on those options, follow the appropriate link from [G-2] graph twoway. For an example using recast (), see Continuous covariates.
You may specify recast() within a plotopts() or plot\#opts() option. It is better, however, to specify it as documented here, outside those options. When specified outside those options, you have greater access to the plot-specific rendition options of your specified plottype.

\section*{CI plot}
ciopts (rcap_options) affects the rendition of all confidence interval plots; see [G-3] rcap_options.
These settings may be overridden for specific confidence interval plots with the ci\#opts() option. ci\#opts (rcap_options) affects the rendition of the \#th confidence interval; see [G-3] rcap_options. recastci(plottype) specifies that confidence intervals be plotted using plottype. plottype may be rarea, rbar, rspike, rcap, rcapsym, rline, rconnected, or rscatter; see [G-2] graph twoway. When recastci() is specified, the plot-rendition options appropriate to the specified plottype may be used in lieu of rcap_options. For details on those options, follow the appropriate link from [G-2] graph twoway.
For an example using recastci(), see Continuous covariates.
You may specify recastci() within a ciopts() or ci\#opts() option. It is better, however, to specify it as documented here, outside those options. When specified outside those options, you have greater access to the plot-specific rendition options of your specified plottype.
mcompare (method) specifies the method for confidence intervals that account for multiple comparisons within a factor-variable term. The default is determined by the margins results stored in \(r()\). If marginsplot is working from margins results stored in e(), the default is mcompare (noadjust).
level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is determined by the margins results stored in \(r()\). If marginsplot is working from margins results stored in e(), the default is level(95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals.

\section*{Pairwise}

These options have an effect only when the pwcompare option was specified on the preceding margins command.
unique specifies that only unique pairwise comparisons be plotted. The default is to plot all pairwise comparisons, including those that are mirror images of each other-"male" versus "female" and "female" versus "male". margins reports only the unique pairwise comparisons. unique
also changes the default xdimension() for graphs of pairwise comparisons from the reference categories ( \(\_\mathrm{pw} 0\) ) to the comparisons of each pairwise category ( \(\_\mathrm{pw}\) ).
Unique comparisons are often preferred with horizontal graphs that put all pairwise comparisons on the \(x\) axis, whereas including the full matrix of comparisons is preferred for charts showing the reference groups on an axis and the comparison groups as plots; see Pairwise comparisons and Horizontal is sometimes better.
csort specifies that comparison categories are sorted first, and then reference categories are sorted within comparison category. The default is to sort reference categories first, and then sort comparison categories within reference categories. This option has an observable effect only when \(\quad\) pw is also specified in one of the dimension options. It then determines the order of the labeling in the dimension where _pw is specified.
addplot (plot) provides a way to add other plots to the generated graph; see [G-3] addplot_option. For an example using addplot(), see Adding scatterplots of the data.
If multiple graphs are drawn by a single marginsplot command or if plot specifies plots with multiple \(y\) variables, for example, scatter y1 y2 \(x\), then the graph's legend will not clearly identify all the plots and will require customization using the legend() option; see [G-3] legend_options.

Y axis, X axis, Titles, Legend, Overall, By
twoway_options are any of the options documented in [G-3] twoway_options. These include options for titling the graph (see [G-3] title_options); for saving the graph to disk (see [G-3] saving_option); for controlling the labeling and look of the axes (see [G-3] axis_options); for controlling the look, contents, position, and organization of the legend (see [G-3] legend_options); for adding lines (see [G-3] added_line_options) and text (see [G-3] added_text_options); and for controlling other aspects of the graph's appearance (see [G-3] twoway_options).

The label() suboption of the legend() option has no effect on marginsplot. Use the order () suboption instead.
byopts (byopts) affects the appearance of the combined graph when bydimension() is specified or when the default graph has subgraphs, including the overall graph title, the position of the legend, and the organization of subgraphs. See [G-3] by_option.

\section*{Remarks and examples}

Remarks are presented under the following headings:

\author{
Introduction \\ Dataset \\ Profile plots \\ Interaction plots \\ Contrasts of margins-effects (discrete marginal effects) \\ Three-way interactions \\ Continuous covariates \\ Plots at every value of a continuous covariate \\ Contrasts of at() groups-discrete effects \\ Controlling the graph's dimensions \\ Pairwise comparisons \\ Horizontal is sometimes better \\ Marginal effects \\ Plotting a subset of the results from margins \\ Advanced usage \\ Plots with multiple terms \\ Plots with multiple at() options \\ Adding scatterplots of the data \\ Video examples
}

\section*{Introduction}
marginsplot is a post-margins command. It graphs the results of the margins command, whether those results are marginal means, predictive margins, marginal effects, contrasts, pairwise comparisons, or other statistics; see [R] margins.

By default, the margins are plotted on the \(y\) axis, and all continuous and factor covariates specified in the margins command will usually be placed on the \(x\) axis or used to identify plots. Exceptions are discussed in the following sections and in Addendum: Advanced uses of dimlist below.
marginsplot produces classic plots, such as profile plots and interaction plots. Beyond that, anything that margins can compute, marginsplot can graph.

We will be using some relatively complicated margins commands with little explanation of the syntax. We will also avoid lengthy interpretations of the results of margins. See [R] margins for the complete syntax of margins and discussions of its results.

All graphs in this entry were drawn using the s2gcolor scheme; see [G-4] Scheme s2.
Mitchell (2021) and Baldwin (2019) show in many examples how to use marginsplot to understand a fitted model.

\section*{Dataset}

For continuity, we will use one dataset for most examples-the Second National Health and Nutrition Examination Survey (NHANES II) (McDowell et al. 1981). NHANES II is part of a study to assess the health and nutritional status of adults and children in the United States. It is designed to be a nationally representative sample of the US population. This particular sample is from 1976 to 1980.

The survey nature of the dataset-weights, strata, and sampling units-will be ignored in our analyses. We are discussing graphing, not survey statistics. If you would like to see the results with the appropriate adjustments for the survey design, just add svy: before each estimation command, and if you wish, add vce (unconditional) as an option to each margins command. See [R] margins, particularly the discussion and examples under Obtaining margins with survey data and representative
samples, for reasons why you probably would want to add vce(unconditional) when analyzing survey data. For the most part, adjusting for survey design produces moderately larger confidence intervals and relatively small changes in point estimates.

\section*{Profile plots}

What does my estimation say about how my response varies as one (or more) of my covariates changes? That is the question that is answered by profile plots. Profile plots are also referred to as plots of estimated (or expected, or least-squares) means, though that is unnecessarily restrictive when considering models of binary, count, and ordered outcomes. In the latter cases, we might prefer to say they plot conditional expectations of responses, where a response might be a probability.

What we do with the other covariates depends on the questions we wish to answer. Sometimes, we wish to hold other covariates at fixed values, and sometimes we wish to average the response over their values. margins can do either, so you can graph either.

We can fit a fully factorial two-way ANOVA of systolic blood pressure on age group and sex using the NHANES II data.
. use https://www.stata-press.com/data/r18/nhanes2
. anova bpsystol agegrp\#\#sex
\begin{tabular}{|c|c|c|c|c|c|}
\hline & \[
\begin{array}{ll}
\text { Number of obs } & = \\
\text { Root MSE } & =
\end{array}
\] & \[
\begin{array}{r}
10,351 \\
20.2209
\end{array}
\] & \multicolumn{2}{|l|}{\begin{tabular}{l}
R-squared = \\
Adj R-squared =
\end{tabular}} & \[
\begin{aligned}
& 0.2497 \\
& 0.2489
\end{aligned}
\] \\
\hline Source & Partial SS & df & MS & F & Prob>F \\
\hline Model & 1407229.3 & 11 & 127929.93 & 312.88 & 0.0000 \\
\hline agegrp & 1243037.8 & 5 & 248607.56 & 608.02 & 0.0000 \\
\hline sex & 27728.379 & 1 & 27728.379 & 67.81 & 0.0000 \\
\hline agegrp\#sex & 88675.043 & 5 & 17735.009 & 43.37 & 0.0000 \\
\hline Residual & 4227440.7 & 10,339 & 408.88294 & & \\
\hline Total & 5634670 & 10,350 & 544.41256 & & \\
\hline
\end{tabular}

If you are more comfortable with regression than ANOVA, then type
```

. regress bpsystol agegrp\#\#sex

```

The anova and regress commands fit identical models. The output from anova displays all the terms in the model and thus tends to be more conducive to exploration with margins and marginsplot.

We estimate the predictive margins of systolic blood pressure for each age group using margins.
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline \multicolumn{5}{|l|}{Predictive margins} & \multicolumn{2}{|l|}{\multirow[t]{2}{*}{Number of obs \(=10,351\)}} \\
\hline \multicolumn{5}{|l|}{Expression: Linear prediction, predict()} & & \\
\hline & \multicolumn{3}{|l|}{\begin{tabular}{l}
Delta-method \\
Margin std. err.
\end{tabular}} & \(P>|t|\) & \multicolumn{2}{|l|}{[95\% conf. interval]} \\
\hline agegrp & & & & & & \\
\hline 20-29 & 117.2684 & . 419845 & 279.31 & 0.000 & 116.4454 & 118.0914 \\
\hline 30-39 & 120.2383 & . 5020813 & 239.48 & 0.000 & 119.2541 & 121.2225 \\
\hline 40-49 & 126.9255 & . 56699 & 223.86 & 0.000 & 125.8141 & 128.0369 \\
\hline 50-59 & 135.682 & . 5628593 & 241.06 & 0.000 & 134.5787 & 136.7853 \\
\hline 60-69 & 141.5285 & . 3781197 & 374.30 & 0.000 & 140.7873 & 142.2696 \\
\hline 70+ & 148.1096 & . 6445073 & 229.80 & 0.000 & 146.8463 & 149.373 \\
\hline
\end{tabular}

The six predictive margins are just the averages of the predictions over the estimation sample, holding agegrp to each of its six levels. If this were a designed experiment rather than survey data, we might wish to assume the cells are balanced-that they have the same number of observations-and thus estimate what are often called expected means or least-squares means. To do that, we would simply add the asbalanced option to the margins command. The NHANES II data are decidedly unbalanced over sex\#agegrp cells. So much so that it is unreasonable to assume the cells are balanced.

We graph the results:
```

. marginsplot
Variables that uniquely identify margins: agegrp

```

Predictive margins of agegrp with \(95 \%\) CIs


Profile plots are often drawn without confidence intervals (CIs). The CIs may be removed by adding the noci option. We prefer to see the CIs.

Disciplines vary widely in their use of the term profile plot. Some disciplines consider any connected plot of a response over values of other variables to be a profile plot. By that definition, most graphs in this entry are profile plots.

\section*{Interaction plots}

Interaction plots are often used to explore the form of an interaction. The interaction term in our ANOVA results is highly significant. Are the interaction effects also large enough to matter? What form do they take? We can answer these questions by fixing agegrp and sex to each possible combination of the two covariates and estimating the margins for those cells.

Then, we can graph the results:
```

. marginsplot
Variables that uniquely identify margins: agegrp sex

```


It is clear that the effect of age differs by sex-there is an interaction. If there were no interaction, then the two lines would be parallel.

While males start out with higher systolic blood pressure, females catch up to the males as age increases and may even surpass males in the upper age groups. We say "may" because we cannot tell if the differences are statistically significant. The CIs overlap for the top three age groups. It is tempting to conclude from this overlap that the differences are not statistically significant. Do not fall into this trap. Likewise, do not fall into the trap that the first three age groups are different because their CIs do not overlap. The CIs are for the point estimates, not the differences. There is a covariance between the differences that we must consider if we are to make statements about those differences.

\section*{Contrasts of margins-effects (discrete marginal effects)}

To assess the differences, all we need to do is ask margins to contrast the sets of effects that we just estimated; see [R] margins, contrast. With only two groups in sex, it does not matter much which contrast operator we choose. We will use the reference contrast. It will compare the difference between males and females, with males (the first category) as the reference category.
. margins r.sex@agegrp
Contrasts of adjusted predictions \(\quad\) Number of obs \(=10,351\)
Expression: Linear prediction, predict()
\begin{tabular}{r|rrr}
\hline & \(d f\) & F & \(\mathrm{P}>\mathrm{F}\) \\
\hline sex@agegrp & & & \\
(Female vs Male) 20-29 & 1 & 224.92 & 0.0000 \\
(Female vs Male) 30-39 & 1 & 70.82 & 0.0000 \\
(Female vs Male) 40-49 & 1 & 12.15 & 0.0005 \\
(Female vs Male) 50-59 & 1 & 0.47 & 0.4949 \\
(Female vs Male) 60-69 & 1 & 3.88 & 0.0488 \\
(Female vs Male) 70+ & 1 & 6.37 & 0.0116 \\
Joint & 6 & 53.10 & 0.0000 \\
Denominator & 10339 & & \\
\hline
\end{tabular}
\begin{tabular}{l|rrrr}
\hline & \multicolumn{4}{|c}{\begin{tabular}{c} 
Delta-method \\
std. err.
\end{tabular}} \\
\hline Contrast & [95\% conf. interval] \\
\hline (Female vs Male) 20-29 & -12.60132 & .8402299 & -14.24833 & -10.9543 \\
(Female vs Male) 30-39 & -8.461161 & 1.005448 & -10.43203 & -6.490288 \\
(Female vs Male) 40-49 & -3.956451 & 1.134878 & -6.181031 & -1.731871 \\
(Female vs Male) 50-59 & -.7699782 & 1.128119 & -2.981309 & 1.441353 \\
(Female vs Male) 60-69 & 1.491684 & .756906 & .0080022 & 2.975367 \\
(Female vs Male) 70+ & 3.264762 & 1.293325 & .729594 & 5.79993 \\
\hline
\end{tabular}

Because we are looking for effects that are different from 0 , we will add a reference line at 0 to our graph.
. marginsplot, yline(0)
Variables that uniquely identify margins: agegrp


We can now say that females' systolic blood pressure is substantially and significantly lower than males' in the first three age groups but is significantly higher in the last two age groups. Despite the overlapping CIs for the last two age groups in the interaction graph, the effect of sex is significant in these age groups.

The terminology for what we just estimated and graphed varies widely across disciplines. Those versed in design of experiments refer to these values as contrasts or effects. Economists and some other social scientists call them marginal or partial effects. The latter groups might be more comfortable if we avoided the whole concept of contrasts and instead estimated the effects by typing
```

. margins agegrp, dydx(sex)

```

This will produce estimates that are identical to those shown above, and we can graph them by typing marginsplot.

The advantage of using the contrast notation and thinking in contrasts is most evident when we take marginal effects with respect to a categorical covariate with more than two levels. Marginal effects for each level of the covariate will be taken with respect to a specified base level. Contrasts are much more flexible. Using the r. operator, we can reproduce the marginal-effects results by taking derivatives with respect to a reference level (as we saw above.) We can also estimate the marginal effect of first moving from level 1 to level 2, then from level 2 to level 3, then from level 3 to level \(4, \ldots\) using the ar . or "reverse adjacent" operator. Adjacent effects (marginal effects) can be valuable when evaluating an ordinal covariate, such as agegrp in our current model. For a discussion of contrasts, see \([R]\) contrast and \([R]\) margins, contrast.

\section*{Three-way interactions}
marginsplot can handle any number of covariates in your margins command. Consider the three-way ANOVA model that results from adding an indicator for whether an individual has been diagnosed with diabetes. We will fully interact the new covariate with the others in the model.
\begin{tabular}{|c|c|c|c|c|c|}
\hline & \[
\begin{array}{ll}
\text { Number of obs } & = \\
\text { Root MSE } & =
\end{array}
\] & \[
\begin{aligned}
& 10,349 \\
& 20.131
\end{aligned}
\] & \multicolumn{2}{|l|}{\begin{tabular}{l}
R -squared \(=\) \\
Adj R-squared =
\end{tabular}} & \[
\begin{aligned}
& 0.2572 \\
& 0.2556
\end{aligned}
\] \\
\hline Source & Partial SS & df & MS & F & Prob>F \\
\hline Model & 1448983.2 & 23 & 62999.268 & 155.45 & 0.0000 \\
\hline agegrp & 107963.58 & 5 & 21592.716 & 53.28 & 0.0000 \\
\hline sex & 1232.7927 & 1 & 1232.7927 & 3.04 & 0.0812 \\
\hline agegrp\#sex & 11679.592 & 5 & 2335.9185 & 5.76 & 0.0000 \\
\hline diabetes & 7324.9892 & 1 & 7324.9892 & 18.07 & 0.0000 \\
\hline agegrp\#diabetes & 5484.5462 & 5 & 1096.9092 & 2.71 & 0.0189 \\
\hline sex\#diabetes & 102.98824 & 1 & 102.98824 & 0.25 & 0.6142 \\
\hline agegrp\#sex\#diabetes & 4863.1497 & 5 & 972.62994 & 2.40 & 0.0349 \\
\hline Residual & 4184296.9 & 10,325 & 405.25878 & & \\
\hline Total & 5633280 & 10,348 & 544.38346 & & \\
\hline
\end{tabular}

The three-way interaction is significant, as is the main effect of diabetes and its interaction with agegrp.

Again, if you are more comfortable with regression than ANOVA, you may type
```

. regress bpsystol agegrp\#\#sex\#\#diabetes

```

The margins and marginsplot results will be the same.

We estimate the expected cell means for each combination of agegrp, sex, and diabetes, and then graph the results by typing
```

. margins agegrp\#sex\#diabetes
(output omitted)
. marginsplot
Variables that uniquely identify margins: agegrp sex diabetes

```

Adjusted predictions of agegrp\#sex\#diabetes with \(95 \% \mathrm{Cls}\)


The graph is busy and difficult to interpret.
We can make it better by putting those with diabetes on one subgraph and those without on another:
```

. marginsplot, by(diabetes)
Variables that uniquely identify margins: agegrp sex diabetes

```

Adjusted predictions of agegrp\#sex\#diabetes with 95\% Cls


We notice much larger CIs for diabetics. That is not surprising because our sample contains only 499 diabetics compared with 9,850 nondiabetics.

A more interesting way to arrange the plots is by grouping the subgraphs on sex:
```

. marginsplot, by(sex)
Variables that uniquely identify margins: agegrp sex diabetes

```

Adjusted predictions of agegrp\#sex\#diabetes with 95\% Cls


Aside from increased systolic blood pressure in the upper-age groups, which we saw earlier, it appears that those with diabetes are at greater risk of higher systolic blood pressure for many upper-age groups. We can check that by having margins estimate the differences between diabetics and nondiabetics, and graphing the results.
```

. margins r.diabetes@agegrp\#sex
(output omitted)
. marginsplot, by(sex) yline(0)
Variables that uniquely identify margins: agegrp sex

```

Contrasts of adjusted predictions of diabetes@agegrp\#sex with 95\% CIs


With CIs above 0 for six of eight age groups over 40, this graph provides evidence that diabetes is related to higher blood pressure in those over 40.

\section*{Continuous covariates}
margins and marginsplot are just as useful with continuous covariates as they are with factor variables. As a variation on our ANOVA/regression models, let's move to a logistic regression, using as our dependent variable an indicator for whether a person has high blood pressure. We introduce a continuous covariate-body mass index (BMI), a measure of weight relative to height. High BMI is often associated with high blood pressure. We will allow the effect of BMI to vary across sexes, age groups, and sex/age combinations by fully interacting the covariates.
. logistic highbp sex\#\#agegrp\#\#c.bmi
If we wished, we could perform all the analyses above on this model. Instead of estimating margins, contrasts, and marginal effects on the level of systolic blood pressure, we would be estimating margins, contrasts, and marginal effects on the probability of having high blood pressure. You can see those results by repeating any of the prior commands that involve sex and agegrp. In this section, we will focus on the continuous covariate bmi.

With continuous covariates, rather than specify them in the marginlist of margins, we specify the specific values at which we want the covariate evaluated in an at () option. at () options are very flexible, and there are many ways to specify values; see Syntax of at() in [R] margins.

BMI in our sample ranges from 12.4 to 61.1. Let's estimate the predictive margins for males and females at levels of BMI from 10 through 65 at intervals of 5 and graph the results:
```

. margins sex, at(bmi=(10(5)65))
(output omitted)
. marginsplot, xlabel(10(10)60)
Variables that uniquely identify margins: bmi sex

```


We added the xlabel (10(10)60) option to improve the labeling of the \(x\) axis. You may add any twoway_options (see [G-3] twoway_options) to the marginsplot command.

For a given BMI, males are generally more susceptible to high blood pressure, though the effect is attenuated by the logistic response when the probabilities approach 0 or 1 .

Because bmi is continuous, we might prefer to see the response graphed using a line. We might also prefer that the CIs be plotted as areas. We change the plottype of the response by using the recast() option and the plottype of the CI by using the recastci() option:
```

. marginsplot, xlabel(10(10)60) recast(line) recastci(rarea)

```
Variables that uniquely identify margins: bmi sex


The CIs are a little dark for our tastes. You can dim them a bit by reducing the intensity of their color. Adding ciopts (color(*.8)) to our marginsplot command will do that. Any plot option accepted by twoway rarea (see [G-2] graph twoway rarea) may be specified in a ciopts() option.

Given their confidence regions, the male and female profiles appear to be statistically different over most of the range of BMI. As with the profiles of categorical covariates, we can check that assertion by contrasting the two profiles on sex and graphing the results. Let's improve the smoothness of the response by specifying intervals of 1 instead of 5 .
```

. margins r.sex, at(bmi=(10(1)65))
(output omitted)
. marginsplot, xlabel(10(10)60) recast(line) recastci(rarea)
Variables that uniquely identify margins: bmi

```

Contrasts of predictive margins of sex with \(95 \% \mathrm{Cls}\)


We see that the difference between the sexes is largest at a BMI of about 35 and that the sexes respond more similarly with very high and very low BMI. This shape is largely determined by the response of the logistic function, which is attenuated near probabilities 0 and 1 , combined with the fact that the lowest measured BMIs are associated with low probabilities of high blood pressure and the highest measured BMIs are associated with high probabilities of high blood pressure.

As when we contrasted profiles of categorical variables, different disciplines will think of this graph differently. Those familiar with designed experiments will be comfortable with the terms used above-this is a contrast of profiles, or a profile of effects, or a profile of a contrast. Many social scientists will prefer to think of this as a graph of marginal or partial effects. For them, this is a plot of the discrete marginal effect of being female for various levels of BMI. They can obtain an identical graph, with labeling more appropriate for the marginal effect's interpretation, by typing
```

. margins, at(bmi=(10(1)65)) dydx(sex)
. marginsplot, xlabel(10(10)60) recast(line) recastci(rarea)

```

We can also plot profiles of the response of BMI by levels of another continuous covariate (rather than by the categorical variable sex). To do so, we will need another continuous variable in our model. We have been using age groups as a covariate to emphasize the treatment of categorical variables and to allow the effect of age to be flexible. Our dataset also has age recorded in integer years. We replace agegrp with continuous age in our logistic regression.
```

. logistic highbp sex\#\#c.age\#\#c.bmi

```

We can now obtain profiles of BMI for different ages by specifying ranges for both bmi and age in a single at () option on the margins command:
```

. margins sex, at(bmi=(10(5)60) age=(20(10)70))

```

With six ages specified, we have many profiles, so we will dispense with the CIs by adding the noci option:
```

. marginsplot, noci by(sex)
Variables that uniquely identify margins: bmi age sex

```

Adjusted predictions of sex


Our model seems to indicate that males have a sharper reaction to body mass indices than do females. Likewise, younger subjects display a sharper response, while older subjects have a more gradual response with earlier onset. That interpretation might be a result of our parametric treatment of age. As it turns out, the interpretation holds if we allow age to take more flexible forms or return to our use of age groups, which allows each of seven age groups to have unique BMI profiles. Here are the commands to perform that analysis:
```

. logistic highbp sex\#\#agegrp\#\#c.bmi
(output omitted)
. margins sex\#agegrp, at(bmi=(10(5)60))
(output omitted)
. marginsplot, noci by(sex)
Variables that uniquely identify margins: bmi sex agegrp

```

Adjusted predictions of sex\#agegrp


\section*{Plots at every value of a continuous covariate}

In some cases, the specific values of a continuous covariate are important, and we want to plot the response at those specific values. Return to our logistic example with age treated as a continuous covariate.
```

. logistic highbp sex\#\#c.age\#\#c.bmi

```

We can use a programming trick to extract all the values of age and then supply them in an at () option, just as we would any list of values.
. levelsof age
. margins sex, at(age=('r(levels)'))
See [ P\(]\) levelsof for a discussion of the levelsof command. levelsof returns in \(r\) (levels) the sorted list of unique values of the specified varlist, in our case, age.

We can then plot the results using marginsplot.
This is not a very interesting trick when using our age variable, which is recorded as integers from 20 to 74 , but the approach will work with almost any continuous variable. In our model, bmi might seem more interesting, but there are 9,941 unique values of bmi in our dataset. A graph cannot resolve so many different values. For that reason, we usually recommend against plotting at every value of a covariate. Instead, graph at reasonable values over the range of the covariate by using the at () option, as we did earlier. This trick is best reserved for variables with a few, or at most a few dozen, unique values.

\section*{Contrasts of at() groups-discrete effects}

We have previously contrasted across the values of factor variables in our model. Put another way, we have estimated the discrete marginal effects of factor variables. We can do the same for the levels of variables in at () specifications and across separate at () specifications.

Returning to one of our logistic models and its margins, we earlier estimated the predictive margins of BMI at 5 -unit intervals for both sexes. These are the commands we typed:
```

. logistic highbp sex\#\#agegrp\#\#c.bmi
. margins sex, at(bmi=(10(5)65))
. marginsplot, xlabel(10(10)60)

```

We can estimate the discrete effects by sex of bmi moving from 10 to 15 , then from 15 to \(20, \ldots\), and then from 60 to 65 by contrasting the levels of the at () groups using the reverse-adjacent contrast operator (ar.). We specify the operator within the atcontrast() suboption of the contrast() option. We need to specify one other option. By default, margins, contrast will apply a contrast to all variables in its marginlist when a contrast has been requested. In this case, we do not want to contrast across sexes but rather to contrast across the levels of BMI within each sex. To prevent margins from contrasting across the sexes, we specify the marginswithin option. Our margins command is
```

. margins sex, at(bmi=(10(5)65)) contrast(atcontrast(ar._at) marginswithin)

```

And we graph the results using marginsplot:
```

. marginsplot
Variables that uniquely identify margins: bmi sex

```

Contrasts of predictive margins of sex with \(95 \% \mathrm{Cls}\)


The graph shows the contrasts (or if you prefer, discrete changes) in the probability of high blood pressure by sex as one increases BMI in 5-unit increments.

We can even estimate contrasts (discrete effects) across at () options. To start, let's compare the age-group profiles of the probability of high blood pressure for those in the 25 th and 75 th percentile of BMI.
```

. margins agegrp, at((p25) bmi) at((p75) bmi)
(output omitted)
. marginsplot
Variables that uniquely identify margins: agegrp _atopt
Multiple at() options specified:
_atoption=1: (p25) bmi
_atoption=2: (p75) bmi

```

Predictive margins of agegrp with \(95 \% \mathrm{Cls}\)


For each age group, people whose BMI is at the 75th percentile have a much higher probability of high blood pressure than those at the 25 th percentile. What is that difference in probability and its CI? To contrast across the percentiles of BMI within age groups, we again specify a contrast operator on the at () groups using atcontrast (), and we also tell margins to perform that contrast within the levels of the marginlist by using the marginswithin option.
```

. margins agegrp, at((p25) bmi) at((p75) bmi)
> contrast(atcontrast(r._at) marginswithin)
(output omitted)
. marginsplot
Variables that uniquely identify margins: agegrp _atopt
Multiple at() options specified:
_atoption=1: (p25) bmi
_atoption=2: (p75) bmi

```

Contrasts of predictive margins of agegrp with \(95 \%\) Cls


The differences in probability between 25th and 75th BMI percentiles are clearly significantly greater than 0 . The differences appear to be smallest for those in the youngest and oldest age groups.

\section*{Controlling the graph's dimensions}

Thus far, marginsplot has miraculously done almost exactly what we want in most cases. The things we want on the \(x\) axis have been there, the choice of plots has made sense, etc. Some of that luck sprang from the relatively simple analyses we were performing, and some was from careful specification of our margins command. Sometimes, we will not be so lucky.

Consider the following regress, margins, and marginsplot commands:
```

. regress bpsystol agegrp\#\#sex\#\#c.bmi
(output omitted)
. margins agegrp, over(sex) at(bmi=(10(10)60))
(output omitted)
. marginsplot
Variables that uniquely identify margins: agegrp _atopt
Multiple at() options specified:
_atoption=1: (p25) bmi
_atoption=2: (p75) bmi

```


By default, marginsplot places the levels of the first multilevel at() specification on the \(x\) axis and then usually plots the levels of all remaining variables as connected lines. That is what we see in the graph above-bmi, the at () variable, is on the \(x\) axis, and each combination of agegrp and sex is plotted as a separate connected line. If there is no multilevel at () specification, then the first variable in marginlist becomes the \(x\) axis. There are many more rules, but it is usually best to simply type marginsplot and see what happens. If you do not like marginsplot's choices, change them.

What if we wanted agegrp on the \(x\) axis instead of BMI? We tell marginsplot to make that change by specifying agegrp in the xdimension() option:
```

. marginsplot, xdimension(agegrp)
Variables that uniquely identify margins: bmi agegrp sex

```


We have been suppressing the Results window output for marginsplot, but that output is helpful if we want to change how things are plotted. You may specify any variable used in your margins command in any of the dimension options-xdimension(), plotdimension(), bydimension(), and graphdimension(). (In fact, there are some pseudovariables that you may also specify in some cases; see Addendum: Advanced uses of dimlist for details.) marginsplot tries to help you narrow your choices by listing a set of variables that uniquely identify all your margins. You are not restricted to this list.

We have a different \(x\) axis and a different set of plots, but our graph is still busy and difficult to read. We can make it better by creating separate graph panels for each sex. We do that by adding a bydimension() option with sex as the argument.
```

. marginsplot, xdimension(agegrp) bydimension(sex)
Variables that uniquely identify margins: bmi agegrp sex

```

Predictive margins of agegrp with \(95 \%\) Cls


The patterns and the differences between males and females are now easier to see.
If our interest is in comparing males and females, we might even choose to create a separate panel for each level of BMI:
```

. marginsplot, xdimension(agegrp) bydimension(bmi) xlabel(, angle(45))

```
Variables that uniquely identify margins: bmi agegrp sex

Predictive margins of agegrp with \(95 \%\) Cls


The \(x\)-axis labels did not fit, so we angled them.
We leave you to explore the use of the graphdimension() option. It is much like bydimension() but creates separate graphs rather than separate panels. Operationally, the plotdimension() option is rarely used. All variables not in the \(x\) dimension and not specified elsewhere become the plotted connected lines.

You will likely use the dimension options frequently. This is one of the rare cases where we recommend using the minimal abbreviations of the options- x() for xdimension(), plot() for plotdimension(), by() for bydimension(), and graph() for graphdimension(). The abbreviations are easy to read and just as meaningful as the full option names. The full names exist to reinforce the relationship between the dimension options.

\section*{Pairwise comparisons}
marginsplot can graph the results of margins, pwcompare; see [R] margins, pwcompare. We return to one of our ANOVA examples. Here we request pairwise comparisons with the pwcompare option of margins, and we request Bonferroni-adjusted CIs with the mcompare() option:
```

. anova bpsystol agegrp\#\#sex
(output omitted)
. margins agegrp, pwcompare mcompare(bonferroni)
(output omitted)
. marginsplot
Variables that uniquely identify margins: _pw1 _pw0
i_Pw enumerates all pairwise comparisons; _pw0 enumerates the reference
categories; _pw1 enumerates the comparison categories.

```


Each connected line plot in the graph represents a reference age-group category for the pairwise comparison. The ticks on the \(x\) axis represent comparison age-group categories. So, each plot is a profile for a reference category showing its comparison to each other category.

\section*{Horizontal is sometimes better}

Another interesting way to graph pairwise comparisons is to simply plot each comparison and label the two categories being compared. This type of graph works better if it is oriented horizontally rather than vertically.

Continuing with the example above, we will switch the graph to horizontal. We will also make several changes to display the graph better. We specify that only unique comparisons be plotted. The graph above plotted both 1 versus 2 and 2 versus 1 , which are the same comparison with opposite signs. We add a reference line at 0 because we are interested in comparisons that differ from 0 . This graph looks better without the connecting lines, so we add the option recast (scatter). We also reverse the \(y\) scale so that the smallest levels of age group appear at the top of the axis.
```

. marginsplot, horizontal unique xline(0) recast(scatter) yscale(reverse)
Variables that uniquely identify margins: _pw1 _pw0
i_pw enumerates all pairwise comparisons; _pw0 enumerates the reference
categories; _pw1 enumerates the comparison categories.

```

Pairwise comparisons of predictive margins of agegrp with \(95 \% \mathrm{Cls}\)


All the comparisons differ from 0 , so all our age groups are statistically different from each other.
The horizontal option can be useful outside of pairwise comparisons. Profile plots are usually oriented vertically. However, when your covariates have long labels or there are many levels at which the margins are being evaluated, the graph may be easier to read when rendered horizontally.

\section*{Marginal effects}

We have seen how to graph discrete effects for factor variables and continuous variables by using contrasts, and optionally by using the dydx() option of margins: Contrasts of margins-effects (discrete marginal effects) and Continuous covariates. Let's now consider graphing instantaneous marginal effects for continuous covariates. Begin by refitting our logistic model of high blood pressure as a function of sex, age, and BMI:

We estimate the average marginal effect of BMI on the probability of high blood pressure for each age group and then graph the results by typing
```

. margins agegrp, dydx(bmi)
(output omitted)
. marginsplot
Variables that uniquely identify margins: agegrp

```

Average marginal effects of bmi with \(95 \%\) Cls


These are the conditional expectations of the marginal effects treating everyone in the sample as though they were in each age group. We can estimate fully conditional marginal effects that do not depend on averaging over the sample by also margining on our one remaining covariate-sex.
```

. margins agegrp\#sex, dydx(bmi)
(output omitted)
. marginsplot
Variables that uniquely identify margins: agegrp sex

```

Average marginal effects of bmi with \(95 \%\) Cls


The effect of BMI on the probability of high blood pressure looks to increase with age for females. The marginal effect is higher for males than females in the younger age groups but then decreases with age for males after the 40-49 age group.

You may want to test for differences in the marginal effect of BMI for males and females by contrasting across sexes within agegrp:
```

. margins r.sex@agegrp, dydx(bmi)

```

\section*{Plotting a subset of the results from margins}
marginsplot plots all the margins produced by the preceding margins command. If you want a graph that does not include all the margins, then enter a margins command that produces a reduced set of margins. Obvious ways to reduce the number of margins include not specifying some factors or interactions in the marginlist of margins, not specifying some at () or over () options, or reducing the values specified in an at () option. A less obvious technique uses selection lists in factor operators to select specific sets of levels from factor variables specified in the marginlist.

\section*{Instead of typing}
. margins agegrp
which will give you margins for all six age groups in our sample, type
. margins i(2/4).agegrp
which will give you only three margins-those for groups 2, 3, and 4. See [U] 11.4.3.4 Selecting levels.

\section*{Advanced usage}
margins is incredibly flexible in the statistics it can estimate and in the grouping of those estimates. Many of the estimates that margins can produce do not make convincing graphs. marginsplot plots the results of any margins command, regardless of whether the resulting graph is easily interpreted. Here we demonstrate some options that can make complicated margins into graphs that are somewhat more useful than those produced by marginsplot's defaults. Others may find truly useful applications for these approaches.

\section*{Plots with multiple terms}

Margins plots are rarely interesting when you specify multiple terms on your margins command, for example, margins a b. Such plots often compare things that are not comparable. The defaults for marginsplot rarely produce useful plots with multiple terms. Perhaps the most interesting graph in such cases puts all the levels of all the terms together on the vertical axis and plots their margins on the horizontal axis. We do that by including the marginlist from margins in an xdimension() option on marginsplot. The long labels on such graphs look better with a horizontal orientation, and there is no need to connect the margin estimates, so we specify the recast (scatter) option.

Using one of our ANOVA examples from earlier,
```

. anova bpsystol agegrp\#\#sex
(output omitted)

- margins agegrp sex
(output omitted)
. marginsplot, xdimension(agegrp sex) horizontal recast(scatter)
Variables that uniquely identify margins: agegrp sex

```


The "asobserved" notations in the \(y\)-axis labels are informing us that, for example, when the margin for females is evaluated, the values of age group are taken as they are observed in the dataset. The margin is computed as an average over those values.

\section*{Plots with multiple at() options}

Some disciplines like to compute margins at the means of other covariates in their model and others like to compute the response for each observation and then take the means of the response. These correspond to the margins options at ((mean) _all) and at ((asobserved) _all). For responses that are linear functions of the coefficients, such as predict after regress, the two computations yield identical results. For responses that are nonlinear functions of the coefficients, the two computations estimate different things.

Using one of our logistic models of high blood pressure,
```

. logistic highbp sex\#\#agegrp\#\#c.bmi

```
and computing both sets of margins for each age group,
```

. margins agegrp, at((mean) _all) at((asobserved) _all)

```
we can use marginsplot to compare the approaches:
```

. marginsplot
Variables that uniquely identify margins: agegrp _atopt
Multiple at() options specified:
_atoption=1: (mean)_all
_atoption=2: (asobserved) _all

```

Predictive margins of agegrp with \(95 \% \mathrm{Cls}\)


For the first three age groups, the probabilities of high blood pressure are lower at the means of sex and bpi than are the mean probabilities of high blood pressure averaged over the observed values of sex and bpi. The reverse is true for the last three age groups, although the values are very similar in these older age groups.

Such comparisons come up even more frequently when evaluating marginal effects. We can estimate the marginal effects of sex at each age group and graph the results by adding dydx (sex) to our margins command:
```

. margins agegrp, at((mean) _all) at((asobserved) _all) dydx(sex)
(output omitted)
. marginsplot
Variables that uniquely identify margins: agegrp _atopt
Multiple at() options specified:
_atoption=1: (mean)_all
_atoption=2: (asobserved) _all

```

Average marginal effects of 2 .sex with \(95 \%\) Cls


The average marginal effect is smaller for most age groups, but the CIs for both sets of estimates are wide. Can we tell the difference between the estimates? To answer that, we use the now-familiar tactic of taking the contrast of our estimated marginal-effects profiles. That means adding contrast (atjoint marginswithin) to our margins command. We will also add mcompare (bonferroni) to account for the fact that we will be comparing six contrasts.
```

. margins agegrp, at((mean) _all) at((asobserved) _all) dydx(sex)
> contrast(atjoint marginswithin) mcompare(bonferroni)

```

We will also add the familiar reference line at 0 to our graph of the contrasts.
```

. marginsplot, yline(0)
Variables that uniquely identify margins: agegrp _atopt
Multiple at() options specified:
_atoption=1: (mean)_all
_atoption=2: (asobserved) _all

```

Contrasts of average marginal effects of 2.sex with \(95 \%\) Cls


While the difference in the estimates of marginal effects is not large, we can distinguish the estimates for the 30-39 and 70+ age groups.

The at() option of margins provides far more flexibility than demonstrated above. It can be used to evaluate a response or marginal effect at almost any point of interest or combinations of such points. See Syntax of at() in [R] margins.

\section*{Adding scatterplots of the data}

We can add scatterplots of the observed data to our plots of the margins. The NHANES II dataset is too large for this to be interesting, so for this example, we will use auto.dta. We fit mileage on whether the care is foreign and on a quadratic in the weight of the car. We convert the weight into tons (US definition) to improve the scaling, and we format the new tons variable to improve its labels on the graph. For our graph, we create separate variables for mileage of domestic and of foreign cars. We fit a fully interacted model so that the effect of weight on mileage can be different for foreign and for domestic cars.
```

. use https://www.stata-press.com/data/r18/auto
. generate tons = weight/2000
. format tons %6.2f
. separate mpg, by(foreign)
. regress mpg foreign\#\#c.tons\#\#c.tons

```

We then estimate the margins over the range of tons, using the option over(foreign) to obtain separate estimates for foreign and domestic cars.
```

. margins, at(tons=(.8(.05)2.4)) over(foreign)

```

Adding scatterplots of mileage for domestic and foreign cars is easy. We insert into an addplot () option of marginsplot the same scatterplot syntax for twoway that we would type to produce a scatterplot of the data:
. marginsplot, addplot(scatter mpg0 tons \| scatter mpg1 tons) recast(line) noci
Variables that uniquely identify margins: tons foreign


Many will be surprised that the mileage profile is higher in 1978 for domestic (US built) cars. Is the difference significant?
```

. margins, at(tons=(.8(.05)2.4)) over(r.for)
(output omitted)
. marginsplot, yline(0)
Variables that uniquely identify margins: tons

```

Contrasts of predictive margins with \(95 \% \mathrm{Cls}\)


As we did earlier, we contrast the two profiles. We can discern some difference between the two profiles for midweight vehicles, but otherwise there is insufficient information to believe mileage differs across domestic and foreign cars.

\section*{Video examples}

Profile plots and interaction plots, part 1: A single categorical variable
Profile plots and interaction plots, part 2: A single continuous variable
Profile plots and interaction plots, part 3: Interactions between categorical variables
Profile plots and interaction plots, part 4: Interactions of continuous and categorical variables
Profile plots and interaction plots, part 5: Interactions of two continuous variables

\section*{Addendum: Advanced uses of dimlist}
dimlist specifies the dimensions from the immediately preceding margins command that are to be used for the marginsplot's \(x\) axis, plots, subgraphs, and graphs. dimlist may contain:
\begin{tabular}{|c|c|}
\hline dim & Description \\
\hline varname & Any variable referenced in the preceding margins command. \\
\hline _equation & If the estimation command being analyzed is multivariate and margins automatically produced estimates for more than one dependent-variable equation, then dimlist may contain _equation to enumerate those equations. \\
\hline _outcome & If the estimation command being analyzed is ordinal and margins automatically produced estimates for more than one outcome level, then dimlist may contain _outcome to enumerate those outcomes. \\
\hline _predict & If the preceding margins command included multiple predict() options, then dimlist may contain _predict to enumerate those predict() options. \\
\hline at (varname) & If a variable is specified in both the marginlist or the over() option and in the at () option of margins, then the two uses can be distinguished in marginsplot by typing the at () variables as at (varname) in dimlist. \\
\hline _deriv & If the preceding margins command included a dydx(), eyex(), dyex(), or eydx() option, dimlist may also contain _deriv to specify all the variables over which derivatives were taken. \\
\hline _term & If the preceding margins command included multiple terms (for example, margins a b), then dimlist may contain _term to enumerate those terms. \\
\hline _atopt & If the preceding margins command included multiple at () options, then dimist may contain _atopt to enumerate those at () options. \\
\hline
\end{tabular}

When the pairwise option is specified on margins, you may specify dimensions that enumerate the pairwise comparisons.
_pw enumerates all the pairwise comparisons
_pw0 enumerates the reference categories of the comparisons
_pw1 enumerates the comparison categories of the comparisons

\section*{Acknowledgments}

We thank Philip B. Ender (retired) of UCLA Academic Technology Services for his programs that demonstrated what could be done in this area. We also thank Michael N. Mitchell, author of the Stata Press books Data Management Using Stata: A Practical Handbook, Interpreting and Visualizing Regression Models Using Stata, Stata for the Behavioral Sciences, and A Visual Guide to Stata Graphics, for his generous advice and comprehensive insight into the application of margins and their plots.

\section*{References}

Baldwin, S. 2019. Psychological Statistics and Psychometrics Using Stata. College Station, TX: Stata Press.
Bruun, N. H. 2019. Visualizing effect modification on contrasts. Stata Journal 19: 566-580.
Jann, B. 2014. Plotting regression coefficients and other estimates. Stata Journal 14: 708-737.
Lindsey, C. 2016. Estimating covariate effects after gmm. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2016/10/04/estimating-covariate-effects-after-gmm/.

MacDonald, K. 2018. Exploring results of nonparametric regression models. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2018/06/18/exploring-results-of-nonparametric-regression-models/.
McDowell, A., A. Engel, J. T. Massey, and K. Maurer. 1981. Plan and operation of the Second National Health and Nutrition Examination Survey, 1976-1980. Vital and Health Statistics 1(15): 1-144.

Mitchell, M. N. 2015. Stata for the Behavioral Sciences. College Station, TX: Stata Press.
——. 2021. Interpreting and Visualizing Regression Models Using Stata. 2nd ed. College Station, TX: Stata Press.
Pinzon, E. 2016. Effects of nonlinear models with interactions of discrete and continuous variables: Estimating, graphing, and interpreting. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2016/07/12/effects-for-nonlinear-models-with-interactions-of-discrete-and-continuous-variables-estimating-graphing-and-interpreting/.
Royston, P. 2013. marginscontplot: Plotting the marginal effects of continuous predictors. Stata Journal 13: 510-527.
Williams, R. 2012. Using the margins command to estimate and interpret adjusted predictions and marginal effects. Stata Journal 12: 308-331.

\section*{Also see}
[R] margins - Marginal means, predictive margins, and marginal effects
[R] margins, contrast - Contrasts of margins
[R] margins, pwcompare - Pairwise comparisons of margins
[R] margins postestimation - Postestimation tools for margins

\section*{Title}

Maximize - Details of iterative maximization
\begin{tabular}{lll} 
Description & Syntax & Maximization options \\
Remarks and examples & Stored results & Methods and formulas \\
References & Also see &
\end{tabular}

\section*{Description}

All Stata commands maximize likelihood functions using moptimize() and optimize(); see Methods and formulas below. Commands use the Newton-Raphson method with step halving and special fixups when they encounter nonconcave regions of the likelihood. For details, see [M-5] moptimize( ) and [M-5] optimize( ). For more information about programming maximum likelihood estimators in ado-files and Mata, see [R] ml and Pitblado, Poi, and Gould (2024).

\section*{Syntax}
\[
\text { mle_cmd } \ldots[\text {, options }]
\]
options
difficult
technique (algorithm_spec)
iterate(\#)
[no]log
trace
gradient
showstep
hessian
showtolerance
tolerance(\#)
ltolerance(\#)
nrtolerance(\#)
qtolerance(\#)
nonrtolerance
from(init_specs)

Description
use a different stepping algorithm in nonconcave regions maximization technique
perform maximum of \# iterations; default is iterate (300) display an iteration \(\log\) of the log likelihood; typically, the default display current parameter vector in iteration log display current gradient vector in iteration log report steps within an iteration in iteration \(\log\) display current negative Hessian matrix in iteration log report the calculated result that is compared to the effective convergence criterion
tolerance for the coefficient vector; see Options for the defaults tolerance for the log likelihood; see Options for the defaults tolerance for the scaled gradient; see Options for the defaults when specified with algorithms bhhh, dfp, or bfgs, the \(\mathbf{q}-\mathbf{H}\) matrix is used as the final check for convergence rather than nrtolerance() and the \(\mathbf{H}\) matrix; seldom used
ignore the nrtolerance() option
initial values for the coefficients
algorithm_spec is

> algorithm [ \# [ algorithm [\#] ]...]
algorithm is \(\{\mathrm{nr} \mid\) bhhh \(|\mathrm{dfp}|\) bfgs \(\}\)
init_specs is one of
\[
\begin{aligned}
& \text { matname }[, \text { skip copy }] \\
& \{[\text { eqname }:] \text { name }=\# \mid / \text { eqname }=\#\}[\ldots] \\
& \#[\# \ldots], \text { copy }
\end{aligned}
\]

\section*{Maximization options}
difficult specifies that the likelihood function is likely to be difficult to maximize because of nonconcave regions. When the message "not concave" appears repeatedly, ml's standard stepping algorithm may not be working well. difficult specifies that a different stepping algorithm be used in nonconcave regions. There is no guarantee that difficult will work better than the default; sometimes it is better and sometimes it is worse. You should use the difficult option only when the default stepper declares convergence and the last iteration is "not concave" or when the default stepper is repeatedly issuing "not concave" messages and producing only tiny improvements in the log likelihood.
technique (algorithm_spec) specifies how the likelihood function is to be maximized. The following algorithms are allowed. For details, see Pitblado, Poi, and Gould (2024).
technique ( nr ) specifies Stata's modified Newton-Raphson (NR) algorithm.
technique (bhhh) specifies the Berndt-Hall-Hall-Hausman (BHHH) algorithm.
technique (dfp) specifies the Davidon-Fletcher-Powell (DFP) algorithm.
technique (bfgs) specifies the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm.
The default is technique (nr).
You can switch between algorithms by specifying more than one in the technique() option. By default, an algorithm is used for five iterations before switching to the next algorithm. To specify a different number of iterations, include the number after the technique in the option. For example, specifying technique (bhhh 10 nr 1000 ) requests that ml perform 10 iterations with the BHHH algorithm followed by 1000 iterations with the NR algorithm, and then switch back to BHHH for 10 iterations, and so on. The process continues until convergence or until the maximum number of iterations is reached.
iterate(\#) specifies the maximum number of iterations. When the number of iterations equals iterate(), the optimizer stops and presents the current results. If convergence is declared before this threshold is reached, it will stop when convergence is declared. Specifying iterate (0) is useful for viewing results evaluated at the initial value of the coefficient vector. Specifying iterate ( 0 ) and from() together allows you to view results evaluated at a specified coefficient vector; however, not all commands allow the from() option. The default value of iterate(\#) for both estimators programmed internally and estimators programmed with ml is the number set using set maxiter, which is 300 by default.
log and nolog specify whether an iteration log showing the progress of the log likelihood is to be displayed. For most commands, the log is displayed by default, and nolog suppresses it; see set iterlog in [R] set iter. For a few commands (such as the svy maximum likelihood estimators), you must specify log to see the log.
trace adds to the iteration \(\log\) a display of the current parameter vector.
gradient adds to the iteration \(\log\) a display of the current gradient vector.
showstep adds to the iteration log a report on the steps within an iteration. This option was added so that developers at StataCorp could view the stepping when they were improving the ml optimizer code. At this point, it mainly provides entertainment.
hessian adds to the iteration \(\log\) a display of the current negative Hessian matrix.
showtolerance adds to the iteration \(\log\) the calculated value that is compared with the effective convergence criterion at the end of each iteration. Until convergence is achieved, the smallest calculated value is reported.
shownrtolerance is a synonym of showtolerance.

Below, we describe the three convergence tolerances. Convergence is declared when the nrtolerance() criterion is met and either the tolerance() or the ltolerance() criterion is also met.
tolerance(\#) specifies the tolerance for the coefficient vector. When the relative change in the coefficient vector from one iteration to the next is less than or equal to tolerance(), the tolerance() convergence criterion is satisfied.
tolerance (1e-4) is the default for estimators programmed with ml .
tolerance ( \(1 \mathrm{e}-6\) ) is the default.
ltolerance(\#) specifies the tolerance for the log likelihood. When the relative change in the log likelihood from one iteration to the next is less than or equal to ltolerance(), the ltolerance() convergence is satisfied.
ltolerance(0) is the default for estimators programmed with ml .
ltolerance (1e-7) is the default.
nrtolerance(\#) specifies the tolerance for the scaled gradient. Convergence is declared when \(\mathbf{g H}^{-1} \mathbf{g}^{\prime}<\) nrtolerance (). The default is nrtolerance (1e-5).
qtolerance(\#) when specified with algorithms bhhh, dfp, or bfgs uses the \(\mathbf{q}-\mathbf{H}\) matrix as the final check for convergence rather than nrtolerance() and the \(\mathbf{H}\) matrix.

Beginning with Stata 12, by default, Stata now computes the \(\mathbf{H}\) matrix when the \(\mathbf{q}-\mathbf{H}\) matrix passes the convergence tolerance, and Stata requires that \(\mathbf{H}\) be concave and pass the nrtolerance() criterion before concluding convergence has occurred.
qtolerance() provides a way for the user to obtain Stata's earlier behavior.
nonrtolerance specifies that the default nrtolerance() criterion be turned off.
from() specifies initial values for the coefficients. Not all estimators in Stata support this option. You can specify the initial values in one of three ways: by specifying the name of a vector containing the initial values (for example, from(b0), where b0 is a properly labeled vector); by specifying coefficient names with the values (for example, from (age=2.1/sigma=7.4)); or by specifying a list of values (for example, from (2.17.4, copy)). from() is intended for use when doing
bootstraps (see [R] bootstrap) and in other special situations (for example, with iterate(0)). Even when the values specified in from() are close to the values that maximize the likelihood, only a few iterations may be saved. Poor values in from() may lead to convergence problems.
skip specifies that any parameters found in the specified initialization vector that are not also found in the model be ignored. The default action is to issue an error message.
copy specifies that the list of values or the initialization vector be copied into the initial-value vector by position rather than by name.

\section*{Remarks and examples}

Only in rare circumstances would you ever need to specify any of these options, except nolog. The nolog option is useful for reducing the amount of output appearing in log files; also see set iterlog in [ R ] set iter.

The following is an example of an iteration log:
```

Iteration 0: Log likelihood = -3791.0251
Iteration 1: Log likelihood = -3761.738
Iteration 2: Log likelihood = -3758.0632 (not concave)
Iteration 3: Log likelihood = -3758.0447
Iteration 4: Log likelihood = -3757.5861
Iteration 5: Log likelihood = -3757.474
Iteration 6: Log likelihood = -3757.4613
Iteration 7: Log likelihood = -3757.4606
Iteration 8: Log likelihood = -3757.4606
(table of results omitted)

```

At iteration 8, the model converged. The message "not concave" at the second iteration is notable. This example was produced using the heckman command; its likelihood is not globally concave, so it is not surprising that this message sometimes appears. The other message that is occasionally seen is "backed up". Neither of these messages should be of any concern unless they appear at the final iteration.

If a "not concave" message appears at the last step, there are two possibilities. One is that the result is valid, but there is collinearity in the model that the command did not otherwise catch. Stata checks for obvious collinearity among the independent variables before performing the maximization, but strange collinearities or near collinearities can sometimes arise between coefficients and ancillary parameters. The second, more likely cause for a "not concave" message at the final step is that the optimizer entered a flat region of the likelihood and prematurely declared convergence.

If a "backed up" message appears at the last step, there are also two possibilities. One is that Stata found a perfect maximum and could not step to a better point; if this is the case, all is fine, but this is a highly unlikely occurrence. The second is that the optimizer worked itself into a bad concave spot where the computed gradient and Hessian gave a bad direction for stepping.

If either of these messages appears at the last step, perform the maximization again with the gradient option. If the gradient goes to zero, the optimizer has found a maximum that may not be unique but is a maximum. From the standpoint of maximum likelihood estimation, this is a valid result. If the gradient is not zero, it is not a valid result, and you should try tightening up the convergence criterion, or try ltol ( 0 ) tol ( \(1 \mathrm{e}-7\) ) to see if the optimizer can work its way out of the bad region.

If you get repeated "not concave" steps with little progress being made at each step, try specifying the difficult option. Sometimes difficult works wonderfully, reducing the number of iterations and producing convergence at a good (that is, concave) point. Other times, difficult works poorly, taking much longer to converge than the default stepper.

\section*{Stored results}

Maximum likelihood estimators store the following in e():
Scalars
e(N)
\(e(k)\)
\(e\left(k \_e q\right)\)
\(e\left(k \_e q-m o d e l\right)\)
\(e\left(k \_d v\right)\)
\(e\left(d f \_m\right)\)
\(e(r 2-p)\)
\(e(l l)\)
\(e\left(l l \_0\right)\)
\(e\left(N \_c l u s t\right)\)
\(e(c h i 2)\)
\(e(p)\)
\(e(r a n k)\)
\(e(r a n k 0)\)
\(e(i c)\)
\(e(r c)\)
\(e(c o n v e r g e d)\)
number of observations
number of parameters
number of equations in \(e(b)\)
number of equations in overall model test
number of dependent variables
model degrees of freedom
pseudo- \(R^{2}\)
log likelihood
log likelihood, constant-only model
number of clusters
\(\chi^{2}\)
\(p\)-value for model test
rank of \(e(V)\)
always stored always stored usually stored usually stored
usually stored
always stored
sometimes stored
always stored
stored when constant-only model is fit
stored when vce(cluster clustvar) is specified; see [U] 20.22 Obtaining robust variance estimates
usually stored
usually stored
always stored
rank of \(e(V)\) for constant-only model stored when constant-only model is fit
number of iterations usually stored
return code usually stored
1 if converged, 0 otherwise usually stored
Macros
```

e(cmd)
e(cmdline)
e(depvar)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(chi2type)
e(vce)
e(vcetype)
e(opt)
e(which)

```
e(ml_method)
e(user)
\(e\) (technique)
e(ml_method)
e(user)
\(e(t e c h n i q u e)\)
\(e(s i n g u l a r H m e t h o d) ~ m-m a r q u a r d t ~ o r ~ h y b r i d ; ~ m e t h o d ~\)
e(crittype)
e(properties)
e(predict)
name of command
command as typed
names of dependent variables
weight type
weight expression
title in estimation output name of cluster variable

Wald or LR; type of model \(\chi^{2}\) test vcetype specified in vce() title used to label Std. err. type of optimization
\(\max\) or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program used when Hessian is singular
name of command
command as typed
names of dependent variables
weight type
weight expression
from technique() option
optimization criterion
estimator properties
program used to implement predict usually stored
always stored
always stored
always stored
stored when weights are specified or implied
stored when weights are specified or implied
usually stored by commands using ml
stored when vce(cluster clustvar) is specified;
see [U] 20.22 Obtaining robust variance estimates
usually stored
stored when command allows vce()
sometimes stored
always stored
always stored
always stored by commands using ml
always stored
sometimes stored
sometimes stored \({ }^{1}\)
always stored \({ }^{1}\)
always stored
\begin{tabular}{|c|c|c|}
\hline Macros, continued e(deriv_useminbound) & off or on; whether minimum bounds are used for step sizes in numerical derivative computations & always stored by ml model \\
\hline \multicolumn{3}{|l|}{Matrices} \\
\hline e(b) & coefficient vector & always stored \\
\hline e(Cns) & constraints matrix & sometimes stored \\
\hline e(ilog) & iteration \(\log\) (up to 20 iterations) & usually stored \\
\hline e(gradient) & gradient vector & usually stored \\
\hline e (V) & variance-covariance matrix of the estimators & always stored \\
\hline e (V_modelbased) & model-based variance & only stored when \(e(V)\) is robust, cluster-robust, bootstrap, or jackknife variance \\
\hline e(deriv_minbound) & minimum values for step sizes & only stored when e(deriv_useminbound) is on \\
\hline \multicolumn{3}{|l|}{Functions} \\
\hline e(sample) & marks estimation sample & always stored \\
\hline
\end{tabular}
1. Type ereturn list, all to view these results; see \([P]\) return.

See Stored results in the manual entry for any maximum likelihood estimator for a list of returned results.

\section*{Methods and formulas}

Optimization is currently performed by moptimize() and optimize(), with the former implemented in terms of the latter; see [M-5] moptimize() and [M-5] optimize(). Some estimators use moptimize() and optimize() directly, and others use the ml ado-file interface to moptimize().

Prior to Stata 11, Stata had three separate optimization engines: an internal one used by estimation commands implemented in C code; ml implemented in ado-code separately from moptimize() and used by most estimators; and moptimize() and optimize() used by a few recently written estimators. These days, the internal optimizer and the old version of ml are used only under version control. In addition, arch and arima (see [TS] arch and [TS] arima) are currently implemented using the old ml .

Let \(L_{1}\) be the log likelihood of the full model (that is, the log-likelihood value shown on the output), and let \(L_{0}\) be the log likelihood of the "constant-only" model. The likelihood-ratio \(\chi^{2}\) model test is defined as \(2\left(L_{1}-L_{0}\right)\). The pseudo- \(R^{2}\) (McFadden 1974) is defined as \(1-L_{1} / L_{0}\). This is simply the log likelihood on a scale where 0 corresponds to the "constant-only" model and 1 corresponds to perfect prediction for a discrete model (in which case the overall log likelihood is 0 ).

Some maximum likelihood routines can report coefficients in an exponentiated form, for example, odds ratios in logistic. Let \(b\) be the unexponentiated coefficient, \(s\) its standard error, and \(b_{0}\) and \(b_{1}\) the reported confidence interval for \(b\). In exponentiated form, the point estimate is \(e^{b}\), the standard error \(e^{b} s\), and the confidence interval \(e^{b_{0}}\) and \(e^{b_{1}}\). The displayed \(Z\) (or \(t\) ) statistics and \(p\)-values are the same as those for the unexponentiated results. This is justified because \(e^{b}=1\) and \(b=0\) are equivalent hypotheses, and normality is more likely to hold in the \(b\) metric.

\section*{References}

McFadden, D. L. 1974. Conditional logit analysis of qualitative choice behavior. In Frontiers in Econometrics, ed. P. Zarembka, 105-142. New York: Academic Press.

Pitblado, J. S., B. P. Poi, and W. W. Gould. 2024. Maximum Likelihood Estimation with Stata. 5th ed. College Station, TX: Stata Press.

\section*{Also see}
[R] ml - Maximum likelihood estimation
[R] set iter - Control iteration settings
[SVY] ml for svy - Maximum pseudolikelihood estimation for survey data
[M-5] moptimize( ) - Model optimization
[M-5] optimize( ) - Function optimization

\section*{Title}
mean - Estimate means
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
mean produces estimates of means, along with standard errors.

\section*{Quick start}

Mean, standard error, and \(95 \%\) confidence interval for v1
mean v1
Also compute statistics for v2
mean v1 v2
Same as above, but for each level of categorical variable catvar1
mean v 1 v 2 , over(catvar1)
Weighting by probability weight wvar
mean v1 v2 [pweight=wvar]
Population mean using svyset data
svy: mean v3
Subpopulation means for each level of categorical variable catvar2 using svyset data svy: mean v3, over(catvar2)

Test equality of two subpopulation means
svy: mean v3, over(catvar2)
test v3@1.catvar2 = v3@2.catvar2

\section*{Menu}

Statistics \(>\) Summaries, tables, and tests \(>\) Summary and descriptive statistics \(>\) Means

\section*{Syntax}
```

mean varlist [if][in][weight][,options]

```

\section*{options}

Description

Model
stdize (varname)
stdweight (varname) nostdrescale
if/in/over over (varlisto \({ }_{o}\) )

SE/Cluster vce(vcetype)

\section*{Reporting}
level(\#)
noheader
display_options
coeflegend
variable identifying strata for standardization
weight variable for standardization
do not rescale the standard weight variable
group over subpopulations defined by varlisto
vcetype may be analytic, cluster clustvar, bootstrap, or jackknife
set confidence level; default is level (95)
suppress table header
control column formats, line width, display of omitted variables and base and empty cells, and factor-variable labeling
display legend instead of statistics
varlist may contain factor variables; see [U] 11.4.3 Factor variables.
bootstrap, collect, jackknife, mi estimate, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands.
vce (bootstrap) and vce(jackknife) are not allowed with the mi estimate prefix; see [MI] mi estimate.
Weights are not allowed with the bootstrap prefix; see \([R]\) bootstrap.
aweights are not allowed with the jackknife prefix; see \([R]\) jackknife.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, aweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
coeflegend does not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options}
\(\qquad\) Model
stdize (varname) specifies that the point estimates be adjusted by direct standardization across the strata identified by varname. This option requires the stdweight() option.
stdweight (varname) specifies the weight variable associated with the standard strata identified in the stdize() option. The standardization weights must be constant within the standard strata. nostdrescale prevents the standardization weights from being rescaled within the over() groups. This option requires stdize() but is ignored if the over() option is not specified.
over (varlist \(t_{o}\) ) specifies that estimates be computed for multiple subpopulations, which are identified by the different values of the variables in varlist \({ }_{o}\). Only numeric, nonnegative, integer-valued variables are allowed in over \(\left(\right.\) varlist \(\left._{o}\right)\).
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (analytic), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
vce(analytic), the default, uses the analytically derived variance estimator associated with the sample mean.
\(\qquad\) Reporting
level (\#); see [R] Estimation options.
noheader prevents the table header from being displayed.
display_options: noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), and nolstretch; see [R] Estimation options.

The following option is available with mean but is not shown in the dialog box:
coeflegend; see [R] Estimation options.

\section*{Remarks and examples}

\section*{> Example 1}

Using the fuel data from example 3 of \([\mathrm{R}]\) ttest, we estimate the average mileage of the cars without the fuel treatment (mpg1) and those with the fuel treatment (mpg2).
```

. use https://www.stata-press.com/data/r18/fuel

```
. mean mpg1 mpg2
Mean estimation \(\quad\) Number of obs \(=12\)
\begin{tabular}{r|rrrr}
\hline & Mean & Std. err. & [95\% conf. interval] \\
\hline mpg1 & 21 & .7881701 & 19.26525 & 22.73475 \\
mpg2 & 22.75 & .9384465 & 20.68449 & 24.81551 \\
\hline
\end{tabular}

Using these results, we can test the equality of the mileage between the two groups of cars.
```

. test mpg1 = mpg2
( 1) mpg1 - mpg2 = 0
F( 1, 11) = 5.04
Prob > F = 0.0463

```

\section*{> Example 2}

In example 1, the joint observations of mpg1 and mpg2 were used to estimate a covariance between their means.
```

. matrix list e(V)
symmetric e(V)[2,2]
mpg1 mpg2
mpg1 . }6212121
mpg2 . }4469697\mathrm{ . }8806818

```

If the data were organized this way out of convenience but the two variables represent independent samples of cars (coincidentally of the same sample size), we should reshape the data and use the over() option to ensure that the covariance between the means is zero.
```

. use https://www.stata-press.com/data/r18/fuel
. stack mpg1 mpg2, into(mpg) clear
. rename _stack trt
. label define trt_lab 1 "without" 2 "with"
. label values trt trt_lab
. label var trt "Fuel treatment"
. mean mpg, over(trt)
Mean estimation Number of obs = 24

|  | Mean | Std. err. | [95\% conf. interval] |  |
| :---: | ---: | :---: | :---: | :---: |
| c.mpg@trt |  |  |  |  |
| without | 21 | .7881701 | 19.36955 | 22.63045 |
| with | 22.75 | .9384465 | 20.80868 | 24.69132 |

. matrix list e(V)
symmetric e(V)[2,2]
c.mpg@ c.mpg@
1.trt 2.trt
c.mpg@1.trt . }6212121
c.mpg@2.trt 0 . 88068182

```

Now, we can test the equality of the mileage between the two independent groups of cars.
```

. test mpg@1.trt = mpg@2.trt
( 1) c.mpg@1bn.trt - c.mpg@2.trt = 0
F( 1, 23) = 2.04
Prob > F = 0.1667

```

\section*{> Example 3: standardized means}

Suppose that we collected the blood pressure data from example 2 of \([R]\) dstdize, and we wish to obtain standardized high blood pressure rates for each city in 1990 and 1992, using, as the standard, the age, sex, and race distribution of the four cities and two years combined. Our rate is really the mean of a variable that indicates whether a sampled individual has high blood pressure. First, we generate the strata and weight variables from our standard distribution, and then use mean to compute the rates.
```

. use https://www.stata-press.com/data/r18/hbp, clear
. egen strata = group(age race sex) if inlist(year, 1990, 1992)
(675 missing values generated)
. by strata, sort: gen stdw = _N
. mean hbp, over(city year) stdize(strata) stdweight(stdw)
Mean estimation
N. of std strata = 24 Number of obs = 455

```
\begin{tabular}{r|rrrr}
\hline & Mean & Std. err. & [95\% conf. interval] \\
\hline c.hbp@city\#year & & & & \\
11990 & .058642 & .0296273 & .0004182 & .1168657 \\
11992 & .0117647 & .0113187 & -.0104789 & .0340083 \\
21990 & .0488722 & .0238958 & .0019121 & .0958322 \\
21992 & .014574 & .007342 & .0001455 & .0290025 \\
31990 & .1011211 & .0268566 & .0483425 & .1538998 \\
31992 & .0810577 & .0227021 & .0364435 & .1256719 \\
51990 & .0277778 & .0155121 & -.0027066 & .0582622 \\
5 & 1992 & .0548926 & 0 &. \\
\hline
\end{tabular}

The standard error of the high blood pressure rate estimate is missing for city 5 in 1992 because there was only one individual with high blood pressure; that individual was the only person observed in the stratum of white males 30-35 years old.

By default, mean rescales the standard weights within the over() groups. In the following, we use the nostdrescale option to prevent this, thus reproducing the results in \([R]\) dstdize.
```

. mean hbp, over(city year) stdize(strata) stdweight(stdw) nostdrescale
Mean estimation
N. of std strata = 24 Number of obs = 455

```
\begin{tabular}{r|rrrr}
\hline & Mean & Std. err. & [95\% conf. interval] \\
\hline c.hbp@city\#year & & & & \\
11990 & .0073302 & .0037034 & .0000523 & .0146082 \\
11992 & .0015432 & .0014847 & -.0013745 & .004461 \\
21990 & .0078814 & .0038536 & .0003084 & .0154544 \\
21992 & .0025077 & .0012633 & .000025 & .0049904 \\
31990 & .0155271 & .0041238 & .007423 & .0236312 \\
31992 & .0081308 & .0022772 & .0036556 & .012606 \\
51990 & .0039223 & .0021904 & -.0003822 & .0082268 \\
5 & 1992 & .0088735 & 0 &. \\
\hline
\end{tabular}

\section*{> Example 4: profile plots and contrasts}

The first example in \([\mathrm{R}]\) marginsplot shows how to use margins and marginsplot to get profile plots from a linear regression. We can similarly explore the data using marginsplot after mean with the over () option. Here we use marginsplot to plot the means of systolic blood pressure for each age group.
. use https://www.stata-press.com/data/r18/nhanes2, clear
. mean bpsystol, over (agegrp)
Mean estimation \(\quad\) Number of obs \(=10,351\)
\begin{tabular}{r|rrrr}
\hline & Mean & Std. err. & [95\% conf. interval] \\
\hline c.bpsystol@agegrp & & & & \\
\(20-29\) & 117.3466 & .3247329 & 116.71 & 117.9831 \\
\(30-39\) & 120.2374 & .4095845 & 119.4345 & 121.0402 \\
\(40-49\) & 126.9442 & .532033 & 125.9013 & 127.9871 \\
\(50-59\) & 135.6754 & .6061842 & 134.4872 & 136.8637 \\
\(60-69\) & 141.5227 & .4433527 & 140.6537 & 142.3918 \\
\(70+\) & 148.1765 & .8321116 & 146.5454 & 149.8076 \\
\hline
\end{tabular}
```

. marginsplot
Variables that uniquely identify means: agegrp

```


We see that the mean systolic blood pressure increases with age. We can use contrast to formally test whether each mean is different from the mean in the previous age group using the ar. contrast operator; see \([R]\) contrast for more information on this command.
. contrast ar.agegrp\#c.bpsystol, effects nowald
Contrasts of means
\begin{tabular}{c|cccccc}
\hline & Contrast & Std. err. & t & \(\mathrm{P}>\mid \mathrm{tI}\) & [95\% conf. interval] \\
\hline \begin{tabular}{c} 
agegrp\# \\
c.bpsystol \\
(30-39 \\
vs
\end{tabular} & & & & & & \\
\begin{tabular}{c}
\(20-29)\) \\
\((40-49\) \\
vs
\end{tabular} & 2.89081 & .5226958 & 5.53 & 0.000 & 1.866225 & 3.915394 \\
\begin{tabular}{c}
\(30-39)\) \\
\((50-59\) \\
vs
\end{tabular} & 6.706821 & .6714302 & 9.99 & 0.000 & 5.390688 & 8.022954 \\
\begin{tabular}{c}
\(40-49)\) \\
\((60-69\) \\
vs
\end{tabular} & 8.731263 & .8065472 & 10.83 & 0.000 & 7.150275 & 10.31225 \\
\begin{tabular}{c}
\(50-59)\) \\
\((70+\) \\
vs
\end{tabular} & 5.847282 & .7510133 & 7.79 & 0.000 & 4.375151 & 7.319413 \\
\begin{tabular}{c}
\(60-69)\)
\end{tabular} & 6.653743 & .9428528 & 7.06 & 0.000 & 4.80557 & 8.501917 \\
\hline
\end{tabular}

The first row of the output reports that the mean systolic blood pressure for the 30-39 age group is 2.89 higher than the mean for the \(20-29\) age group. The mean for the \(40-49\) age group is 6.71 higher than the mean for the \(30-39\) age group, and so on. Each of these differences is significantly different from zero.

We can include both agegrp and sex in the over() option to estimate means separately for men and women in each age group.
. mean bpsystol, over (agegrp sex)
Mean estimation \(\quad\) Number of obs \(=10,351\)
\begin{tabular}{r|rrrr}
\hline & Mean & Std. err. & [95\% conf. interval] \\
\hline c.bpsystol@agegrp\#sex & & & & \\
\(20-29 \# M a l e\) & 123.8862 & .4528516 & 122.9985 & 124.7739 \\
\(20-29 \#\) Female & 111.2849 & .3898972 & 110.5206 & 112.0492 \\
30-39\#Male & 124.6818 & .5619855 & 123.5802 & 125.7834 \\
30-39\#Female & 116.2207 & .5572103 & 115.1284 & 117.3129 \\
\(40-49 \# M a l e\) & 129.0033 & .7080788 & 127.6153 & 130.3912 \\
\(40-49 \#\) Female & 125.0468 & .7802558 & 123.5174 & 126.5763 \\
\(50-59 \# M a l e\) & 136.0864 & .855435 & 134.4096 & 137.7632 \\
\(50-59 \#\) Female & 135.3164 & .8556015 & 133.6393 & 136.9935 \\
60-69\#Male & 140.7451 & .6059786 & 139.5572 & 141.9329 \\
60-69\#Female & 142.2368 & .6427981 & 140.9767 & 143.4968 \\
\(70+\# M a l e\) & 146.3951 & 1.141126 & 144.1583 & 148.6319 \\
\(70+\#\) Female & 149.6599 & 1.189975 & 147.3273 & 151.9924 \\
\hline
\end{tabular}
. marginsplot
Variables that uniquely identify means: agegrp sex
Estimated means of bpsystol with \(95 \%\) Cls


Are the means different for men and women within each age group? We can again perform the tests using contrast. This time, we will use r.sex to obtain contrasts comparing men and women and use @agegrp to request that the tests are performed for each age group.
. contrast r.sex\#c.bpsystol@agegrp, effects nowald
Contrasts of means
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline & Contrast & Std. err. & t & \(P>|t|\) & [95\% conf & interval] \\
\hline \multicolumn{7}{|l|}{\begin{tabular}{l}
sex@agegrp\# \\
c.bpsystol \\
(Female
\end{tabular}} \\
\hline \[
\begin{array}{r}
\text { vs } \\
\text { Male) }
\end{array}
\] & & & & & & \\
\hline \[
\begin{array}{r}
20-29 \\
\text { (Female }
\end{array}
\] & -12.60132 & . 5975738 & -21.09 & 0.000 & -13.77268 & -11.42996 \\
\hline \multicolumn{7}{|l|}{} \\
\hline 30-39 & -8.461161 & . 7913981 & -10.69 & 0.000 & -10.01245 & -6.909868 \\
\hline \multicolumn{7}{|l|}{(Female} \\
\hline \multicolumn{7}{|l|}{vs} \\
\hline 40-49 & -3.956451 & 1.053648 & -3.76 & 0.000 & -6.021805 & -1.891097 \\
\hline \multicolumn{7}{|l|}{(Female \(\quad\) ( \({ }^{\text {a }}\)} \\
\hline \multicolumn{7}{|l|}{vs} \\
\hline \multicolumn{7}{|l|}{Male)} \\
\hline 50-59 & -. 7699782 & 1.209886 & -0.64 & 0.525 & -3.141588 & 1.601631 \\
\hline \multicolumn{7}{|l|}{(Female 1.601631} \\
\hline \multicolumn{7}{|l|}{vs} \\
\hline Male) & & & & & & \\
\hline 60-69 & 1.491684 & . 8834022 & 1.69 & 0.091 & -. 2399545 & 3.223323 \\
\hline \multicolumn{7}{|l|}{(Female} \\
\hline \multicolumn{7}{|l|}{vs} \\
\hline \multicolumn{7}{|l|}{Male)} \\
\hline 70+ & 3.264762 & 1.648699 & 1.98 & 0.048 & . 0329927 & 6.496531 \\
\hline
\end{tabular}

Using a 0.05 significance level, we find that the mean systolic blood pressure is different for men and women in all age groups except the fifties and sixties.

\section*{Video example}

Descriptive statistics in Stata

\section*{Stored results}
mean stores the following in e() :
Scalars
\(\mathrm{e}(\mathrm{N}) \quad\) number of observations
e(N_over) number of subpopulations
e(N_stdize) number of standard strata
e(N_clust) number of clusters
e(k_eq)
e(df_r)
number of equations in \(e(b)\)
sample degrees of freedom
e(rank)
rank of e(V)
Macros
e(cmd)
e(cmdline)
e(varlist)
e(stdize)
e(stdweight)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(over)
e(vce)
e(vcetype)
e(properties)
mean
command as typed
varlist
varname from stdize()
varname from stdweight()
weight type
weight expression
title in estimation output
name of cluster variable
varlist from over()
vcetype specified in vce()
title used to label Std. err.
e(estat_cmd)
b V
\(e\) (marginsnotok)
program used to implement estat
predictions disallowed by margins
Matrices
e(b)
e(V)
vector of mean estimates
(co)variance estimates
vector of standard deviation estimates
vector of numbers of nonmissing observations
number of nonmissing observations within the standard strata
standardizing proportions
error code corresponding to \(e(b)\)
Functions
```

    e(sample) marks estimation sample
    ```

In addition to the above, the following is stored in r() :

\section*{Matrices}
\(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r -class command is run after the estimation command.

\section*{Methods and formulas}

Methods and formulas are presented under the following headings:
```

The mean estimator
Survey data
The survey mean estimator
The standardized mean estimator
The poststratified mean estimator
The standardized poststratified mean estimator
Subpopulation estimation

```

\section*{The mean estimator}

Let \(y\) be the variable on which we want to calculate the mean and \(y_{j}\) an individual observation on \(y\), where \(j=1, \ldots, n\) and \(n\) is the sample size. Let \(w_{j}\) be the weight, and if no weight is specified, define \(w_{j}=1\) for all \(j\). For aweights, the \(w_{j}\) are normalized to sum to \(n\). See The survey mean estimator for pweighted data.

Let \(W\) be the sum of the weights
\[
W=\sum_{j=1}^{n} w_{j}
\]

The mean is defined as
\[
\bar{y}=\frac{1}{W} \sum_{j=1}^{n} w_{j} y_{j}
\]

The default variance estimator for the mean is
\[
\widehat{V}(\bar{y})=\frac{1}{W(W-1)} \sum_{j=1}^{n} w_{j}\left(y_{j}-\bar{y}\right)^{2}
\]

The standard error of the mean is the square root of the variance.
If \(x, x_{j}\), and \(\bar{x}\) are similarly defined for another variable (observed jointly with \(y\) ), the covariance estimator between \(\bar{x}\) and \(\bar{y}\) is
\[
\widehat{\operatorname{Cov}}(\bar{x}, \bar{y})=\frac{1}{W(W-1)} \sum_{j=1}^{n} w_{j}\left(x_{j}-\bar{x}\right)\left(y_{j}-\bar{y}\right)
\]

\section*{Survey data}

See [SVY] Variance estimation, [SVY] Direct standardization, and [SVY] Poststratification for discussions that provide background information for the following formulas. The following formulas are derived from the fact that the mean is a special case of the ratio estimator where the denominator variable is one, \(x_{j}=1\); see [ R\(]\) ratio.

\section*{The survey mean estimator}

Let \(Y_{j}\) be a survey item for the \(j\) th individual in the population, where \(j=1, \ldots, M\) and \(M\) is the size of the population. The associated population mean for the item of interest is \(\bar{Y}=Y / M\) where
\[
Y=\sum_{j=1}^{M} Y_{j}
\]

Let \(y_{j}\) be the survey item for the \(j\) th sampled individual from the population, where \(j=1, \ldots, m\) and \(m\) is the number of observations in the sample.

The estimator for the mean is \(\bar{y}=\widehat{Y} / \widehat{M}\), where
\[
\widehat{Y}=\sum_{j=1}^{m} w_{j} y_{j} \quad \text { and } \quad \widehat{M}=\sum_{j=1}^{m} w_{j}
\]
and \(w_{j}\) is a sampling weight. The score variable for the mean estimator is
\[
z_{j}(\bar{y})=\frac{y_{j}-\bar{y}}{\widehat{M}}=\frac{\widehat{M} y_{j}-\widehat{Y}}{\widehat{M^{2}}}
\]

\section*{The standardized mean estimator}

Let \(D_{g}\) denote the set of sampled observations that belong to the \(g\) th standard stratum and define \(I_{D_{g}}(j)\) to indicate if the \(j\) th observation is a member of the \(g\) th standard stratum; where \(g=1, \ldots\), \(L_{D}\) and \(L_{D}\) is the number of standard strata. Also, let \(\pi_{g}\) denote the fraction of the population that belongs to the \(g\) th standard stratum, thus \(\pi_{1}+\cdots+\pi_{L_{D}}=1 . \pi_{g}\) is derived from the stdweight () option.

The estimator for the standardized mean is
\[
\bar{y}^{D}=\sum_{g=1}^{L_{D}} \pi_{g} \frac{\widehat{Y}_{g}}{\widehat{M}_{g}}
\]
where
\[
\widehat{Y}_{g}=\sum_{j=1}^{m} I_{D_{g}}(j) w_{j} y_{j} \quad \text { and } \quad \widehat{M}_{g}=\sum_{j=1}^{m} I_{D_{g}}(j) w_{j}
\]

The score variable for the standardized mean is
\[
z_{j}\left(\bar{y}^{D}\right)=\sum_{g=1}^{L_{D}} \pi_{g} I_{D_{g}}(j) \frac{\widehat{M}_{g} y_{j}-\widehat{Y}_{g}}{\widehat{M}_{g}^{2}}
\]

\section*{The poststratified mean estimator}

Let \(P_{k}\) denote the set of sampled observations that belong to poststratum \(k\) and define \(I_{P_{k}}(j)\) to indicate if the \(j\) th observation is a member of poststratum \(k\); where \(k=1, \ldots, L_{P}\) and \(L_{P}\) is the number of poststrata. Also let \(M_{k}\) denote the population size for poststratum \(k . P_{k}\) and \(M_{k}\) are identified by specifying the poststrata() and postweight () options on svyset; see [SVY] svyset.

The estimator for the poststratified mean is
\[
\bar{y}^{P}=\frac{\widehat{Y}^{P}}{\widehat{M}^{P}}=\frac{\widehat{Y}^{P}}{M}
\]
where
\[
\widehat{Y}^{P}=\sum_{k=1}^{L_{P}} \frac{M_{k}}{\widehat{M}_{k}} \widehat{Y}_{k}=\sum_{k=1}^{L_{P}} \frac{M_{k}}{\widehat{M}_{k}} \sum_{j=1}^{m} I_{P_{k}}(j) w_{j} y_{j}
\]
and
\[
\widehat{M}^{P}=\sum_{k=1}^{L_{P}} \frac{M_{k}}{\widehat{M}_{k}} \widehat{M}_{k}=\sum_{k=1}^{L_{P}} M_{k}=M
\]

The score variable for the poststratified mean is
\[
z_{j}\left(\bar{y}^{P}\right)=\frac{z_{j}\left(\widehat{Y}^{P}\right)}{M}=\frac{1}{M} \sum_{k=1}^{L_{P}} I_{P_{k}}(j) \frac{M_{k}}{\widehat{M}_{k}}\left(y_{j}-\frac{\widehat{Y}_{k}}{\widehat{M}_{k}}\right)
\]

\section*{The standardized poststratified mean estimator}

The estimator for the standardized poststratified mean is
\[
\bar{y}^{D P}=\sum_{g=1}^{L_{D}} \pi_{g} \frac{\widehat{Y}_{g}^{P}}{\widehat{M}_{g}^{P}}
\]
where
\[
\widehat{Y}_{g}^{P}=\sum_{k=1}^{L_{p}} \frac{M_{k}}{\widehat{M}_{k}} \widehat{Y}_{g, k}=\sum_{k=1}^{L_{p}} \frac{M_{k}}{\widehat{M}_{k}} \sum_{j=1}^{m} I_{D_{g}}(j) I_{P_{k}}(j) w_{j} y_{j}
\]
and
\[
\widehat{M}_{g}^{P}=\sum_{k=1}^{L_{p}} \frac{M_{k}}{\widehat{M}_{k}} \widehat{M}_{g, k}=\sum_{k=1}^{L_{p}} \frac{M_{k}}{\widehat{M}_{k}} \sum_{j=1}^{m} I_{D_{g}}(j) I_{P_{k}}(j) w_{j}
\]

The score variable for the standardized poststratified mean is
\[
z_{j}\left(\bar{y}^{D P}\right)=\sum_{g=1}^{L_{D}} \pi_{g} \frac{\widehat{M}_{g}^{P} z_{j}\left(\widehat{Y}_{g}^{P}\right)-\widehat{Y}_{g}^{P} z_{j}\left(\widehat{M}_{g}^{P}\right)}{\left(\widehat{M}_{g}^{P}\right)^{2}}
\]
where
\[
z_{j}\left(\widehat{Y}_{g}^{P}\right)=\sum_{k=1}^{L_{P}} I_{P_{k}}(j) \frac{M_{k}}{\widehat{M}_{k}}\left\{I_{D_{g}}(j) y_{j}-\frac{\widehat{Y}_{g, k}}{\widehat{M}_{k}}\right\}
\]
and
\[
z_{j}\left(\widehat{M}_{g}^{P}\right)=\sum_{k=1}^{L_{P}} I_{P_{k}}(j) \frac{M_{k}}{\widehat{M}_{k}}\left\{I_{D_{g}}(j)-\frac{\widehat{M}_{g, k}}{\widehat{M}_{k}}\right\}
\]

\section*{Subpopulation estimation}

Let \(S\) denote the set of sampled observations that belong to the subpopulation of interest, and define \(I_{S}(j)\) to indicate if the \(j\) th observation falls within the subpopulation.

The estimator for the subpopulation mean is \(\bar{y}^{S}=\widehat{Y}^{S} / \widehat{M}^{S}\), where
\[
\widehat{Y}^{S}=\sum_{j=1}^{m} I_{S}(j) w_{j} y_{j} \quad \text { and } \quad \widehat{M}^{S}=\sum_{j=1}^{m} I_{S}(j) w_{j}
\]

Its score variable is
\[
z_{j}\left(\bar{y}^{S}\right)=I_{S}(j) \frac{y_{j}-\bar{y}^{S}}{\widehat{M}^{S}}=I_{S}(j) \frac{\widehat{M}^{S} y_{j}-\widehat{Y}^{S}}{\left(\widehat{M}^{S}\right)^{2}}
\]

The estimator for the standardized subpopulation mean is
\[
\bar{y}^{D S}=\sum_{g=1}^{L_{D}} \pi_{g} \frac{\widehat{Y}_{g}^{S}}{\widehat{M}_{g}^{S}}
\]
where
\[
\widehat{Y}_{g}^{S}=\sum_{j=1}^{m} I_{D_{g}}(j) I_{S}(j) w_{j} y_{j} \quad \text { and } \quad \widehat{M}_{g}^{S}=\sum_{j=1}^{m} I_{D_{g}}(j) I_{S}(j) w_{j}
\]

Its score variable is
\[
z_{j}\left(\bar{y}^{D S}\right)=\sum_{g=1}^{L_{D}} \pi_{g} I_{D_{g}}(j) I_{S}(j) \frac{\widehat{M}_{g}^{S} y_{j}-\widehat{Y}_{g}^{S}}{\left(\widehat{M}_{g}^{S}\right)^{2}}
\]

The estimator for the poststratified subpopulation mean is
\[
\bar{y}^{P S}=\frac{\widehat{Y}^{P S}}{\widehat{M}^{P S}}
\]
where
\[
\widehat{Y}^{P S}=\sum_{k=1}^{L_{P}} \frac{M_{k}}{\widehat{M}_{k}} \widehat{Y}_{k}^{S}=\sum_{k=1}^{L_{P}} \frac{M_{k}}{\widehat{M}_{k}} \sum_{j=1}^{m} I_{P_{k}}(j) I_{S}(j) w_{j} y_{j}
\]
and
\[
\widehat{M}^{P S}=\sum_{k=1}^{L_{P}} \frac{M_{k}}{\widehat{M}_{k}} \widehat{M}_{k}^{S}=\sum_{k=1}^{L_{P}} \frac{M_{k}}{\widehat{M}_{k}} \sum_{j=1}^{m} I_{P_{k}}(j) I_{S}(j) w_{j}
\]

Its score variable is
\[
z_{j}\left(\bar{y}^{P S}\right)=\frac{\widehat{M}^{P S} z_{j}\left(\widehat{Y}^{P S}\right)-\widehat{Y}^{P S} z_{j}\left(\widehat{M}^{P S}\right)}{\left(\widehat{M}^{P S}\right)^{2}}
\]
where
\[
z_{j}\left(\widehat{Y}^{P S}\right)=\sum_{k=1}^{L_{P}} I_{P_{k}}(j) \frac{M_{k}}{\widehat{M}_{k}}\left\{I_{S}(j) y_{j}-\frac{\widehat{Y}_{k}^{S}}{\widehat{M}_{k}}\right\}
\]
and
\[
z_{j}\left(\widehat{M}^{P S}\right)=\sum_{k=1}^{L_{P}} I_{P_{k}}(j) \frac{M_{k}}{\widehat{M}_{k}}\left\{I_{S}(j)-\frac{\widehat{M}_{k}^{S}}{\widehat{M}_{k}}\right\}
\]

The estimator for the standardized poststratified subpopulation mean is
\[
\bar{y}^{D P S}=\sum_{g=1}^{L_{D}} \pi_{g} \frac{\widehat{Y}_{g}^{P S}}{\widehat{M}_{g}^{P S}}
\]
where
\[
\widehat{Y}_{g}^{P S}=\sum_{k=1}^{L_{p}} \frac{M_{k}}{\widehat{M}_{k}} \widehat{Y}_{g, k}^{S}=\sum_{k=1}^{L_{p}} \frac{M_{k}}{\widehat{M}_{k}} \sum_{j=1}^{m} I_{D_{g}}(j) I_{P_{k}}(j) I_{S}(j) w_{j} y_{j}
\]
and
\[
\widehat{M}_{g}^{P S}=\sum_{k=1}^{L_{p}} \frac{M_{k}}{\widehat{M}_{k}} \widehat{M}_{g, k}^{S}=\sum_{k=1}^{L_{p}} \frac{M_{k}}{\widehat{M}_{k}} \sum_{j=1}^{m} I_{D_{g}}(j) I_{P_{k}}(j) I_{S}(j) w_{j}
\]

Its score variable is
\[
z_{j}\left(\bar{y}^{D P S}\right)=\sum_{g=1}^{L_{D}} \pi_{g} \frac{\widehat{M}_{g}^{P S} z_{j}\left(\widehat{Y}_{g}^{P S}\right)-\widehat{Y}_{g}^{P S} z_{j}\left(\widehat{M}_{g}^{P S}\right)}{\left(\widehat{M}_{g}^{P S}\right)^{2}}
\]
where
\[
z_{j}\left(\widehat{Y}_{g}^{P S}\right)=\sum_{k=1}^{L_{P}} I_{P_{k}}(j) \frac{M_{k}}{\widehat{M}_{k}}\left\{I_{D_{g}}(j) I_{S}(j) y_{j}-\frac{\widehat{Y}_{g, k}^{S}}{\widehat{M}_{k}}\right\}
\]
and
\[
z_{j}\left(\widehat{M}_{g}^{P S}\right)=\sum_{k=1}^{L_{P}} I_{P_{k}}(j) \frac{M_{k}}{\widehat{M}_{k}}\left\{I_{D_{g}}(j) I_{S}(j)-\frac{\widehat{M}_{g, k}^{S}}{\widehat{M}_{k}}\right\}
\]

\section*{References}

Bakker, A. 2003. The early history of average values and implications for education. Journal of Statistics Education 11(1). https://www.amstat.org/publications/jse/v11n1/bakker.html.
Cochran, W. G. 1977. Sampling Techniques. 3rd ed. New York: Wiley.
Manski, C. F., and M. Tabord-Meehan. 2017. Evaluating the maximum MSE of mean estimators with missing data. Stata Journal 17: 723-735.

Stuart, A., and J. K. Ord. 1994. Kendall's Advanced Theory of Statistics: Distribution Theory, Vol. 1. 6th ed. London: Arnold.

\section*{Also see}
[R] mean postestimation - Postestimation tools for mean
[R] ameans - Arithmetic, geometric, and harmonic means
[R] proportion - Estimate proportions
\([\mathrm{R}]\) ratio - Estimate ratios
[R] summarize - Summary statistics
[R] total - Estimate totals
[MI] Estimation - Estimation commands for use with mi estimate
[SVY] Direct standardization - Direct standardization of means, proportions, and ratios
[SVY] Poststratification - Poststratification for survey data
[SVY] Subpopulation estimation - Subpopulation estimation for survey data
[SVY] svy estimation - Estimation commands for survey data
[SVY] Variance estimation - Variance estimation for survey data
[U] 20 Estimation and postestimation commands

\section*{Title}
mean postestimation - Postestimation tools for mean

Postestimation commands estat sd Also see

\section*{Postestimation commands}

The following postestimation commands are of special interest after mean:
\begin{tabular}{ll} 
Command & Description \\
\hline estat sd & standard deviation estimates \\
\hline
\end{tabular}

The following standard postestimation commands are also available:
\begin{tabular}{ll} 
Command & Description \\
\hline contrast & contrasts and ANOVA-style joint tests of estimates \\
estat vce & \begin{tabular}{l} 
variance-covariance matrix of the estimators (VCE)
\end{tabular} \\
estat (svy) & \begin{tabular}{l} 
postestimation statistics for survey data \\
cataloging estimation results
\end{tabular} \\
estimates & table of estimation results \\
etable & \begin{tabular}{l} 
Hausman's specification test \\
hausman \\
lincom
\end{tabular} \\
marginsplot estimates, standard errors, testing, and inference for linear combinations of \\
nlcom & \begin{tabular}{l} 
graph the results from mean \\
point estimates, standard errors, testing, and inference for nonlinear combinations
\end{tabular} \\
pwcompare & \begin{tabular}{l} 
of coefficients
\end{tabular} \\
pairwise comparisons of estimates \\
test & \begin{tabular}{l} 
Wald tests of simple and composite linear hypotheses
\end{tabular} \\
testnl & Wald tests of nonlinear hypotheses
\end{tabular}

\section*{estat sd}

\section*{Description for estat sd}
estat sd reports standard deviations based on the estimation results from mean. estat sd is not appropriate with estimation results that used direct standardization.
estat sd can also report subpopulation standard deviations based on estimation results from svy: mean; see [SVY] estat.

\section*{Menu for estat sd}

Statistics \(>\) Postestimation

\section*{Syntax for estat sd}
estat sd [, variance]

\section*{Option for estat sd}
variance requests that the variance be displayed instead of the standard deviation.

\section*{Stored results for estat sd}
estat sd stores the following in \(r()\) :
Matrices
\begin{tabular}{ll}
\(r(\) mean \()\) & vector of mean estimates \\
\(r(s d)\) & vector of standard deviation estimates \\
\(r(\) variance \()\) & vector of variance estimates
\end{tabular}

\section*{Also see}
[R] mean - Estimate means
[SVY] estat - Postestimation statistics for survey data
[U] 20 Estimation and postestimation commands

\section*{Title}
mfp - Multivariable fractional polynomial models
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Acknowledgments \\
References & Also see & &
\end{tabular}

\section*{Description}
mfp selects the multivariable fractional polynomial (MFP) model that best predicts the outcome variable from the right-hand-side variables in xvarlist.

For univariate fractional polynomials, \(f p\) can be used to fit a wider range of models than \(\mathrm{mf} p\). See [R] fp for more details.

\section*{Quick start}

Find optimal MFP model for regression of y on fractional polynomials of \(\mathrm{x} 1, \mathrm{x} 2\), and x 3 mfp: regress y x1 x2 x3

Same as above, but search only powers of \(-1,-0.5,1\), and 2
mfp , xpowers (-1 -. 5 1 2) : regress y x1 x2 x3
Allow a maximum of 2 degrees of freedom for all covariates
mfp, dfdefault(2): regress y x1 x2 x3
Allow 4 degrees of freedom for x 1 and 2 degrees of freedom for x 2 and x 3
mfp, dfdefault(2) df(x1:4): regress y x1 x2 x3
Same as above
mfp, \(d f(x 1: 4, x 2\) x3:2): regress \(y\) x1 \(x 2\) x3
Use a \(10 \%\) significance level when testing between fractional polynomials of different degrees
mfp, alpha(0.1): regress y x1 x2 x3
Perform backward selection using a nominal \(p\)-value of 0.05 for all variables
mf p, select(0.05): regress y x1 x2 x3
Same as above, but force x 3 into the model by setting its nominal \(p\)-value to 1
mfp, select(0.05, x3:1): regress y x1 x2 x3

Note: In the above examples, regress could be replaced with any estimation command allowing the mfp prefix.

\section*{Menu}

Statistics \(>\) Linear models and related \(>\) Fractional polynomials \(>\) Multivariable fractional polynomial models

\section*{Syntax}
\(\operatorname{mfp}[\), options \(]:\) regression_cmd \(\left[\right.\) yvar \(_{1}\left[\right.\) yvar \(\left.\left._{2}\right]\right]\) xvarlist [if] [in] [weight] [, regression_cmd_options]
regression_cmd may be clogit, glm, intreg, logistic, logit, mlogit, nbreg, ologit, oprobit, poisson, probit, qreg, regress, rreg, stcox, stcrreg, streg, or xtgee.
\(y v a r_{1}\) is not allowed for streg, stcrreg, and stcox. For these commands, you must first stset your data.
\(y v a r_{1}\) and \(y v a r_{2}\) must both be specified when regression_cmd is intreg.
xvarlist has elements of type varlist or (varlist) or both, for example, x1 x2 (x3 x4 x5). Elements enclosed in parentheses are tested jointly for inclusion in the model and are not eligible for fractional polynomial transformation.
options
Description
Model 2
sequential
cycles(\#)
dfdefault(\#)
center (cent_list)
alpha(alpha_list)
df (df_list)
powers(numlist)
Adv. model
xorder ( \(+|-| n\) )
select (select_list)
xpowers (xp_list)
zero (varlist)
catzero (varlist)
all
Reporting
level(\#)
display_options
use the Royston and Altman model-selection algorithm; default uses closed-test procedure
maximum number of iteration cycles; default is cycles (5) default maximum degrees of freedom; default is dfdefault (4) specification of centering for the independent variables
\(p\)-values for testing between FP models; default is alpha(0.05)
degrees of freedom for each predictor
list of FP powers to use; default is powers \((-2-1(.5) 123)\)
order of entry into model-selection algorithm; default is xorder ( + ) nominal \(p\)-values for selection on each predictor FP powers for each predictor
treat nonpositive values of specified predictors as zero when FP is transformed
add indicator variable for specified predictors
include out-of-sample observations in generated variables
set confidence level; default is level (95)
control column formats and line width
regression_cmd_options Description
Adv. model
regression_cmd_options
options appropriate to the regression command in use
collect is allowed; see [U] 11.1.10 Prefix commands.
All weight types supported by regression_cmd are allowed; see [U] 11.1.6 weight.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands. fp generate may be used to create new variables containing fractional polynomial powers. See \([\mathrm{R}] \mathbf{f p}\).

\section*{Options}

\section*{Model 2 \\ sequential chooses the sequential fractional polynomial (FP) selection algorithm (see Methods of FP model selection).}
cycles(\#) sets the maximum number of iteration cycles permitted. cycles(5) is the default.
dfdefault(\#) determines the default maximum degrees of freedom (df) for a predictor. The default is dfdefault (4) (second-degree FP).
center (cent_list) defines the centering of the covariates xvar \(_{1}\), xvar \(_{2}, \ldots\) of \(x v a r l i s t\). The default is center (mean), except for binary covariates, where it is center(\#), with \# being the lower of the two distinct values of the covariate. A typical item in cent_list is varlist: \{mean|\#|no\}. Items are separated by commas. The first item is special in that varlist is optional, and if it is omitted, the default is reset to the specified value (mean, \#, or no). For example, center (no, age:mean) sets the default to no (that is, no centering) and the centering of age to mean.
alpha(alpha_list) sets the significance levels for testing between FP models of different degrees. The rules for alpha_list are the same as those for \(d f \_l i s t\) in the df () option (see below). The default nominal \(p\)-value (significance level, selection level) is 0.05 for all variables.

Example: alpha(0.01) specifies that all variables have an FP selection level of \(1 \%\).
Example: alpha(0.05, weight:0.1) specifies that all variables except weight have an FP selection level of \(5 \%\); weight has a level of \(10 \%\).
\(\mathrm{df}\left(d f \_l i s t\right)\) sets the df for each predictor. The df (not counting the regression constant, _cons) is twice the degree of the FP, so, for example, an xvar fit as a second-degree FP (FP2) has 4 df. The first item in df_list may be either \# or varlist:\#. Subsequent items must be varlist:\#. Items are separated by commas, and varlist is specified in the usual way for variables. With the first type of item, the df for all predictors is taken to be \#. With the second type of item, all members of varlist (which must be a subset of xvarlist) have \# df.
The default number of degrees of freedom for a predictor of type varlist specified in xvarlist but not in \(d f\) _list is assigned according to the number of distinct (unique) values of the predictor, as follows:
\begin{tabular}{cl}
\hline \# of distinct values & Default df \\
\hline 1 & (invalid predictor) \\
\(2-3\) & 1 \\
\(4-5\) & min(2, dfdefault()) \\
\(\geq 6\) & dfdefault () \\
\hline
\end{tabular}

Example: df (4)
All variables have 4 df .
Example: df (2, weight displ:4)
weight and displ have 4 df ; all other variables have 2 df .
Example: df (weight displ:4, mpg:2)
weight and displ have 4 df , mpg has 2 df ; all other variables have default df.
powers (numlist) is the set of FP powers to be used. The default set is \(-2,-1,-0.5,0,0.5,1,2\), 3 (0 means log).
xorder \((+|-| n)\) determines the order of entry of the covariates into the model-selection algorithm. The default is xorder (+), which enters them in decreasing order of significance in a multiple linear regression (most significant first). xorder ( - ) places them in reverse significance order, whereas xorder ( \(n\) ) respects the original order in xvarlist.
select (select_list) sets the nominal p-values (significance levels) for variable selection by backward elimination. A variable is dropped if its removal causes a nonsignificant increase in deviance. The rules for select_list are the same as those for \(d f\) _list in the \(d f()\) option (see above). Using the default selection level of 1 for all variables forces them all into the model. Setting the nominal \(p\)-value to be 1 for a given variable forces it into the model, leaving others to be selected or not. The nominal \(p\)-value for elements of xvarlist bound by parentheses is specified by including (varlist) in select_list.

Example: select (0.05)
All variables have a nominal \(p\)-value of \(5 \%\).
Example: select(0.05, weight:1)
All variables except weight have a nominal \(p\)-value of \(5 \%\); weight is forced into the model.
Example: select (a (b c):0.05)
All variables except \(a, b\), and \(c\) are forced into the model. \(b\) and \(c\) are tested jointly with 2 df at the \(5 \%\) level, and a is tested singly at the \(5 \%\) level.
xpowers (xp_list) sets the permitted FP powers for covariates individually. The rules for \(x p \_l i s t\) are the same as for \(d f \_l i s t\) in the df() option. The default selection is the same as that for the powers() option.
Example: xpowers (-1 0 1)
All variables have powers \(-1,0,1\).
Example: xpowers (x5:-1 0 1)
All variables except x 5 have default powers; x 5 has powers \(-1,0,1\).
zero (varlist) treats negative and zero values of members of varlist as zero when FP transformations are applied. By default, such variables are subjected to a preliminary linear transformation to avoid negative and zero values, as described in the scale option of \([\mathrm{R}] \mathbf{f p}\). varlist must be part of xvarlist.
catzero(varlist) is a variation on zero(); see Zeros and zero categories below. varlist must be part of xvarlist.
regression_cmd_options may be any of the options appropriate to regression_cmd.
all includes out-of-sample observations when generating the FP variables. By default, the generated FP variables contain missing values outside the estimation sample.

Reporting
level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals. display_options: cformat (\%fmt), pformat (\%fmt), sformat(\%fmt), and nolstretch; see [R] Estimation options.

\section*{Remarks and examples}

Remarks are presented under the following headings:
Iteration report
Estimation algorithm
Methods of FP model selection
Zeros and zero categories
For elements in xvarlist not enclosed in parentheses, mfp leaves variables in the data named Ixvar__1, Ixvar__2, .., where xvar represents the first four letters of the name of \(x^{2} a_{1}\), and so on, for \(\mathrm{xvar}_{2}, \mathrm{xvar}_{3}\), etc. The new variables contain the best-fitting FP powers of \(\mathrm{xvar}_{1}, x \mathrm{xvar}_{2}, \ldots\).

\section*{Iteration report}

By default, for each continuous predictor, \(x\), mfp compares null, linear, and FP1 models for \(x\) with an FP2 model. The deviance for each of these nested submodels is given in the column labeled "Deviance". The line labeled "Final" gives the deviance for the selected model and its powers. All the other predictors currently selected are included, with their transformations (if any). For models specified as having 1 df , the only choice is whether the variable enters the model.

\section*{Estimation algorithm}

The estimation algorithm in mfp processes the xvars in turn. Initially, mfp silently arranges xvarlist in order of increasing \(p\)-value (that is, of decreasing statistical significance) for omitting each predictor from the model comprising xvarlist, with each term linear. The aim is to model relatively important variables before unimportant ones. This approach may help to reduce potential model-fitting difficulties caused by collinearity or, more generally, "concurvity" among the predictors. See the xorder() option above for details on how to change the ordering.

At the initial cycle, the best-fitting FP function for xvar \(_{1}\) (the first of \(x\) varlist) is determined, with all the other variables assumed to be linear. Either the default or the alternative procedure is used (see Methods of FP model selection below). The functional form (but not the estimated regression coefficients) for xvar \(_{1}\) is kept, and the process is repeated for \(x_{v a r_{2}, x^{\prime}}\) var \(_{3}\), etc. The first iteration concludes when all the variables have been processed in this way. The next cycle is similar, except that the functional forms from the initial cycle are retained for all variables except the one currently being processed.

A variable whose functional form is prespecified to be linear (that is, to have 1 df ) is tested for exclusion within the above procedure when its nominal \(p\)-value (selection level) according to select () is less than 1 ; otherwise, it is included.

Updating of FP functions and candidate variables continues until the functions and variables included in the overall model do not change (convergence). Convergence is usually achieved within 1-4 cycles.

\section*{Methods of FP model selection}
mfp includes two algorithms for FP model selection, both of which combine backward elimination with the selection of an FP function. For each continuous variable in turn, they start from a mostcomplex permitted FP model and attempt to simplify the model by reducing the degree. The default algorithm resembles a closed-test procedure, a sequence of tests maintaining the overall type I error rate at a prespecified nominal level, such as \(5 \%\). All significance tests are approximate; therefore, the algorithm is not precisely a closed-test procedure (Royston and Sauerbrei 2008, chap. 6).

The closed-test algorithm for choosing an FP model with maximum permitted degree \(m=2\) (that is, an FP2 model with 4 df ) for one continuous predictor, \(x\), is as follows:
1. Inclusion: Test FP2 against the null model for \(x\) on 4 df at the significance level determined by select (). If \(x\) is significant, continue; otherwise, drop \(x\) from the model.
2. Nonlinearity: Test FP2 against a straight line in \(x\) on 3 df at the significance level determined by alpha(). If significant, continue; otherwise, stop, with the chosen model for \(x\) being a straight line.
3. Simplification: Test FP2 against FP1 on 2 df at the significance level determined by alpha(). If significant, the final model is FP2; otherwise, it is FP1.
The first step is omitted if \(x\) is to be retained in the model, that is, if its nominal \(p\)-value, according to the select () option, is 1 .

An alternative algorithm is available with the sequential option, as originally suggested by Royston and Altman (1994):
1. Test FP2 against FP1 on 2 df at the alpha() significance level. If significant, the final model is FP2; otherwise, continue.
2. Test FP1 against a straight line on 1 df at the alpha() level. If significant, the final model is FP1; otherwise, continue.
3. Test a straight line against omitting \(x\) on 1 df at the select() level. If significant, the final model is a straight line; otherwise, drop \(x\).

The final step is omitted if \(x\) is to be retained in the model, that is, if its nominal \(p\)-value, according to the select () option, is 1 .

If \(x\) is uninfluential, the overall type I error rate of this procedure is about double that of the closed-test procedure, for which the rate is close to the nominal value. This inflated type I error rate confers increased apparent power to detect nonlinear relationships.

\section*{Zeros and zero categories}

The zero() option permits fitting an FP model to the positive values of a covariate, taking nonpositive values as zero. An application is the assessment of the effect of cigarette smoking as a risk factor in an epidemiological study. Nonsmokers may be qualitatively different from smokers, so the effect of smoking (regarded as a continuous variable) may not be continuous between one and zero cigarettes. To allow for this, the risk may be modeled as constant for the nonsmokers and as an FP function of the number of cigarettes for the smokers:
. generate byte nonsmokr \(=\) cond(n_cigs==0, 1, 0) if n_cigs != .
. mfp, zero(n_cigs) df(4, nonsmokr:1): logit case n_cigs nonsmokr age
Omission of zero(n_cigs) would cause n_cigs to be transformed before analysis by the addition of a suitable constant, probably 1.

A closely related approach involves the catzero() option. The command
```

. mfp, catzero(n_cigs): logit case n_cigs age

```
would achieve a similar result to the previous command but with important differences. First, mfp would create the equivalent of the binary variable nonsmokr automatically and include it in the model. Second, the two smoking variables would be treated as one predictor in the model. With the select () option active, the two variables would be tested jointly for inclusion in the model. A modified version is described in Royston and Sauerbrei (2008, sec. 4.15).

\section*{> Example 1}

We illustrate two of the analyses performed by Sauerbrei and Royston (1999). We use brcancer.dta, which contains prognostic factors data from the German Breast Cancer Study Group of patients with node-positive breast cancer. The response variable is recurrence-free survival time (rectime), and the censoring variable is censrec. There are 686 patients with 299 events. We use Cox regression to predict the log hazard of recurrence from prognostic factors of which five are continuous ( \(\mathrm{x} 1, \mathrm{x} 3, \mathrm{x} 5, \mathrm{x} 6, \mathrm{x} 7\) ) and three are binary ( \(\mathrm{x} 2, \mathrm{x} 4 \mathrm{a}, \mathrm{x} 4 \mathrm{~b}\) ). Hormonal therapy (hormon) is known to reduce recurrence rates and is forced into the model. We use mfp to build a model from the initial set of eight predictors by using the backfitting model-selection algorithm. We set the nominal \(p\)-value for variable and FP selection to 0.05 for all variables except hormon, which it is set to 1 :
```

. use https://www.stata-press.com/data/r18/brcancer
(German breast cancer data)
. stset rectime, fail(censrec)
(output omitted)

```
. mfp, alpha(.05) select(.05, hormon:1) : stcox x 1 x 2 x 3 x 4 a x 4 b x 5 x 6 x 7 hormon, > nohr

Deviance for model with all terms untransformed = 3471.637, 686 observations
\begin{tabular}{lllllllll} 
Variable & Model (vs.) & Deviance & Dev diff. & P & Powers & (vs.) \\
\hline x5 & null & FP2 & 3503.610 & 61.366 & \(0.000 *\) &. & .5 & 3 \\
& Lin. & & 3471.637 & 29.393 & \(0.000+\) & 1 & \\
& FP1 & & 3449.203 & 6.959 & \(0.031+\) & 0 & \\
x6 & Final & & 3442.244 & & & .5 & 3 & \\
& null & FP2 & 3464.113 & 29.917 & \(0.000 *\) &. & -2.5 \\
& Lin. & & 3442.244 & 8.048 & \(0.045+\) & 1 & \\
& FP1 & & 3435.550 & 1.354 & 0.508 & .5 & \\
& Final & & 3435.550 & & & .5 &
\end{tabular}
[hormon included with 1 df in model]
\begin{tabular}{lllllllll} 
x4a & null & Lin. & 3440.749 & 5.199 & \(0.023 *\) &. & 1 \\
& Final & & 3435.550 & & & 1 & \\
x3 & null & FP2 & 3436.832 & 3.560 & 0.469 &. & -23 \\
& Final & & 3436.832 & & &. & \\
x2 & null & Lin. & 3437.589 & 0.756 & 0.384 &. & 1 \\
& Final & & 3437.589 & & &. & \\
x4b & null & Lin. & 3437.848 & 0.259 & 0.611 &. & 1 \\
& Final & & 3437.848 & & &. & \\
x1 & null & FP2 & 3437.893 & 18.085 & \(0.001 *\) &. & \(-2-.5\) \\
& Lin. & & 3437.848 & 18.040 & \(0.000+\) & 1 & \\
& FP1 & & 3433.628 & 13.820 & \(0.001+\) & -2 & \\
& Final & & 3419.808 & & & \(-2-.5\) & \\
x7 & null & FP2 & 3420.805 & 3.715 & 0.446 &. & -.5 & 3
\end{tabular}
\begin{tabular}{llrrllll} 
End of Cycle 1: Deviance \(=\) & 3420.805 \\
x5 & null & FP2 & 3494.867 & 74.143 & \(0.000 *\) & & \\
& Lin. & & 3451.795 & 31.071 & \(0.000+\) & 1 & -2 \\
& FP1 & & 3428.023 & 7.299 & \(0.026+\) & 0 & \\
& Final & & 3420.724 & & & -2 & -1 \\
x6 & null & FP2 & 3452.093 & 32.704 & \(0.000 *\) &. & 0 \\
& Lin. & & 3427.703 & 8.313 & \(0.040+\) & 1 & 0 \\
& FP1 & 3420.724 & 1.334 & 0.513 & .5 & \\
& Final & 3420.724 & & & .5 &
\end{tabular}
[hormon included with 1 df in model]
\begin{tabular}{|c|c|c|c|c|c|c|c|}
\hline \multirow[t]{2}{*}{x4a} & null & Lin. & 3425.310 & 4.586 & 0.032* & & \multirow[t]{2}{*}{1} \\
\hline & Final & & 3420.724 & & & 1 & \\
\hline \multirow[t]{2}{*}{x3} & null & \multirow[t]{2}{*}{FP2} & 3420.724 & \multirow[t]{2}{*}{5.305} & \multirow[t]{2}{*}{0.257} & . & \multirow[t]{2}{*}{-. 50} \\
\hline & Final & & 3420.724 & & & . & \\
\hline \multirow[t]{2}{*}{x 2} & null & \multirow[t]{2}{*}{Lin.} & 3420.724 & \multirow[t]{2}{*}{0.214} & \multirow[t]{2}{*}{0.644} & . & \multirow[t]{2}{*}{1} \\
\hline & Final & & 3420.724 & & & . & \\
\hline \multirow[t]{2}{*}{x4b} & null & \multirow[t]{2}{*}{Lin.} & 3420.724 & \multirow[t]{2}{*}{0.145} & \multirow[t]{2}{*}{0.703} & . & \multirow[t]{2}{*}{1} \\
\hline & Final & & 3420.724 & & & - & \\
\hline \multirow[t]{4}{*}{x1} & null & \multirow[t]{4}{*}{FP2} & 3440.057 & 19.333 & 0.001* & . & \multirow[t]{4}{*}{-2-.5} \\
\hline & Lin. & & 3440.038 & 19.314 & 0.000+ & 1 & \\
\hline & FP1 & & 3436.949 & \multirow[t]{2}{*}{16.225} & \multirow[t]{2}{*}{0.000+} & -2 & \\
\hline & Final & & 3420.724 & & & -2 -. 5 & \\
\hline x7 & null & FP2 & 3420.724 & 2.152 & 0.708 & . & -13 \\
\hline & Final & & 3420.724 & & & & \\
\hline
\end{tabular}

Fractional polynomial fitting algorithm converged after 2 cycles.

Transformations of covariates:
-> gen double Ix1_-1 = \(X^{\wedge}-2-.0355294635\) if e(sample)
-> gen double Ix1__2 = \(X^{\wedge}-.5-.4341573547\) if e(sample)
(where: \(\mathrm{X}=\mathrm{x} 1 / 10\) )
-> gen double Ix5__1 = \(X^{\wedge}-2-3.983723313\) if e(sample)
-> gen double Ix5__2 = X^-1-1.99592668 if e(sample)
(where: \(X=x 5 / 10\) )
-> gen double Ix6_-1 = \(X^{\wedge} .5-.3331600619\) if e(sample)
(where: \(X=(x 6+1) / 1000)\)
Final multivariable fractional polynomial model for _t
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline \multirow[t]{2}{*}{Variable} & \multicolumn{3}{|l|}{-_Initial__} & \multicolumn{3}{|l|}{_-Final} \\
\hline & df & Select & Alpha & Status & df & Powers \\
\hline x 1 & 4 & 0.0500 & 0.0500 & in & 4 & -2-. 5 \\
\hline x2 & 1 & 0.0500 & 0.0500 & out & 0 & \\
\hline x3 & 4 & 0.0500 & 0.0500 & out & 0 & \\
\hline x 4 a & 1 & 0.0500 & 0.0500 & in & 1 & 1 \\
\hline x4b & 1 & 0.0500 & 0.0500 & out & 0 & \\
\hline x5 & 4 & 0.0500 & 0.0500 & in & 4 & -2 -1 \\
\hline x 6 & 4 & 0.0500 & 0.0500 & in & 2 & . 5 \\
\hline x7 & 4 & 0.0500 & 0.0500 & out & 0 & \\
\hline hormon & 1 & 1.0000 & 0.0500 & in & 1 & 1 \\
\hline
\end{tabular}

Cox regression -- Breslow method for ties
\begin{tabular}{llll} 
Entry time _t0 & Number of obs & \(=\) & 686 \\
& LR chi2 \((7)\) & \(=\) & 155.62 \\
& Prob > chi2 & \(=\) & 0.0000 \\
Log likelihood \(=-1710.3619\) & & Pseudo R2 & \(=\)
\end{tabular} 0.0435
\begin{tabular}{r|rlrlrr}
\hline _t & Coefficient & Std. err. & \multicolumn{1}{c}{z} & \(\mathrm{P}>|\mathrm{z}|\) & \multicolumn{1}{c}{ [95\% conf. interval] } \\
\hline Ix1__1 & 44.73377 & 8.256682 & 5.42 & 0.000 & 28.55097 & 60.91657 \\
Ix1__2 & -17.92302 & 3.909611 & -4.58 & 0.000 & -25.58571 & -10.26032 \\
x4a & .5006982 & .2496324 & 2.01 & 0.045 & .0114276 & .9899687 \\
Ix5__1 & .0387904 & .0076972 & 5.04 & 0.000 & .0237041 & .0538767 \\
Ix5__2 & -.5490645 & .0864255 & -6.35 & 0.000 & -.7184554 & -.3796736 \\
Ix6__1 & -1.806966 & .3506314 & -5.15 & 0.000 & -2.494191 & -1.119741 \\
hormon & -.4024169 & .1280843 & -3.14 & 0.002 & -.6534575 & -.1513763 \\
\hline
\end{tabular}

Deviance \(=3420.724\).
Some explanation of the output from the model-selection algorithm is desirable. Consider the first few lines of output in the iteration log:
\begin{tabular}{lllllllll} 
1. Deviance for model with all terms untransformed \(=3471.637,686\) & observations \\
Variable & Model (vs.) & Deviance & Dev diff. & P & Powers & (vs.) \\
\hline 2. x5 & null & FP2 & 3503.610 & 61.366 & \(0.000 *\) & & .5 & .5 \\
3. & lin. & & 3471.637 & 29.393 & \(0.000+\) & 1 & \\
4. & FP1 & & 3449.203 & 6.959 & \(0.031+\) & 0 & \\
5. & Final & 3442.244 & & & .5 &
\end{tabular}

Line 1 gives the deviance ( \(-2 \times \log\) partial likelihood) for the Cox model with all terms linear, the place where the algorithm starts. The model is modified variable by variable in subsequent steps. The most significant linear term turns out to be x 5 , which is therefore processed first. Line 2 compares the best-fitting FP2 for \(x 5\) with a model omitting \(x 5\). The FP has powers \((0.5,3)\), and the test for inclusion of \(x 5\) is highly significant. The reported deviance of \(3,503.610\) is of the null model, not for the FP2 model. The deviance for the FP2 model may be calculated by subtracting the deviance
difference (Dev diff.) from the reported deviance, giving 3,503.610-61.366 \(=3,442.244\). Line 3 shows that the FP2 model is also a significantly better fit than a straight line (lin.) and line 4 that FP2 is also somewhat better than FP1 \((p=0.031)\). Thus at this stage in the model-selection procedure, the final model for x 5 (line 5) is FP2 with powers \((0.5,3)\). The overall model with an FP2 for x 5 and all other terms linear has a deviance of \(3,442.244\).

After all the variables have been processed (cycle 1) and reprocessed (cycle 2) in this way, convergence is achieved because the functional forms (FP powers and variables included) after cycle 2 are the same as they were after cycle 1 . The model finally chosen is Model II as given in tables 3 and 4 of Sauerbrei and Royston (1999). Because of scaling of variables, the regression coefficients reported there are different, but the model and its deviance are identical. The model includes x 1 with powers \((-2,-0.5), \mathrm{x} 4 \mathrm{a}, \mathrm{x} 5\) with powers \((-2,-1)\), and x 6 with power 0.5 . There is strong evidence of nonlinearity for x 1 and for x 5 , the deviance differences for comparison with a straight-line model (FP2 vs lin.) being, respectively, 19.3 and 31.1 at convergence (cycle 2). Predictors x 2 , x 3 , x 4 b , and \(x 7\) are dropped, as may be seen from their status out in the table Final multivariable fractional polynomial model for _t (the assumed depvar when using stcox).

All predictors except x 4 a and hormon, which are binary, have been centered on the mean of the original variable. For example, the mean of x 1 (age) is 53.05 years. The first FP-transformed variable for x 1 is \(\mathrm{x} 1^{\wedge}-2\) and is created by the expression generate double Ix1_-1 = \(\mathrm{X}^{\wedge}-2-.0355\) if e(sample). The value 0.0355 is obtained from \((53.05 / 10)^{-2}\). The division by 10 is applied automatically to improve the scaling of the regression coefficient for Ix1__1.

According to Sauerbrei and Royston (1999), medical knowledge dictates that the estimated risk function for \(x 5\) (number of positive nodes), which was based on the above FP with powers \((-2,-1)\), should be monotonic, but it was not. They improved Model II by estimating a preliminary exponential transformation, \(\mathrm{x} 5 \mathrm{e}=\exp (-0.12 \cdot \mathrm{x} 5)\), for x 5 and fitting a degree 1 FP for x 5 e , thus obtaining a monotonic risk function. The value of -0.12 was estimated univariately using nonlinear Cox regression with the ado-file boxtid (Royston and Ambler 1999b, 1999d). To ensure a negative exponent, Sauerbrei and Royston (1999) restricted the powers for x5e to be positive. Their Model III may be fit by using the following command:
```

. mfp, alpha(.05) select(.05, hormon:1) df(x5e:2) xpowers(x5e:0.5 1 2 3):
> stcox x1 x2 x3 x4a x4b x5e x6 x7 hormon

```

Other than the customization for x 5 e , the command is the same as it was before. The resulting model is as reported in table 4 of Sauerbrei and Royston (1999):
. use https://www.stata-press.com/data/r18/brcancer, clear
(German breast cancer data)
. stset rectime, fail(censrec)
(output omitted)
. mfp, alpha(.05) select(.05, hormon:1) df(x5e:2) xpowers(x5e:0.5 123 ):
> stcox x1 x2 x3 x4a x4b x5e x6 x7 hormon, nohr
(output omitted)
Final multivariable fractional polynomial model for _t
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline \multirow[t]{2}{*}{Variable} & \multicolumn{3}{|l|}{_-Initial__} & \multirow[t]{2}{*}{Status} & \multicolumn{2}{|l|}{\multirow[t]{2}{*}{\[
\underset{\text { Final }}{ }
\]}} \\
\hline & df & Select & Alpha & & & \\
\hline x 1 & 4 & 0.0500 & 0.0500 & in & 4 & -2-. 5 \\
\hline x2 & 1 & 0.0500 & 0.0500 & out & 0 & \\
\hline x3 & 4 & 0.0500 & 0.0500 & out & 0 & \\
\hline x4a & 1 & 0.0500 & 0.0500 & in & 1 & 1 \\
\hline x4b & 1 & 0.0500 & 0.0500 & out & 0 & \\
\hline x5e & 2 & 0.0500 & 0.0500 & in & 1 & 1 \\
\hline x 6 & 4 & 0.0500 & 0.0500 & in & 2 & . 5 \\
\hline x7 & 4 & 0.0500 & 0.0500 & out & 0 & \\
\hline hormon & 1 & 1.0000 & 0.0500 & in & 1 & 1 \\
\hline
\end{tabular}

Cox regression -- Breslow method for ties
Entry time _t0
\begin{tabular}{llr} 
Number of obs & \(=\) & 686 \\
LR chi2 \((6)\) & \(=\) & 153.11 \\
Prob > chi2 & \(=\) & 0.0000 \\
Pseudo R2 & \(=\) & 0.0428
\end{tabular}
\begin{tabular}{r|rlllrr}
\hline _t & Coefficient & Std. err. & \multicolumn{1}{c}{\(z\)} & P>|z| & [95\% conf. interval] \\
\hline Ix1__1 & 43.55382 & 8.253433 & 5.28 & 0.000 & 27.37738 & 59.73025 \\
Ix1__2 & -17.48136 & 3.911882 & -4.47 & 0.000 & -25.14851 & -9.814212 \\
x4a & .5174351 & .2493739 & 2.07 & 0.038 & .0286713 & 1.006199 \\
Ix5e__1 & -1.981213 & .2268903 & -8.73 & 0.000 & -2.425909 & -1.536516 \\
Ix6__1 & -1.84008 & .3508432 & -5.24 & 0.000 & -2.52772 & -1.15244 \\
hormon & -.3944998 & .128097 & -3.08 & 0.002 & -.6455654 & -.1434342 \\
\hline
\end{tabular}

Deviance \(=3423.237\).

\section*{Stored results}

In addition to what regression_cmd stores, mfp stores the following in e():
Scalars
e(fp_nx) number of predictors in xvarlist
e(fp_dev)
deviance of final model fit
e(Fp_id\#)
e(Fp_fd\#)
e(Fp_al\#)
e(Fp_se\#)
initial degrees of freedom for the \#th element of xvarlist
final degrees of freedom for the \#th element of xvarlist
FP selection level for the \#th element of xvarlist
backward elimination selection level for the \#th element of xvarlist
Macros
e(fp_cmd)
fracpoly
e(fp_cmd2)
mf \(p\)
e(cmdline)
command as typed
e(fracpoly)
command used to fit the selected model using fracpoly
e(fp_fvl)
variables in final model
yvar \(_{1}\left(\right.\) yvar \(\left._{2}\right)\)
estimation command options
e(fp_opts)
first variable in xvarlist
second variable in xvarlist
e(fp_x2)
e(fp_xN)
last variable in xvarlist, \(N=\mathrm{e}\left(\mathrm{fp} \_\mathrm{nx}\right)\)
e(fp_k1)
power for first variable in xvarlist (*)
e(fp_k2)
power for second variable in xvarlist (*)
\(\mathrm{e}\left(\mathrm{fp} \_\mathrm{k} N\right) \quad\) power for last var. in xvarlist \((*), N=\mathrm{e}\left(\mathrm{fp} \_\mathrm{nx}\right)\)
Note: (*) contains '.' if the variable is not selected in the final model.
In addition to the above, the following is stored in \(r()\) :
Matrices
\(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r -class command is run after the estimation command.

\section*{Acknowledgments}
\(m f p\) is an update of mfracpol by Royston and Ambler (1998).

\section*{References}

Ambler, G., and P. Royston. 2001. Fractional polynomial model selection procedures: Investigation of Type I error rate. Journal of Statistical Computation and Simulation 69: 89-108. https://doi.org/10.1080/00949650108812083.

Royston, P., and D. G. Altman. 1994. Regression using fractional polynomials of continuous covariates: Parsimonious parametric modelling. Journal of the Royal Statistical Society, Series C 43: 429-467. https://doi.org/10.2307/2986270.

Royston, P., and G. Ambler. 1998. sg81: Multivariable fractional polynomials. Stata Technical Bulletin 43: 24-32. Reprinted in Stata Technical Bulletin Reprints, vol. 8, pp. 123-132. College Station, TX: Stata Press.

1999a. sg112: Nonlinear regression models involving power or exponential functions of covariates. Stata Technical Bulletin 49: 25-30. Reprinted in Stata Technical Bulletin Reprints, vol. 9, pp. 173-179. College Station, TX: Stata Press.
__. 1999b. sg81.1: Multivariable fractional polynomials: Update. Stata Technical Bulletin 49: 17-23. Reprinted in Stata Technical Bulletin Reprints, vol. 9, pp. 161-168. College Station, TX: Stata Press.

1999c. sg112.1: Nonlinear regression models involving power or exponential functions of covariates: Update. Stata Technical Bulletin 50: 26. Reprinted in Stata Technical Bulletin Reprints, vol. 9, p. 180. College Station, TX: Stata Press.
_-. 1999d. sg81.2: Multivariable fractional polynomials: Update. Stata Technical Bulletin 50: 25. Reprinted in Stata Technical Bulletin Reprints, vol. 9, p. 168. College Station, TX: Stata Press.

Royston, P., and W. Sauerbrei. 2007. Multivariable modeling with cubic regression splines: A principled approach. Stata Journal 7: 45-70.
——. 2008. Multivariable Model-Building: A Pragmatic Approach to Regression Analysis Based on Fractional Polynomials for Modelling Continuous Variables. Chichester, UK: Wiley.
——. 2009a. Two techniques for investigating interactions between treatment and continuous covariates in clinical trials. Stata Journal 9: 230-251.
——. 2009b. Bootstrap assessment of the stability of multivariable models. Stata Journal 9: 547-570.
——. 2016. mfpa: Extension of mfp using the ACD covariate transformation for enhanced parametric multivariable modeling. Stata Journal 16: 72-87.

Sauerbrei, W., and P. Royston. 1999. Building multivariable prognostic and diagnostic models: Transformation of the predictors by using fractional polynomials. Journal of the Royal Statistical Society, Series A 162: 71-94. https://doi.org/10.1111/1467-985X.00122.
2002. Corrigendum: Building multivariable prognostic and diagnostic models: Transformation of the predictors by using fractional polynomials. Journal of the Royal Statistical Society, Series A 165: 399-400. https://doi.org/10.1111/1467-985X.02026.

\section*{Also see}
[R] mfp postestimation - Postestimation tools for mfp
[R] fp - Fractional polynomial regression

\section*{[U] 20 Estimation and postestimation commands}

\section*{Title}
mfp postestimation - Postestimation tools for mfp
\begin{tabular}{ll} 
Postestimation commands & fracplot and fracpred \\
Methods and formulas & Also see
\end{tabular}

\section*{Postestimation commands}

The following postestimation commands are of special interest after mfp:
\begin{tabular}{ll} 
Command & Description \\
\hline fracplot & plot data and fit from most recently fit fractional polynomial model \\
fracpred & create variable containing prediction, deviance residuals, or SEs of fitted values
\end{tabular}

The following standard postestimation commands are also available if available after regression_cmd:
\begin{tabular}{ll} 
Command & Description \\
\hline estat ic & \begin{tabular}{l} 
Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian in- \\
formation criteria (AIC, CAIC, AICc, and BIC)
\end{tabular} \\
estat summarize & \begin{tabular}{l} 
summary statistics for the estimation sample
\end{tabular} \\
estat vce & variance-covariance matrix of the estimators (VCE) \\
estimates & \begin{tabular}{l} 
cataloging estimation results \\
table of estimation results
\end{tabular} \\
etable & \begin{tabular}{l} 
point estimates, standard errors, testing, and inference for linear combinations of \\
coefficients
\end{tabular} \\
lincom & \begin{tabular}{l} 
link test for model specification \\
likelihood-ratio test
\end{tabular} \\
lrtest & \begin{tabular}{l} 
point estimates, standard errors, testing, and inference for nonlinear combinations \\
of coefficients
\end{tabular} \\
nlcom & Wald tests of simple and composite linear hypotheses \\
test & Wald tests of nonlinear hypotheses \\
testnl &
\end{tabular}

\section*{fracplot and fracpred}

\section*{Description for fracplot and fracpred}
fracplot plots the data and fit, with \(95 \%\) confidence limits, from the most recently fit fractional polynomial (FP) model. The data and fit are plotted against varname, which may be \(x v a r_{1}\) or another of the covariates \(\left(x v a r_{2}, \ldots\right.\), or a variable from xvarlist). If varname is not specified, xvar \({ }_{1}\) is assumed.
fracpred creates newvar containing the fitted index or deviance residuals for the whole model, or the fitted index or its standard error for varname, which may be \(x^{2 v a r}{ }_{1}\) or another covariate.

\section*{Menu for fracplot and fracpred}

\section*{fracplot}
```

Statistics > Linear models and related > Fractional polynomials > Multivariable fractional polynomial plot

```

\section*{fracpred}

Statistics > Linear models and related > Fractional polynomials > Multivariable fractional polynomial prediction

\section*{Syntax for fracplot and fracpred}

Plot data and fit from most recently fit fractional polynomial model
fracplot [varname] [if] [in] [, fracplot_options]

Create variable containing the prediction, deviance residuals, or SEs of fitted values
fracpred newvar [, fracpred_options]
fracplot_options Description
Plot
marker_options
change look of markers (color, size, etc.)
marker_label_options add marker labels; change look or position

Fitted line
lineopts(cline_options) affect rendition of the fitted line
CI plot
ciopts(area_options) affect rendition of the confidence bands
Add plots
addplot (plot) add other plots to the generated graph
Y axis, X axis, Titles, Legend, Overall
twoway_options
any options other than by () documented in [G-3] twoway_options
\begin{tabular}{ll} 
fracpred_options & Description \\
\hline for(varname) & compute prediction for varname \\
dresid & compute deviance residuals \\
stdp & compute standard errors of the fitted values varname
\end{tabular}
fracplot is not allowed after mfp with clogit, mlogit, or stcrreg. fracpred, dresid is not allowed after mfp with clogit, mlogit, or stcrreg.

\section*{Options for fracplot}
\(\qquad\) Plot
marker_options affect the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G-3] marker_options.
marker_label_options specify if and how the markers are to be labeled; see [G-3] marker_label_options.
\(\qquad\) Fitted line
lineopts(cline_options) affect the rendition of the fitted line; see [G-3] cline_options.
\(\qquad\) Cl plot
ciopts(area_options) affect the rendition of the confidence bands; see [G-3] area_options.
\(\underset{\text { addplot (plot) provides a way to add other plots to the generated graph. See [G-3] addplot_option. }}{\text { Add plots }}\)

Y axis, X axis, Titles, Legend, Overall
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

\section*{Options for fracpred}
for (varname) specifies (partial) prediction for variable varname. The fitted values are adjusted to the value specified by the center() option in mfp.
dresid specifies that deviance residuals be calculated.
stdp specifies calculation of the standard errors of the fitted values varname, adjusted for all the other predictors at the values specified by center ().

\section*{Remarks and examples}
fracplot actually produces a component-plus-residual plot. For normal-error models with constant weights and one covariate, this amounts to a plot of the observations with the fitted line inscribed. For other normal-error models, weighted residuals are calculated and added to the fitted values.

For models with additional covariates, the line is the partial linear predictor for the variable in question (xvar \({ }_{1}\) or a covariate) and includes the intercept \(\beta_{0}\).

For generalized linear and Cox models, the fitted values are plotted on the scale of the "index" (linear predictor). Deviance residuals are added to the (partial) linear predictor to give component-plus-residual values. These values are plotted as small circles.

\section*{\(>\) Example 1}

In example 1 of \([\mathrm{R}] \mathbf{m f p}\), we used Cox regression to predict the \(\log\) hazard of breast cancer recurrence from prognostic factors of which five are continuous ( \(\mathrm{x} 1, \mathrm{x} 3, \mathrm{x} 5, \mathrm{x} 6, \mathrm{x} 7\) ) and three are binary ( \(\mathrm{x} 2, \mathrm{x} 4 \mathrm{a}, \mathrm{x} 4 \mathrm{~b}\) ). We also controlled for hormonal therapy (hormon). We used mfp to build a model from the initial set of eight predictors by using the backfitting model-selection algorithm. The nominal \(p\)-value for variable and FP selection was set to 0.05 for all variables except hormon, which is set to 1 .
```

. use https://www.stata-press.com/data/r18/brcancer
(German breast cancer data)
. stset rectime, fail(censrec)
(output omitted)
. mfp, alpha(.05) select(.05, hormon:1): stcox x1 x2 x3 x4a x4b x5 x6 x7 hormon,
> nohr
(output omitted)

```

We can use fracplot to produce component-plus-residual plots of the continuous variables. We produce the component-plus-residual plot for x 1 with fracplot by specifying x 1 after the command name.
```

. fracplot x1

```

Fractional polynomial (-2 -.5), adjusted for covariates


We use fracpred with the stdp option to predict the standard error of the fractional polynomial prediction for x 1 . The standard error prediction will be stored in variable sepx1. We specify that prediction is made for x 1 with the for () option. After prediction, we use summarize to show how the standard error estimate varies over different values of x 1 .
```

. fracpred sepx1, stdp for(x1)

```
. summarize sepx1
\begin{tabular}{r|rrrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline sepx1 & 686 & .0542654 & .0471993 & .0003304 & .6862065
\end{tabular}

\section*{Methods and formulas}

The general definition of an FP, accommodating possible repeated powers, may be written for functions \(H_{1}(x), \ldots, H_{m}(x)\) as
\[
\beta_{0}+\sum_{j=1}^{m} \beta_{j} H_{j}(x)
\]
where \(H_{1}(x)=x^{\left(p_{1}\right)}\) and for \(j=2, \ldots, m\),
\[
H_{j}(x)= \begin{cases}x^{\left(p_{j}\right)} & \text { if } p_{j} \neq p_{j-1} \\ H_{j-1}(x) \log x & \text { if } p_{j}=p_{j-1}\end{cases}
\]

For example, an FP of degree 3 with powers \((1,3,3)\) has \(H_{1}(x)=x, H_{2}(x)=x^{3}\), and \(H_{3}(x)=\) \(x^{3} \log x\) and equals \(\beta_{0}+\beta_{1} x+\beta_{2} x^{3}+\beta_{3} x^{3} \log x\).

The component-plus-residual values graphed by fracplot are calculated as follows: Let the data consist of triplets \(\left(y_{i}, x_{i}, \mathbf{z}_{i}\right), i=1, \ldots, n\), where \(\mathbf{z}_{i}\) is the vector of covariates for the \(i\) th observation, after applying possible fractional polynomial transformation and adjustment as described earlier. Let \(\widehat{\eta}_{i}=\widehat{\beta}_{0}+\left\{\mathbf{H}\left(x_{i}\right)-\mathbf{H}\left(x_{0}\right)\right\}^{\prime} \widehat{\boldsymbol{\beta}}+\mathbf{z}_{i}{ }_{\gamma}\) be the linear predictor from the FP model, as given by the fracpred command or, equivalently, by the predict command with the xb option, following mfp. Here \(\mathbf{H}\left(x_{i}\right)=\left\{H_{1}\left(x_{i}\right), \ldots, H_{m}\left(x_{i}\right)\right\}^{\prime}\) is the vector of FP functions described above, \(\mathbf{H}\left(x_{0}\right)=\left\{H_{1}\left(x_{0}\right), \ldots, H_{m}\left(x_{0}\right)\right\}^{\prime}\) is the vector of adjustments to \(x_{0}\) (often, \(x_{0}\) is chosen to be the mean of the \(\left.x_{i}\right), \widehat{\boldsymbol{\beta}}\) is the estimated parameter vector, and \(\widehat{\gamma}\) is the estimated parameter vector for the covariates. The values \(\widehat{\eta}_{i}^{*}=\widehat{\beta}_{0}+\left\{\mathbf{H}\left(x_{i}\right)-\mathbf{H}\left(x_{0}\right)\right\}^{\prime} \widehat{\boldsymbol{\beta}}\) represent the behavior of the FP model for \(x\) at fixed values \(\mathbf{z}=\mathbf{0}\) of the (adjusted) covariates. The \(i\) th component-plus-residual is defined as \(\widehat{\eta}_{i}^{*}+d_{i}\), where \(d_{i}\) is the deviance residual for the \(i\) th observation. For normal-errors models, \(d_{i}=\sqrt{w_{i}}\left(y_{i}-\widehat{\eta}_{i}\right)\), where \(w_{i}\) is the case weight (or 1 , if weight is not specified). For logistic, Cox, and generalized linear regression models, see [R] logistic, [R] probit, [ST] stcox, and [R] glm for the formula for \(d_{i}\). The formula for poisson models is the same as that for glm with family (poisson). For stcox, \(d_{i}\) is the partial martingale residual (see [ST] stcox postestimation).
fracplot plots the values of \(d_{i}\) and the curve represented by \(\widehat{\eta}_{i}^{*}\) against \(x_{i}\). The confidence interval for \(\widehat{\eta}_{i}^{*}\) is obtained from the variance-covariance matrix of the entire model and takes into account the uncertainty in estimating \(\beta_{0}, \boldsymbol{\beta}\), and \(\gamma\) (but not in estimating the FP powers for \(x\) ).
fracpred with the for (varname) option calculates the predicted index at \(x_{i}=x_{0}\) and \(\mathbf{z}_{i}=\mathbf{0}\); that is, \(\widehat{\eta}_{i}=\widehat{\beta}_{0}+\left\{\mathbf{H}\left(x_{i}\right)-\mathbf{H}\left(x_{0}\right)\right\}^{\prime} \widehat{\boldsymbol{\beta}}\). The standard error is calculated from the variance-covariance matrix of \(\left(\widehat{\beta}_{0}, \widehat{\boldsymbol{\beta}}\right)\), again ignoring estimation of the powers.

\section*{Also see}
[R] mfp - Multivariable fractional polynomial models
[U] 20 Estimation and postestimation commands

\section*{Title}
misstable - Tabulate missing values
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Also see
\end{tabular}

\section*{Description}
misstable makes tables that help you understand the pattern of missing values in your data.

\section*{Quick start}

Tables with counts of missing values
Missing observations in v1, v2, and v3
misstable summarize v1 v2 v3
Missing observations in v1-v3 for cases where v4 \(>10\) misstable summarize v1 v2 v3 if v4>10

All variables with missing values misstable summarize

Include variables with no missing values in the table misstable summarize, all

Create 3 missing-data indicator variables with prefix m for v1, v2, and v3
misstable summarize v1 v2 v3, generate(m)

Tables of missing-value patterns
Missing-value patterns for v1, v2, and v3
misstable patterns v1 v2 v3
Same as above, but for all variables with missing values misstable patterns

Show variables in the order listed in the command misstable patterns v1 v3 v2, asis

Show frequencies, rather than percentages, in the output misstable patterns v1 v2 v3, frequency

Missing-value patterns displayed as trees
A tree view of missing-value patterns for v1, v2, and v3
misstable tree v1 v2 v3

For all variables with missing values
misstable tree

Nesting patterns of missing values
Missing values for \(v 1, \mathrm{v} 2\), and v 3
misstable nested v1 v2 v3
Same as above, but for all variables with missing values
misstable nested
Treat extended missing values (.a, .b,..., .z) as nonmissing
misstable nested v1 v2 v3, exok

\section*{Menu}

Statistics \(>\) Summaries, tables, and tests \(>\) Other tables \(>\) Tabulate missing values

\section*{Syntax}

Report counts of missing values
```

misstable summarize [varlist] [if] [in] [, summarize_options]

```

Report pattern of missing values
```

misstable patterns [varlist] [if] [in] [, patterns_options]

```

Present a tree view of the pattern of missing values
```

misstable tree [varlist] [if] [in] [, tree_options]

```

List the nesting rules that describe the missing-value pattern
```

misstable nested [varlist] [if] [in] [, nested_options]

```
summarize_options Description
\begin{tabular}{ll} 
all & \begin{tabular}{l} 
show all variables \\
show zeros in table
\end{tabular} \\
showzeros & senerate (stub [, exok] \()\) \\
generate missing-value indicators
\end{tabular}
\begin{tabular}{ll} 
patterns_options & Description \\
\hline asis & use variables in order given \\
frequency & report frequencies instead of percentages \\
\hline exok & treat \(. a, . b, \ldots, . \mathrm{z}\) as nonmissing \\
replace & replace data in memory with dataset of patterns \\
clear & okay to replace even if original unsaved \\
bypatterns & list by patterns rather than by frequency
\end{tabular}
\begin{tabular}{ll} 
tree_options & Description \\
\hline asis & use variables in order given \\
frequency & report frequencies instead of percentages \\
\hline exok & treat \(. a, . b, \ldots, . z\) as nonmissing
\end{tabular}
nested_options

Description
exok treat \(. a, . b, \ldots, . z\) as nonmissing

In addition, programmer's option nopreserve is allowed with all syntaxes; see [P] nopreserve option. collect is allowed; see [U] 11.1.10 Prefix commands.

\section*{Options}

Options are presented under the following headings:
Options for misstable summarize
Options for misstable patterns Options for misstable tree Option for misstable nested Common options

\section*{Options for misstable summarize}
all specifies that the table should include all the variables specified or all the variables in the dataset. The default is to include only numeric variables that contain missing values.
showzeros specifies that zeros in the table should display as 0 rather than being omitted.
generate (stub [, exok]) requests that a missing-value indicator newvar, a new binary variable containing 0 for complete observations and 1 for incomplete observations, be generated for every numeric variable in varlist containing missing values. If the all option is specified, missing-value indicators are created for all the numeric variables specified or for all the numeric variables in the dataset. If exok is specified within generate(), the extended missing values .a, .b, ..., .z are treated as if they do not designate missing.
For each variable in varlist, newvar is the corresponding variable name varname prefixed with stub. If the total length of stub and varname exceeds 32 characters, newvar is abbreviated so that its name does not exceed 32 characters.

\section*{Options for misstable patterns}
asis, frequency, and exok - see Common options below.
replace specifies that the data in memory be replaced with a dataset corresponding to the table just displayed; see misstable patterns under Remarks and examples below.
clear is for use with replace; it specifies that it is okay to change the data in memory even if they have not been saved to disk.
bypatterns specifies the table be ordered by pattern rather than by frequency. That is, bypatterns specifies that patterns containing one incomplete variable be listed first, followed by those for two incomplete variables, and so on. The default is to list the most frequent pattern first, followed by the next most frequent pattern, etc.

\section*{Options for misstable tree}
asis, frequency, and exok - see Common options below.

\section*{Option for misstable nested}
exok - see Common options below.

\section*{Common options}
asis specifies that the order of the variables in the table be the same as the order in which they are specified on the misstable command. The default is to order the variables by the number of missing values, and within that, by the amount of overlap of missing values.
frequency specifies that the table should report frequencies instead of percentages.
exok specifies that the extended missing values .a, .b, ..., .z should be treated as if they do not designate missing. Some users use extended missing values to designate values that are missing for a known and valid reason.
nopreserve is a programmer's option allowed with all misstable commands; see [P] nopreserve option.

\section*{Remarks and examples}

Remarks are presented under the following headings:
```

misstable summarize
misstable patterns
misstable tree
misstable nested
Execution time of misstable nested

```

In what follows, we will use data from a 125-observation, fictional, student-satisfaction survey:
. use https://www.stata-press.com/data/r18/studentsurvey
(Student survey)
. summarize
Variable

The m 1 and m 2 variables record the student's satisfaction with teaching and with academics. comment is a string variable recording any comments the student might have had.

\section*{misstable summarize}
> Example 1
misstable summarize reports counts of missing values:
. misstable summarize
Obs<.
\begin{tabular}{r|rrr|rrr}
\cline { 4 - 6 } Variable & Obs=. & Obs>. & Obs<. & \begin{tabular}{l} 
Unique \\
values
\end{tabular} & Min & Max \\
\hline age & 3 & & 122 & 5 & 17 & 21 \\
female & 3 & 122 & 2 & 0 & 1 \\
dept & 9 & & 116 & 4 & 1 & 4 \\
\hline
\end{tabular}

Stata provides 27 different missing values, namely, ., .a, .b, ... .z. The first of those, ., is often called system missing. The remaining missing values are called extended missings. The nonmissing and missing values are ordered nonmissing \(<.<. \mathrm{a}<. \mathrm{b}<\cdots<. \mathrm{z}\). Thus, reported in the column "Obs=." are counts of system missing values; in the column "Obs \(>\).", extended missing values; and in the column "Obs \(<\).", nonmissing values.

The rightmost portion of the table is included to remind you how the variables are encoded.
Our data contain seven variables and yet misstable reported only three of them. The omitted variables contain no missing values or are string variables. Even if we specified the varlist explicitly, those variables would not appear in the table unless we specified the all option.

We can also create missing-value indicators for each of the variables above using the generate() option:
\begin{tabular}{|c|c|c|c|c|}
\hline Variable name & Storage type & Display format & \begin{tabular}{l}
Value \\
label
\end{tabular} & Variable label \\
\hline miss_age & byte & \%8.0g & & (age>=.) \\
\hline miss_female & byte & \%8.0g & & (female>=.) \\
\hline miss_dept & byte & \%8.0g & & (dept>=.) \\
\hline
\end{tabular}

For each variable containing missing values, the generate() option creates a new binary variable containing 0 for complete observations and 1 for incomplete observations. In our example, three new missing-value indicators are generated, one for each of the incomplete variables age, female, and dept. The naming convention of generate() is to prefix the corresponding variable names with the specified stub, which is miss_ in this example.

Missing-value indicators are useful, for example, for checking whether data are missing completely at random. They are also often used within the multiple-imputation context to identify the observed and imputed data; see [MI] Intro substantive for a general introduction to multiple imputation. Within Stata's multiple-imputation commands, an incomplete value is identified by the system missing value, a dot. By default, misstable summarize, generate() marks the extended missing values as incomplete values, as well. You can use exok within generate() to treat extended missing values as complete when creating missing-value identifiers.

\section*{misstable patterns}

\section*{> Example 2}
misstable patterns reports the pattern of missing values:


There are three patterns in these data: \((1,1,1),(1,1,0)\), and \((0,0,0)\). By default, the rows of the table are ordered by frequency. In larger tables that have more patterns, it is sometimes useful to order the rows by pattern. We could have obtained that by typing mi misstable patterns, bypatterns.

In a pattern, 1 indicates that all values of the variable are nonmissing and 0 indicates that all values are missing. Thus, pattern \((1,1,1)\) means no missing values, and \(93 \%\) of our data have that pattern. There are two patterns in which variables are missing, \((1,1,0)\) and \((0,0,0)\). Pattern \((1,1,0)\) means that age is nonmissing, female is nonmissing, and dept is missing. The order of the variables in the patterns appears in the key at the bottom of the table. Five percent of the observations have pattern \((1,1,0)\). The remaining \(2 \%\) have pattern \((0,0,0)\), meaning that all three variables contain missing.

As with misstable summarize, only numeric variables that contain missing are listed, so had we typed misstable patterns comments age female offcampus dept, we still would have obtained the same table. Variables that are automatically omitted contain no missing values or are string variables.

The variables in the table are ordered from lowest to highest frequency of missing values, although you cannot see that from the information presented in the table. The variables are ordered this way even if you explicitly specify the varlist with a different ordering. Typing misstable patterns dept female age would produce the same table as above. Specify the asis option if you want the variables in the order in which you specify them.

You can obtain a dataset of the patterns by specifying the replace option:

. list

The differences between the dataset and the printed table are that 1 ) the dataset always records frequency and 2) the rows are reversed.

\section*{misstable tree}

\section*{> Example 3}
misstable tree presents a tree view of the pattern of missing values:
```

. use https://www.stata-press.com/data/r18/studentsurvey, clear
(Student survey)
. misstable tree, frequency
Nested pattern of missing values
dept age female
9 3
6 0
0}
116 0 0
116 0
116
(number missing listed first)

```

In this example, we specified the frequency option to see the table in frequency rather than percentage terms. In the table, each column sums to the total number of observations in the data, 125. Variables are ordered from those with the most missing values to those with the least. Start with the first column. The dept variable is missing in 9 observations and, farther down, the table reports that it is not missing in 116 observations.

Go back to the first row and read across, but only to the second column. The dept variable is missing in 9 observations. Within those 9 , age is missing in 3 of them and is not missing in the remaining 6. Reading down the second column, within the 116 observations that dept is not missing, age is missing in 0 and not missing in 116 .

Reading straight across the first row again, dept is missing in 9 observations, and within the 9 , age is missing in 3 , and within the 3 , female is also missing in 3 . Skipping down just a little, within the 6 observations for which dept is missing and age is not missing, female is not missing, too.

\section*{misstable nested}
> Example 4
misstable nested lists the nesting rules that describe the missing-value pattern,
```

. misstable nested
1. female(3) <-> age(3) -> dept(9)

```

This line says that in observations in which female is missing, so is age missing, and vice versa, and in observations in which age (or female) is missing, so is dept. The numbers in parentheses are counts of the missing values. The female variable happens to be missing in 3 observations, and the same is true for age; the dept variable is missing in 9 observations. Thus, dept is missing in the 3 observations for which age and female are missing, and in 6 more observations, too.

In these data, it turns out that the missing-value pattern can be summarized in one statement. In a larger dataset, you might see something like this:
```

. misstable nested

```
1. female(50) <-> age(50) -> dept(120)
2. female(50) -> m1 (58)
3. offcampus(11)
misstable nested accounts for every missing value. In the above, in addition to female <-> age \(\rightarrow\) dept, we have that female \(->\) m1, and we have offcampus, the last all by itself. The last line says that the 11 missing values in offcampus are not themselves nested in the missing value of any other variable, nor do they imply the missing values in another variable. In some datasets, all the statements will be of this last form.

In our data, however, we have one statement:
```

. misstable nested

```
1. female(3) <-> age(3) -> dept(9)

When the missing-value pattern can be summarized in one misstable nested statement, the pattern of missing values in the data is said to be monotone.

\section*{Execution time of misstable nested}

The execution time of misstable nested is affected little by the number of observations but can grow quickly with the number of variables, depending on the fraction of missing values within variable. The execution time of the example above, which has 3 variables containing missing, is instant. In worst-case scenarios, with 500 variables, the time might be 25 seconds; with 1,000 variables, the execution time might be closer to an hour.

In situations where misstable nested takes a long time to complete, it will produce thousands of rules that will defy interpretation. A 523-variable dataset we have seen ran in 20 seconds and produced 8,040 rules. Although we spotted a few rules in the output that did not surprise us, such as the year of the date being missing implied that the month and the day were also missing, mostly the output was not helpful.

If you have such a dataset, we recommend you run misstable on groups of variables that you have reason to believe the pattern of missing values might be related.

\section*{Stored results}
misstable summarize stores the following values of the last variable summarized in \(r()\) :

\section*{Scalars}
```

    r(N_eq_dot) number of observations containing .
    r(N_gt_dot) number of observations containing .a, .b, ..., .z
    r(N_lt_dot) number of observations containing nonmissing
    r(K_uniq) number of unique, nonmissing values
    r(min) variable's minimum value
    r(max) variable's maximum value
    r(vartype) numeric, string, or none
    ```
Macros
\(r\) (K_uniq) contains . if the number of unique, nonmissing values is greater than 500. \(r\) (vartype) contains none if no variables are summarized, and in that case, the value of the scalars are all set to missing (.). Programmers intending to access results after misstable summarize should specify the all option.
misstable patterns stores the following in \(r()\) :
Scalars
\[
\begin{array}{ll}
r\left(N \_c o m p l e t e\right) & \text { number of complete observations } \\
r\left(N \_i n c o m p l e t e\right) & \text { number of incomplete observations } \\
r(K) & \text { number of patterns }
\end{array}
\]

Macros
\(r\) (vars) variables used in order presented
\(r\) ( \(N\) _complete) and \(r\) ( \(N\) _incomplete) are defined with respect to the variables specified if variables were specified and otherwise, defined with respect to all the numeric variables in the dataset. \(r\) ( \(N\) _complete) is the number of observations that contain no missing values.
misstable tree stores the following in \(r()\) :
Macros
\(r\) (vars) variables used in order presented
misstable nested stores the following in r() :
Scalars
\(r(K) \quad\) number of statements
Macros
\(r\) (stmt1) first statement
\(r\) (stmt2) second statement
- .
\(r(\) stmt' \(r(K)\) ') last statement
\(r\) (stmt1wc) \(\quad r\) (stmt1) with missing-value counts
\(r\) (vars) variables considered
A statement is encoded "varname", "varname op varname", or "varname op varname op varname", and so on; \(o p\) is either "->" or "<->".

\section*{Also see \\ [MI] mi misstable - Tabulate pattern of missing values \\ [R] summarize - Summary statistics \\ [R] tabulate oneway - One-way table of frequencies \\ [R] tabulate twoway - Two-way table of frequencies}

\section*{Title}
ml — Maximum likelihood estimation
\begin{tabular}{llll} 
Description & Syntax & Options & Remarks and examples \\
Stored results & Methods and formulas & References & Also see
\end{tabular}

\section*{Description}
ml model defines the current problem.
ml clear clears the current problem definition. This command is rarely used because when you type ml model, any previous problem is automatically cleared.
ml query displays a description of the current problem.
ml check verifies that the log-likelihood evaluator you have written works. We strongly recommend using this command.
ml search searches for (better) initial values. We recommend using this command.
ml plot provides a graphical way of searching for (better) initial values.
ml init provides a way to specify initial values.
ml report reports \(\ln L\) 's values, gradient, and Hessian at the initial values or current parameter estimates, \(\mathbf{b}_{0}\).
ml trace traces the execution of the user-defined log-likelihood evaluation program.
ml count counts the number of times the user-defined log-likelihood evaluation program is called; this command is seldom used. ml count clear clears the counter. ml count on turns on the counter. ml count without arguments reports the current values of the counter. ml count off stops counting calls.
ml maximize maximizes the likelihood function and reports results. Once ml maximize has successfully completed, the previously mentioned ml commands may no longer be used unless noclear is specified. ml graph and ml display may be used whether or not noclear is specified.
ml graph graphs the log-likelihood values against the iteration number.
ml display redisplays results.
ml footnote displays a warning message when the model did not converge within the specified number of iterations.
ml score creates new variables containing the equation-level scores. The variables generated by ml score are equivalent to those generated by specifying the score() option of ml maximize (and ml model maximize).
progname is the name of a Stata program you write to evaluate the log-likelihood function.
funcname () is the name of a Mata function you write to evaluate the log-likelihood function.
In this documentation, progname and funcname () are referred to as the user-written evaluator, the likelihood evaluator, or sometimes simply as the evaluator. The program you write is written in the style required by the method you choose. The methods are lf, d0, d1, d2, lf0, lf1, lf2, and gf0. Thus, if you choose to use method lf, your program is called a method-lf evaluator.

Method-lf evaluators are required to evaluate the observation-by-observation \(\log\) likelihood \(\ln \ell_{j}\), \(j=1, \ldots, N\).

Method-d0 evaluators are required to evaluate the overall \(\log\) likelihood \(\ln L\). Method-d1 evaluators are required to evaluate the overall \(\log\) likelihood and its gradient vector \(\mathbf{g}=\partial \ln L / \partial \mathbf{b}\). Method-d2 evaluators are required to evaluate the overall log likelihood, its gradient, and its Hessian matrix \(H=\partial^{2} \ln L / \partial \mathbf{b} \partial \mathbf{b}^{\prime}\).
Method-lf0 evaluators are required to evaluate the observation-by-observation \(\log\) likelihood \(\ln \ell_{j}\), \(j=1, \ldots, N\). Method-lf1 evaluators are required to evaluate the observation-by-observation \(\log\) likelihood and its equation-level scores \(g_{j i}=\partial \ln \ell / \partial \mathbf{x}_{j i} \mathbf{b}_{i}\). Method-lf2 evaluators are required to evaluate the observation-by-observation log likelihood, its equation-level scores, and its Hessian matrix \(H=\partial^{2} \ln \ell / \partial \mathbf{b} \partial \mathbf{b}^{\prime}\).
Method-gf0 evaluators are required to evaluate the summable pieces of the \(\log\) likelihood \(\ln \ell_{k}\), \(k=1, \ldots, K\).
mleval is a subroutine used by evaluators of methods d0, d1, d2, lf0, lf1, lf2, and gf0 to evaluate the coefficient vector, \(\mathbf{b}\), that they are passed.
mlsum is a subroutine used by evaluators of methods \(\mathrm{d} 0, \mathrm{~d} 1\), and d 2 to define the value, \(\ln L\), that is to be returned.
mlvecsum is a subroutine used by evaluators of methods d 1 and d 2 to define the gradient vector, \(\mathbf{g}\), that is to be returned. It is suitable for use only when the likelihood function meets the linear-form restrictions.
mlmatsum is a subroutine used by evaluators of methods d2 and lf2 to define the Hessian matrix, \(\mathbf{H}\), that is to be returned. It is suitable for use only when the likelihood function meets the linear-form restrictions.
mlmatbysum is a subroutine used by evaluator of method d2 to help define the Hessian matrix, \(\mathbf{H}\), that is to be returned. It is suitable for use when the likelihood function contains terms made up of grouped sums, such as in panel-data models. For such models, use mlmatsum to compute the observation-level outer products and mlmatbysum to compute the group-level outer products. mlmatbysum requires that the data be sorted by the variable identified in the by () option.

\section*{Syntax}
ml model in interactive mode
ml model method progname eq \([\) eq ...] [if][in] [weight]
[, model_options svy diparm_options]
ml model method funcname() eq \([\) eq...] [if] [in] [weight]
[, model_options svy diparm_options]
ml model in noninteractive mode
ml model method progname eq [eq...] [if] [in] [weight], maximize [model_options svy diparm_options noninteractive_options]
ml model method funcname() eq [eq...] [if] [in] [weight], maximize [model_options svy diparm_options noninteractive_options]

Noninteractive mode is invoked by specifying the maximize option. Use maximize when ml will be used as a subroutine of another ado-file or program and you want to carry forth the problem, from definition to posting of results, in one command.
ml clear
ml query
ml check
ml search \(\quad\left[[/]\right.\) eqname \(\left.[:] \#_{l b} \#_{u b}\right][\ldots]\) [, search_options \(]\)
ml plot \([\) eqname: \(]\) name \([\#[\#[\#]]][\), saving(filename \([\), replace \(]\) )]
ml init \(\quad\{[\) eqname \(:]\) name \(=\# \mid\) eqname \(=\#\}[. .\).
ml init \# [\#...], copy
ml init matname [, copy skip]
ml report
ml trace \(\quad\{\) on \(\mid\) off \(\}\)
ml count \(\quad[\) clear \(\mid\) on \(\mid\) off \(]\)
ml maximize [, ml_maximize_options display_options eform_option]
ml graph [\#][, saving(filename[, replace])]
ml display [, display_options eform_option]
ml footnote
ml score newvar \([\) if \(][i n]\) [, equation(eqname) missing]
ml score newvarlist [if] [in] [, missing]
ml score [type] stub* [if] [in] [, missing]
where method is one of
\begin{tabular}{llll} 
lf & d0 & lf0 & gf0 \\
& d1 & lf1 & \\
& d1debug & lf1debug & \\
& d2 & lf2 & \\
& d2debug & lf2debug &
\end{tabular}
or method can be specified using one of the longer, more descriptive names
\begin{tabular}{ll} 
method & Longer name \\
\hline lf & linearform \\
d0 & derivative0 \\
d1 & derivative1 \\
d1debug & derivative1debug \\
d2 & derivative2 \\
d2debug & derivative2debug \\
lf0 & linearform0 \\
lf1 & linearform1 \\
lf1debug & linearform1debug \\
lf2 & linearform2 \\
lf2debug & linearform2debug \\
gf0 & generalform0 \\
\hline
\end{tabular}
\(e q\) is the equation to be estimated, enclosed in parentheses, and optionally with a name to be given to the equation, preceded by a colon,
\[
\left([\text { eqname }: ~]\left[\text { varlist }_{y}=\right]\left[\text { varlist }_{x}\right][\text {, eq_options }]\right)
\]
or eq is the name of a parameter, such as sigma, with a slash in front
/eqname which is equivalent to (eqname:, freeparm)
and diparm_options is one or more diparm(diparm_args) options where diparm_args is either __sep__ or anything accepted by the "undocumented" _diparm command; see help _diparm.
eq_options

\section*{Description}
noconstant
offset (varname \({ }_{o}\) )
exposure ( varname \(_{e}\) )
freeparm
do not include an intercept in the equation
include varname \(e_{o}\) in model with coefficient constrained to 1 include \(\ln \left(\right.\) varname \(\left._{e}\right)\) in model with coefficient constrained to 1 eqname is a free parameter
model_options
Description
\begin{tabular}{|c|c|}
\hline group(varname) & use varname to identify groups \\
\hline vce(vcetype) & vcetype may be robust, cluster clustvar, oim, or opg \\
\hline constraints (numlist) & constraints by number to be applied \\
\hline constraints (matname) & matrix that contains the constraints to be applied \\
\hline nocnsnotes & do not display notes when constraints are dropped \\
\hline title(string) & place a title on the estimation output \\
\hline nopreserve & do not preserve the estimation subsample in memory \\
\hline collinear & keep collinear variables within equations \\
\hline missing & keep observations containing variables with missing values \\
\hline \(\left.\mathrm{lf0} 0 \#_{k} \#_{l l}\right)\) & number of parameters and log-likelihood value of the constant-only model \\
\hline continue & \begin{tabular}{l}
specifies that a model has been fit and sets the initial values \\
\(\mathbf{b}_{0}\) for the model to be fit based on those results
\end{tabular} \\
\hline waldtest(\#) & perform a Wald test; see Options for use with ml model in interactive or noninteractive mode below \\
\hline obs(\#) & number of observations \\
\hline crittype (string) & describe the criterion optimized by ml \\
\hline subpop(varname) & compute estimates for the single subpopulation \\
\hline nosvyadjust & carry out Wald test as \(W / k \sim F(k, d)\) \\
\hline technique(nr) & Stata's modified Newton-Raphson (NR) algorithm \\
\hline technique (bhhh) & Berndt-Hall-Hall-Hausman (BHHH) algorithm \\
\hline technique (dfp) & Davidon-Fletcher-Powell (DFP) algorithm \\
\hline technique (bfgs) & Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm \\
\hline deriv(min_spec) & specify the minimum step size for computing numerical derivatives \\
\hline
\end{tabular}
noninteractive_options
init (ml_init_args)
search (on)
search(norescale)
search (quietly)
search (off)
repeat (\#)
bounds ( \(m l\) _search_bounds)
nowarning
novce
negh
score (newvars)
maximize_options

Description
set the initial values \(\mathbf{b}_{0}\)
equivalent to ml search, repeat ( 0 ) ; the default
equivalent to ml search, repeat (0) norescale
same as search(on), except that output is suppressed
prevents calling ml search
ml search's repeat() option; see below
specify bounds for ml search
suppress "convergence not achieved" message of iterate (0)
substitute the zero matrix for the variance matrix
indicates that the evaluator returns the negative Hessian matrix
new variables containing the contribution to the score
control the maximization process; seldom used
search_options
Description
repeat(\#) norescale
maximize_options
restart use random actions to find starting values; not recommended
number of random attempts to find better initial-value vector; default is repeat (10) in interactive mode and repeat ( 0 ) in noninteractive mode do not rescale to improve parameter vector; not recommended control the maximization process; seldom used
ml_maximize_options
nowarning
novce
negh
score (newvars \(\mid\) stub*)
nooutput
noskipline
noclear
maximize_options

Description
suppress "convergence not achieved" message of iterate (0) substitute the zero matrix for the variance matrix indicates that the evaluator returns the negative Hessian matrix new variables containing the contribution to the score suppress display of final results suppress display of blank line before iteration log do not clear ml problem definition after model has converged control the maximization process; seldom used
display_options
Description
noheader
nofootnote
level(\#)
first
neq(\#)
showeqns
plus
nocnsreport
noomitted
vsquish
noemptycells
baselevels
allbaselevels
cformat (\% fint)
pformat (\% fmt)
sformat (\%fmt)
nolstretch
coeflegend
suppress header display above the coefficient table suppress footnote display below the coefficient table set confidence level; default is level (95)
display coefficient table reporting results for first equation only display coefficient table reporting first \# equations display equation names in the coefficient table display coefficient table ending in dashes-plus-sign-dashes suppress constraints display above the coefficient table suppress display of omitted variables
suppress blank space separating factor-variable terms or time-series-operated variables from other variables suppress empty cells for interactions of factor variables
report base levels of factor variables and interactions display all base levels of factor variables and interactions
format the coefficients, standard errors, and confidence limits in the coefficient table
format the \(p\)-values in the coefficient table
format the test statistics in the coefficient table
do not automatically widen the coefficient table to accommodate longer variable names
display legend instead of statistics
eform_option
Description
eform (string)
eform
hr
shr
irr
or
rrr
display exponentiated coefficients; column title is "string"
display exponentiated coefficients; column title is "exp(b)"
report hazard ratios
report subhazard ratios
report incidence-rate ratios
report odds ratios
report relative-risk ratios
fweights, aweights, iweights, and pweights are allowed; see [U] 11.1.6 weight. With all but method lf, you must write your likelihood-evaluation program carefully if pweights are to be specified, and pweights may not be specified with method d0, d1, d1debug, d2, or d2debug. See Pitblado, Poi, and Gould (2024, chap. 7) for details.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.
To redisplay results, type ml display.

\section*{Syntax of subroutines for use by evaluator programs}
```

mleval newvar $=$ vecname $[$, eq(\#) $]$
mleval scalarname $=$ vecname, scalar $[\operatorname{eq}(\#)]$
mlsum $\quad$ scalarname $_{\operatorname{lnf}}=\exp [$ if $][$, noweight $]$
mlvecsum $\quad$ scalarname $_{\operatorname{lnf}}$ rowvecname $=\exp [$ if $][$, eq(\#) $]$
mlmatsum $\quad$ scalarname $_{\operatorname{lnf}}$ matrixname $=\exp [$ if $][$, eq $(\#[, \#])]$
mlmatbysum $^{\text {scalarname }_{\operatorname{lnf}}}$ matrixname varname ${ }_{a}$ varname $_{b}\left[\right.$ varname $\left._{c}\right][$ if $]$,
by (varname) [ eq(\#[,\#])]

```

\section*{Syntax of user-written evaluator}

\section*{Summary of notation}

The log-likelihood function is \(\ln L\left(\theta_{1 j}, \theta_{2 j}, \ldots, \theta_{E j}\right)\), where \(\theta_{i j}=\mathbf{x}_{i j} \mathbf{b}_{i}, j=1, \ldots, N\) indexes observations, and \(i=1, \ldots, E\) indexes the linear equations defined by ml model. If the likelihood satisfies the linear-form restrictions, it can be decomposed as \(\ln L=\sum_{j=1}^{N} \ln \ell\left(\theta_{1 j}, \theta_{2 j}, \ldots, \theta_{E j}\right)\).

\section*{Method-lf evaluators}
program progname
version 18.0 // (or version 18.5 for StataNow) args lnfj theta1 theta2 ...
// if you need to create any intermediate results:
tempvar tmp1 tmp2...
quietly generate double 'tmp1' = ...
quietly replace ' 1 nfj f =..
where
' \(\operatorname{lnf} \mathrm{j}\) ' variable to be filled in with observation-by-observation values of \(\ln \ell_{j}\)
'theta1' variable containing evaluation of first equation \(\theta_{1 j}=\mathbf{x}_{1 j} \mathbf{b}_{1}\)
'theta2' variable containing evaluation of second equation \(\theta_{2 j}=\mathbf{x}_{2 j} \mathbf{b}_{2}\)

\section*{Method-d0 evaluators}
program progname
version 18.0
// (or version 18.5 for StataNow)
args todo b lnf
tempvar thetal theta2...
mleval 'theta1' = 'b', eq(1)
mleval 'theta2' \(=\) ' \(b\) ', eq(2) // if there is a \(\theta_{2}\)
// if you need to create any intermediate results:
tempvar tmp1 tmp2...
generate double 'tmp1' = ...
...
mlsum ' 1 nf ' \(=\ldots\)
end
where
```

'todo' always contains 0 (may be ignored)
'b' full parameter row vector b=(b
'lnf' scalar to be filled in with overall }\operatorname{ln}

```

\section*{Method-d1 evaluators}
```

program progname
version 18.0 // (or version 18.5 for StataNow)
args todo b lnf g
tempvar thetal theta2...
mleval 'theta1' = 'b', eq(1)
mleval 'theta2' = 'b', eq(2) // if there is a }\mp@subsup{0}{2}{
// if you need to create any intermediate results:
tempvar tmp1 tmp2...
generate double 'tmp1' = ...
mlsum 'lnf' = ...
if ('todo'==0 | 'lnf'>=.) exit
tempname d1 d2 ...
mlvecsum 'lnf' 'd1' = formula for }\partial\operatorname{ln}\mp@subsup{\ell}{j}{}/\partial\mp@subsup{0}{1j}{}\mathrm{ , eq(1)
mlvecsum 'lnf' 'd2' = formula for }\partial\operatorname{ln}\mp@subsup{\ell}{j}{}/\partial\mp@subsup{0}{2j}{}\mathrm{ , eq(2)
matrix 'g' = ('d1','d2', ... )
end

```
```

'todo' contains 0 or 1

```
'todo' contains 0 or 1
                        0=>'lnf'to be filled in;
                        1=>'lnf' and 'g' to be filled in
''b' full parameter row vector b=( }\mp@subsup{\mathbf{b}}{1}{},\mp@subsup{\mathbf{b}}{2}{},\ldots,\mp@subsup{\mathbf{b}}{E}{}
'lnf' scalar to be filled in with overall }\operatorname{ln}
'g' row vector to be filled in with overall }\mathbf{g}=\partial\operatorname{ln}L/\partial\mathbf{b
```

where

## Method-d2 evaluators

```
program progname
    version 18.0 // (or version 18.5 for StataNow)
    args todo b lnf g H
    tempvar thetal theta2...
    mleval 'theta1' = 'b', eq(1)
    mleval 'theta2' = 'b', eq(2) // if there is a }\mp@subsup{0}{2}{
    // if you need to create any intermediate results:
    tempvar tmp1 tmp2...
    generate double 'tmpl' = ...
    mlsum 'lnf' = ...
    if ('todo'==0 | 'lnf'>=.) exit
    tempname d1 d2 ...
    mlvecsum 'lnf' 'd1' = formula for }\partial\operatorname{ln}\mp@subsup{\ell}{j}{}/\partial\mp@subsup{0}{1j}{}\mathrm{ , eq(1)
    mlvecsum 'lnf' 'd2' = formula for }\partial\operatorname{ln}\mp@subsup{\ell}{j}{}/\partial\mp@subsup{0}{2j}{}\mathrm{ , eq(2)
    matrix 'g' = ('d1','d2', ... )
    if ('todo'==1 | 'lnf'>=.) exit
    tempname d11 d12 d22 ...
    mlmatsum 'lnf' 'd11' = formula for }\mp@subsup{\partial}{}{2}\operatorname{ln}\mp@subsup{\ell}{j}{}/\partial\mp@subsup{0}{1j}{2}\mathrm{ , eq(1)
    mlmatsum 'lnf' 'd12' = formula for }\mp@subsup{\partial}{}{2}\operatorname{ln}\mp@subsup{\ell}{j}{}/\partial\mp@subsup{0}{1j}{}\partial\mp@subsup{0}{2j}{}\mathrm{ , eq(1,2)
    mlmatsum 'lnf' 'd22' = formula for }\mp@subsup{\partial}{}{2}\operatorname{ln}\mp@subsup{\ell}{j}{}/\partial\mp@subsup{0}{2j}{2}\mathrm{ , eq(2)
    matrix 'H' = ('d11','d12', ... \ 'd12'','d22', ... )
end
```

where

```
'todo' contains 0, 1, or 2
                        0=>'lnf' to be filled in;
                        1=> 'lnf' and 'g' to be filled in;
                        2=>'lnf', 'g', and 'H' to be filled in
'b' full parameter row vector }\mathbf{b}=(\mp@subsup{\mathbf{b}}{1}{},\mp@subsup{\mathbf{b}}{2}{},\ldots,\mp@subsup{\mathbf{b}}{E}{}
'lnf' scalar to be filled in with overall }\operatorname{ln}
'g' row vector to be filled in with overall g=\partial ln L/\partialb
''H' matrix to be filled in with overall Hessian H=\partial' ln L/\partial\mathbf{b}\partial\mp@subsup{\mathbf{b}}{}{\prime}
```


## Method-lf0 evaluators

program progname
version 18.0 // (or version 18.5 for StataNow)
args todo b lnfj
tempvar thetal theta2...
mleval 'theta1' = 'b', eq(1)
mleval 'theta2' $=$ ' $b$ ', eq(2) // if there is a $\theta_{2}$
// if you need to create any intermediate results:
tempvar tmp1 tmp2...
generate double 'tmp1' = ...
quietly replace 'lnfj' = ...
end
where

| 'todo' | always contains 0 (may be ignored) |
| :--- | :--- |
| ' $b$ ' | full parameter row vector $\mathbf{b}=\left(\mathbf{b}_{1}, \mathbf{b}_{2}, \ldots, \mathbf{b}_{E}\right)$ |
| ' $\operatorname{lnf} \mathrm{j}$, | variable to be filled in with observation-by-observation values of $\ln \ell_{j}$ |

## Method-lf1 evaluators

program progname
version 18.0 // (or version 18.5 for StataNow)
args todo b lnfj g1 g2 ...
tempvar thetal theta2...
mleval 'theta1' = 'b', eq(1)
mleval 'theta2' $=$ ' $b$ ', eq(2) // if there is a $\theta_{2}$
// if you need to create any intermediate results:
tempvar tmp1 tmp2...
generate double 'tmp1' = ...
quietly replace ' $\operatorname{lnf} \mathrm{j}$ ' = ...
if ('todo'==0) exit
quietly replace ' g 1 ' = formula for $\partial \ln \ell_{j} / \partial \theta_{1 j}$
quietly replace ' g 2 ' $=$ formula for $\partial \ln \ell_{j} / \partial \theta_{2 j}$
end
where

```
'todo' contains 0 or 1
                        0=>'lnfj'to be filled in;
                        1=>'lnfj', 'g1', 'g2',..., to be filled in
' b
'lnfj' variable to be filled in with observation-by-observation values of }\operatorname{ln}\mp@subsup{\ell}{j}{
'g1'' variable to be filled in with }\partial\operatorname{ln}\mp@subsup{\ell}{j}{}/\partial\mp@subsup{0}{1j}{
'g2' variable to be filled in with }\partial\operatorname{ln}\mp@subsup{\ell}{j}{}/\partial\mp@subsup{0}{2j}{
```


## Method-lf2 evaluators

```
program progname
    version 18.0 // (or version 18.5 for StataNow)
    args todo b lnfj g1 g2 ... H
    tempvar thetal theta2 ...
    mleval 'theta1' = 'b', eq(1)
    mleval 'theta2' = 'b', eq(2) // if there is a }\mp@subsup{0}{2}{
    // if you need to create any intermediate results:
    tempvar tmp1 tmp2...
    generate double 'tmp1' = ...
    quietly replace 'lnfj' = ...
    if ('todo'==0) exit
    quietly replace 'g1' = formula for }\partial\operatorname{ln}\mp@subsup{\ell}{j}{}/\partial\mp@subsup{0}{1j}{
    quietly replace 'g2' = formula for }\partial\operatorname{ln}\mp@subsup{\ell}{j}{}/\partial\mp@subsup{0}{2j}{
    if ('todo'==1) exit
    tempname d11 d12 d22 lnf ...
    mlmatsum 'lnf' 'd11' = formula for }\mp@subsup{\partial}{}{2}\operatorname{ln}\mp@subsup{\ell}{j}{}/\partial\mp@subsup{0}{1j}{2}\mathrm{ , eq(1)
    mlmatsum 'lnf' 'd12' = formula for }\mp@subsup{\partial}{}{2}\operatorname{ln}\mp@subsup{\ell}{j}{}/\partial\mp@subsup{0}{1j}{}\partial\mp@subsup{0}{2j}{}\mathrm{ , eq (1,2)
    mlmatsum 'lnf' 'd22' = formula for }\mp@subsup{\partial}{}{2}\operatorname{ln}\mp@subsup{\ell}{j}{}/\partial\mp@subsup{0}{2j}{2}\mathrm{ , eq(2)
    matrix 'H' = ('d11','d12', ... \ 'd12'','d22', ... )
end
```

where

```
'todo' contains 0 or 1
    0=>'lnfj'to be filled in;
    1=>'lnfj',' 'g1', 'g2',..., to be filled in
    2=>'lnfj', 'g1', 'g2',..., and 'H' to be filled in
''b' full parameter row vector b}=(\mp@subsup{\mathbf{b}}{1}{},\mp@subsup{\mathbf{b}}{2}{},\ldots,\mp@subsup{\mathbf{b}}{E}{}
'Infj' scalar to be filled in with observation-by-observation }\operatorname{ln}
'g1' variable to be filled in with }\partial\operatorname{ln}\mp@subsup{\ell}{j}{}/\partial\mp@subsup{0}{1j}{
'g2' variable to be filled in with }\partial\operatorname{ln}\mp@subsup{\ell}{j}{}/\partial\mp@subsup{0}{2j}{
''H' matrix to be filled in with overall Hessian H=\partial ' ln L/\partial\mathbf{b}\partial\mp@subsup{\mathbf{b}}{}{\prime}
```


## Method-gf0 evaluators

```
program progname
    version 18.0 // (or version 18.5 for StataNow)
    args todo b lnfj
    tempvar thetal theta2 ...
    mleval 'theta1' = 'b', eq(1)
    mleval 'theta2' = 'b', eq(2) // if there is a }\mp@subsup{0}{2}{
    // if you need to create any intermediate results:
    tempvar tmp1 tmp2...
    generate double 'tmp1' = ...
    quietly replace 'lnfj' = ...
end
where
'todo' always contains 0 (may be ignored)
'b' full parameter row vector b=( b
' lnfj' variable to be filled in with the values of the log-likelihood }\operatorname{ln}\mp@subsup{\ell}{j}{
```


## Global macros for use by all evaluators

```
$ML_y1 name of first dependent variable
$ML_y2 name of second dependent variable, if any
$ML_samp variable containing 1 if observation to be used; 0 otherwise
$ML_w variable containing weight associated with observation or 1 if no weights specified
```

Method-lf evaluators can ignore $\$ \mathrm{ML}$ _samp, but restricting calculations to the $\$$ ML_samp==1 subsample will speed execution. Method-lf evaluators must ignore \$ML_w; application of weights is handled by the method itself.

Methods d0, d1, d2, lf0, lf1, lf2, and gf0 can ignore \$ML_samp as long as ml model's nopreserve option is not specified. These methods will run more quickly if nopreserve is specified. These evaluators can ignore \$ML_w only if they use mlsum, mlvecsum, mlmatsum, and mlmatbysum to produce all final results.

## Options

Options are presented under the following headings:

```
Options for use with ml model in interactive or noninteractive mode
Options for use with ml model in noninteractive mode
Options for use when specifying equations
Options for use with ml search
Option for use with ml plot
Options for use with ml init
Options for use with ml maximize
Option for use with ml graph
Options for use with ml display
Options for use with mleval
Option for use with mlsum
Option for use with mlvecsum
Option for use with mlmatsum
Options for use with mlmatbysum
Options for use with ml score
```


## Options for use with $\mathbf{m l}$ model in interactive or noninteractive mode

group (varname) specifies the numeric variable that identifies groups. This option is typically used to identify panels for panel-data models.
vce (vcetype) specifies the type of standard error reported, which includes types that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that are derived from asymptotic theory (oim, opg); see [R] vce_option.
vce(robust), vce(cluster clustvar), pweight, and svy will work with evaluators of methods lf, lf0, lf1, lf2, and gf0; all you need to do is specify them.
These options will not work with evaluators of methods d 0 , d 1 , or d 2 , and specifying these options will produce an error message.
constraints (numlist $\mid$ matname) specifies the linear constraints to be applied during estimation. constraints (numlist) specifies the constraints by number. Constraints are defined by using the constraint command; see $[\mathrm{R}]$ constraint. constraint (matname) specifies a matrix that contains the constraints.
nocnsnotes prevents notes from being displayed when constraints are dropped. A constraint will be dropped if it is inconsistent, contradicts other constraints, or causes some other error when the constraint matrix is being built. Constraints are checked in the order in which they are specified.
title(string) specifies the title for the estimation output when results are complete.
nopreserve specifies that ml need not ensure that only the estimation subsample is in memory when the user-written likelihood evaluator is called. nopreserve is irrelevant when you use method lf.
For the other methods, if nopreserve is not specified, ml saves the data in a file (preserves the original dataset) and drops the irrelevant observations before calling the user-written evaluator. This way, even if the evaluator does not restrict its attentions to the \$ML_samp==1 subsample, results will still be correct. Later, ml automatically restores the original dataset.
ml need not go through these machinations for method lf because the user-written evaluator calculates observation-by-observation values, and ml itself sums the components.
ml goes through these machinations if and only if the estimation sample is a subsample of the data in memory. If the estimation sample includes every observation in memory, ml does not preserve the original dataset. Thus, programmers must not alter the original dataset unless they preserve the data themselves.

We recommend that interactive users of ml not specify nopreserve; the speed gain is not worth the possibility of getting incorrect results.
We recommend that programmers specify nopreserve, but only after verifying that their evaluator really does restrict its attentions solely to the $\$$ ML_samp $==1$ subsample.
collinear specifies that ml not remove the collinear variables within equations. There is no reason to leave collinear variables in place, but this option is of interest to programmers who, in their code, have already removed collinear variables and do not want ml to waste computer time checking again.
missing specifies that observations containing variables with missing values not be eliminated from the estimation sample. There are two reasons you might want to specify missing:

Programmers may wish to specify missing because, in other parts of their code, they have already eliminated observations with missing values and do not want ml to waste computer time looking again.

You may wish to specify missing if your model explicitly deals with missing values. Stata's heckman command is a good example of this. In such cases, there will be observations where missing values are allowed and other observations where they are not-where their presence should cause the observation to be eliminated. If you specify missing, it is your responsibility to specify an if exp that eliminates the irrelevant observations.
$\operatorname{lf} 0\left(\#_{k} \#_{l l}\right)$ is typically used by programmers. It specifies the number of parameters and log-likelihood value of the constant-only model so that ml can report a likelihood-ratio test rather than a Wald test. These values may have been analytically determined, or they may have been determined by a previous fitting of the constant-only model on the estimation sample.
Also see the continue option directly below.
If you specify lf0(), it must be safe for you to specify the missing option, too, else how did you calculate the log likelihood for the constant-only model on the same sample? You must have identified the estimation sample, and done so correctly, so there is no reason for ml to waste time rechecking your results. All of which is to say, do not specify 1 ffO() unless you are certain your code identifies the estimation sample correctly.
lf0(), even if specified, is ignored if vce(robust), vce(cluster clustvar), pweight, or svy is specified because, in that case, a likelihood-ratio test would be inappropriate.
continue is typically specified by programmers and does two things:
First, it specifies that a model has just been fit by either ml or some other estimation command, such as logit, and that the likelihood value stored in e(ll) and the number of parameters stored in e(b) as of that instant are the relevant values of the constant-only model. The current value of the log likelihood is used to present a likelihood-ratio test unless vce(robust), vce (cluster clustvar), pweight, svy, or constraints() is specified. A likelihood-ratio test is inappropriate when vce(robust), vce(cluster clustvar), pweight, or svy is specified. We suggest using lrtest when constraints() is specified; see [R] Irtest.
Second, continue sets the initial values, $\mathbf{b}_{0}$, for the model about to be fit according to the $\mathrm{e}(\mathrm{b})$ currently stored.

The comments made about specifying missing with 1 fO 0() apply equally well here.
waldtest(\#) is typically specified by programmers. By default, ml presents a Wald test, but that is overridden if the lf0() or continue option is specified. A Wald test is performed if vce (robust), vce(cluster clustvar), or pweight is specified.
waldtest ( 0 ) prevents even the Wald test from being reported.
waldtest ( -1 ) is the default. It specifies that a Wald test be performed by constraining all coefficients except the intercept to 0 in the first equation. Remaining equations are to be unconstrained. A Wald test is performed if neither $\operatorname{lf0} 0()$ nor continue was specified, and a Wald test is forced if vce(robust), vce(cluster clustvar), or pweight was specified.
waldtest ( $k$ ) for $k \leq-1$ specifies that a Wald test be performed by constraining all coefficients except intercepts to 0 in the first $|k|$ equations; remaining equations are to be unconstrained. A Wald test is performed if neither $\operatorname{lfO}()$ nor continue was specified, and a Wald test is forced if vce(robust), vce(cluster clustvar), or pweight was specified.
waldtest ( $k$ ) for $k \geq 1$ works like the options above, except that it forces a Wald test to be reported even if the information to perform the likelihood-ratio test is available and even if none of vce(robust), vce(cluster clustvar), or pweight was specified. waldtest ( $k$ ), $k \geq 1$, may not be specified with 1 fO ().
obs(\#) is used mostly by programmers. It specifies that the number of observations reported and ultimately stored in $e(N)$ be \#. Ordinarily, ml works that out for itself. Programmers may want to specify this option when, for the likelihood evaluator to work for $N$ observations, they first had to modify the dataset so that it contained a different number of observations.
crittype (string) is used mostly by programmers. It allows programmers to supply a string (up to 32 characters long) that describes the criterion that is being optimized by ml. The default is "Log likelihood" for nonrobust and "Log pseudolikelihood" for robust estimation.
svy indicates that ml is to pick up the svy settings set by svyset and use the robust variance estimator. This option requires the data to be svyset; see [SVY] svyset. svy may not be specified with vce() or weights.
subpop(varname) specifies that estimates be computed for the single subpopulation defined by the observations for which varname $\neq 0$. Typically, varname $=1$ defines the subpopulation, and varname $=0$ indicates observations not belonging to the subpopulation. For observations whose subpopulation status is uncertain, varname should be set to missing ('.'). This option requires the svy option.
nosvyadjust specifies that the model Wald test be carried out as $W / k \sim F(k, d)$, where $W$ is the Wald test statistic, $k$ is the number of terms in the model excluding the constant term, $d$ is the total number of sampled PSUs minus the total number of strata, and $F(k, d)$ is an $F$ distribution with $k$ numerator degrees of freedom and $d$ denominator degrees of freedom. By default, an adjusted Wald test is conducted: $(d-k+1) W /(k d) \sim F(k, d-k+1)$. See Korn and Graubard (1990) for a discussion of the Wald test and the adjustments thereof. This option requires the svy option.
technique (algorithm_spec) specifies how the likelihood function is to be maximized. The following algorithms are currently implemented in ml. For details, see Pitblado, Poi, and Gould (2024).
technique (nr) specifies Stata's modified Newton-Raphson (NR) algorithm.
technique (bhhh) specifies the Berndt-Hall-Hall-Hausman (BHHH) algorithm.
technique (dfp) specifies the Davidon-Fletcher-Powell (DFP) algorithm.
technique (bfgs) specifies the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm.
The default is technique ( nr ).
You can switch between algorithms by specifying more than one in the technique() option. By default, ml will use an algorithm for five iterations before switching to the next algorithm. To specify a different number of iterations, include the number after the technique in the option. For example, technique (bhhh 10 nr 1000 ) requests that ml perform 10 iterations using the BHHH algorithm, followed by 1,000 iterations using the NR algorithm, and then switch back to BHHH for

10 iterations, and so on. The process continues until convergence or until reaching the maximum number of iterations.
deriv(min_spec) specifies whether to use minimum values of step sizes for computing numerical derivatives and optionally specifies the minimum values.
deriv(nomin), the default, suppresses the use of minimum values.
deriv(min) sets the minimum step size to $1 \mathrm{e}-6$ for computing first-order derivatives and $1 \mathrm{e}-4$ for computing second-order derivatives.
deriv( $\left.\min \left(\#_{1}, \#_{2}\right)\right)$ sets the minimum step size to $\#_{1}$ for computing the first-order derivatives and $\#_{2}$ for computing the second-order derivatives. Specifying deriv(min(1e-6, 1e-4)) is equivalent to specifying deriv (min).
deriv(min(matname)) sets the minimum step size to the values in the $1 \times 2$ matrix matname. The value in the first column of the row vector will set the minimum step size for the firstorder derivatives, and the value in the second column will set the minimum step size for the second-order derivatives.

## Options for use with ml model in noninteractive mode

The following extra options are for use with ml model in noninteractive mode. Noninteractive mode is for programmers who use ml as a subroutine and want to issue one command that will carry forth the estimation from start to finish.
maximize is required. It specifies noninteractive mode.
init (ml_init_args) sets the initial values, $\mathbf{b}_{0}$. ml_init_args are whatever you would type after the ml init command.
search (on | norescale| quietly|off) specifies whether ml search is to be used to improve the initial values. search(on) is the default and is equivalent to separately running ml search, repeat (0). search (norescale) is equivalent to separately running ml search, repeat (0) norescale. search (quietly) is equivalent to search(on), except that it suppresses ml search's output. search (off) prevents calling ml search.
repeat (\#) is ml search's repeat() option. repeat(0) is the default.
bounds ( $m l$ _search_bounds) specifies the search bounds. ml_search_bounds is specified as
[eqn_name] lower_bound upper_bound ... [eqn_name] lower_bound upper_bound for instance, bounds (100 100 lnsigma 010 ). The ml model command issues ml search ml_search_bounds, repeat(\#). Specifying search bounds is optional.
nowarning, novce, negh, and score() are ml maximize's equivalent options.
maximize_options: difficult, technique (algorithm_spec), iterate (\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#),
nrtolerance(\#), nonrtolerance, and from(init_specs); see [R] Maximize. These options are seldom used.

## Options for use when specifying equations

noconstant specifies that the equation not include an intercept.
offset $\left(\right.$ varname $\left._{o}\right)$ specifies that the equation be $\mathbf{x b}+$ varname $_{o}$-that it include varname ${ }_{o}$ with coefficient constrained to be 1 .
exposure ( varname $_{e}$ ) is an alternative to offset (varname ${ }_{o}$ ); it specifies that the equation be $\mathbf{x b}+\ln \left(\right.$ varname $\left._{e}\right)$. The equation is to include $\ln \left(\right.$ varname $\left._{e}\right)$ with coefficient constrained to be 1 .
freeparm specifies that the associated eqname is a free parameter. The corresponding full column name on $\mathrm{e}(\mathrm{b})$ will be /eqname instead of eqname :_cons. This option is not allowed with varlist $_{x}$.

## Options for use with ml search

repeat (\#) specifies the number of random attempts that are to be made to find a better initial-value vector. The default is repeat (10).
repeat ( 0 ) specifies that no random attempts be made. More precisely, repeat (0) specifies that no random attempts be made if the first initial-value vector is a feasible starting point. If it is not, ml search will make random attempts, even if you specify repeat(0), because it has no alternative. The repeat () option refers to the number of random attempts to be made to improve the initial values. When the initial starting value vector is not feasible, ml search will make up to 1,000 random attempts to find starting values. It stops when it finds one set of values that works and then moves into its improve-initial-values logic.
repeat $(k), k>0$, specifies the number of random attempts to be made to improve the initial values.
restart specifies that random actions be taken to obtain starting values and that the resulting starting values not be a deterministic function of the current values. Generally, you should not specify this option because, with restart, ml search intentionally does not produce as good a set of starting values as it could. restart is included for use by the optimizer when it gets into serious trouble. The random actions ensure that the optimizer and ml search, working together, do not cause an endless loop.
restart implies norescale, which is why we recommend that you do not specify restart. In testing, sometimes rescale worked so well that, even after randomization, the rescaler would bring the starting values right back to where they had been the first time and thus defeat the intended randomization.
norescale specifies that ml search not engage in its rescaling actions to improve the parameter vector. We do not recommend specifying this option because rescaling tends to work so well.
maximize_options: [no]log and trace; see [R] Maximize. These options are seldom used.

## Option for use with ml plot

saving (filename[, replace]) specifies that the graph be saved in filename.gph. See [G-3] saving_option.

## Options for use with ml init

copy specifies that the list of numbers or the initialization vector be copied into the initial-value vector by position rather than by name.
skip specifies that any parameters found in the specified initialization vector that are not also found in the model be ignored. The default action is to issue an error message.

## Options for use with ml maximize

nowarning is allowed only with iterate(0). nowarning suppresses the "convergence not achieved" message. Programmers might specify iterate(0) nowarning when they have a vector $\mathbf{b}$ already containing the final estimates and want ml to calculate the variance matrix and postestimation results. Then, specify init(b) search(off) iterate(0) nowarning nolog.
novce is allowed only with iterate ( 0 ) . novce substitutes the zero matrix for the variance matrix, which in effect posts estimation results as fixed constants.
negh indicates that the evaluator returns the negative Hessian matrix. By default, ml assumes d2 and lf2 evaluators return the Hessian matrix.
score (newvars $\mid$ stub*) creates new variables containing the contributions to the score for each equation and ancillary parameter in the model; see [U] 20.23 Obtaining scores.
If score(newvars) is specified, the newvars must contain $k$ new variables. For evaluators of methods lf, lf0, lf1, and lf2, $k$ is the number of equations. For evaluators of method gf0, $k$ is the number of parameters. If score (stub*) is specified, variables named stub1, stub2, ..., stubk are created.

For evaluators of methods lf, lf0, lf1, and lf2, the first variable contains $\partial \ln \ell_{j} / \partial\left(\mathbf{x}_{1 j} \mathbf{b}_{1}\right)$, the second variable contains $\partial \ln \ell_{j} / \partial\left(\mathbf{x}_{2 j} \mathbf{b}_{2}\right)$, and so on.
For evaluators of method gf0, the first variable contains $\partial \ln \ell_{j} / \partial \mathbf{b}_{1}$, the second variable contains $\partial \ln \ell_{j} / \partial \mathbf{b}_{2}$, and so on.
nooutput suppresses display of results. This option is different from prefixing ml maximize with quietly in that the iteration log is still displayed (assuming that nolog is not specified).
noskipline suppresses display of a blank line before the iteration log. This is useful in programs when there is a header preceding the iteration $\log$ and a blank line is not wanted after the header.
noclear specifies that the ml problem definition not be cleared after the model has converged. Perhaps you are having convergence problems and intend to run the model to convergence. If so, use ml search to see if those values can be improved, and then restart the estimation.
maximize_options: difficult, iterate(\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), and nonrtolerance; see [R] Maximize. These options are seldom used.
display_options; see Options for use with ml display below.
eform_option; see Options for use with ml display below.

## Option for use with ml graph

saving (filename[, replace]) specifies that the graph be saved in filename.gph. See [G-3] saving_option.

## Options for use with ml display

noheader suppresses the header display above the coefficient table that displays the final log-likelihood value, the number of observations, and the model significance test.
nofootnote suppresses the footnote display below the coefficient table, which displays a warning if the model fit did not converge within the specified number of iterations. Use ml footnote to display the warning if 1) you add to the coefficient table using the plus option or 2) you have your own footnotes and want the warning to be last.
level(\#) is the standard confidence-level option. It specifies the confidence level, as a percentage, for confidence intervals of the coefficients. The default is level (95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals.
first displays a coefficient table reporting results for the first equation only, and the report makes it appear that the first equation is the only equation. This option is used by programmers who estimate ancillary parameters in the second and subsequent equations and who wish to report the values of such parameters themselves.
neq(\#) is an alternative to first. neq(\#) displays a coefficient table reporting results for the first \# equations. This option is used by programmers who estimate ancillary parameters in the \# + 1 and subsequent equations and who wish to report the values of such parameters themselves.
showeqns is a seldom-used option that displays the equation names in the coefficient table. ml display uses the numbers stored in $e\left(k_{-} e q\right)$ and $e\left(k \_a u x\right)$ to determine how to display the coefficient table. e(k_eq) identifies the number of equations, and e( $k$ _aux) identifies how many of these are for ancillary parameters. The first option is implied when showeqns is not specified and all but the first equation are for ancillary parameters.
plus displays the coefficient table, but rather than ending the table in a line of dashes, ends it in dashes-plus-sign-dashes. This is so that programmers can write additional display code to add more results to the table and make it appear as if the combined result is one table. Programmers typically specify plus with the first or neq() options. This option implies nofootnote.
nocnsreport suppresses the display of constraints above the coefficient table. This option is ignored if constraints were not used to fit the model.
noomitted specifies that variables that were omitted because of collinearity not be displayed. The default is to include in the table any variables omitted because of collinearity and to label them as "(omitted)".
vsquish specifies that the blank space separating factor-variable terms or time-series-operated variables from other variables in the model be suppressed.
noemptycells specifies that empty cells for interactions of factor variables not be displayed. The default is to include in the table interaction cells that do not occur in the estimation sample and to label them as "(empty)".
baselevels and allbaselevels control whether the base levels of factor variables and interactions are displayed. The default is to exclude from the table all base categories.
baselevels specifies that base levels be reported for factor variables and for interactions whose bases cannot be inferred from their component factor variables.
allbaselevels specifies that all base levels of factor variables and interactions be reported.
cformat ( $\%$ fmt) specifies how to format coefficients, standard errors, and confidence limits in the coefficient table.
pformat (\%fmt) specifies how to format $p$-values in the coefficient table.
sformat ( $\%$ fmt) specifies how to format test statistics in the coefficient table.
nolstretch specifies that the width of the coefficient table not be automatically widened to accommodate longer variable names. The default, lstretch, is to automatically widen the coefficient table up to the width of the Results window. Specifying lstretch or nolstretch overrides the setting given by set lstretch. If set lstretch has not been set, the default is lstretch.
coeflegend specifies that the legend of the coefficients and how to specify them in an expression be displayed rather than displaying the statistics for the coefficients.
eform_option: eform(string), eform, hr, shr, irr, or, and rrr display the coefficient table in exponentiated form: for each coefficient, $\exp (b)$ rather than $b$ is displayed, and standard errors and confidence intervals are transformed. string is the table header that will be displayed above the transformed coefficients and must be 11 characters or shorter in length-for example, eform("Odds ratio"). The options eform, hr, shr, irr, or, and rrr provide a default string equivalent to "exp(b)", "Haz. ratio", "SHR", "IRR", "Odds ratio", and "RRR", respectively. These options may not be combined.
ml display looks at e (k_eform) to determine how many equations are affected by an eform_option; by default, only the first equation is affected. Type ereturn list, all to view e(k_eform); see [P] ereturn.

## Options for use with mleval

eq(\#) specifies the equation number, $i$, for which $\theta_{i j}=\mathbf{x}_{i j} \mathbf{b}_{i}$ is to be evaluated. eq(1) is assumed if eq() is not specified.
scalar asserts that the $i$ th equation is known to evaluate to a constant, meaning that the equation was specified as (), (name:), or /name on the ml model statement. If you specify this option, the new variable created is created as a scalar. If the $i$ th equation does not evaluate to a scalar, an error message is issued.

## Option for use with mlsum

noweight specifies that weights (\$ML_w) be ignored when summing the likelihood function.

## Option for use with mlvecsum

eq(\#) specifies the equation for which a gradient vector $\partial \ln L / \partial \mathbf{b}_{i}$ is to be constructed. The default is eq(1).

## Option for use with mlmatsum

eq(\#[,\#]) specifies the equations for which the Hessian matrix is to be constructed. The default is eq (1), which is the same as eq $(1,1)$, which means $\partial^{2} \ln L / \partial \mathbf{b}_{1} \partial \mathbf{b}_{1}^{\prime}$. Specifying eq $(i, j)$ results in $\partial^{2} \ln L / \partial \mathbf{b}_{i} \partial \mathbf{b}_{j}^{\prime}$.

## Options for use with mimatbysum

by (varname) is required and specifies the group variable.
eq (\#[,\#]) specifies the equations for which the Hessian matrix is to be constructed. The default is eq (1), which is the same as eq $(1,1)$, which means $\partial^{2} \ln L / \partial \mathbf{b}_{1} \partial \mathbf{b}_{1}^{\prime}$. Specifying eq $(i, j)$ results in $\partial^{2} \ln L / \partial \mathbf{b}_{i} \partial \mathbf{b}_{j}^{\prime}$.

## Options for use with ml score

equation(eqname) identifies from which equation the observation scores are to come. This option may be used only when generating one variable.
missing specifies that observations containing variables with missing values not be eliminated from the estimation sample.

## Remarks and examples

For a thorough discussion of ml, see the fifth edition of Maximum Likelihood Estimation with Stata (Pitblado, Poi, and Gould 2024). The book provides a tutorial introduction to ml , notes on advanced programming issues, and a discourse on maximum likelihood estimation from both theoretical and practical standpoints. See Survey options and ml at the end of Remarks and examples for examples of the new svy options. For more information about survey estimation, see [SVY] Survey, [SVY] svy estimation, and [SVY] Variance estimation.
ml requires that you write a program that evaluates the log-likelihood function and, possibly, its first and second derivatives. The style of the program you write depends upon the method you choose. Methods lf, lf0, d0, and gf0 require that your program evaluate the log likelihood only. Methods d1 and lf1 require that your program evaluate the log likelihood and its first derivatives. Methods d2 and lf2 requires that your program evaluate the log likelihood and its first and second derivatives. Methods lf, lf0, d0, and gf0 differ from each other in that, with methods if and lf0, your program is required to produce observation-by-observation $\log$-likelihood values $\ln \ell_{j}$ and it is assumed that $\ln L=\sum_{j} \ln \ell_{j}$; with method d0, your program is required to produce only the overall value $\ln L$; and with method gf0, your program is required to produce the summable pieces of the log likelihood, such as those in panel-data models.

Once you have written the program-called an evaluator-you define a model to be fit using ml model and obtain estimates using ml maximize. You might type

```
. ml model ...
. ml maximize
```

but we recommend that you type

```
. ml model ...
. ml check
. ml search
. ml maximize
```

ml check verifies your evaluator has no obvious errors, and ml search finds better initial values.
You fill in the ml model statement with 1) the method you are using, 2) the name of your program, and 3) the "equations". You write your evaluator in terms of $\theta_{1}, \theta_{2}, \ldots$, each of which has a linear equation associated with it. That linear equation might be as simple as $\theta_{i}=b_{0}$, it might be $\theta_{i}=b_{1} \mathrm{mpg}+b_{2}$ weight $+b_{3}$, or it might omit the intercept $b_{3}$. The equations are specified in parentheses on the ml model line.

Suppose that you are using method lf and the name of your evaluator program is myprog. The statement

```
. ml model lf myprog (mpg weight)
```

would specify one equation with $\theta_{i}=b_{1} \mathrm{mpg}+b_{2}$ weight $+b_{3}$. If you wanted to omit $b_{3}$, you would type
. ml model lf myprog (mpg weight, nocons)
and if all you wanted was $\theta_{i}=b_{0}$, you would type
. ml model lf myprog ()
With multiple equations, you list the equations one after the other; so, if you typed

```
. ml model lf myprog (mpg weight) ()
```

you would be specifying $\theta_{1}=b_{1} \mathrm{mpg}+b_{2}$ weight $+b_{3}$ and $\theta_{2}=b_{4}$. You would write your likelihood in terms of $\theta_{1}$ and $\theta_{2}$. If the model was linear regression, $\theta_{1}$ might be the $\mathbf{x b}$ part and $\theta_{2}$ the variance of the residuals.

When you specify the equations, you also specify any dependent variables. If you typed

```
. ml model lf myprog (price = mpg weight) ()
```

price would be the one and only dependent variable, and that would be passed to your program in $\$ M L-y 1$. If your model had two dependent variables, you could type

```
. ml model lf myprog (price displ = mpg weight) ()
```

Then, \$ML_y1 would be price and \$ML_y2 would be displ. You can specify however many dependent variables are necessary and specify them on any equation. It does not matter on which equation you specify them; the first one specified is placed in \$ML_y1, the second in \$ML_y2, and so on.

## > Example 1: Method If

Using method lf, we want to produce observation-by-observation values of the log likelihood. The probit log-likelihood function is

$$
\begin{aligned}
\ln \ell_{j} & = \begin{cases}\ln \Phi\left(\theta_{1 j}\right) & \text { if } y_{j}=1 \\
\ln \Phi\left(-\theta_{1 j}\right) & \text { if } y_{j}=0\end{cases} \\
\theta_{1 j} & =\mathbf{x}_{j} \mathbf{b}_{1}
\end{aligned}
$$

The following is the method-lf evaluator for this likelihood function:

```
program myprobit
    version 18.0 // (or version 18.5 for StataNow)
    args lnf theta1
    quietly replace 'lnf' = ln(normal('theta1')) if $ML_y1==1
    quietly replace 'lnf' = ln(normal(-'theta1')) if $ML_y1==0
end
```

If we wanted to fit a model of foreign on mpg and weight, we would type the following commands. The 'foreign =' part specifies that $y$ is foreign. The 'mpg weight' part specifies that $\theta_{1 j}=b_{1} \mathrm{mpg}_{j}+b_{2}$ weight $_{j}+b_{3}$.

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. ml model lf myprobit (foreign = mpg weight)
. ml maximize
Initial: Log likelihood = -51.292891
Alternative: Log likelihood = -45.055272
Rescale: Log likelihood = -45.055272
Iteration 0: Log likelihood = -45.055272
Iteration 1: Log likelihood = -27.904125
Iteration 2: Log likelihood = -26.858643
Iteration 3: Log likelihood = -26.844199
Iteration 4: Log likelihood = -26.844189
Iteration 5: Log likelihood = -26.844189
```

Number of obs $=74$
Wald chi2(2) $=20.75$
Prob > chi2 $=0.0000$

Log likelihood = -26.844189

| foreign | Coefficient | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | ---: | ---: |
| mpg | -.1039503 | .0515689 | -2.02 | 0.044 | -.2050235 | -.0028772 |
| weight | -.0023355 | .0005661 | -4.13 | 0.000 | -.003445 | -.0012261 |
| _cons | 8.275464 | 2.554142 | 3.24 | 0.001 | 3.269438 | 13.28149 |

## Example 2: Method If for two-equation, two-dependent-variable model

A two-equation, two-dependent-variable model is a little different. Rather than receiving one $\theta$, our program will receive two. Rather than there being one dependent variable in \$ML_y1, there will be dependent variables in \$ML_y1 and \$ML_y2. For instance, the Weibull regression log-likelihood function is

$$
\begin{aligned}
\ln \ell_{j} & =-\left(t_{j} e^{-\theta_{1 j}}\right) \exp \left(\theta_{2 j}\right) \\
\theta_{1 j} & =d_{j}\left\{\theta_{2 j}-\theta_{1 j}+\left(e^{\theta_{2 j}}-1\right)\left(\ln t_{j}-\theta_{1 j}\right)\right\} \\
\theta_{2 j} & =s
\end{aligned}
$$

where $t_{j}$ is the time of failure or censoring and $d_{j}=1$ if failure and 0 if censored. We can make the log likelihood a little easier to program by introducing some extra variables:

$$
\begin{aligned}
p_{j} & =\exp \left(\theta_{2 j}\right) \\
M_{j} & =\left\{t_{j} \exp \left(-\theta_{1 j}\right)\right\}^{p_{j}} \\
R_{j} & =\ln t_{j}-\theta_{1 j} \\
\ln \ell_{j} & =-M_{j}+d_{j}\left\{\theta_{2 j}-\theta_{1 j}+\left(p_{j}-1\right) R_{j}\right\}
\end{aligned}
$$

The method-lf evaluator for this is

```
program myweib
    version 18.0 // (or version 18.5 for StataNow)
    args lnf theta1 theta2
    tempvar p M R
    quietly generate double 'p' = exp('theta2')
    quietly generate double 'M' = ($ML_y1*exp(-'theta1')) ''p'
    quietly generate double ' }R\mathrm{ ' = ln($ML_y1)-'theta1'
    quietly replace 'lnf' = -'M' + $ML_y2*('theta2'-'theta1' + ('p'-1)*'R')
end
```

We can fit a model by typing

```
. ml model lf myweib (studytime died = i.drug age) ()
. ml maximize
```

Note that we specified '()' for the second equation. The second equation corresponds to the Weibull shape parameter $s$, and the linear combination we want for $s$ contains just an intercept. Alternatively, we could type

```
. ml model lf myweib (studytime died = i.drug age) /s
```

Typing /s means the same thing as typing (s:, freeparm). The s, either after a slash or in parentheses before a colon, labels the equation. The leading slash, /, is a shortcut for specifying that " $s$ " is a free parameter. Free parameters are labeled using this shortcut notation, /s, in the column names of $e(b)$ and in the estimation results. If we instead specified ( $s:$ ), then $s$ would be the equation label for a constant linear equation and would be labeled as $s$ :_cons in the column names of $e(b)$ and in the estimation results.

```
. use https://www.stata-press.com/data/r18/cancer, clear
(Patient survival in drug trial)
. ml model lf myweib (studytime died = i.drug age) /s
. ml maximize
Initial: Log likelihood = -744
Alternative: Log likelihood = -356.14276
Rescale: Log likelihood = -200.80201
Rescale eq: Log likelihood = -136.69232
Iteration 0: Log likelihood = -136.69232 (not concave)
Iteration 1: Log likelihood = -124.11726
Iteration 2: Log likelihood = -113.9591
Iteration 3: Log likelihood = -110.30683
Iteration 4: Log likelihood = -110.26748
Iteration 5: Log likelihood = -110.26736
Iteration 6: Log likelihood = -110.26736
```

| Number of obs | $=$ | 48 |
| :--- | :--- | ---: |
| Wald chi2(3) | $=35.25$ |  |
| Prob $>\operatorname{chi2}$ | $=0.0000$ |  |

Log likelihood = -110.26736 Prob > chi2 = 0.0000

|  | Coefficient | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | :--- | :---: | :---: | ---: | ---: |
| drug |  |  |  |  |  |  |
| Other | 1.012966 | .2903917 | 3.49 | 0.000 | .4438086 | 1.582123 |
| NA | 1.45917 | .2821195 | 5.17 | 0.000 | .9062261 | 2.012114 |
| age | -.0671728 | .0205688 | -3.27 | 0.001 | -.1074868 | -.0268587 |
| _cons | 6.060723 | 1.152845 | 5.26 | 0.000 | 3.801188 | 8.320259 |
| $/ \mathrm{s}$ | .5573333 | .1402154 | 3.97 | 0.000 | .2825162 | .8321504 |

## > Example 3: Method d0

Method-d0 evaluators receive $\mathbf{b}=\left(\mathbf{b}_{1}, \mathbf{b}_{\mathbf{2}}, \ldots, \mathbf{b}_{E}\right)$, the coefficient vector, rather than the already evaluated $\theta_{1}, \theta_{2}, \ldots, \theta_{E}$, and they are required to evaluate the overall $\log$-likelihood $\ln L$ rather than $\ln \ell_{j}, j=1, \ldots, N$.

Use mleval to produce the thetas from the coefficient vector.
Use mlsum to sum the components that enter into $\ln L$.

In the case of Weibull, $\ln L=\sum \ln \ell_{j}$, and our method-d0 evaluator is

```
program weib0
    version 18.0 // (or version 18.5 for StataNow)
    args todo b lnf
    tempvar theta1 theta2
    mleval 'theta1' = 'b', eq(1)
    mleval 'theta2' = 'b', eq(2)
    local t "$ML_y1" // this is just for readability
    local d "$ML_y2"
    tempvar p M R
    quietly generate double 'p' = exp('theta2')
    quietly generate double 'M' = ('t'*exp(-'theta1')) ''p'
    quietly generate double 'R' = ln('t')-'theta1'
    mlsum 'lnf' = -'M' + 'd'*('theta2'-'theta1' + ('p'-1)*'R')
end
```

To fit our model using this evaluator, we would type

```
. ml model dO weibO (studytime died = i.drug age) /s
. ml maximize
```


## - Technical note

Method d0 does not require $\ln L=\sum_{j} \ln \ell_{j}, j=1, \ldots, N$, as method if does. Your likelihood function might have independent components only for groups of observations. Panel-data estimators have a $\log$-likelihood value $\ln L=\sum_{i} \ln L_{i}$, where $i$ indexes the panels, each of which contains multiple observations. Conditional logistic regression has $\ln L=\sum_{k} \ln L_{k}$, where $k$ indexes the risk pools. Cox regression has $\ln L=\sum_{(t)} \ln L_{(t)}$, where $(t)$ denotes the ordered failure times.

To evaluate such likelihood functions, first calculate the within-group log-likelihood contributions. This usually involves generate and replace statements prefixed with by, as in

```
tempvar sumd
by group: generate double 'sumd' = sum($ML_y1)
```

Structure your code so that the log-likelihood contributions are recorded in the last observation of each group. Say that a variable is named 'cont'. To sum the contributions, code

```
tempvar last
quietly by group: generate byte 'last' = (_n==_N)
mlsum 'lnf' = 'cont' if 'last'
```

You must inform mlsum which observations contain log-likelihood values to be summed. First, you do not want to include intermediate results in the sum. Second, mlsum does not skip missing values. Rather, if mlsum sees a missing value among the contributions, it sets the overall result, 'lnf', to missing. That is how ml maximize is informed that the likelihood function could not be evaluated at the particular value of $\mathbf{b} . \mathrm{ml}$ maximize will then take action to escape from what it thinks is an infeasible area of the likelihood function.

When the likelihood function violates the linear-form restriction $\ln L=\sum_{j} \ln \ell_{j}, j=1, \ldots, N$, with $\ln \ell_{j}$ being a function solely of values within the $j$ th observation, use method d0. In the following examples, we will demonstrate methods d 1 and d 2 with likelihood functions that meet this linear-form restriction. The d1 and d2 methods themselves do not require the linear-form restriction, but the utility routines mlvecsum and mlmatsum do. Using method d1 or d2 when the restriction is violated is difficult; however, mlmatbysum may be of some help for method-d2 evaluators.

## > Example 4: Method d1

Method-d1 evaluators are required to produce the gradient vector $\mathbf{g}=\partial \ln L / \partial \mathbf{b}$, as well as the overall log-likelihood value. Using mlvecsum, we can obtain $\partial \ln L / \partial \mathbf{b}$ from $\partial \ln L / \partial \theta_{i}, i=$ $1, \ldots, E$. The derivatives of the Weibull log-likelihood function are

$$
\begin{aligned}
\frac{\partial \ln \ell_{j}}{\partial \theta_{1 j}} & =p_{j}\left(M_{j}-d_{j}\right) \\
\frac{\partial \ln \ell_{j}}{\partial \theta_{2 j}} & =d_{j}-R_{j} p_{j}\left(M_{j}-d_{j}\right)
\end{aligned}
$$

The method-d1 evaluator for this is

```
program weib1
    version 18.0 // (or version 18.5 for StataNow)
    args todo b lnf g // g is new
    tempvar t1 t2
    mleval 't1' = 'b', eq(1)
    mleval 't2' = 'b', eq(2)
    local t "$ML_y1"
    local d "$ML_y2"
    tempvar p M R
    quietly generate double 'p' = exp('t2')
    quietly generate double 'M' = ('t'*exp(-'t1')) ^'p'
    quietly generate double 'R' = ln('t')-'t1'
    mlsum 'lnf' = -'M' + 'd'*('t2'-'t1' + ('p'-1)*'R')
    if ('todo'==0 | 'lnf'>=.) exit /* <-- new */
    tempname d1 d2 /* <-- new */
    mlvecsum 'lnf' 'd1' = 'p'*('M'-'d'), eq(1) /* <-- new */
    mlvecsum 'lnf' 'd2' = 'd' - 'R'*'p'*('M'-'d'), eq(2) /* <-- new */
    matrix 'g' = ('d1','d2') /* <-- new */
end
```

We obtained this code by starting with our method-d0 evaluator and then adding the extra lines that method d1 requires. To fit our model using this evaluator, we could type

```
. ml model d1 weib1 (studytime died = drug2 drug3 age) /s
```

. ml maximize
but we recommend substituting method d1debug for method d1 and typing

```
. ml model d1debug weib1 (studytime died = drug2 drug3 age) /s
. ml maximize
```

Method d1debug will compare the derivatives we calculate with numerical derivatives and thus verify that our program is correct. Once we are certain the program is correct, then we would switch from method d1debug to method d1.

## Example 5: Method d2

Method-d2 evaluators are required to produce $\mathbf{H}=\partial^{2} \ln L / \partial \mathbf{b} \partial \mathbf{b}^{\prime}$, the Hessian matrix, as well as the gradient and log-likelihood value. mlmatsum will help calculate $\partial^{2} \ln L / \partial \mathbf{b} \partial \mathbf{b}^{\prime}$ from the second derivatives with respect to $\theta$. For the Weibull model, these second derivatives are

$$
\begin{aligned}
\frac{\partial^{2} \ln \ell_{j}}{\partial \theta_{1 j}^{2}} & =-p_{j}^{2} M_{j} \\
\frac{\partial^{2} \ln \ell_{j}}{\partial \theta_{1 j} \partial \theta_{2 j}} & =p_{j}\left(M_{j}-d_{j}+R_{j} p_{j} M_{j}\right) \\
\frac{\partial^{2} \ln \ell_{j}}{\partial \theta_{2 j}^{2}} & =-p_{j} R_{j}\left(R_{j} p_{j} M_{j}+M_{j}-d_{j}\right)
\end{aligned}
$$

The method-d2 evaluator is

```
program weib2
    version 18.0 // (or version 18.5 for StataNow)
    args todo b lnf g H // H added
    tempvar t1 t2
    mleval 't1' = 'b', eq(1)
    mleval 't2' = 'b', eq(2)
    local t "$ML_y1"
    local d "$ML_y2"
    tempvar p M R
    quietly generate double 'p' = exp('t2')
    quietly generate double 'M' = ('t'*exp(-'t1')) -'p'
    quietly generate double 'R' = ln('t')-'t1'
    mlsum 'lnf' = -'M' + 'd'*('t2'-'t1' + ('p'-1)*'R')
    if ('todo'==0 | 'lnf'>=.) exit
    tempname d1 d2
    mlvecsum 'lnf' 'd1' = 'p'*('M'-'d'), eq(1)
    mlvecsum 'lnf' 'd2' = 'd' - 'R'*'p'*('M'-'d'), eq(2)
    matrix 'g' = ('d1','d2')
    if ('todo'==1 | 'lnf'>=.) exit // new from here down
    tempname d11 d12 d22
    mlmatsum 'lnf' 'd11' = -'p'^2 * 'M', eq(1)
    mlmatsum 'lnf' 'd12' = 'p'*('M'-'d' + 'R'*'p'*'M'), eq(1,2)
    mlmatsum 'lnf, 'd22' = -'p'*'R'*('R'*'p'*'M' + 'M' - 'd') , eq(2)
    matrix 'H' = ('d11','d12' \ 'd12'','d22')
end
```

We started with our previous method-d1 evaluator and added the lines that method d2 requires. We could now fit a model by typing
. ml model d2 weib2 (studytime died = drug2 drug3 age) /s
. ml maximize
but we would recommend substituting method d2debug for method d2 and typing
. ml model d2debug weib2 (studytime died = drug2 drug3 age) /s
. ml maximize
Method d2debug will compare the first and second derivatives we calculate with numerical derivatives and thus verify that our program is correct. Once we are certain the program is correct, then we would switch from method d2debug to method d2.

As we stated earlier, to produce the robust variance estimator with method lf, there is nothing to do except specify vce(robust), vce(cluster clustvar), or pweight. For methods d0, d1, and d2, these options do not work. If your likelihood function meets the linear-form restrictions, you can use methods lf0, lf1, and lf2, then these options will work. The equation scores are defined as

$$
\frac{\partial \ln \ell_{j}}{\partial \theta_{1 j}}, \frac{\partial \ln \ell_{j}}{\partial \theta_{2 j}}, \ldots
$$

Your evaluator will be passed variables, one for each equation, which you fill in with the equation scores. For both method lf1 and lf2, these variables are passed in the fourth and subsequent positions of the argument list. That is, you must process the arguments as

```
args todo b lnf g1 g2 ... H
```

Note that for method lf1, the ' H ' argument is not used and can be ignored.

## > Example 6: Robust variance estimates

If you have used mlvecsum in your evaluator of method d1 or d2, it is easy to turn it into an evaluator of method lf1 or lf2 that allows the computation of the robust variance estimator. The expression that you specified on the right-hand side of mlvecsum is the equation score.

Here we turn the program that we gave earlier in the method-d1 example into a method-lf1 evaluator that allows vce(robust), vce(cluster clustvar), or pweight.

```
program weib1
    version 18.0 // (or version 18.5 for StataNow)
    args todo b lnfj g1 g2 // g1 and g2 are new
    tempvar t1 t2
    mleval 't1' = 'b', eq(1)
    mleval 't2' = 'b', eq(2)
    local t "$ML_y1"
    local d "$ML_y2"
    tempvar p M R
    quietly generate double 'p' = exp('t2')
    quietly generate double ' M' = ('t'*exp(-'t1')) ^'p'
    quietly generate double 'R' = ln('t')-'t1'
    quietly replace 'lnfj' = -'M' + 'd'*('t2'-'t1' + ('p'-1)*'R')
    if ('todo'==0) exit
    quietly replace 'g1' = 'p'*('M'-'d') /* <-- new */
    quietly replace 'g2' = 'd' - 'R'*'p'*('M'-'d') /* <-- new */
end
```

To fit our model and get the robust variance estimates, we type

```
. ml model lf1 weib1 (studytime died = drug2 drug3 age) /s, vce(robust)
. ml maximize
```


## Survey options and ml

ml can handle stratification, poststratification, multiple stages of clustering, and finite population corrections. Specifying the svy option implies that the data come from a survey design and also implies that the survey linearized variance estimator is to be used; see [SVY] Variance estimation.

## $>$ Example 7

Suppose that we are interested in a probit analysis of data from a survey in which q1 is the answer to a yes/no question and x 1 , x 2 , x 3 are demographic responses. The following is a lf 2 evaluator for the probit model that meets the requirements for vce(robust) (linear form and computes the scores).

```
program mylf2probit
    version 18.0 // (or version 18.5 for StataNow)
    args todo b lnfj g1 H
    tempvar z Fz lnf
    mleval 'z' = 'b'
    quietly generate double 'Fz' = normal( 'z') if $ML_y1 == 1
    quietly replace 'Fz' = normal(-'z') if $ML_y1 == 0
    quietly replace ''lnfj' = log('Fz')
    if ('todo'==0) exit
    quietly replace 'g1' = normalden('z')/'Fz' if $ML_y1 == 1
    quietly replace 'g1' = -normalden('z')/'Fz' if $ML_y1 == 0
    if ('todo'==1) exit
    mlmatsum 'lnf' 'H' = -'g1'*('g1'+'z'), eq(1,1)
end
```

To fit a model, we svyset the data, then use svy with ml .

```
. svyset psuid [pw=w], strata(strid)
. ml model lf2 mylf2probit (q1 = x1 x2 x3), svy
. ml maximize
```

We could also use the subpop() option to make inferences about the subpopulation identified by the variable sub:

```
. svyset psuid [pw=w], strata(strid)
. ml model lf2 mylf2probit (q1 = x1 x2 x3), svy subpop(sub)
. ml maximize
```


## Stored results

For results stored by ml without the svy option, see [R] Maximize.
For results stored by ml with the svy option, see [SVY] svy.

## Methods and formulas

ml is implemented using moptimize(); see [M-5] moptimize().

## References

Korn, E. L., and B. I. Graubard. 1990. Simultaneous testing of regression coefficients with complex survey data: Use of Bonferroni $t$ statistics. American Statistician 44: 270-276. https://doi.org/10.2307/2684345.
Pitblado, J. S., B. P. Poi, and W. W. Gould. 2024. Maximum Likelihood Estimation with Stata. 5th ed. College Station, TX: Stata Press.
Royston, P. 2007. Profile likelihood for estimation and confidence intervals. Stata Journal 7: 376-387.

## Also see

[R] Maximize - Details of iterative maximization
[R] mlexp - Maximum likelihood estimation of user-specified expressions
[R] nl - Nonlinear least-squares estimation
[M-5] moptimize() - Model optimization
[M-5] optimize() — Function optimization
[U] 20 Estimation and postestimation commands

## Title

mlexp - Maximum likelihood estimation of user-specified expressions

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

mlexp performs maximum likelihood estimation of models that satisfy the linear-form restrictions, that is, models for which you can write the log likelihood for an individual observation and for which the overall log likelihood is the sum of the individual observations' log likelihoods.

You express the observation-level log-likelihood function by using a substitutable expression. Unlike models fit using ml , you do not need to do any programming. However, ml can fit classes of models that cannot be fit by mlexp.

## Quick start

Linear regression of y on x 1 and x 2

```
mlexp (ln(normalden(y, {xb: x1 x2 _cons}, {sigma})))
```

Same as above

```
mlexp (ln(normalden(y, {b0} + {b1}*x1 + {b2}*x2, {sigma})))
```

Same as above, and set initial values for b2 and sigma

```
mlexp (ln(normalden(y, {b0} + {b1}*x1 + {b2=0.5}*x2, {sigma=3})))
```

Constrain sigma and the coefficient on x 1 to be positive

```
mlexp (ln(normalden(y, {b0} + exp({lnb1})*x1 + {b2}*x2, exp({lnsigma}))))
nlcom (b1: exp(_b[lnb1:_cons])) (sigma: exp(_b[lnsigma:_cons]))
```

Omit observations with missing values for $\mathrm{y}, \mathrm{x} 1$, or x 2

```
mlexp (ln(normalden(y, {xb: x1 x2 _cons}, {sigma}))), variables(y x1 x2)
```


## Menu

Statistics $>$ Other $>$ Maximum likelihood estimation of expression

## Syntax

$$
\operatorname{mlexp}(\text { lexp })[i f][\text { in }][\text { weight }][\text {, options }]
$$

where lexp is a substitutable expression representing the log-likelihood function.
options
Model
variables (varlist)
from(initial_values)
constraints (numlist)
Derivatives
$\underline{\text { derivative }(/ n a m e ~=~ d e x p) ~}$

SE/Robust
vce (vcetype)

Reporting
level(\#)
title (string)
title2 (string)
display_options

## Maximization

maximize_options
debug
collinear
coeflegend

Description
specify variables in model
specify initial values for parameters
apply specified linear constraints
specify derivative of lexp with respect to parameter name; can be specified more than once
vcetype may be oim, opg, robust, cluster clustvar, bootstrap, or jackknife
set confidence level; default is level (95)
display string as title above the table of parameter estimates display string as subtitle
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
control the maximization process; seldom used
display debug output
keep collinear variables
display legend instead of statistics
lexp and dexp may contain factor variables and time-series operators; see [U] 11.4.3 Factor variables and [U] 11.4.4 Timeseries varlists.
bootstrap, by, collect, jackknife, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands.
Weights are not allowed with the bootstrap prefix; see $[R]$ bootstrap.
aweights are not allowed with the jackknife prefix; see $[R]$ jackknife.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
aweights, fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
debug, collinear, and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.
lexp and dexp are substitutable expressions, Stata expressions that also contain parameters to be estimated. The parameters are enclosed in curly braces and must satisfy the naming requirements for variables; \{beta\} is an example of a parameter. The notation $\{l c: v a r l i s t\}$ is allowed for linear combinations of multiple covariates and their parameters. For example, \{xb: mpg price turn _cons\} defines a linear combination of the variables mpg, price, turn, and _cons (the constant term). See Substitutable expressions under Remarks and examples below.

## Options

variables (varlist) specifies the variables in the model. mlexp excludes observations for which any of these variables has missing values. If you do not specify variables (), then mlexp assumes all observations are valid. mlexp will exit with an error message if the log likelihood cannot be calculated at the initial values for any observation.
from(initial_values) specifies the initial values to begin the estimation. You can specify parameter names and values, or you can specify a $1 \times k$ matrix, where $k$ is the number of parameters in the model. For example, to initialize alpha to 1.23 and delta to 4.57 , you would type

```
mlexp ..., from(alpha=1.23 delta=4.57) ...
```

or equivalently

```
matrix define initval = (1.23, 4.57)
```

mlexp ..., from(initval) ...

Initial values declared in the from() option override any that are declared within substitutable expressions. If you specify a parameter that does not appear in your model, mlexp exits with an error. If you specify a matrix, the values must be in the same order in which the parameters are declared in your model.
constraints (numlist); see [R] Estimation options.

Derivatives
derivative (/name $=$ dexp $)$ specifies the derivative of the observation-level log-likelihood function with respect to parameter name. If you wish to specify analytic derivatives, you must specify derivative() for each parameter in your model.
dexp uses the same substitutable expression syntax as is used to specify the log-likelihood function. If you declare a linear combination in the log-likelihood function, you provide the derivative for the linear combination; mlexp then applies the chain rule for you. See Specifying derivatives under Remarks and examples for examples.
If you do not specify the derivative() option, mlexp calculates derivatives numerically.

SE/Robust
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.

## Reporting

level (\#); see [R] Estimation options.
title(string) specifies an optional title that will be displayed just above the table of parameter estimates.
title2(string) specifies an optional subtitle that will be displayed between the title specified in title() and the table of parameter estimates. If title2() is specified but title() is not, then title2() has the same effect as title().
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap (\#), fvwrapon(style), cformat ( $\%$ fmt ), pformat ( $\%$ fmt ), sformat (\% firt), and nolstretch; see [R] Estimation options.
maximize_options: difficult, technique (algorithm_spec), iterate (\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), and nonrtolerance; see [R] Maximize. These options are seldom used.

The following options are available with mlexp but are not shown in the dialog box:
debug specifies that differences between the numerically computed gradient and the gradient computed from your derivative expression are reported at each iteration. This option is only allowed with the derivative() option.
collinear, coeflegend; see [R] Estimation options.

## Remarks and examples

Remarks are presented under the following headings:

## Introduction

Substitutable expressions
Parameter interpretation using margins
Parameter constraints
Specifying derivatives

## Introduction

mlexp performs maximum likelihood estimation of models that satisfy the linear-form restrictions, that is, models for which you can write the log likelihood for a single observation and for which the overall log likelihood is the sum of the individual observations' log likelihoods. Models designed for use with cross-sectional data usually meet the linear-form restrictions, including linear regression, many discrete choice models, limited-dependent-variable models, and selection models. Examples of models that do not satisfy the linear-form restrictions are random-effects panel-data models (because the likelihood function is defined at the panel level) and Cox proportional hazards models (because the likelihood function is defined for risk sets).

Because of its straightforward syntax and accessibility from the menu system, mlexp is particularly suited to users who are new to Stata and to those using Stata for pedagogical purposes. You specify the log-likelihood function that mlexp is to maximize by using substitutable expressions that are similar to those used by nl , nl sur, and gmm. mlexp allows you to avoid the programming requirements of ml . However, ml can fit classes of models that cannot be fit by mlexp, including those that do not meet the linear-form restrictions. See Pitblado, Poi, and Gould (2024, chap. 2) for examples using mlexp and for further discussion of the advantages of both mlexp and ml.

## Substitutable expressions

Substitutable expressions allow you to distinguish between variables and parameters. There are three rules to follow when defining substitutable expressions:

1. Parameters of the model are bound in curly braces: \{b0\}, \{param\}, etc. Parameter names must follow the same conventions as variable names; see [U] 11.3 Naming conventions.
2. Initial values for parameters are given by including an equal sign and the initial value inside the curly braces: $\{b 0=1\}$, $\{$ param $=3.571\}$, etc.
3. Linear combinations of variables can be included using the notation $\{l c:$ varlist $\}$ : \{xb: mpg price weight _cons\}, \{score: wx z\}, etc. Parameters of linear combinations are initialized to zero.
Substitutable expressions can include any mathematical expression involving scalars and variables. See [U] 13.2 Operators and [U] 13.3 Functions for more information on expressions.

## $>$ Example 1: The gamma density function

In an extract of the March 2014 Current Population Survey downloaded from Integrated Public Use Microdata Series (IPUMS), wage contains the wage and salary income earned in tens of thousands of dollars per year for working individuals; see King et al. (2010). We want to model wage using the two-parameter gamma distribution with shape parameter $\alpha$ and rate parameter $\beta$. The density function for $y>0$ is

$$
f(y)=\frac{\beta^{\alpha}}{\Gamma(\alpha)} y^{\alpha-1} \exp (-\beta y) \quad \beta>0, \alpha>0
$$

so that the $\log$ likelihood for the $i$ th observation is

$$
\ln \left(\ell_{i}\right)=\alpha \ln (\beta)-\ln \{\Gamma(\alpha)\}+(\alpha-1) \ln \left(y_{i}\right)-\beta y_{i}
$$

To fit wage to the two-parameter gamma distribution, we let a be the parameter name that conforms to Stata naming conventions for $\alpha$ and b be the parameter name for $\beta$. We enclose both in $\}$ in our substitutable expression for the log-likelihood function:

```
. use https://www.stata-press.com/data/r18/cpswage
. mlexp ({a}*ln({b}) - lngamma({a}) + ({a}-1)*ln(wage) - {b}*wage)
Initial: Log likelihood = -<inf> (could not be evaluated)
Feasible: Log likelihood = -239367.36
Rescale: Log likelihood = -228037.02
Rescale eq: Log likelihood = -163560.64
Iteration 0: Log likelihood = -163560.64
Iteration 1: Log likelihood = -162820.41
Iteration 2: Log likelihood = -162808.55
Iteration 3: Log likelihood = -162808.55
Maximum likelihood estimation
Log likelihood = -162808.55 Number of obs = 64,748
```

|  | Coefficient | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ---: | :---: | :---: | :---: | :---: | :---: | :---: |
| /a | 1.097287 | .0054134 | 202.70 | 0.000 | 1.086677 | 1.107897 |
| /b | .2406718 | .0014917 | 161.34 | 0.000 | .2377482 | .2435954 |

Because we did not specify initial values, mlexp initialized $\alpha$ and $\beta$ to be 0 . When both parameters are 0 , the log-likelihood function cannot be evaluated, because $\ln (0)$ is undefined. Therefore, in the iteration $\log$ above the coefficient table, we see that mlexp reported the initial log likelihood to be -<inf> (could not be evaluated). When this occurs, mlexp uses a search routine to find alternative initial values that do allow the log-likelihood function to be calculated.

We now initialize a to 1 and b to 0.1 the first time that we type them within the substitutable expression:

| . $m \mathrm{lexp}(\{\mathrm{a}=1\} * \ln (\{\mathrm{~b}=.1\})-\operatorname{lngamma}(\{\mathrm{a}\})+(\{\mathrm{a}\}-1) * \ln$ (wage) - \{b\}*wage) |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Initial: | Log likelihood $=-178608.12$ |  |  |  |  |  |
| Rescale: | Log likelihood $=-178608.12$ |  |  |  |  |  |
| Rescale eq: | Log likelihood = -173389.94 |  |  |  |  |  |
| Iteration 0: | Log likelihood $=-173389.94$ |  |  |  |  |  |
| Iteration 1: | Log likelihood $=-163081.69$ |  |  |  |  |  |
| Iteration 2: | Log likelihood $=-162813.54$ |  |  |  |  |  |
| Iteration 3: | Log likelihood $=-162808.55$ |  |  |  |  |  |
| Iteration 4: | Log likelihood $=-162808.55$ |  |  |  |  |  |
| Maximum likelihood estimation |  |  |  |  |  |  |
| Log likelihood $=-162808.55 \quad$ Number of obs $=64,748$ |  |  |  |  |  |  |
|  | Coefficient | Std. err. | z | $P>\|z\|$ | [95\% conf | interval] |
| /a | 1.097287 | . 0054134 | 202.70 | 0.000 | 1.086677 | 1.107897 |
| /b | . 2406719 | . 0014917 | 161.34 | 0.000 | . 2377483 | . 2435955 |

Even when mlexp can find alternative initial values, specifying your own values facilitates optimization.

By default, initial values of parameters are set to zero, and mlexp performs an iterative search for optimum starting values before beginning maximization. If mlexp cannot find initial parameter values for which it can calculate the log-likelihood function, it will exit with an error message. Restricting the sample or specifying starting values solves this problem.

Use the variables () option, an if qualifier, or an in qualifier to restrict the sample. You can also specify initial values by using the from() option or within the substitutable expression by including an equal sign and the initial value after the parameter. If you specify initial values by using from(), they override whatever initial values are given within the substitutable expression. Regardless of whether you specify initial values, mlexp performs a search procedure for better starting values before commencing the first iteration of the maximization routine.
> Example 2: Linear combinations of covariates
We frequently want to model the parameters as linear combinations of variables. Continuing example 1, we see that the mean of the two-parameter gamma distribution is $\mathbf{E}(y)=\alpha / \beta$. By letting $\alpha=a_{1}$ age $+a_{0}$, we model the mean of wage conditional on age as $\mathbf{E}($ wage age $)=\left(a_{1}\right.$ age $\left.+a_{0}\right) / \beta$. Below, we specify \{a: age _cons\} to model a as a linear combination of age and a constant term.

```
. mlexp ({a:age _cons}*ln({b=.1})-lngamma({a:})+({a:}-1)*ln(wage)-{b}*wage)
Initial: Log likelihood = -<inf> (could not be evaluated)
Feasible: Log likelihood = -239367.36
Rescale: Log likelihood = -228037.02
Rescale eq: Log likelihood = -163560.64
Iteration 0: Log likelihood = -163560.64
Iteration 1: Log likelihood = -160123.65
Iteration 2: Log likelihood = -159838.42
Iteration 3: Log likelihood = -159838.38
Iteration 4: Log likelihood = -159838.38
Maximum likelihood estimation
Log likelihood = -159838.38 Number of obs = 64,748
```

|  | Coefficient | Std. err. | z | P>\|z| | [95\% conf. interval] |  |  |
| ---: | ---: | :--- | :--- | :--- | :--- | :--- | :--- |
| a |  |  |  |  |  |  |  |
|  | age | .0194396 | .0002689 | 72.31 | 0.000 | .0189126 | .0199665 |
| _cons | .4108355 | .0096791 | 42.45 | 0.000 | .3918647 | .4298062 |  |
|  | /b | .2667954 | .0016423 | 162.45 | 0.000 | .2635765 | .2700142 |

The notation \{a: age _cons\} indicates to mlexp that you want a linear combination of the variable age and a constant term. We named the linear combination a, so mlexp names the parameters a: age and a:_cons, respectively.

Once you have declared a linear combination, you can subsequently refer to the linear combination by specifying its name and a colon inside curly braces, as we did in this example. You cannot use the same name for both an individual parameter and a linear combination. However, after a linear combination has been declared, you can refer to the parameter of an individual variable within that linear combination by using the notation $\{l c: z\}$, where $l c$ is the name of the linear combination, and $z$ is the variable whose parameter you want to reference.

## > Example 3: Linear combinations of factor variables

Factor variables and time-series operated variables can be included in a varlist defining a linear combination. Continuing example 2, we want to allow different intercepts for males and females in $\alpha$, and we want different $\beta$ parameters for males and females. We implement this model below by including ibn.female in each linear combination using factor-variable notation.

```
. mlexp ({a:age ibn.female}*ln({b:ibn.female}) - lngamma({a:}) +
> ({a:}-1)*ln(wage) - {b:}*wage)
Initial: Log likelihood = -<inf> (could not be evaluated)
Feasible: Log likelihood = -239367.36
Rescale: Log likelihood = -228037.02
Rescale eq: Log likelihood = -163560.64
Iteration 0: Log likelihood = -163560.64
Iteration 1: Log likelihood = -158999.51
Iteration 2: Log likelihood = -158135.16
Iteration 3: Log likelihood = -158130.94
Iteration 4: Log likelihood = -158130.93
```

Maximum likelihood estimation
Log likelihood $=-158130.93 \quad$ Number of obs $=64,748$

|  | Coefficient | Std. err. | z | $P>\|z\|$ | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| a |  |  |  |  |  |  |
| age | . 020543 | . 0002781 | 73.88 | 0.000 | . 019998 | . 0210881 |
| female |  |  |  |  |  |  |
| Male | . 3917619 | . 0111697 | 35.07 | 0.000 | . 3698696 | . 4136542 |
| Female | . 4475977 | . 0116285 | 38.49 | 0.000 | . 4248063 | . 4703892 |
| b |  |  |  |  |  |  |
| female |  |  |  |  |  |  |
| Male | . 2279862 | . 0018677 | 122.07 | 0.000 | . 2243256 | . 2316468 |
| Female | . 3600628 | . 0030156 | 119.40 | 0.000 | . 3541523 | . 3659733 |

## Parameter interpretation using margins

The margins command can be used after mlexp to estimate the effect of a covariate from a set of parameter estimates. The estimated covariate effects can be conditional on the other covariates or population-averaged effects that average out the other covariates.

## > Example 4

Continuing example 3, we use margins to estimate the average wage if everyone in the sample were male and if everyone in the sample were female. The expression() option is used to specify the formula for the mean. Wages, measured in tens of thousands of dollars, have a gamma distribution, so the mean is a ratio of the $\alpha$ and $\beta$ parameters. The linear prediction is specified in the expression with xb() .

```
. margins i.female, expression(xb(a)/xb(b))
Predictive margins Number of obs = 64,748
Model VCE: OIM
Expression: xb(a)/xb(b)
```

|  | Delta-method <br> Margin <br> std.err. |  |  |  | $\mathbf{z}$ | $\mathrm{P}>\|\mathrm{z}\|$ |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| [95\% conf. interval] |  |  |  |  |  |  |
| female |  |  |  |  |  |  |
| Male | 5.452281 | .0267573 | 203.77 | 0.000 | 5.399838 | 5.504724 |
| Female | 3.607372 | .0178778 | 201.78 | 0.000 | 3.572332 | 3.642412 |

When everyone is male, the estimated average wage is 5.45 ; when everyone is female, the estimated average wage is 3.61 .

## Parameter constraints

## Example 5

In examples 1,2 , and 3 , we were lucky. The two-parameter gamma density function is defined only when both $\alpha$ and $\beta$ are positive. However, mlexp does not know this; when maximizing the log-likelihood function, it will consider all real values for the parameters. Also recall from above, mlexp will exit with an error message if it cannot find parameter values that produce nonmissing values for the likelihood for each sample observation.

We could reparameterize our model so that we avoid having to directly estimate parameters that are restricted. For example, consider the parameter $\alpha>0$, and suppose we define the new parameter $\theta=\ln (\alpha)$ so that $\alpha=\exp (\theta)$. With this parameterization, for any real value of $\theta$ that mlexp might try to use when evaluating the log-likelihood function, $\alpha$ is guaranteed to be positive. Below we apply this parameterization to the model in example 1.

```
. mlexp (exp({lna})*{lnb} - lngamma(exp({lna}))
> + (exp({lna})-1)*ln(wage) - exp({lnb})*wage)
Initial: Log likelihood = -295203.41
Alternative: Log likelihood = -249215.67
Rescale: Log likelihood = -230376.58
Rescale eq: Log likelihood = -166588.81
Iteration 0: Log likelihood = -166588.81
Iteration 1: Log likelihood = -162845.95
Iteration 2: Log likelihood = -162808.56
Iteration 3: Log likelihood = -162808.55
Iteration 4: Log likelihood = -162808.55
Maximum likelihood estimation
```

Log likelihood $=-162808.55 \quad$ Number of obs $=64,748$

|  | Coefficient | Std. err. | $z$ | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| /lna | .0928409 | .0049334 | 18.82 | 0.000 | .0831716 | .1025103 |
| /lnb | -1.424321 | .0061979 | -229.81 | 0.000 | -1.436469 | -1.412173 |

We can use nlcom to obtain the back-transformed parameter estimates, but first we need to know how to refer to the parameters estimated by mlexp. We can replay the results and request the coefficient legend.
. mlexp, coeflegend
Maximum likelihood estimation
Log likelihood $=-162808.55 \quad$ Number of obs $=64,748$

|  | Coefficient | Legend |
| :--- | ---: | :--- |
| /lna | .0928409 -b[lna:_cons] <br> /lnb -1.424321 | _b[lnb:_cons] |

Now, we see that we can refer to _b[lna:_cons] and _b[lnb:_cons] in nlcom.

```
. nlcom (a: exp(_b[lna:_cons])) (b: exp(_b[lnb:_cons]))
    a: exp(_b[lna:_cons])
    b: exp(_b[lnb:_cons])
```

|  | Coefficient | Std. err. | $z$ | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| a | 1.097287 | .0054134 | 202.70 | 0.000 | 1.086677 | 1.107897 |
| b | .2406718 | .0014917 | 161.34 | 0.000 | .2377482 | .2435954 |

The optimal value of the log-likelihood function and the back-transformed parameter estimates match those reported in example 1.

However, when you impose nonlinear constraints on linear combinations of covariates, you change the model. These nonlinear constraints change the model because they are not invertible functions of the original parameters. For example, we can use the exponential function to ensure $\alpha>0$ and $\beta>0$ in the model from example 3.

```
. mlexp (exp({a:age ibn.female})*{b:ibn.female} - lngamma(exp({a:}))
> + (exp({a:})-1)*\operatorname{ln}(\mathrm{ wage) - exp({b:})*wage)}
Initial: Log likelihood = -295203.41
Alternative: Log likelihood = -249215.67
Rescale: Log likelihood = -230376.58
Rescale eq: Log likelihood = -166588.81
Iteration 0: Log likelihood = -166588.81
Iteration 1: Log likelihood = -160273.12
Iteration 2: Log likelihood = -158868.39
Iteration 3: Log likelihood = -158864.02
Iteration 4: Log likelihood = -158864.02
Maximum likelihood estimation
Log likelihood = -158864.02 Number of obs = 64,748
```

|  | Coefficient | Std. err. | z | $P>\|z\|$ | [95\% conf. | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| a |  |  |  |  |  |  |
| age | . 0126713 | . 0001785 | 70.98 | 0.000 | . 0123214 | . 0130212 |
| female |  |  |  |  |  |  |
| Male | -. 3470203 | . 009978 | -34.78 | 0.000 | -. 3665769 | -. 3274638 |
| Female | -. 3177398 | . 0103376 | -30.74 | 0.000 | -. 338001 | -. 2974786 |
| b |  |  |  |  |  |  |
| female |  |  |  |  |  |  |
| Male | -1.502222 | . 0085431 | -175.84 | 0.000 | -1.518966 | -1.485478 |
| Female | -1.059358 | . 0087602 | -120.93 | 0.000 | -1.076528 | -1.042189 |

The value of the log-likelihood function is now $-158,864.02$, which is less than the value $-158,130.93$, which was reported in example 3. The constrained parameterization that keeps $\alpha>0$ and $\beta>0$ has changed the model, and the optimal value of the log-likelihood function in the constrained model is less than the optimal value in the unconstrained model.

Because neither the unconstrained nor the constrained parameterization generates missing predicted values for $\alpha$ or $\beta$, we can choose between them. However, if any of the predicted values for $\alpha$ or $\beta$ produce missing values for the log-likelihood function using the unconstrained parameterization, we could not compute the log-likelihood function for the unconstrained parameterization, and we would have to use a constrained parameterization.

## Specifying derivatives

By default, mlexp calculates derivatives of the log-likelihood function numerically using an algorithm that produces accurate results. However, mlexp will fit your model more quickly (and even more accurately) if you specify analytic derivatives.

You specify derivatives by using substitutable expressions in much the same way as you specify the log-likelihood function. If you specify a linear combination in your log-likelihood function, then you supply a derivative with respect to that linear combination; mlexp then uses the chain rule to obtain the derivatives with respect to the individual parameters.

We will illustrate how to specify derivatives using the probit model for dichotomous outcomes. The log-likelihood function for the probit model is often written as

$$
\ln \ell_{i}= \begin{cases}\ln \Phi\left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\right) & y_{i}=1 \\ \ln \Phi\left(-\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\right) & y_{i}=0\end{cases}
$$

using the fact that $1-\Phi\left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\right)=\Phi\left(-\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\right)$, where $\Phi(\cdot)$ is the cumulative standard normal distribution function. If we use the trick suggested by Greene ( 2018 , 742 , fn. 16), we can simplify the loglikelihood function, making the derivative calculation easier. Let $q_{i}=2 y_{i}-1$. Then, we can write the log-likelihood function as

$$
\ln \ell_{i}=\ln \Phi\left(q_{i} \mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\right)
$$

and the first derivative as

$$
\frac{\partial \ln \ell_{i}}{\partial \boldsymbol{\beta}}=\frac{q_{i} \phi\left(q_{i} \mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\right)}{\Phi\left(q_{i} \mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\right)} \mathbf{x}_{i}
$$

- Example 6: Probit with a linear combination

Now, let's fit a probit model of the indicator for whether an individual is below the official poverty level offpov on age, female, and a constant. We could specify the parameters and independent variables individually, but we will use a linear combination instead. First, note that

$$
\frac{\partial \ln \ell_{i}}{\partial \mathbf{x}_{i}^{\prime} \boldsymbol{\beta}}=\frac{q_{i} \phi\left(q_{i} \mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\right)}{\Phi\left(q_{i} \mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\right)}
$$

When you specify a linear combination of variables, you specify the derivative with respect to the linear combination. That way, if you change the variables in the linear combination, you do not need to change the derivative. To see why this is the case, consider the function $f\left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\right)$, where $\mathbf{x}_{i}^{\prime} \beta$ is a linear combination. Then, using the chain rule, we see that

$$
\frac{\partial f\left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\right)}{\partial \beta_{j}}=\frac{\partial f\left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\right)}{\partial \mathbf{x}_{i}^{\prime} \boldsymbol{\beta}} \times \frac{\partial \mathbf{x}_{i}^{\prime} \boldsymbol{\beta}}{\partial \beta_{j}}=\frac{\partial f\left(\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}\right)}{\partial \mathbf{x}_{i}^{\prime} \boldsymbol{\beta}} \times x_{i j}
$$

Once the derivative with respect to the linear combination is known, mlexp can then multiply it by each of the variables in the linear combination to get the full set of derivatives with respect to the parameters needed to maximize the likelihood function. Moreover, the derivative with respect to the linear combination does not depend on the variables within the linear combination, so even if you change the variables in it, you will not need to modify the specification of the corresponding derivative() option.

We type

```
. generate int q = 2*offpov - 1
. mlexp (ln(normal(q*({xb:age i.female _cons})))),
> deriv(/xb = q*normalden(q*{xb:})/normal(q*{xb:}))
Initial: Log likelihood = -44879.894
Alternative: Log likelihood = -27407.718
Rescale: Log likelihood = -17888.147
Iteration 0: Log likelihood = -17888.147
Iteration 1: Log likelihood = -15805.22
Iteration 2: Log likelihood = -15427.598
Iteration 3: Log likelihood = -15427.04
Iteration 4: Log likelihood = -15427.04
Maximum likelihood estimation
Log likelihood = -15427.04 Number of obs = 64,748
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline & Coefficient & Std. err. & z & \(P>|z|\) & \multicolumn{2}{|l|}{[95\% conf. interval]} \\
\hline xb & & & & & & \\
\hline age & -. 0179139 & . 0005852 & -30.61 & 0.000 & -. 0190608 & -. 016767 \\
\hline female & & & & & & \\
\hline Female & . 1514014 & . 0154565 & 9.80 & 0.000 & . 1211072 & . 1816955 \\
\hline _cons & -. 8773462 & . 0244956 & -35.82 & 0.000 & -. 9253567 & -. 8293357 \\
\hline
\end{tabular}
```

After defining q, we specified the log-likelihood function using q and the linear combination xb . We also use xb in specifying the derivative.

## Stored results

mlexp stores the following in e():

Scalars
e(N)
e(k)
e(k_aux)
e(k_eq)
e(k_eq_model)
e(df_m)
e(11)
e(N_clust)
$e$ (rank)
e(ic)
e(rc)
e(converged)
Macros
e(cmd)
e(cmdline)
e(lexp)
e(wtype)
e(wexp)
e(usrtitle)
e(usrtitle2)
e(clustvar)
e(vce)
e(vcetype)
$e$ (params)
e(hasderiv)
$e\left(d_{-} j\right)$
e(rhs)
e(opt)
e(ml_method)
$e$ (technique)
e(singularHmethod)
e(crittype)
e(properties)
e(estat_cmd)
e(predict)
e(marginsok)
e(marginsnotok)
e(marginsprop)
e(asbalanced)
e(asobserved)
Matrices
e(b)
e(Cns)
e(ilog)
$e$ (init)
e(gradient)
e(V)
e(V_modelbased)
Functions
e(sample)
number of observations
number of parameters
number of ancillary parameters
number of equations in $e(b)$
number of equations in overall model test
model degrees of freedom
log likelihood
number of clusters
rank of e(V)
number of iterations
return code
1 if converged, 0 otherwise

## mlexp

command as typed
likelihood expression
weight type
weight expression
user-specified title
user-specified secondary title
name of cluster variable
vcetype specified in vce()
title used to label Std. err.
names of parameters
yes, if derivative() is specified
derivative expression for parameter $j$
contents of variables()
type of optimization
type of ml method
maximization technique
m-marquardt or hybrid; method used when Hessian is singular ${ }^{1}$
optimization criterion ${ }^{1}$
b V
program used to implement estat
program used to implement predict
predictions allowed by margins
predictions disallowed by margins
signals to the margins command
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
constraints matrix
iteration $\log$ (up to 20 iterations)
initial values
gradient vector
variance-covariance matrix of the estimators
model-based variance
marks estimation sample

1. Type ereturn list, all to view these results; see [P] return.

In addition to the above, the following is stored in r() :
Matrices
$r$ (table) matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

Optimization is carried out using moptimize(); see [M-5] moptimize().

## References

Drukker, D. M. 2015a. Efficiency comparisons by Monte Carlo simulation. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2015/10/13/efficiency-comparisons-by-monte-carlo-simulation/.
——. 2015b. Maximum likelihood estimation by mlexp: A chi-squared example. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2015/10/08/maximum-likelihood-estimation-by-mlexp-a-chi-squared-example/.
-_. 2015c. Understanding the generalized method of moments (GMM): A simple example. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2015/12/03/understanding-the-generalized-method-of-moments-gmm-a-simple-example/.

Greene, W. H. 2018. Econometric Analysis. 8th ed. New York: Pearson.
King, M., S. Ruggles, J. T. Alexander, S. Flood, K. Genadek, M. B. Schroeder, B. Trampe, and R. Vick. 2010. Integrated Public Use Microdata Series, Current Population Survey: Version 3.0 [Machine-readable database]. Minneapolis, MN: Minnesota Population Center [producer and distributor].

Lindsey, C. 2015a. Probit model with sample selection by mlexp. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2015/10/22/probit-model-with-sample-selection-by-mlexp/.
—. 2015b. Using mlexp to estimate endogenous treatment effects in a heteroskedastic probit model. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2015/12/10/using-mlexp-to-estimate-endogenous-treatment-effects-in-a-heteroskedastic-probit-model/.
—. 2015c. Using mlexp to estimate endogenous treatment effects in a probit model. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2015/11/05/using-mlexp-to-estimate-endogenous-treatment-effects-in-a-probitmodel/.

Lindsey, C., and E. Pinzon. 2016. Multiple equation models: Estimation and marginal effects using mlexp. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2016/06/14/multiple-equation-models-estimation-and-marginal-effects-using-mlexp/.
Pitblado, J. S., B. P. Poi, and W. W. Gould. 2024. Maximum Likelihood Estimation with Stata. 5th ed. College Station, TX: Stata Press.

Rajbhandari, A. 2015. Estimating parameters by maximum likelihood and method of moments using mlexp and gmm. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2015/10/15/estimating-parameters-by-maximum-likelihood-and-method-of-moments-using-mlexp-and-gmm/.

## Also see

[R] mlexp postestimation - Postestimation tools for mlexp
[R] gmm - Generalized method of moments estimation
[R] Maximize - Details of iterative maximization
[R] ml - Maximum likelihood estimation
[R] nl - Nonlinear least-squares estimation
[R] nlsur - Estimation of nonlinear systems of equations
[U] 20 Estimation and postestimation commands

## Title

mlexp postestimation - Postestimation tools for mlexp

Postestimation commands predict margins Also see

## Postestimation commands

The following postestimation commands are available after mlexp:

| Command | Description |
| :---: | :---: |
| contrast | contrasts and ANOVA-style joint tests of estimates |
| estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estimates | cataloging estimation results |
| etable | table of estimation results |
| lincom | point estimates, standard errors, testing, and inference for linear combinations of coefficients |
| * Irtest | likelihood-ratio test |
| margins | marginal means, predictive margins, marginal effects, and average marginal effects |
| marginsplot | graph the results from margins (profile plots, interaction plots, etc.) |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients |
| predict | linear predictions and scores |
| predictnl | point estimates, standard errors, testing, and inference for generalized predictions |
| pwcompare | pairwise comparisons of estimates |
| suest | seemingly unrelated estimation |
| test | Wald tests of simple and composite linear hypotheses |
| testnl | Wald tests of nonlinear hypotheses |

[^31]
## predict

## Description for predict

predict creates a new variable containing predictions such as linear predictions and equation-level scores.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

```
predict [type] newvar [if] [in] [, xb equation(#eqno|eqname)]
predict [type] stub* [if] [in], scores
```

Scores are only available for observations within the estimation sample.

## Options for predict

xb calculates the linear prediction.
equation(\#eqno $\mid$ eqname) specifies the equation for which the linear prediction is desired. Specifying equation(\#1) indicates that the calculation be made for the first equation. Specifying equation(demand) indicates that the calculation be made for the equation named demand, assuming there is an equation named demand in the model.

If you specify one new variable name and omit equation(), results are the same as if you had specified equation(\#1).
For more information on using predict after multiple-equation estimation commands, see $[R]$ predict.
scores calculates the equation-level score variables. The $j$ th new variable will contain the scores for the $j$ th equation of the model.

## margins

## Description for margins

margins estimates margins of response for linear predictions.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...)...] [options]
```

statistic Description
xb
linear prediction
xb defaults to the first equation.

## Also see

[R] mlexp - Maximum likelihood estimation of user-specified expressions
[U] 20 Estimation and postestimation commands

## Title

mlogit - Multinomial (polytomous) logistic regression

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

mlogit fits a multinomial logit (MNL) model for a categorical dependent variable with outcomes that have no natural ordering. The actual values taken by the dependent variable are irrelevant. The MNL model is also known as the polytomous logistic regression model. Some people refer to conditional logistic regression as multinomial logistic regression. If you are one of them, see [R] clogit.

## Quick start

MNL model of y on x 1 , x 2 , and categorical variable a
mlogit y x1 x2 i.a
Same as above, but use $\mathrm{y}=1$ as the base outcome even if 1 is not the most frequent mlogit y x1 x2 i.a, baseoutcome(1)

Report results as relative-risk ratios
mlogit y x1 x2 i.a, rrr
Constrain coefficient of x 1 to be equal for second and third outcomes

```
constraint 1 [#2=#3]:x1
mlogit y x1 x2 i.a, constraints(1)
```


## Menu

Statistics $>$ Categorical outcomes $>$ Multinomial logistic regression

## Syntax

mlogit depvar [indepvars] [if] [in] [weight] [, options]
options
Description

## Model

noconstant
baseoutcome(\#)
constraints (constraints)
SE/Robust
vce (vcetype)

Reporting
level(\#)
rrr
nocnsreport
display_options
suppress constant term
value of depvar that will be the base outcome
apply specified linear constraints
vcetype may be oim, opg, robust, cluster clustvar, bootstrap,
or jackknife
set confidence level; default is level (95)
report relative-risk ratios
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

Maximization
maximize_options
collinear
coeflegend
control the maximization process; seldom used
keep collinear variables
display legend instead of statistics
indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
indepvars may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bayes, bootstrap, by, collect, fmm, fp, jackknife, mfp, mi estimate, rolling, statsby, and svy are allowed;
see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: mlogit and [FMM] fmm: mlogit.
vce (bootstrap) and vce (jackknife) are not allowed with the mi estimate prefix; see [MI] mi estimate.
Weights are not allowed with the bootstrap prefix; see $[R]$ bootstrap.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

$\xlongequal[{\text { noconstant; see [R] Estimation options. }}]{\text { Model }}$
baseoutcome(\#) specifies the value of depvar to be treated as the base outcome. The default is to choose the most frequent outcome.
constraints (constraints); see [R] Estimation options.
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.

If specifying vce(bootstrap) or vce(jackknife), you must also specify baseoutcome().

## Reporting

level(\#); see [R] Estimation options.
rrr reports the estimated coefficients transformed to relative-risk ratios, that is, $e^{b}$ rather than $b$; see
Description of the model below for an explanation of this concept. Standard errors and confidence intervals are similarly transformed. This option affects how results are displayed, not how they are estimated. rrr may be specified at estimation or when replaying previously estimated results.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap (\#), fvwrapon(style), cformat (\%fmt), pformat (\% fmt), sformat (\% fmt), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate(\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from (init_specs); see [R] Maximize. These options are seldom used.

The following options are available with mlogit but are not shown in the dialog box: collinear, coeflegend; see [R] Estimation options.

## Remarks and examples

Remarks are presented under the following headings:
mlogit fits maximum likelihood models with discrete dependent (left-hand-side) variables when the dependent variable takes on more than two outcomes and the outcomes have no natural ordering. If the dependent variable takes on only two outcomes, estimates are identical to those produced by logistic or logit; see [R] logistic or [R] logit. If the outcomes are ordered, see [R] ologit. See $[R]$ logistic for a list of related estimation commands.

## Description of the model

For an introduction to MNL models, see Greene (2018, 829-833), Hosmer, Lemeshow, and Sturdivant (2013, 269-289), Long (1997, chap. 6), Long and Freese (2014, chap. 8), and Treiman (2009, 336-341). For a description emphasizing the difference in assumptions and data requirements for conditional and multinomial logit, see Davidson and MacKinnon (1993).

Consider the outcomes $1,2,3, \ldots, m$ recorded in $y$, and the explanatory variables $X$. Assume that there are $m=3$ outcomes: "buy an American car", "buy a Japanese car", and "buy a European car". The values of $y$ are then said to be "unordered". Even though the outcomes are coded 1, 2, and 3, the numerical values are arbitrary because $1<2<3$ does not imply that outcome 1 (buy American) is less than outcome 2 (buy Japanese) is less than outcome 3 (buy European). This unordered categorical property of $y$ distinguishes the use of mlogit from regress (which is appropriate for a continuous dependent variable), from ologit (which is appropriate for ordered categorical data), and from logit (which is appropriate for two outcomes, which can be thought of as ordered).

In the MNL model, you estimate a set of coefficients, $\beta^{(1)}, \beta^{(2)}$, and $\beta^{(3)}$, corresponding to each outcome:

$$
\begin{aligned}
& \operatorname{Pr}(y=1)=\frac{e^{X \beta^{(1)}}}{e^{X \beta^{(1)}}+e^{X \beta^{(2)}}+e^{X \beta^{(3)}}} \\
& \operatorname{Pr}(y=2)=\frac{e^{X \beta^{(2)}}}{e^{X \beta^{(1)}}+e^{X \beta^{(2)}}+e^{X \beta^{(3)}}} \\
& \operatorname{Pr}(y=3)=\frac{e^{X \beta^{(3)}}}{e^{X \beta^{(1)}}+e^{X \beta^{(2)}}+e^{X \beta^{(3)}}}
\end{aligned}
$$

The model, however, is unidentified in the sense that there is more than one solution to $\beta^{(1)}, \beta^{(2)}$, and $\beta^{(3)}$ that leads to the same probabilities for $y=1, y=2$, and $y=3$. To identify the model, you arbitrarily set one of $\beta^{(1)}, \beta^{(2)}$, or $\beta^{(3)}$ to 0 -it does not matter which. That is, if you arbitrarily set $\beta^{(1)}=0$, the remaining coefficients $\beta^{(2)}$ and $\beta^{(3)}$ will measure the change relative to the $y=1$ group. If you instead set $\beta^{(2)}=0$, the remaining coefficients $\beta^{(1)}$ and $\beta^{(3)}$ will measure the change relative to the $y=2$ group. The coefficients will differ because they have different interpretations, but the predicted probabilities for $y=1,2$, and 3 will still be the same. Thus, either parameterization will be a solution to the same underlying model.

Setting $\beta^{(1)}=0$, the equations become

$$
\begin{aligned}
& \operatorname{Pr}(y=1)=\frac{1}{1+e^{X \beta^{(2)}}+e^{X \beta^{(3)}}} \\
& \operatorname{Pr}(y=2)=\frac{e^{X \beta^{(2)}}}{1+e^{X \beta^{(2)}}+e^{X \beta^{(3)}}} \\
& \operatorname{Pr}(y=3)=\frac{e^{X \beta^{(3)}}}{1+e^{X \beta^{(2)}}+e^{X \beta^{(3)}}}
\end{aligned}
$$

The relative probability of $y=2$ to the base outcome is

$$
\frac{\operatorname{Pr}(y=2)}{\operatorname{Pr}(y=1)}=e^{X \beta^{(2)}}
$$

Let's call this ratio the relative risk, and let's further assume that $X$ and $\beta_{k}^{(2)}$ are vectors equal to $\left(x_{1}, x_{2}, \ldots, x_{k}\right)$ and $\left(\beta_{1}^{(2)}, \beta_{2}^{(2)}, \ldots, \beta_{k}^{(2)}\right)^{\prime}$, respectively. The ratio of the relative risk for a one-unit change in $x_{i}$ is then

$$
\frac{e^{\beta_{1}^{(2)} x_{1}+\cdots+\beta_{i}^{(2)}\left(x_{i}+1\right)+\cdots+\beta_{k}^{(2)} x_{k}}}{e^{\beta_{1}^{(2)} x_{1}+\cdots+\beta_{i}^{(2)} x_{i}+\cdots+\beta_{k}^{(2)} x_{k}}}=e^{\beta_{i}^{(2)}}
$$

Thus, the exponentiated value of a coefficient is the relative-risk ratio for a one-unit change in the corresponding variable (risk is measured as the risk of the outcome relative to the base outcome).

## Fitting unconstrained models

## > Example 1: A first example

We have data on the type of health insurance available to 616 psychologically depressed subjects in the United States (Tarlov et al. 1989; Wells et al. 1989). The insurance is categorized as either an indemnity plan (that is, regular fee-for-service insurance, which may have a deductible or coinsurance rate) or a prepaid plan (a fixed up-front payment allowing subsequent unlimited use as provided, for instance, by an HMO). The third possibility is that the subject has no insurance whatsoever. We wish to explore the demographic factors associated with each subject's insurance choice. One of the demographic factors in our data is the race of the participant, coded as white or nonwhite:

```
. use https://www.stata-press.com/data/r18/sysdsn1
(Health insurance data)
. tabulate insure nonwhite, chi2 col
```

| Key |
| :--- |
| frequency |
| column percentage |


| Type of insurance | Nonwhite |  | Total |
| :---: | :---: | :---: | :---: |
| Indemnity | 251 | 43 | 294 |
|  | 50.71 | 35.54 | 47.73 |
| Prepaid | 208 | 69 | 277 |
|  | 42.02 | 57.02 | 44.97 |
| Uninsure | 36 | 9 | 45 |
|  | 7.27 | 7.44 | 7.31 |
| Total | 495 | 121 | 616 |
|  | 100.00 | 100.00 | 100.00 |
|  | Pearson chi2(2) = 9.55 |  | $\operatorname{Pr}=0$ |

Although insure appears to take on the values Indemnity, Prepaid, and Uninsure, it actually takes on the values 1,2 , and 3 . The words appear because we have associated a value label with the numeric variable insure; see [U] 12.6.3 Value labels.

When we fit an MNL model, we can tell mlogit which outcome to use as the base outcome, or we can let mlogit choose. To fit a model of insure on nonwhite, letting mlogit choose the base outcome, we type

| Iteration 0: Log likelihood = -556.59502 |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Iteration 1: Log likelihood $=-551.78935$ |  |  |  |  |  |  |
| Iteration 2: Log likelihood $=-551.78348$ |  |  |  |  |  |  |
| Iteration 3: Log likelihood = -551.78348 |  |  |  |  |  |  |
| Multinomial logistic regression |  |  |  |  | Number of obs | $=616$ |
|  |  |  |  |  | LR chi2(2) | $=9.62$ |
|  |  |  |  |  | Prob > chi2 | $=0.0081$ |
| Log likelihood = -551.78348 |  |  |  |  | Pseudo R2 | $=0.0086$ |
| insure | Coefficient | Std. err. | z | $P>\|z\|$ | [95\% conf. | interval] |
| Indemnity | (base outcome) |  |  |  |  |  |
| Prepaid |  |  |  |  |  |  |
| nonwhite | . 6608212 | . 2157321 | 3.06 | 0.002 | . 2379942 | 1.083648 |
| _cons | -. 1879149 | . 0937644 | -2.00 | 0.045 | -. 3716896 | -. 0041401 |
| Uninsure |  |  |  |  |  |  |
| nonwhite | . 3779586 | . 407589 | 0.93 | 0.354 | -. 4209011 | 1.176818 |
| _cons | -1.941934 | . 1782185 | -10.90 | 0.000 | -2.291236 | -1.592632 |

mlogit chose the indemnity outcome as the base outcome and presented coefficients for the outcomes prepaid and uninsured. According to the model, the probability of prepaid for whites (nonwhite $=0$ ) is

$$
\operatorname{Pr}(\text { insure }=\text { Prepaid })=\frac{e^{-.188}}{1+e^{-.188}+e^{-1.942}}=0.420
$$

Similarly, for nonwhites, the probability of prepaid is

$$
\operatorname{Pr}(\text { insure }=\operatorname{Prepaid})=\frac{e^{-.188+.661}}{1+e^{-.188+.661}+e^{-1.942+.378}}=0.570
$$

These results agree with the column percentages presented by tabulate because the mlogit model is fully saturated. That is, there are enough terms in the model to fully explain the column percentage in each cell. The model $\chi^{2}$ and the tabulate $\chi^{2}$ are in almost perfect agreement; both test that the column percentages of insure are the same for both values of nonwhite.

## > Example 2: Specifying the base outcome

By specifying the baseoutcome () option, we can control which outcome of the dependent variable is treated as the base. Left to its own, mlogit chose to make outcome 1, indemnity, the base outcome. To make outcome 2, prepaid, the base, we would type

```
. mlogit insure nonwhite, base(2)
Iteration 0: Log likelihood = -556.59502
Iteration 1: Log likelihood = -551.78935
Iteration 2: Log likelihood = -551.78348
Iteration 3: Log likelihood = -551.78348
```

Multinomial logistic regression Number of obs = 616
LR chi2(2) $=9.62$
Prob > chi2 $=0.0081$
Pseudo R2 $=0.0086$

| insure | Coefficient | Std. err. | $z$ | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| :---: | ---: | ---: | ---: | ---: | ---: | ---: |
| Indemnity <br> nonwhite <br> _cons | -.6608212 <br> .1879149 | .2157321 | -3.06 | 0.002 | -1.083648 | -.2379942 |
| Prepaid | (base outcome) | 2.00 | 0.045 | .0041401 | .3716896 |  |
| Uninsure <br> nonwhite <br> _cons | -.2828627 | .3977302 | -0.71 | 0.477 | -1.0624 | .4966742 |

The baseoutcome() option requires that we specify the numeric value of the outcome, so we could not type base (Prepaid).

Although the coefficients now appear to be different, the summary statistics reported at the top are identical. With this parameterization, the probability of prepaid insurance for whites is

$$
\operatorname{Pr}(\text { insure }=\operatorname{Prepaid})=\frac{1}{1+e^{.188}+e^{-1.754}}=0.420
$$

This is the same answer we obtained previously.

## > Example 3: Displaying relative-risk ratios

By specifying rrr, which we can do at estimation time or when we redisplay results, we see the model in terms of relative-risk ratios:

| . mlogit, rrr |  |  |
| :--- | :--- | ---: |
| Multinomial logistic regression | Number of obs $=$ | 616 |
|  | LR chi2 $(2)$ | $=$ |
|  | Prob $>$ chi2 | $=0.62$ |
| Log likelihood $=-551.78348$ | Pseudo R2 | $=0.0081$ |


| insure | RRR | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| :---: | ---: | :---: | :---: | :---: | :---: | :---: |
| Indemnity <br> nonwhite <br> _cons | 1.206731 | .1131483 | 2.00 | 0.045 | 1.004149 | 1.450183 |
| Prepaid | (base outcome) |  |  |  |  |  |
| Uninsure <br> nonwhite <br> _cons | .7536233 | .2997387 | -0.71 | 0.477 | .3456255 | 1.643247 |

[^32]Looked at this way, the relative risk of choosing an indemnity over a prepaid plan is 0.516 for nonwhites relative to whites.

To illustrate, from the output and discussions of examples 1 and 2 we find that

$$
\operatorname{Pr}(\text { insure }=\text { Indemnity } \mid \text { white })=\frac{1}{1+e^{-.188}+e^{-1.942}}=0.507
$$

and thus the relative risk of choosing indemnity over prepaid (for whites) is

$$
\frac{\operatorname{Pr}(\text { insure }=\text { Indemnity } \mid \text { white })}{\operatorname{Pr}(\text { insure }=\text { Prepaid } \mid \text { white })}=\frac{0.507}{0.420}=1.207
$$

For nonwhites,

$$
\operatorname{Pr}(\text { insure }=\text { Indemnity } \mid \text { not white })=\frac{1}{1+e^{-.188+.661}+e^{-1.942+.378}}=0.355
$$

and thus the relative risk of choosing indemnity over prepaid (for nonwhites) is

$$
\frac{\operatorname{Pr}(\text { insure }=\text { Indemnity } \mid \text { not white })}{\operatorname{Pr}(\text { insure }=\text { Prepaid } \mid \text { not white })}=\frac{0.355}{0.570}=0.623
$$

The ratio of these two relative risks, hence the name "relative-risk ratio", is $0.623 / 1.207=0.516$, as given in the output under the heading "RRR".

## - Technical note

In models where only two categories are considered, the mlogit model reduces to standard logit. Consequently, the exponentiated regression coefficients, labeled as RRR within mlogit, are equal to the odds ratios as given when the or option is specified under logit; see [R] logit.

As such, always referring to mlogit's exponentiated coefficients as odds ratios may be tempting. However, the discussion in example 3 demonstrates that doing so would be incorrect. In general mlogit models, the exponentiated coefficients are ratios of relative risks, not ratios of odds.

## > Example 4: Model with continuous and multiple categorical variables

One of the advantages of mlogit over tabulate is that we can include continuous variables and multiple categorical variables in the model. In examining the data on insurance choice, we decide that we want to control for age, gender, and site of study (the study was conducted in three sites):

```
. mlogit insure age male nonwhite i.site
Iteration 0: Log likelihood = -555.85446
Iteration 1: Log likelihood = -534.67443
Iteration 2: Log likelihood = -534.36284
Iteration 3: Log likelihood = -534.36165
Iteration 4: Log likelihood = -534.36165
Multinomial logistic regression Number of obs = 615
LR chi2(10) = 42.99
Prob > chi2 = 0.0000
Pseudo R2 = 0.0387
```

| insure | Coefficient | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Indemnity | (base outcome) |  |  |  |  |  |
| Prepaid |  |  |  |  |  |  |
| age | -.011745 | .0061946 | -1.90 | 0.058 | -.0238862 | .0003962 |
| male | .5616934 | .2027465 | 2.77 | 0.006 | .1643175 | .9590693 |
| nonwhite | .9747768 | .2363213 | 4.12 | 0.000 | .5115955 | 1.437958 |
| site |  |  |  |  |  |  |
| 2 | .1130359 | .2101903 | 0.54 | 0.591 | -.2989296 | .5250013 |
| 3 | -.5879879 | .2279351 | -2.58 | 0.010 | -1.034733 | -.1412433 |
| _cons | .2697127 | .3284422 | 0.82 | 0.412 | -.3740222 | .9134476 |
| age | -.0077961 | .0114418 | -0.68 | 0.496 | -.0302217 | .0146294 |
| male | .4518496 | .3674867 | 1.23 | 0.219 | -.268411 | 1.17211 |
| Uninsure | .2170589 | .4256361 | 0.51 | 0.610 | -.6171725 | 1.05129 |
| nonwhite |  |  |  |  |  |  |
| site | -1.211563 | .4705127 | -2.57 | 0.010 | -2.133751 | -.2893747 |
| 2 |  |  |  |  |  |  |
| 3 | -.2078123 | .3662926 | -0.57 | 0.570 | -.9257327 | .510108 |
| _cons | -1.286943 | .5923219 | -2.17 | 0.030 | -2.447872 | -.1260134 |

These results suggest that the inclination of nonwhites to choose prepaid care is even stronger than it was without controlling. We also see that subjects in site 2 are less likely to be uninsured.

## Fitting constrained models

mlogit can fit models with subsets of coefficients constrained to be zero, with subsets of coefficients constrained to be equal both within and across equations, and with subsets of coefficients arbitrarily constrained to equal linear combinations of other estimated coefficients.

Before fitting a constrained model, you define the constraints with the constraint command; see $[R]$ constraint. Once the constraints are defined, you estimate using mlogit, specifying the constraint() option. Typing constraint(4) would use the constraint you previously saved as 4. Typing constraint $(1,4,6)$ would use the previously stored constraints 1,4 , and 6 . Typing constraint $(1-4,6)$ would use the previously stored constraints $1,2,3,4$, and 6 .

Sometimes, you will not be able to specify the constraints without knowing the omitted outcome. In such cases, assume that the omitted outcome is whatever outcome is convenient for you, and include the baseoutcome() option when you specify the mlogit command.

## > Example 5: Specifying constraints to test hypotheses

We can use constraints to test hypotheses, among other things. In our insurance-choice model, let's test the hypothesis that there is no distinction between having indemnity insurance and being uninsured. Indemnity-style insurance was the omitted outcome, so we type

```
. test [Uninsure]
( 1) [Uninsure] age = 0
( 2) [Uninsure]male = 0
( 3) [Uninsure]nonwhite = 0
(4) [Uninsure]1b.site = 0
( 5) [Uninsure]2.site = 0
( 6) [Uninsure]3.site = 0
    Constraint 4 dropped
    chi2( 5) = 9.31
    Prob > chi2 = 0.0973
```

If indemnity had not been the omitted outcome, we would have typed test [Uninsure=Indemnity].
The results produced by test are an approximation based on the estimated covariance matrix of the coefficients. Because the probability of being uninsured is low, the log likelihood may be nonlinear for the uninsured. Conventional statistical wisdom is not to trust the asymptotic answer under these circumstances but to perform a likelihood-ratio test instead.

To use Stata's lrtest (likelihood-ratio test) command, we must fit both the unconstrained and constrained models. The unconstrained model is the one we have previously fit. Following the instruction in $[\mathrm{R}]$ Irtest, we first store the unconstrained model results:

```
. estimates store unconstrained
```

To fit the constrained model, we must refit our model with all the coefficients except the constant set to 0 in the Uninsure equation. We define the constraint and then refit:
. constraint 1 [Uninsure]
. mlogit insure age male nonwhite i.site, constraints(1)
Iteration 0: Log likelihood $=-555.85446$
Iteration 1: Log likelihood $=-539.80523$
Iteration 2: Log likelihood $=-539.75644$
Iteration 3: Log likelihood = -539.75643
$\begin{array}{ll}\text { Multinomial logistic regression } & \text { Number of obs }=\begin{array}{r}615 \\ \end{array} \\ & \text { Wald chi2 }(5)=29.70\end{array}$
Log likelihood = -539.75643
Prob > chi2 $=0.0000$
( 1) [Uninsure]o.age $=0$
( 2) [Uninsure]o.male $=0$
( 3) [Uninsure]o.nonwhite $=0$
( 4) [Uninsure]2o.site $=0$
( 5) [Uninsure] 30. site $=0$

| insure | Coefficient | Std. err. | z | $P>\|z\|$ | [95\% con | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Indemnity | (base outcome) |  |  |  |  |  |
| Prepaid |  |  |  |  |  |  |
| age | -. 0107025 | . 0060039 | -1.78 | 0.075 | -. 0224699 | . 0010649 |
| male | . 4963616 | . 1939683 | 2.56 | 0.010 | . 1161907 | . 8765324 |
| nonwhite | . 9421369 | . 2252094 | 4.18 | 0.000 | . 5007346 | 1.383539 |
| site |  |  |  |  |  |  |
| 2 | . 2530912 | . 2029465 | 1.25 | 0.212 | -. 1446767 | . 6508591 |
| 3 | -. 5521773 | . 2187237 | -2.52 | 0.012 | -. 9808678 | -. 1234869 |
| _cons | . 1792752 | . 3171372 | 0.57 | 0.572 | -. 4423023 | . 8008527 |
| Uninsure |  |  |  |  |  |  |
| age | 0 | (omitted) |  |  |  |  |
| male | 0 | (omitted) |  |  |  |  |
| nonwhite | 0 | (omitted) |  |  |  |  |
| site |  |  |  |  |  |  |
| 2 | 0 | (omitted) |  |  |  |  |
| 3 | 0 | (omitted) |  |  |  |  |
| _cons | -1.87351 | . 1601099 | -11.70 | 0.000 | -2.18732 | -1.5597 |

We can now perform the likelihood-ratio test:

```
. lrtest unconstrained .
Likelihood-ratio test
Assumption: . nested within unconstrained
    LR chi2(5) = 10.79
Prob > chi2 = 0.0557
```

The likelihood-ratio $\chi^{2}$ is 10.79 with 5 degrees of freedom-just slightly greater than the magic $p=0.05$ level-so we should not call this difference significant.

## - Technical note

In certain circumstances, you should fit an MNL model with conditional logit; see [R] clogit. With substantial data manipulation, clogit can handle the same class of models with some interesting additions. For example, if we had available the price and deductible of the most competitive insurance plan of each type, mlogit could not use this information, but clogit could.

## Stored results

mlogit stores the following in e():
Scalars
e(N)
e(N_cd)
e(k_out)
e(k)
$e\left(k \_e q\right)$
e(k_eq_model)
e(k_dv)
e(df_m)
e(r2_p)
e(11)
e(11_0)
e(N_clust)
e(chi2)
$e(p)$
e(k_eq_base)
e(baseout)
e(ibaseout)
e(rank)
e(ic)
$e(r c)$
e(converged)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(chi2type)
e(vce)
e(vcetype)
e(eqnames)
e(baselab)
e(opt)
$e$ (which)
e(ml_method)
e(user)
e(technique)
e(properties)
e(predict)
e(marginsnotok)
$e$ (marginsdefault)
e(asbalanced)
e(asobserved)
number of observations
number of completely determined observations
number of outcomes
number of parameters
number of equations in $e(b)$
number of equations in overall model test
number of dependent variables
model degrees of freedom
pseudo- $R^{2}$
log likelihood
log likelihood, constant-only model
number of clusters
$\chi^{2}$
$p$-value for model test
equation number of the base outcome
the value of depvar to be treated as the base outcome
index of the base outcome
rank of e(v)
number of iterations
return code
1 if converged, 0 otherwise
mlogit
command as typed
name of dependent variable
weight type
weight expression
title in estimation output
name of cluster variable
Wald or LR; type of model $\chi^{2}$ test
vcetype specified in vce()
title used to label Std. err.
names of equations
value label corresponding to base outcome
type of optimization
max or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement predict
predictions disallowed by margins
default predict() specification for margins
factor variables fvset as asbalanced
factor variables fvset as asobserved

Matrices
$e(b) \quad$ coefficient vector
e (out) outcome values
e(Cns) constraints matrix
e(ilog) iteration $\log$ (up to 20 iterations)
e(gradient) gradient vector
$e(V) \quad$ variance-covariance matrix of the estimators
e(V_modelbased)
Functions
e(sample) marks estimation sample
In addition to the above, the following is stored in r() :
Matrices
$r$ (table) matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

The MNL model is described in Greene (2018, 829-833).
Suppose that there are $k$ categorical outcomes and-without loss of generality-let the base outcome be 1 . The probability that the response for the $j$ th observation is equal to the $i$ th outcome is

$$
p_{i j}=\operatorname{Pr}\left(y_{j}=i\right)=\left\{\begin{array}{ll}
\frac{1}{1+\sum_{m=2}^{k} \exp \left(\mathbf{x}_{j} \boldsymbol{\beta}_{m}\right)}, & \text { if } \\
i=1 \\
\frac{\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}_{i}\right)}{1+\sum_{m=2}^{k} \exp \left(\mathbf{x}_{j} \boldsymbol{\beta}_{m}\right)}, & \text { if }
\end{array} \quad i>1\right.
$$

where $\mathbf{x}_{j}$ is the row vector of observed values of the independent variables for the $j$ th observation and $\boldsymbol{\beta}_{m}$ is the coefficient vector for outcome $m$. The log pseudolikelihood is

$$
\ln L=\sum_{j} w_{j} \sum_{i=1}^{k} I_{i}\left(y_{j}\right) \ln p_{i k}
$$

where $w_{j}$ is an optional weight and

$$
I_{i}\left(y_{j}\right)=\left\{\begin{array}{l}
1, \text { if } y_{j}=i \\
0, \text { otherwise }
\end{array}\right.
$$

Newton-Raphson maximum likelihood is used; see [R] Maximize.
For constrained equations, the set of constraints is orthogonalized, and a subset of maximizable parameters is selected. For example, a parameter that is constrained to zero is not a maximizable parameter. If two parameters are constrained to be equal to each other, only one is a maximizable parameter.

Let $\mathbf{r}$ be the vector of maximizable parameters. $\mathbf{r}$ is physically a subset of the solution parameters, b. A matrix, $\mathbf{T}$, and a vector, $\mathbf{m}$, are defined as

$$
\mathbf{b}=\operatorname{Tr}+\mathbf{m}
$$

so that

$$
\begin{aligned}
\frac{\partial f}{\partial \mathbf{b}} & =\frac{\partial f}{\partial \mathbf{r}} \mathbf{T}^{\prime} \\
\frac{\partial^{2} f}{\partial \mathbf{b}^{2}} & =\mathbf{T} \frac{\partial^{2} f}{\partial \mathbf{r}^{2}} \mathbf{T}^{\prime}
\end{aligned}
$$

$\mathbf{T}$ consists of a block form in which one part is a permutation of the identity matrix and the other part describes how to calculate the constrained parameters from the maximizable parameters.

This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce(robust) and vce(cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.
mlogit also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

## References

Davidson, R., and J. G. MacKinnon. 1993. Estimation and Inference in Econometrics. New York: Oxford University Press.

Greene, W. H. 2018. Econometric Analysis. 8th ed. New York: Pearson.
Haan, P., and A. Uhlendorff. 2006. Estimation of multinomial logit models with unobserved heterogeneity using maximum simulated likelihood. Stata Journal 6: 229-245.

Hole, A. R. 2007. Fitting mixed logit models by using maximum simulated likelihood. Stata Journal 7: 388-401.
Hosmer, D. W., Jr., S. A. Lemeshow, and R. X. Sturdivant. 2013. Applied Logistic Regression. 3rd ed. Hoboken, NJ: Wiley.

Long, J. S. 1997. Regression Models for Categorical and Limited Dependent Variables. Thousand Oaks, CA: Sage.
Long, J. S., and J. Freese. 2014. Regression Models for Categorical Dependent Variables Using Stata. 3rd ed. College Station, TX: Stata Press.

Tarlov, A. R., J. E. Ware, Jr., S. Greenfield, E. C. Nelson, E. Perrin, and M. Zubkoff. 1989. The medical outcomes study. An application of methods for monitoring the results of medical care. Journal of the American Medical Association 262: 925-930. https://doi.org/10.1001/jama.1989.03430070073033.
Treiman, D. J. 2009. Quantitative Data Analysis: Doing Social Research to Test Ideas. San Francisco: Jossey-Bass.
Vach, W., C. Alder, and S. Pichler. 2022. Analyzing coarsened categorical data with or without probabilistic information. Stata Journal 22: 158-194.

Wells, K. B., R. D. Hays, M. A. Burnam, W. H. Rogers, S. Greenfield, and J. E. Ware, Jr. 1989. Detection of depressive disorder for patients receiving prepaid or fee-for-service care. Results from the Medical Outcomes Survey. Journal of the American Medical Association 262: 3298-3302. https://doi.org/10.1001/jama.1989.03430230083030.

Xu, J., and J. S. Long. 2005. Confidence intervals for predicted outcomes in regression models for categorical outcomes. Stata Journal 5: 537-559.

## Also see

[R] mlogit postestimation - Postestimation tools for mlogit
[R] clogit - Conditional (fixed-effects) logistic regression
$[\mathrm{R}]$ logistic - Logistic regression, reporting odds ratios
[R] logit - Logistic regression, reporting coefficients
[R] mprobit - Multinomial probit regression
$[\mathrm{R}]$ ologit - Ordered logistic regression
[R] slogit - Stereotype logistic regression
[BAYES] bayes: mlogit - Bayesian multinomial logistic regression
[CM] cmrologit — Rank-ordered logit choice model
[CM] nlogit - Nested logit regression
[FMM] fmm: mlogit - Finite mixtures of multinomial (polytomous) logistic regression models
[MI] Estimation - Estimation commands for use with mi estimate
[SVY] svy estimation - Estimation commands for survey data
[XT] xtmlogit - Fixed-effects and random-effects multinomial logit models
[U] 20 Estimation and postestimation commands

## Title

mlogit postestimation - Postestimation tools for mlogit

| Postestimation commands | predict | margins |
| :--- | :--- | :--- |
| Reference | Also see |  |

## Postestimation commands

The following postestimation commands are available after mlogit:

| Command | Description |
| :---: | :---: |
| contrast | contrasts and ANOVA-style joint tests of estimates |
| estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estat (svy) | postestimation statistics for survey data |
| estimates | cataloging estimation results |
| etable | table of estimation results |
| * forecast | dynamic forecasts and simulations |
| * hausman | Hausman's specification test |
| lincom | point estimates, standard errors, testing, and inference for linear combinations of coefficients |
| * 1 rtest | likelihood-ratio test |
| margins | marginal means, predictive margins, marginal effects, and average marginal effects |
| marginsplot | graph the results from margins (profile plots, interaction plots, etc.) |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients |
| predict | probabilities, linear predictions and their SEs, etc. |
| predictnl | point estimates, standard errors, testing, and inference for generalized predictions |
| pwcompare | pairwise comparisons of estimates |
| suest | seemingly unrelated estimation |
| test | Wald tests of simple and composite linear hypotheses |
| testnl | Wald tests of nonlinear hypotheses |

[^33]
## predict

## Description for predict

predict creates a new variable containing predictions such as probabilities, linear predictions, and standard errors.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

predict $[$ type $]\{$ stub* $\mid$ newvar $\mid$ newvarlist $\}[i f][$ in $][$, statistic outcome(outcome)]
predict $[$ type $]$ stub* $[i f][i n]$, scores
statistic Description
Main
$\mathrm{pr} \quad$ predicted probabilities; the default
$\mathrm{xb} \quad$ linear prediction
stdp standard error of the linear prediction
stddp standard error of the difference in two linear predictions
You specify one or $k$ new variables with pr , where $k$ is the number of outcomes. If you specify one new variable and you do not specify outcome(), then outcome(\#1) is assumed.
You specify one new variable with xb , stdp, and stddp. If you do not specify outcome(), then outcome(\#1) is assumed. You must specify outcome() with the stddp option.
These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

## Options for predict

Main
pr , the default, computes the predicted probabilities for all outcomes or for a specific outcome. To compute probabilities for all outcomes, you specify $k$ new variables, where $k$ is the number of categories of the dependent variable. Alternatively, you can specify stub*; in which case, pr will store predicted probabilities in variables stub1, stub2, ..., stubk. To compute the probability for a specific outcome, you specify one new variable and, optionally, the outcome value in option outcome(); if you omit outcome(), the first outcome value, outcome(\#1), is assumed.

Say that you fit a model by typing estimation_cmd y x1 x2, and y takes on four values. Then, you could type predict p1 p2 p3 p4 to obtain all four predicted probabilities; alternatively, you could type predict p* to generate the four predicted probabilities. To compute specific probabilities one at a time, you can type predict p1, outcome(\#1) (or simply predict p1), predict p2, outcome(\#2), and so on. See option outcome() for other ways to refer to outcome values.
xb calculates the linear prediction. You must also specify the outcome (outcome) option.
stdp calculates the standard error of the linear prediction. You must also specify the outcome (outcome) option.
stddp calculates the standard error of the difference in two linear predictions. You must specify the outcome (outcome) option, and here you specify the two particular outcomes of interest inside the parentheses, for example, predict sed, stddp outcome $(1,3)$.
outcome(outcome) specifies for which outcome the predicted probabilities are to be calculated. outcome() should contain either one value of the dependent variable or one of $\# 1, \# 2, \ldots$, with \#1 meaning the first category of the dependent variable, \#2 meaning the second category, etc. outcome() is not allowed with scores.
scores calculates equation-level score variables. The number of score variables created will be one less than the number of outcomes in the model. If the number of outcomes in the model were $k$, then
the first new variable will contain $\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}_{1}\right)$;
the second new variable will contain $\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}_{2}\right)$;
the $(k-1)$ th new variable will contain $\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}_{k-1}\right)$.

## margins

## Description for margins

margins estimates margins of response for probabilities and linear predictions.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...) ...] [options]
```

statistic
Description

| default | probabilities for each outcome |
| :--- | :--- |
| pr | probability for a specified outcome |
| xb | linear prediction for a specified outcome |
| stdp | not allowed with margins |
| stddp | not allowed with margins |

pr and xb default to the first outcome.
Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

## Remarks and examples

Remarks are presented under the following headings:
Obtaining predicted values
Calculating marginal effects
Testing hypotheses about coefficients

## Obtaining predicted values

## > Example 1: Obtaining predicted probabilities

After estimation, we can use predict to obtain predicted probabilities, index values, and standard errors of the index, or differences in the index. For instance, in example 4 of [ R ] mlogit, we fit a model of insurance choice on various characteristics. We can obtain the predicted probabilities for outcome 1 by typing

```
. use https://www.stata-press.com/data/r18/sysdsn1
(Health insurance data)
. mlogit insure age i.male i.nonwhite i.site
    (output omitted)
. predict p1 if e(sample), outcome(1)
(option pr assumed; predicted probability)
(29 missing values generated)
```

. summarize p1

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| p1 | 615 | .4764228 | .1032279 | .1698142 | .71939 |

We added the i. prefix to the male, nonwhite, and site variables to explicitly identify them as factor variables. That makes no difference in the estimated results, but we will take advantage of it in later examples. We also included if e (sample) to restrict the calculation to the estimation sample. In example 4 of [R] mlogit, the multinomial logit model was fit on 615 observations, so there must be missing values in our dataset.

Although we typed outcome(1), specifying 1 for the indemnity outcome, we could have typed outcome (Indemnity). For instance, to obtain the probabilities for prepaid, we could type

```
. predict p2 if e(sample), outcome(Prepaid)
(option pr assumed; predicted probability)
(29 missing values generated)
. summarize p2
\begin{tabular}{r|rrrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline p2 & 615 & .4504065 & .1125962 & .1964103 & .7885724
\end{tabular}
```

We must specify the label exactly as it appears in the underlying value label (or how it appears in the mlogit output), including capitalization.

Here we have used predict to obtain probabilities for the same sample on which we estimated. That is not necessary. We could use another dataset that had the independent variables defined (in our example, age, male, nonwhite, and site) and use predict to obtain predicted probabilities; here, we would not specify if e(sample).

## > Example 2: Obtaining index values

predict can also be used to obtain the index values - the $\sum x_{i} \widehat{\beta}_{i}^{(k)}$ — as well as the probabilities:

```
. predict idx1, outcome(Indemnity) xb
(1 missing value generated)
```

. summarize idx1

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | :---: | :---: | :---: | :---: | ---: |
| idx1 | 643 | 0 | 0 | 0 | 0 |

The indemnity outcome was our base outcome - the outcome for which all the coefficients were set to 0 -so the index is always 0 . For the prepaid and uninsured outcomes, we type

```
. predict idx2, outcome(Prepaid) xb
(1 missing value generated)
. predict idx3, outcome(Uninsure) xb
(1 missing value generated)
. summarize idx2 idx3
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline idx2 & 643 & -.0566113 & .4962973 & -1.298198 & 1.700719 \\
idx3 & 643 & -1.980747 & .6018139 & -3.112741 & -.8258458
\end{tabular}
```

We can obtain the standard error of the index by specifying the stdp option:

```
. predict se2, outcome(Prepaid) stdp
(1 missing value generated)
. list p2 idx2 se2 in 1/5
```

| p2 | idx2 | se2 |
| ---: | ---: | ---: |
| .3709022 | -.4831167 | .2437772 |
| .4977667 | .055111 | .1694686 |
| .4113073 | -.1712106 | .1793498 |
| .5424927 | .3788345 | .2513701 |
| . | -.0925817 | .1452616 |

We obtained the probability, p2, in the previous example.

Finally, predict can calculate the standard error of the difference in the index values between two outcomes with the stddp option:

```
. predict se_2_3, outcome(Prepaid,Uninsure) stddp
(1 missing value generated)
. list idx2 idx3 se_2_3 in 1/5
```

| idx2 | idx3 | se_2_3 |
| ---: | ---: | ---: |
| -.4831167 | -3.073253 | .5469354 |
| .055111 | -2.715986 | .4331918 |
| -.1712106 | -1.579621 | .3053815 |
| .3788345 | -1.462007 | .4492552 |
| -.0925817 | -2.814022 | .4024784 |

In the first observation, the difference in the indexes is $-0.483-(-3.073)=2.59$. The standard error of that difference is 0.547 .

## > Example 3: Interpreting results using predictive margins

It is more difficult to interpret the results from mlogit than those from clogit or logit because there are multiple equations. For example, suppose that one of the independent variables in our model takes on the values 0 and 1 , and we are attempting to understand the effect of this variable. Assume that the coefficient on this variable for the second outcome, $\beta^{(2)}$, is positive. We might then be tempted to reason that the probability of the second outcome is higher if the variable is 1 rather than 0 . Most of the time, that will be true, but occasionally we will be surprised. The probability of some other outcome could increase even more (say, $\beta^{(3)}>\beta^{(2)}$ ), and thus the probability of outcome 2 would actually fall relative to that outcome. We can use predict to help interpret such results.

Continuing with our previously fit insurance-choice model, we wish to describe the model's predictions by race. For this purpose, we can use the method of predictive margins (also known as recycled predictions), in which we vary characteristics of interest across the whole dataset and average the predictions. That is, we have data on both whites and nonwhites, and our individuals have other characteristics as well. We will first pretend that all the people in our data are white but hold their other characteristics constant. We then calculate the probabilities of each outcome. Next we will pretend that all the people in our data are nonwhite, still holding their other characteristics constant. Again, we calculate the probabilities of each outcome. The difference in those two sets of calculated probabilities, then, is the difference due to race, holding other characteristics constant.

```
. gen byte nonwhold=nonwhite // save real race
. replace nonwhite=0 // make everyone white
(126 real changes made)
. predict wpind, outcome(Indemnity) // predict probabilities
(option pr assumed; predicted probability)
(1 missing value generated)
. predict wpp, outcome(Prepaid)
(option pr assumed; predicted probability)
(1 missing value generated)
. predict wpnoi, outcome(Uninsure)
(option pr assumed; predicted probability)
(1 missing value generated)
. replace nonwhite=1 // make everyone nonwhite
(644 real changes made)
```

```
. predict nwpind, outcome(Indemnity)
(option pr assumed; predicted probability)
(1 missing value generated)
. predict nwpp, outcome(Prepaid)
(option pr assumed; predicted probability)
(1 missing value generated)
. predict nwpnoi, outcome(Uninsure)
(option pr assumed; predicted probability)
(1 missing value generated)
. replace nonwhite=nonwhold // restore real race
(518 real changes made)
```

. summarize wp* nwp*, sep(3)

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| wpind | 643 | .5141673 | .0872679 | .3092903 | .71939 |
| wpp | 643 | .4082052 | .0993286 | .1964103 | .6502247 |
| wpnoi | 643 | .0776275 | .0360283 | .0273596 | .1302816 |
| nwpind | 643 | .3112809 | .0817693 | .1511329 | .535021 |
| nwpp | 643 | .630078 | .0979976 | .3871782 | .8278881 |
| nwpnoi | 643 | .0586411 | .0287185 | .0209648 | .0933874 |

In example 1 of $[\mathrm{R}]$ mlogit, we presented a cross-tabulation of insurance type and race. Those values were unadjusted. The means reported above are the values adjusted for age, sex, and site. Combining the results gives

|  | Unadjusted |  | Adjusted |  |
| :--- | :---: | :---: | :---: | :---: |
|  | white | nonwhite | white | nonwhite |
| Indemnity | 0.51 | 0.36 | 0.51 | 0.31 |
| Prepaid | 0.42 | 0.57 | 0.41 | 0.63 |
| Uninsured | 0.07 | 0.07 | 0.08 | 0.06 |

We find, for instance, after adjusting for age, sex, and site, that although $57 \%$ of nonwhites in our data had prepaid plans, $63 \%$ of nonwhites chose prepaid plans.

Computing predictive margins by hand was instructive, but we can compute these values more easily using the margins command (see [R] margins). The two margins for the indemnity outcome can be estimated by typing

| Predictive margins <br> Model VCE: OIM |  |  |  |  | Number of obs $=643$ |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Expression: Pr(insure==Indemnity), predict(outcome(Indemnity)) |  |  |  |  |  |  |
|  | Margin | lta-meth std. err | z | $P>\|z\|$ | [95\% conf | interval] |
| nonwhite |  |  |  |  |  |  |
| 0 | . 5141673 | . 0223485 | 23.01 | 0.000 | . 470365 | . 5579695 |
| 1 | . 3112809 | . 0418049 | 7.45 | 0.000 | . 2293448 | . 393217 |

margins also estimates the standard errors and confidence intervals of the margins. By default, margins uses only the estimation sample. We added the noesample option so that margins would use the entire sample and produce results comparable with our earlier analysis.

We can use marginsplot to graph the results from margins:

```
. marginsplot
Variables that uniquely identify margins: nonwhite
```

Predictive margins of nonwhite with $95 \%$ Cls


The margins for the other two outcomes can be computed by typing

```
. margins nonwhite, predict(outcome(Prepaid)) noesample
    (output omitted)
. margins nonwhite, predict(outcome(Uninsure)) noesample
    (output omitted)
```

The margins for each outcome is computed when no outcome is specified. For example,

```
. margins nonwhite, noesample
(output omitted)
```


## - Technical note

You can use predict to classify predicted values and compare them with the observed outcomes to interpret a multinomial logit model. This is a variation on the notions of sensitivity and specificity for logistic regression. Here we will classify indemnity and prepaid as definitely predicting indemnity, definitely predicting prepaid, and ambiguous.

```
. predict indem, outcome(Indemnity) index // obtain indexes
(1 missing value generated)
. predict prepaid, outcome(Prepaid) index
(1 missing value generated)
. gen diff = prepaid-indem // obtain difference
(1 missing value generated)
. predict sediff, outcome(Indemnity,Prepaid) stddp // & its standard error
(1 missing value generated)
. gen type = 1 if diff/sediff < -1.96 // definitely indemnity
(504 missing values generated)
. replace type = 3 if diff/sediff > 1.96
// definitely prepaid
(100 real changes made)
```

| . replace type $=2$ if type>=. \& diff/sediff < . (404 real changes made) |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: |
| . label def <br> . label val <br> . tabulate | pe 1 "Def type type ure type | Ind" 2 "Am | iguous" 3 | Prep" |
| Type of insurance | Def Ind | type <br> Ambiguous | Def Prep | Total |
| Indemnity | 78 | 183 | 33 | 294 |
| Prepaid | 44 | 177 | 56 | 277 |
| Uninsure | 12 | 28 | 5 | 45 |
| Total | 134 | 388 | 94 | 616 |

We can see that the predictive power of this model is modest. There are many misclassifications in both directions, though there are more correctly classified observations than misclassified observations.

Also, the uninsured look overwhelmingly as though they might have come from the indemnity system rather than from the prepaid system.

## Calculating marginal effects

## > Example 4

We have already noted that the coefficients from multinomial logit can be difficult to interpret because they are relative to the base outcome. Another way to evaluate the effect of covariates is to examine the marginal effect of changing their values on the probability of observing an outcome.

The margins command can be used for this too. We can estimate the marginal effect of each covariate on the probability of observing the first outcome-indemnity insurance-by typing

| Average marginal effects Model VCE: OIM |  |  | Number of obs $=615$ |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Expression: $\operatorname{Pr}$ (insure==Indemnity), predict(outcome(Indemnity)) dy/dx wrt: age 1.male 1.nonwhite 2.site 3.site |  |  |  |  |  |  |
|  | Delta-metho $d y / d x$ std. err. |  | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| age | . 0026655 | . 001399 | 1.91 | 0.057 | -. 0000765 | . 0054074 |
| 1.male | -. 1295734 | . 0450945 | -2.87 | 0.004 | -. 2179571 | -. 0411898 |
| 1.nonwhite | -. 2032404 | . 0482554 | -4.21 | 0.000 | -. 2978192 | -. 1086616 |
| site |  |  |  |  |  |  |
| 2 | . 0070995 | . 0479993 | 0.15 | 0.882 | -. 0869775 | . 1011765 |
| 3 | . 1216165 | . 0505833 | 2.40 | 0.016 | . 022475 | . 220758 |

Note: dy/dx for factor levels is the discrete change from the base level.
By default, margins estimates the average marginal effect over the estimation sample, and that is what we see above. Being male decreases the average probability of having indemnity insurance by 0.130. We also see, from the note at the bottom of the table, that the marginal effect was computed as a discrete change in the probability of being male rather than female. That is why we made male a factor variable when fitting the model.

The $\operatorname{dydx}(*)$ option requested that margins estimate the marginal effect for each regressor, dydx (age) would have produced estimates only for the effect of age. margins has many options for controlling how the marginal effect is computed, including the ability to average over subgroups or to compute estimates for specified values of the regressors; see [R] margins.
margins will compute the marginal effects on each outcome when no outcome is specified.
margins, dydx(*)
(output omitted)

## Testing hypotheses about coefficients

## > Example 5

test tests hypotheses about the coefficients just as after any estimation command; see [R] test. Note, however, test's syntax for dealing with multiple-equation models. Because test bases its results on the estimated covariance matrix, we might prefer a likelihood-ratio test; see example 5 in [R] mlogit for an example of lrtest.

If we simply list variables after the test command, we are testing that the corresponding coefficients are zero across all equations:

```
. test 2.site 3.site
( 1) [Indemnity]2o.site = 0
( 2) [Prepaid]2.site = 0
( 3) [Uninsure]2.site = 0
(4) [Indemnity]3o.site = 0
( 5) [Prepaid]3.site = 0
( 6) [Uninsure]3.site = 0
    Constraint 1 dropped
    Constraint 4 dropped
            chi2( 4) = 19.74
        Prob > chi2 = 0.0006
```

We can test that all the coefficients (except the constant) in an equation are zero by simply typing the outcome in square brackets:

```
. test [Uninsure]
( 1) [Uninsure] age = 0
( 2) [Uninsure]0b.male = 0
( 3) [Uninsure]1.male = 0
(4) [Uninsure]0b.nonwhite = 0
( 5) [Uninsure]1.nonwhite = 0
( 6) [Uninsure]1b.site = 0
( 7) [Uninsure]2.site = 0
( 8) [Uninsure]3.site = 0
    Constraint 2 dropped
        Constraint 4 dropped
        Constraint 6 dropped
            chi2( 5) = 9.31
            Prob > chi2 = 0.0973
```

We specify the outcome just as we do with predict; we can specify the label if the outcome variable is labeled, or we can specify the numeric value of the outcome. We would have obtained the same test as above if we had typed test [3] because 3 is the value of insure for the outcome uninsured.

We can combine the two syntaxes. To test that the coefficients on the site variables are 0 in the equation corresponding to the outcome prepaid, we can type

```
. test [Prepaid]: 2.site 3.site
( 1) [Prepaid]2.site = 0
( 2) [Prepaid]3.site = 0
    chi2( 2) = 10.78
    Prob > chi2 = 0.0046
```

We specified the outcome and then followed that with a colon and the variables we wanted to test.
We can also test that coefficients are equal across equations. To test that all coefficients except the constant are equal for the prepaid and uninsured outcomes, we can type

```
. test [Prepaid=Uninsure]
( 1) [Prepaid]age - [Uninsure] age = 0
( 2) [Prepaid]0b.male - [Uninsure]Ob.male = 0
( 3) [Prepaid]1.male - [Uninsure]1.male = 0
(4) [Prepaid]0b.nonwhite - [Uninsure]0b.nonwhite = 0
( 5) [Prepaid]1.nonwhite - [Uninsure]1.nonwhite = 0
( 6) [Prepaid]1b.site - [Uninsure]1b.site = 0
( 7) [Prepaid]2.site - [Uninsure]2.site = 0
( 8) [Prepaid]3.site - [Uninsure]3.site = 0
    Constraint 2 dropped
        Constraint 4 dropped
        Constraint 6 dropped
            chi2( 5) = 13.80
            Prob > chi2 = 0.0169
```

To test that only the site variables are equal, we can type

```
. test [Prepaid=Uninsure]: 2.site 3.site
( 1) [Prepaid]2.site - [Uninsure]2.site = 0
( 2) [Prepaid]3.site - [Uninsure]3.site = 0
    chi2( 2) = 12.68
    Prob > chi2 = 0.0018
```

Finally, we can test any arbitrary constraint by simply entering the equation and specifying the coefficients as described in [U] 13.5 Accessing coefficients and standard errors. The following hypothesis is senseless but illustrates the point:

```
. test ([Prepaid]age+[Uninsure]2.site)/2 = 2-[Uninsure]1.nonwhite
( 1) . 5*[Prepaid]age + [Uninsure]1.nonwhite + .5*[Uninsure]2.site = 2
    chi2( 1) = 22.45
    Prob > chi2 = 0.0000
```

See [R] test for more information about test. The information there about combining hypotheses across test commands (the accumulate option) also applies after mlogit.

## Reference

Fagerland, M. W., and D. W. Hosmer, Jr. 2012. A generalized Hosmer-Lemeshow goodness-of-fit test for multinomial logistic regression models. Stata Journal 12: 447-453.

## Also see

[R] mlogit - Multinomial (polytomous) logistic regression
[U] 20 Estimation and postestimation commands

## Title

more - The -more - message

## Description Syntax Option Remarks and examples Also see

## Description

set more off, which is the default, tells Stata not to pause or display a -more- message. set more on tells Stata to wait until you press a key before continuing when a -more- message is displayed.
set pagesize \# sets the number of lines between -more- messages.

## Syntax

Tell Stata to pause or not pause for -more- messages
set more $\{$ on $\mid$ off $\}[$, permanently $]$
Set number of lines between -more- messages
set pagesize \#

## Option

permanently specifies that, in addition to making the change right now, the more setting be remembered and become the default setting when you invoke Stata.

## Remarks and examples

When you see -more- at the bottom of the screen,

| Press $\ldots$ | and Stata ... |
| :--- | :--- |
| letter $l$ or Enter | displays the next line |
| letter $q$ | acts as if you pressed Break |
| Spacebar or any other key | displays the next screen |

You can also click on the More button or click on -more- to display the next screen.
-more- is Stata's way of telling you that it has something more to show you but that showing it to you will cause the information on the screen to scroll off.

If you type set more off, -more- conditions will never arise, and Stata's output will scroll by at full speed.

If you type set more on, -more- conditions will be restored at the appropriate places.
If set more is used within a do-file or program, Stata automatically restores the previous set more setting when the do-file or program concludes.

Programmers should see $[\mathrm{P}]$ more for information on the more programming command.

## Also see

[R] query - Display system parameters
[P] creturn - Return c-class values
[ P ] more - Pause until key is pressed
[P] sleep - Pause for a specified time
[U] 7 -more- conditions

## Title

mprobit - Multinomial probit regression

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

mprobit fits a multinomial probit (MNP) model for a categorical dependent variable with outcomes that have no natural ordering. The actual values taken by the dependent variable are irrelevant. The error terms are assumed to be independent, standard normal, random variables. cmmprobit relaxes the independence of irrelevant alternatives assumption by specifying correlated latent-variable errors. cmmprobit also allows heteroskedastic latent-variable errors and alternative-specific independent variables.

## Quick start

Multinomial probit model of y on x 1 , x 2 , and categorical a mprobit y x1 x2 i.a

Same as above, but use as the base outcome $\mathrm{y}=3$
mprobit y x1 x2 i.a, baseoutcome(3)
Probit variance parameterization of differenced latent errors
mprobit y x1 x2 i.a, probitparam
Multiple-imputation estimates with Monte Carlo errors from mi set data mi estimate, mcerror: mprobit y x1 x2 i.a

## Menu

Statistics $>$ Categorical outcomes $>$ Multinomial probit regression

## Syntax

mprobit depvar [indepvars] [if] [in] [weight] [, options]
options
Description

## Model

noconstant
baseoutcome (\#|lbl)
probitparam
constraints (constraints)
SE/Robust
vce (vcetype)

Reporting
level (\#)
nocnsreport
display_options

Integration
intpoints(\#)
Maximization
maximize_options
collinear
coeflegend
suppress constant terms
outcome used to normalize location
use the probit variance parameterization
apply specified linear constraints
vcetype may be oim, robust, cluster clustvar, opg, bootstrap, or jackknife
set confidence level; default is level (95)
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
bayes, bootstrap, by, collect, fp, jackknife, mi estimate, rolling, statsby, and svy are allowed; see
[U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: mprobit.
vce (bootstrap) and vce(jackknife) are not allowed with the mi estimate prefix; see [MI] mi estimate.
Weights are not allowed with the bootstrap prefix; see [R] bootstrap.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

$\qquad$ Model
noconstant suppresses the $J-1$ constant terms.
baseoutcome (\#|lbl) specifies the outcome used to normalize the location of the latent variable. The base outcome may be specified as a number or a label. The default is to use the most frequent outcome. The coefficients associated with the base outcome are zero.
probitparam specifies to use the probit variance parameterization by fixing the variance of the differenced latent errors between the scale and the base alternatives to be one. The default is to
make the variance of the base and scale latent errors one, thereby making the variance of the difference to be two.
constraints (constraints); see [R] Estimation options.

SE/Robust
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.

If specifying vce(bootstrap) or vce(jackknife), you must also specify baseoutcome().

```
Reporting
level(\#), nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.
```

Integration
intpoints(\#) specifies the number of Gaussian quadrature points to use in approximating the likelihood. The default is intpoints (15).

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate (\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from (init_specs); see [R] Maximize. These options are seldom used.

Setting the optimization type to technique (bhhh) resets the default vcetype to vce (opg).
The following options are available with mprobit but are not shown in the dialog box:
collinear, coeflegend; see [R] Estimation options.

## Remarks and examples

The MNP model is used with discrete dependent variables that take on more than two outcomes that do not have a natural ordering. The stochastic error terms for this implementation of the model are assumed to have independent, standard normal distributions. To use mprobit, you must have one observation for each decision maker in the sample. See [CM] emmprobit for another implementation of the MNP model that permits correlated and heteroskedastic errors and is suitable when you have data for each alternative that a decision maker faced.

The MNP model is frequently motivated using a latent-variable framework. The latent variable for the $j$ th alternative, $j=1, \ldots, J$, is

$$
\eta_{i j}=\mathbf{z}_{i} \boldsymbol{\alpha}_{j}+\xi_{i j}
$$

where the $1 \times q$ row vector $\mathbf{z}_{i}$ contains the observed independent variables for the $i$ th decision maker. Associated with $\mathbf{z}_{i}$ are the $J$ vectors of regression coefficients $\boldsymbol{\alpha}_{j}$. The $\xi_{i, 1}, \ldots, \xi_{i, J}$ are distributed independently and identically standard normal. The decision maker chooses the alternative $k$ such that $\eta_{i k} \geq \eta_{i m}$ for $m \neq k$.

Suppose that case $i$ chooses alternative $k$, and take the difference between latent variable $\eta_{i k}$ and the $J-1$ others:

$$
\begin{align*}
v_{i j k} & =\eta_{i j}-\eta_{i k} \\
& =\mathbf{z}_{i}\left(\boldsymbol{\alpha}_{j}-\boldsymbol{\alpha}_{k}\right)+\xi_{i j}-\xi_{i k}  \tag{1}\\
& =\mathbf{z}_{i} \boldsymbol{\gamma}_{j^{\prime}}+\epsilon_{i j^{\prime}}
\end{align*}
$$

where $j^{\prime}=j$ if $j<k$ and $j^{\prime}=j-1$ if $j>k$ so that $j^{\prime}=1, \ldots, J-1 . \operatorname{Var}\left(\epsilon_{i j^{\prime}}\right)=\operatorname{Var}\left(\xi_{i j}-\xi_{i k}\right)=2$ and $\operatorname{Cov}\left(\epsilon_{i j^{\prime}}, \epsilon_{i l^{\prime}}\right)=1$ for $j^{\prime} \neq l^{\prime}$. The probability that alternative $k$ is chosen is

$$
\begin{aligned}
\operatorname{Pr}(i \text { chooses } k) & =\operatorname{Pr}\left(v_{i 1 k} \leq 0, \ldots, v_{i, J-1, k} \leq 0\right) \\
& =\operatorname{Pr}\left(\epsilon_{i 1} \leq-\mathbf{z}_{i} \gamma_{1}, \ldots, \epsilon_{i, J-1} \leq-\mathbf{z}_{i} \gamma_{J-1}\right)
\end{aligned}
$$

Hence, evaluating the likelihood function involves computing probabilities from the multivariate normal distribution. That all the covariances are equal simplifies the problem somewhat; see Methods and formulas for details.

In (1), not all $J$ of the $\boldsymbol{\alpha}_{j}$ are identifiable. To remove the indeterminacy, $\boldsymbol{\alpha}_{l}$ is set to the zero vector, where $l$ is the base outcome as specified in the baseoutcome() option. That fixes the lth latent variable to zero so that the remaining variables measure the attractiveness of the other alternatives relative to the base.

## > Example 1

As discussed in example 1 of $[\mathrm{R}]$ mlogit, we have data on the type of health insurance available to 616 psychologically depressed subjects in the United States (Tarlov et al. 1989; Wells et al. 1989). Patients may have either an indemnity (fee-for-service) plan or a prepaid plan such as an HMO, or the patient may be uninsured. Demographic variables include age, gender, race, and site. Indemnity insurance is the most popular alternative, so mprobit will choose it as the base outcome by default.

```
. use https://www.stata-press.com/data/r18/sysdsn1
(Health insurance data)
. mprobit insure age male nonwhite i.site
Iteration 0: Log likelihood \(=-535.89424\)
Iteration 1: Log likelihood \(=-534.56173\)
Iteration 2: Log likelihood \(=-534.52835\)
Iteration 3: Log likelihood \(=-534.52833\)
Multinomial probit regression \(\quad\) Number of obs \(=615\)
Log likelihood \(=-534.52833\)
Log likelihood \(=-534.52833 \quad\) Prob > chi2 \(=0.0000\)
```

Wald chi2(10) $=40.18$

| insure | Coefficient | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% con | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Indemnity | (base outcome) |  |  |  |  |  |
| Prepaid |  |  |  |  |  |  |
| age | -. 0098536 | . 0052688 | -1.87 | 0.061 | -. 0201802 | . 000473 |
| male | . 4774678 | . 1718316 | 2.78 | 0.005 | . 1406841 | . 8142515 |
| nonwhite | . 8245003 | . 1977582 | 4.17 | 0.000 | . 4369013 | 1.212099 |
| site |  |  |  |  |  |  |
| 2 | . 0973956 | . 1794546 | 0.54 | 0.587 | -. 2543289 | . 4491201 |
| 3 | -. 495892 | . 1904984 | -2.60 | 0.009 | -. 869262 | -. 1225221 |
| _cons | . 22315 | . 2792424 | 0.80 | 0.424 | -. 324155 | . 7704549 |
| Uninsure |  |  |  |  |  |  |
| age | -. 0050814 | . 0075327 | -0.67 | 0.500 | -. 0198452 | . 0096823 |
| male | . 3332637 | . 2432986 | 1.37 | 0.171 | -. 1435929 | . 8101203 |
| nonwhite | . 2485859 | . 2767734 | 0.90 | 0.369 | -. 29388 | . 7910518 |
| site |  |  |  |  |  |  |
| 2 | -. 6899485 | . 2804497 | -2.46 | 0.014 | -1.23962 | -. 1402771 |
| 3 | -. 1788447 | . 2479898 | -0.72 | 0.471 | -. 6648957 | . 3072063 |
| _cons | -. 9855917 | . 3891873 | -2.53 | 0.011 | -1.748385 | -. 2227986 |

The likelihood function for mprobit is derived under the assumption that all decision-making units face the same choice set, which is the union of all outcomes observed in the dataset. If that is not true for your model, then an alternative is to use the cmmprobit command, which does not require this assumption. To do that, you will need to expand the dataset so that each decision maker has $k_{i}$ observations, where $k_{i}$ is the number of alternatives in the choice set faced by decision maker $i$. You will also need to create a binary variable to indicate the choice made by each decision maker. Moreover, you will need to use the correlation(independent) and stddev(homoskedastic) options with cmmprobit unless you have alternative-specific variables.

## Stored results

mprobit stores the following in e():
Scalars
e(N)
e(k_out)
e(k_points)
e(k)
$e\left(k \_\right.$eq)
e(k_eq_model)
e(k_indvars)
e(k_dv)
e(df_m)
e(11)
e(N_clust)
e(chi2)
$e(p)$
e(k_eq_base)
e(baseout)
e(ibaseout)
e(const)
e(probitparam)
e(rank)
$e(i c)$
e(rc)
e(converged)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(indvars)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(chi2type)
e(vce)
e(vcetype)
e(outeqs)
e(out\#)
e(opt)
$e$ (which)
e(ml_method)
e(user)
e(technique)
$e$ (properties)
e(predict)
e(marginsnotok)
$e$ (marginsdefault)
e(asbalanced)
e(asobserved)
Matrices
e(b)
e(outcomes)
e(Cns)
e(ilog)
e(gradient)
e(V)
e(V_modelbased)
Functions
e(sample)
number of observations
number of outcomes
number of quadrature points
number of parameters
number of equations in $e(b)$
number of equations in overall model test
number of independent variables
number of dependent variables
model degrees of freedom
log likelihood
number of clusters
$\chi^{2}$
$p$-value for model test
equation number of the base outcome
the value of depvar to be treated as the base outcome
index of the base outcome
0 if noconstant is specified, 1 otherwise
1 if probitparam is specified, 0 otherwise
rank of e(V)
number of iterations
return code
1 if converged, 0 otherwise
mprobit
command as typed
name of dependent variable
independent variables
weight type
weight expression
title in estimation output
name of cluster variable
Wald, type of model $\chi^{2}$ test
vcetype specified in vce()
title used to label Std. err.
outcome equations
outcome labels, \#=1,...,e(k_out)
type of optimization
max or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement predict
predictions disallowed by margins
default predict() specification for margins
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
outcome values
constraints matrix
iteration $\log$ (up to 20 iterations)
gradient vector
variance-covariance matrix of the estimators
model-based variance
marks estimation sample

In addition to the above, the following is stored in r() :
Matrices
$r$ (table) matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

See Cameron and Trivedi (2005, chap. 15) for a discussion of multinomial models, including multinomial probit. Long and Freese (2014, chap. 8) discuss the multinomial logistic, multinomial probit, and stereotype logistic regression models, with examples using Stata.

As discussed in Remarks and examples, the latent variables for a $J$-alternative model are $\eta_{i j}=$ $\mathbf{z}_{i} \boldsymbol{\alpha}_{j}+\xi_{i j}$, for $j=1, \ldots, J, i=1, \ldots, n$, and $\left\{\xi_{i, 1}, \ldots, \xi_{i, \mathrm{~J}}\right\} \sim$ i.i.d. $N(0,1)$. The experimenter observes alternative $k$ for the $i$ th observation if $\eta_{i k}>\eta_{i l}$ for $l \neq k$. For $j \neq k$, let

$$
\begin{aligned}
v_{i j^{\prime}} & =\eta_{i j}-\eta_{i k} \\
& =\mathbf{z}_{i}\left(\boldsymbol{\alpha}_{j}-\boldsymbol{\alpha}_{k}\right)+\xi_{i j}-\xi_{i k} \\
& =\mathbf{z}_{i} \boldsymbol{\gamma}_{j^{\prime}}+\epsilon_{i j^{\prime}}
\end{aligned}
$$

where $j^{\prime}=j$ if $j<k$ and $j^{\prime}=j-1$ if $j>k$ so that $j^{\prime}=1, \ldots, J-1 . \boldsymbol{\epsilon}_{i}=\left(\epsilon_{i 1}, \ldots, \epsilon_{i, J-1}\right) \sim$ $\operatorname{MVN}(\mathbf{0}, \boldsymbol{\Sigma})$, where

$$
\boldsymbol{\Sigma}=\left(\begin{array}{ccccc}
2 & 1 & 1 & \ldots & 1 \\
1 & 2 & 1 & \ldots & 1 \\
1 & 1 & 2 & \ldots & 1 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
1 & 1 & 1 & \ldots & 2
\end{array}\right)
$$

Denote the deterministic part of the model as $\lambda_{i j^{\prime}}=\mathbf{z}_{i} \gamma_{j^{\prime}}$; the probability that subject $i$ chooses outcome $k$ is

$$
\begin{aligned}
\operatorname{Pr}\left(y_{i}\right. & =k)=\operatorname{Pr}\left(v_{i 1} \leq 0, \ldots, v_{i, J-1} \leq 0\right) \\
& =\operatorname{Pr}\left(\epsilon_{i 1} \leq-\lambda_{i 1}, \ldots, \epsilon_{i, J-1} \leq-\lambda_{i, J-1}\right) \\
& =\frac{1}{(2 \pi)^{(J-1) / 2}|\mathbf{\Sigma}|^{1 / 2}} \int_{-\infty}^{-\lambda_{i 1}} \cdots \int_{-\infty}^{-\lambda_{i, J-1}} \exp \left(-\frac{1}{2} \mathbf{z}^{\prime} \mathbf{\Sigma}^{-1} \mathbf{z}\right) d \mathbf{z}
\end{aligned}
$$

Because of the exchangeable correlation structure of $\boldsymbol{\Sigma}\left(\rho_{i j}=1 / 2\right.$ for all $\left.i \neq j\right)$, we can use Dunnett's (1989) result to reduce the multidimensional integral to one dimension:

$$
\operatorname{Pr}\left(y_{i}=k\right)=\frac{1}{\sqrt{\pi}} \int_{0}^{\infty}\left\{\prod_{j=1}^{J-1} \Phi\left(-z \sqrt{2}-\lambda_{i j}\right)+\prod_{j=1}^{J-1} \Phi\left(z \sqrt{2}-\lambda_{i j}\right)\right\} e^{-z^{2}} d z
$$

Gaussian quadrature is used to approximate this integral, resulting in the $K$-point quadrature formula

$$
\operatorname{Pr}\left(y_{i}=k\right) \approx \frac{1}{2} \sum_{k=1}^{K} w_{k}\left\{\prod_{j=1}^{J-1} \Phi\left(-\sqrt{2 x_{k}}-\lambda_{i j}\right)+\prod_{j=1}^{J-1} \Phi\left(\sqrt{2 x_{k}}-\lambda_{i j}\right)\right\}
$$

where $w_{k}$ and $x_{k}$ are the weights and roots of the Laguerre polynomial of order $K$. In mprobit, $K$ is specified by the intpoints () option.

This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce (robust) and vce (cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.
mprobit also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

## References

Cameron, A. C., and P. K. Trivedi. 2005. Microeconometrics: Methods and Applications. New York: Cambridge University Press.
Dunnett, C. W. 1989. Algorithm AS 251: Multivariate normal probability integrals with product correlation structure. Journal of the Royal Statistical Society, Series C 38: 564-579. https://doi.org/10.2307/2347754.
Haan, P., and A. Uhlendorff. 2006. Estimation of multinomial logit models with unobserved heterogeneity using maximum simulated likelihood. Stata Journal 6: 229-245.

Hole, A. R. 2007. Fitting mixed logit models by using maximum simulated likelihood. Stata Journal 7: 388-401.
Long, J. S., and J. Freese. 2014. Regression Models for Categorical Dependent Variables Using Stata. 3rd ed. College Station, TX: Stata Press.

Tarlov, A. R., J. E. Ware, Jr., S. Greenfield, E. C. Nelson, E. Perrin, and M. Zubkoff. 1989. The medical outcomes study. An application of methods for monitoring the results of medical care. Journal of the American Medical Association 262: 925-930. https://doi.org/10.1001/jama.1989.03430070073033.
Wells, K. B., R. D. Hays, M. A. Burnam, W. H. Rogers, S. Greenfield, and J. E. Ware, Jr. 1989. Detection of depressive disorder for patients receiving prepaid or fee-for-service care. Results from the Medical Outcomes Survey. Journal of the American Medical Association 262: 3298-3302. https://doi.org/10.1001/jama.1989.03430230083030.

## Also see

[R] mprobit postestimation - Postestimation tools for mprobit
$[\mathrm{R}]$ clogit - Conditional (fixed-effects) logistic regression
[R] mlogit - Multinomial (polytomous) logistic regression
[R] ologit - Ordered logistic regression
$[\mathrm{R}]$ oprobit - Ordered probit regression
[BAYES] bayes: mprobit - Bayesian multinomial probit regression
[CM] cmmprobit - Multinomial probit choice model
[CM] nlogit - Nested logit regression
[MI] Estimation - Estimation commands for use with mi estimate
[SVY] svy estimation - Estimation commands for survey data
[XT] xtmlogit — Fixed-effects and random-effects multinomial logit models
[U] 20 Estimation and postestimation commands

## Title

mprobit postestimation - Postestimation tools for mprobit

| Postestimation commands | predict | margins |
| :--- | :--- | :--- |
| References | Also see |  |

## Postestimation commands

The following postestimation commands are available after mprobit:

| Command | Description |
| :---: | :---: |
| contrast | contrasts and ANOVA-style joint tests of estimates |
| estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estat (svy) | postestimation statistics for survey data |
| estimates | cataloging estimation results |
| etable | table of estimation results |
| * forecast | dynamic forecasts and simulations |
| * hausman | Hausman's specification test |
| lincom | point estimates, standard errors, testing, and inference for linear combinations of coefficients |
| * 1 rtest | likelihood-ratio test |
| margins | marginal means, predictive margins, marginal effects, and average marginal effects |
| marginsplot | graph the results from margins (profile plots, interaction plots, etc.) |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients |
| predict | probabilities, linear predictions and their SEs, etc. |
| predictnl | point estimates, standard errors, testing, and inference for generalized predictions |
| pwcompare | pairwise comparisons of estimates |
| suest | seemingly unrelated estimation |
| test | Wald tests of simple and composite linear hypotheses |
| testnl | Wald tests of nonlinear hypotheses |

[^34]
## predict

## Description for predict

predict creates a new variable containing predictions such as probabilities, linear predictions, and standard errors.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

predict [type] \{stub*|newvar|newvarlist \}[if][in][,statistic outcome(outcome)]
predict [type] stub* [if] [in], scores

## statistic Description

## Main

pr predicted probabilities; the default
$\mathrm{xb} \quad$ linear prediction
stdp standard error of the linear prediction
You specify one or $k$ new variables with pr , where $k$ is the number of outcomes. If you specify one new variable and you do not specify outcome(), then outcome(\#1) is assumed.
You specify one new variable with xb and stdp. If you do not specify outcome(), then outcome(\#1) is assumed.
These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

## Options for predict

## Main

pr , the default, computes the predicted probabilities for all outcomes or for a specific outcome. To compute probabilities for all outcomes, you specify $k$ new variables, where $k$ is the number of categories of the dependent variable. Alternatively, you can specify stub*; in which case, pr will store predicted probabilities in variables stub1, stub2, ..., stubk. To compute the probability for a specific outcome, you specify one new variable and, optionally, the outcome value in option outcome(); if you omit outcome(), the first outcome value, outcome(\#1), is assumed.

Say that you fit a model by typing estimation_cmd y x1 x2, and y takes on four values. Then, you could type predict p1 p2 p3 p4 to obtain all four predicted probabilities; alternatively, you could type predict $\mathrm{p}^{*}$ to generate the four predicted probabilities. To compute specific probabilities one at a time, you can type predict p1, outcome(\#1) (or simply predict p1), predict p2, outcome(\#2), and so on. See option outcome() for other ways to refer to outcome values.
xb calculates the linear prediction, $\mathbf{x}_{i} \boldsymbol{\alpha}_{j}$, for alternative $j$ and individual $i$. The index, $j$, corresponds to the outcome specified in outcome().
stdp calculates the standard error of the linear prediction.
outcome(outcome) specifies for which outcome the predicted probabilities are to be calculated. outcome() should contain either one value of the dependent variable or one of $\# 1, \# 2, \ldots$, with \#1 meaning the first category of the dependent variable, \#2 meaning the second category, etc. outcome() is not allowed with scores.
scores calculates equation-level score variables. The $j$ th new variable will contain the scores for the $j$ th fitted equation.

## margins

## Description for margins

margins estimates margins of response for probabilities and linear predictions.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...)...] [options]
```

statistic Description

| default | probabilities for each outcome |
| :--- | :--- |
| pr | probability for a specified outcome |
| xb | linear prediction for a specified outcome |
| stdp | not allowed with margins |

pr and xb default to the first outcome.
Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

## Remarks and examples

Once you have fit a multinomial probit model, you can use predict to obtain probabilities that an individual will choose each of the alternatives for the estimation sample, as well as other samples; see [U] 20 Estimation and postestimation commands and [R] predict.
$>$ Example 1
In example 1 of [R] mprobit, we fit the multinomial probit model to a dataset containing the type of health insurance available to 616 psychologically depressed subjects in the United States (Tarlov et al. 1989; Wells et al. 1989). We can obtain the predicted probabilities by typing

```
. use https://www.stata-press.com/data/r18/sysdsn1
(Health insurance data)
. mprobit insure age male nonwhite i.site
    (output omitted)
. predict p1-p3
(option pr assumed; predicted probabilities)
. list p1-p3 insure in 1/10
```

| p1 | p2 | p3 | insure |
| ---: | ---: | ---: | ---: |
| .5961306 | .3741824 | .029687 | Indemnity |
| .4719296 | .4972289 | .0308415 | Prepaid |
| .4896086 | .4121961 | .0981953 | Indemnity |
| .3730529 | .5416623 | .0852848 | Prepaid |
| .5063069 | .4629773 | .0307158 | . |
| .4768125 | .4923548 | .0308327 | Prepaid |
| .5035672 | .4657016 | .0307312 | Prepaid |
| .3326361 | .5580404 | .1093235 | . |
| .4758165 | .4384811 | .0857024 | Uninsure |
| .5734057 | .3316601 | .0949342 | Prepaid |

insure contains a missing value for observations 5 and 8 . Because of that, those two observations were not used in the estimation. However, because none of the independent variables is missing, predict can still calculate the probabilities. Had we typed

```
. predict p1-p3 if e(sample)
```

predict would have filled in missing values for $\mathrm{p} 1, \mathrm{p} 2$, and p 3 for those observations because they were not used in the estimation.

## References

Tarlov, A. R., J. E. Ware, Jr., S. Greenfield, E. C. Nelson, E. Perrin, and M. Zubkoff. 1989. The medical outcomes study. An application of methods for monitoring the results of medical care. Journal of the American Medical Association 262: 925-930. https://doi.org/10.1001/jama.1989.03430070073033.
Wells, K. B., R. D. Hays, M. A. Burnam, W. H. Rogers, S. Greenfield, and J. E. Ware, Jr. 1989. Detection of depressive disorder for patients receiving prepaid or fee-for-service care. Results from the Medical Outcomes Survey. Journal of the American Medical Association 262: 3298-3302. https://doi.org/10.1001/jama.1989.03430230083030.

## Also see

[R] mprobit - Multinomial probit regression
[U] 20 Estimation and postestimation commands

## Title

nbreg - Negative binomial regression

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options for nbreg | Options for gnbreg | Remarks and examples | Stored results |
| Methods and formulas | References | Also see |  |

## Description

nbreg fits a negative binomial regression model for a nonnegative count dependent variable. In this model, the count variable is believed to be generated by a Poisson-like process, except that the variation is allowed to be greater than that of a true Poisson. This extra variation is referred to as overdispersion.
gnbreg fits a generalization of the negative binomial mean-dispersion model; the shape parameter $\alpha$ may also be parameterized.

## Quick start

Negative binomial model of y on x 1 and categorical variable a nbreg y x1 i.a

Same as above, but report results as incidence-rate ratios
nbreg y x1 i.a, irr
Same as above, and specify exposure variable evar nbreg y x1 i.a, irr exposure(evar)

Generalized negative binomial model with shape parameter $\alpha$ a function of x 2 and x 3 gnbreg y x1 i.a, lnalpha(x2 x3)

Add log of exposure, lnevar, as an offset
gnbreg y x1 i.a, lnalpha(x2 x3) offset(lnevar)

## Menu

nbreg
Statistics $>$ Count outcomes $>$ Negative binomial regression

## gnbreg

Statistics $>$ Count outcomes $>$ Generalized negative binomial regression

## Syntax

Negative binomial regression model
nbreg depvar [indepvars] [if] [in] [weight] [, nbreg_options]

Generalized negative binomial model
gnbreg depvar [indepvars] [if] [in] [weight] [, gnbreg_options]
nbreg_options
Description

Model
noconstant
dispersion(mean)
dispersion (constant)
exposure (varname ${ }_{e}$ )
offset (varname ${ }_{o}$ )
constraints (constraints)
SE/Robust
vce (vcetype)

Reporting
level(\#)
nolrtest
irr
nocnsreport
display_options

Maximization
maximize_options
collinear
coeflegend
suppress constant term
parameterization of dispersion; the default constant dispersion for all observations include $\ln$ (varname $e_{e}$ ) in model with coefficient constrained to 1 include varname ${ }_{o}$ in model with coefficient constrained to 1 apply specified linear constraints
vcetype may be oim, robust, cluster clustvar, opg, bootstrap, or jackknife
set confidence level; default is level (95)
suppress likelihood-ratio test
report incidence-rate ratios
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
control the maximization process; seldom used
keep collinear variables
display legend instead of statistics

## Model

noconstant
Inalpha(varlist)
exposure (varname $e$ )
offset ( varname $_{o}$ )
constraints (constraints)
SE/Robust
vce (vcetype)

Reporting
level(\#)
irr
nocnsreport
display_options
suppress constant term
dispersion model variables
include $\ln \left(\right.$ varname $\left._{e}\right)$ in model with coefficient constrained to 1 include varname ${ }_{o}$ in model with coefficient constrained to 1 apply specified linear constraints
vcetype may be oim, robust, cluster clustvar, opg, bootstrap, or jackknife
set confidence level; default is level (95)
report incidence-rate ratios
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

Maximization
maximize_options
collinear
coeflegend
control the maximization process; seldom used
keep collinear variables
display legend instead of statistics
indepvars and varlist may contain factor variables; see [U] 11.4.3 Factor variables.
depvar, indepvars, varname $_{e}$, and varname ${ }_{O}$ may contain time-series operators (nbreg only); see [U] 11.4.4 Time-series varlists.
bayes, bootstrap, by (nbreg only), collect, fmm (nbreg only), fp (nbreg only), jackknife, mfp (nbreg only), mi estimate, nestreg (nbreg only), rolling, statsby, stepwise, and svy are allowed; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: gnbreg, [BAYES] bayes: nbreg, and [FMM] fmm: nbreg.
vce (bootstrap) and vce (jackknife) are not allowed with the mi estimate prefix; see [MI] mi estimate.
Weights are not allowed with the bootstrap prefix; see [R] bootstrap.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options for nbreg

$\qquad$ Model
noconstant; see [R] Estimation options.
dispersion(mean $\mid$ constant) specifies the parameterization of the model. dispersion(mean), the default, yields a model with dispersion equal to $1+\alpha \exp \left(\mathbf{x}_{j} \boldsymbol{\beta}+\operatorname{offset}_{j}\right)$; that is, the dispersion is a function of the expected mean: $\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}+\operatorname{offset}_{j}\right)$. dispersion (constant) has dispersion equal to $1+\delta$; that is, it is a constant for all observations.
exposure (varname $e_{e}$ ), offset (varname ${ }_{o}$ ), constraints (constraints); see [R] Estimation options.
vce(vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.

Reporting
level(\#); see [R] Estimation options.
nolrtest suppresses fitting the Poisson model. Without this option, a comparison Poisson model is fit, and the likelihood is used in a likelihood-ratio test of the null hypothesis that the dispersion parameter is zero.
irr reports estimated coefficients transformed to incidence-rate ratios, that is, $e^{\beta_{i}}$ rather than $\beta_{i}$. Standard errors and confidence intervals are similarly transformed. This option affects how results are displayed, not how they are estimated or stored. irr may be specified at estimation or when replaying previously estimated results.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\% fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate(\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from(init_specs); see [R] Maximize. These options are seldom used.

Setting the optimization type to technique (bhhh) resets the default vcetype to vce (opg).
The following options are available with nbreg but are not shown in the dialog box: collinear, coeflegend; see [R] Estimation options.

## Options for gnbreg


exposure (varname $e_{e}$ ), offset (varname ${ }_{o}$ ), constraints (constraints); see [R] Estimation options.
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
$\qquad$ Reporting
level(\#); see [R] Estimation options.
irr reports estimated coefficients transformed to incidence-rate ratios, that is, $e^{\beta_{i}}$ rather than $\beta_{i}$. Standard errors and confidence intervals are similarly transformed. This option affects how results are displayed, not how they are estimated or stored. irr may be specified at estimation or when replaying previously estimated results.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate(\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance (\#), nonrtolerance, and from(init_specs); see [R] Maximize. These options are seldom used.

Setting the optimization type to technique (bhhh) resets the default vcetype to vce (opg).
The following options are available with gnbreg but are not shown in the dialog box:
collinear, coeflegend; see [R] Estimation options.

## Remarks and examples

Remarks are presented under the following headings:
Introduction to negative binomial regression
nbreg
gnbreg

## Introduction to negative binomial regression

Negative binomial regression models the number of occurrences (counts) of an event when the event has extra-Poisson variation, that is, when it has overdispersion. The Poisson regression model is

$$
y_{j} \sim \operatorname{Poisson}\left(\mu_{j}\right)
$$

where

$$
\mu_{j}=\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}+\text { offset }_{j}\right)
$$

for observed counts $y_{j}$ with covariates $\mathbf{x}_{j}$ for the $j$ th observation. One derivation of the negative binomial mean-dispersion model is that individual units follow a Poisson regression model, but there is an omitted variable $\zeta_{j}$, such that $e^{\zeta_{j}}$ follows a gamma distribution with mean 1 and variance $\alpha$ :

$$
y_{j} \sim \operatorname{Poisson}\left(\mu_{j}^{*}\right)
$$

where

$$
\mu_{j}^{*}=\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}+\operatorname{offset}_{j}+\zeta_{j}\right)
$$

and

$$
e^{\zeta_{j}} \sim \operatorname{Gamma}(1 / \alpha, \alpha)
$$

With this parameterization, a $\operatorname{Gamma}(a, b)$ distribution will have expectation $a b$ and variance $a b^{2}$.
We refer to $\alpha$ as the overdispersion parameter. The larger $\alpha$ is, the greater the overdispersion. The Poisson model corresponds to $\alpha=0$. nbreg parameterizes $\alpha$ as $\ln \alpha$. gnbreg allows $\ln \alpha$ to be modeled as $\ln \alpha_{j}=\mathbf{z}_{j} \gamma$, a linear combination of covariates $\mathbf{z}_{j}$.
nbreg will fit two different parameterizations of the negative binomial model. The default, described above and also given by the dispersion(mean) option, has dispersion for the $j$ th observation equal to $1+\alpha \exp \left(\mathbf{x}_{j} \boldsymbol{\beta}+\right.$ offset $\left._{j}\right)$. This is seen by noting that the above implies that

$$
\mu_{j}^{*} \sim \operatorname{Gamma}\left(1 / \alpha, \alpha \mu_{j}\right)
$$

and thus

$$
\begin{aligned}
\operatorname{Var}\left(y_{j}\right) & =E\left\{\operatorname{Var}\left(y_{j} \mid \mu_{j}^{*}\right)\right\}+\operatorname{Var}\left\{E\left(y_{j} \mid \mu_{j}^{*}\right)\right\} \\
& =E\left(\mu_{j}^{*}\right)+\operatorname{Var}\left(\mu_{j}^{*}\right) \\
& =\mu_{j}\left(1+\alpha \mu_{j}\right)
\end{aligned}
$$

The alternative parameterization, given by the dispersion (constant) option, has dispersion equal to $1+\delta$; that is, it is constant for all observations. This is so because the constant-dispersion model assumes instead that

$$
\mu_{j}^{*} \sim \operatorname{Gamma}\left(\mu_{j} / \delta, \delta\right)
$$

and thus $\operatorname{Var}\left(y_{j}\right)=\mu_{j}(1+\delta)$. The Poisson model corresponds to $\delta=0$.
For detailed derivations of both models, see Cameron and Trivedi (2013, 80-89). In particular, note that the mean-dispersion model is known as the NB2 model in their terminology, whereas the constant-dispersion model is referred to as the NB1 model.

See Long and Freese (2014) and Cameron and Trivedi (2022, chap. 20) for a discussion of the negative binomial regression model with Stata examples and for a discussion of other regression models for count data.

Hilbe (2011) provides an extensive review of the negative binomial model and its variations, using Stata examples.

## nbreg

It is not uncommon to posit a Poisson regression model and observe a lack of model fit. The following data appeared in Rodríguez (1993):

```
. use https://www.stata-press.com/data/r18/rod93
```

. list, sepby(cohort)

|  |  | cohort | age_mos | deaths |
| :--- | ---: | ---: | ---: | ---: |
| 1. | exposure |  |  |  |
| 1. | $1941-1949$ | 0.5 | 168 | 278.4 |
| 2. | $1941-1949$ | 2.0 | 48 | 538.8 |
| 3. | $1941-1949$ | 4.5 | 63 | 794.4 |
| 4. | $1941-1949$ | 9.0 | 89 | $1,550.8$ |
| 5. | $1941-1949$ | 18.0 | 102 | $3,006.0$ |
| 6. | $1941-1949$ | 42.0 | 81 | $8,743.5$ |
| 7. | $1941-1949$ | 90.0 | 40 | $14,270.0$ |
| 8. | $1960-1967$ | 0.5 | 197 | 403.2 |
| 9. | $1960-1967$ | 2.0 | 48 | 786.0 |
| 10. | $1960-1967$ | 4.5 | 62 | $1,165.3$ |
| 11. | $1960-1967$ | 9.0 | 81 | $2,294.8$ |
| 12. | $1960-1967$ | 18.0 | 97 | $4,500.5$ |
| 13. | $1960-1967$ | 42.0 | 103 | $13,201.5$ |
| 14. | $1960-1967$ | 90.0 | 39 | $19,525.0$ |
|  |  |  |  |  |
| 15. | $1968-1976$ | 0.5 | 195 | 495.3 |
| 16. | $1968-1976$ | 2.0 | 55 | 956.7 |
| 17. | $1968-1976$ | 4.5 | 58 | $1,381.4$ |
| 18. | $1968-1976$ | 9.0 | 85 | $2,604.5$ |
| 19. | $1968-1976$ | 18.0 | 87 | $4,618.5$ |
| 20. | $1968-1976$ | 42.0 | 70 | $9,814.5$ |
| 21. | $1968-1976$ | 90.0 | 10 | $5,802.5$ |

. generate logexp = ln(exposure)
. poisson deaths i.cohort, offset(logexp)
Iteration 0: Log likelihood $=-2160.0544$
Iteration 1: Log likelihood $=-2159.5162$
Iteration 2: Log likelihood $=-2159.5159$
Iteration 3: Log likelihood $=-2159.5159$
Poisson regression

Log likelihood = -2159.5159

| Number of obs | $=$ | 21 |
| :--- | :--- | ---: |
| LR chi2(2) | $=49.16$ |  |
| Prob > chi2 | $=0.0000$ |  |
| Pseudo R2 | $=0.0113$ |  |


| deaths | Coefficient | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :--- | ---: | :--- | ---: | ---: |
| cohort |  |  |  |  |  |  |
| $1960-1967$ | -.3020405 | .0573319 | -5.27 | 0.000 | -.4144089 | -.1896721 |
| $1968-1976$ | .0742143 | .0589726 | 1.26 | 0.208 | -.0413698 | .1897983 |
| _cons | -3.899488 | .0411345 <br> (onfset) | -94.80 | 0.000 | -3.98011 | -3.818866 |
| logexp | 1 |  |  |  |  |  |

```
. estat gof
    Deviance goodness-of-fit = 4190.689
    Prob > chi2(18) = 0.0000
    Pearson goodness-of-fit = 15387.67
    Prob > chi2(18) = 0.0000
```

The extreme significance of the goodness-of-fit $\chi^{2}$ indicates that the Poisson regression model is inappropriate, suggesting to us that we should try a negative binomial model:

| Negative binomial regression |  |  |  |  | Number of obs LR chi2(2) | $\begin{array}{lr} = & 21 \\ = & 0.40 \end{array}$ |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Dispersion: mean <br> Log likelihood = -131.3799 |  |  |  |  | Prob > chi2 | $=0.8171$ |
|  |  |  |  |  | Pseudo R2 | $=0.0015$ |
| deaths | Coefficient | Std. err. | z | $P>\|z\|$ | [95\% conf. | interval] |
| cohort |  |  |  |  |  |  |
| 1960-1967 | -. 2676187 | . 7237203 | -0.37 | 0.712 | -1.686084 | 1.150847 |
| 1968-1976 | -. 4573957 | . 7236651 | -0.63 | 0.527 | -1.875753 | . 9609618 |
| _cons | -2.086731 | . 511856 | -4.08 | 0.000 | -3.08995 | -1.083511 |
| logexp | 1 | (offset) |  |  |  |  |
| /lnalpha | . 5939963 | . 2583615 |  |  | . 0876171 | 1.100376 |
| alpha | 1.811212 | .4679475 |  |  | 1.09157 | 3.005295 |
| LR test of alpha=0: chibar2 (01) = $4056.27 \quad$ Prob $>=$ chibar2 $=0.000$ |  |  |  |  |  |  |

Our original Poisson model is a special case of the negative binomial-it corresponds to $\alpha=0$. nbreg, however, estimates $\alpha$ indirectly, estimating instead $\ln \alpha$. In our model, $\ln \alpha=0.594$, meaning that $\alpha=1.81$ (nbreg undoes the transformation for us at the bottom of the output).

To test $\alpha=0$ (equivalent to $\ln \alpha=-\infty$ ), nbreg performs a likelihood-ratio test. The staggering $\chi^{2}$ value of 4,056 asserts that the probability that we would observe these data conditional on $\alpha=0$ is virtually zero, that is, conditional on the process being Poisson. The data are not Poisson. It is not accidental that this $\chi^{2}$ value is close to the goodness-of-fit statistic from the Poisson regression itself.

## - Technical note

The usual Gaussian test of $\alpha=0$ is omitted because this test occurs on the boundary, invalidating the usual theory associated with such tests. However, the likelihood-ratio test of $\alpha=0$ has been modified to be valid on the boundary. In particular, the null distribution of the likelihood-ratio test statistic is not the usual $\chi_{1}^{2}$, but rather a $50: 50$ mixture of a $\chi_{0}^{2}$ (point mass at zero) and a $\chi_{1}^{2}$, denoted as $\bar{\chi}_{01}^{2}$. See Gutierrez, Carter, and Drukker (2001) for more details.

## - Technical note

The negative binomial model deals with cases in which there is more variation than would be expected if the process were Poisson. The negative binomial model is not helpful if there is less than Poisson variation-if the variance of the count variable is less than its mean. However, underdispersion is uncommon. Poisson models arise because of independently generated events. Overdispersion comes about if some of the parameters (causes) of the Poisson processes are unknown. To obtain underdispersion, the sequence of events somehow would have to be regulated; that is, events would not be independent but controlled based on past occurrences.

## gnbreg

gnbreg is a generalization of nbreg, dispersion(mean). Whereas in nbreg, one $\ln \alpha$ is estimated, gnbreg allows $\ln \alpha$ to vary, observation by observation, as a linear combination of another set of covariates: $\ln \alpha_{j}=\mathbf{z}_{j} \gamma$.

We will assume that the number of deaths is a function of age, whereas the $\ln \alpha$ parameter is a function of cohort. To fit the model, we type

```
. gnbreg deaths age_mos, lnalpha(i.cohort) offset(logexp)
Fitting constant-only model:
Iteration 0: Log likelihood = -187.067 (not concave)
Iteration 1: Log likelihood = -138.13047
Iteration 2: Log likelihood = -133.83164
Iteration 3: Log likelihood = -131.59551
Iteration 4: Log likelihood = -131.5795
Iteration 5: Log likelihood = -131.57948
Iteration 6: Log likelihood = -131.57948
Fitting full model:
Iteration 0: Log likelihood = -124.34327
Iteration 1: Log likelihood = -117.76701
Iteration 2: Log likelihood = -117.56403
Iteration 3: Log likelihood = -117.56164
Iteration 4: Log likelihood = -117.56164
Generalized negative binomial regression Number of obs = 21
    LR chi2(1) = 28.04
    Prob > chi2 = 0.0000
    Pseudo R2 = 0.1065
```

| deaths | Coefficient | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| :---: | ---: | :--- | :--- | :--- | :--- | :--- |
| deaths |  |  |  |  |  |  |
| age_mos |  |  |  |  |  |  |
| _cons | -.0516657 | .0051747 | -9.98 | 0.000 | -.061808 | -.0415233 |
| logexp | -1.867225 | .2227944 <br> (offset) | -8.38 | 0.000 | -2.303894 | -1.430556 |
| lnalpha |  |  |  |  |  |  |
| cohort <br> 1960-1967 <br> 1968-1976 | .0939546 | .7187747 | 0.13 | 0.896 | -1.314818 | 1.502727 |
| _cons | -.4759581 | .5156502 | -0.92 | 0.356 | -1.486614 | .5346978 |

We find that age is a significant determinant of the number of deaths. The standard errors for the variables in the $\ln \alpha$ equation suggest that the overdispersion parameter does not vary across cohorts. We can test this assertion by typing

```
. test 2.cohort 3.cohort
( 1) [lnalpha]2.cohort = 0
( 2) [lnalpha]3.cohort = 0
```



```
    Prob > chi2 = 0.9904
```

There is no evidence of variation by cohort in these data.

## - Technical note

Note the intentional absence of a likelihood-ratio test for $\alpha=0$ in gnbreg. The test is affected by the same boundary condition that affects the comparison test in nbreg; however, when $\alpha$ is parameterized by more than a constant term, the null distribution becomes intractable. For this reason, we recommend using nbreg to test for overdispersion and, if you have reason to believe that overdispersion exists, only then modeling the overdispersion using gnbreg.

## Stored results

nbreg stores the following in e():
Scalars
e(N)
e(k)
e(k_aux)
$e\left(k \_e q\right)$
e(k_eq_model)
e(k_dv)
$e\left(d f \_m\right)$
e(r2_p)
e(11)
e(ll_0)
e(ll_c)
e(alpha)
e(delta)
e(N_clust)
e(chi2)
e(chi2_c)
e(p)
$e$ (rank)
e(rankO)
$e(i c)$
e(rc)
$e$ (converged)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(offset)
e(chi2type)
e(chi2_ct)
e(dispers)
e(vce)
e(vcetype)
e(opt)
e(which)
e(ml_method)
e(user)
e(technique)
e(properties)
e(predict)
e(asbalanced)
e(asobserved)
number of observations
number of parameters
number of auxiliary parameters
number of equations in $e(b)$
number of equations in overall model test
number of dependent variables
model degrees of freedom
pseudo- $R^{2}$
log likelihood
log likelihood, constant-only model
log likelihood, comparison model
value of alpha
value of delta
number of clusters
$\chi^{2}$
$\chi^{2}$ for comparison test
$p$-value for model test
rank of e(V)
rank of $e(V)$ for constant-only model
number of iterations
return code
1 if converged, 0 otherwise
nbreg
command as typed
name of dependent variable
weight type
weight expression
title in estimation output
name of cluster variable
linear offset variable
Wald or LR; type of model $\chi^{2}$ test
Wald or LR; type of model $\chi^{2}$ test corresponding to e(chi2_c)
mean or constant
vcetype specified in vce()
title used to label Std. err.
type of optimization
max or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement predict
factor variables fvset as asbalanced
factor variables fvset as asobserved

```
Matrices
    e(b) coefficient vector
    e(Cns) constraints matrix
    e(ilog) iteration log (up to 20 iterations)
    e(gradient) gradient vector
    e(V) variance-covariance matrix of the estimators
    e(V_modelbased) model-based variance
Functions
    e(sample) marks estimation sample
```

In addition to the above, the following is stored in r() :

```
Matrices
```

    \(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values,
    and confidence intervals
    Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r -class command is run after the estimation command.
gnbreg stores the following in e():
Scalars
e(N)
e(k)
e(k_eq)
e(k_eq_model)
e(k_dv)
e(df_m)
e(r2_p)
e(ll)
e(ll_0)
e(N_clust)
e(chi2)
e(p)
e(rank)
e(rank0)
e(ic)
e(rc)
e(converged)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(offset1)
e(chi2type)
e(vce)
e(vcetype)
e(opt)
e(which)
e(ml_method)
e(user)
e(technique)
e(properties)
e(predict)
e(asbalanced)
e(asobserved)
Matrices
e(b)
number of observations
number of parameters
number of equations in $e(b)$
number of equations in overall model test
number of dependent variables
model degrees of freedom
pseudo- $R^{2}$
log likelihood
log likelihood, constant-only model
number of clusters
$\chi^{2}$
$p$-value for model test
rank of e(V)
rank of e(V) for constant-only model
number of iterations
return code
1 if converged, 0 otherwise
gnbreg
command as typed
name of dependent variable
weight type
weight expression
title in estimation output
name of cluster variable
linear offset variable
Wald or LR; type of model $\chi^{2}$ test
vcetype specified in vce()
title used to label Std. err.
type of optimization
max or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement predict
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector

```
e(gradient)
e(V)
e(V_modelbased)
```

- constraints matrix
e(ilog) iteration $\log$ (up to 20 iterations)

Functions
e(sample) marks estimation sample

In addition to the above, the following is stored in r() :

Matrices
r(table)

> matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r -class command is run after the estimation command.

## Methods and formulas

See [R] poisson and Johnson, Kemp, and Kotz (2005, chap. 4) for an introduction to the Poisson distribution.

Methods and formulas are presented under the following headings:
Mean-dispersion model
Constant-dispersion model

## Mean-dispersion model

A negative binomial distribution can be regarded as a gamma mixture of Poisson random variables. The number of times something occurs, $y_{j}$, is distributed as Poisson $\left(\nu_{j} \mu_{j}\right)$. That is, its conditional likelihood is

$$
f\left(y_{j} \mid \nu_{j}\right)=\frac{\left(\nu_{j} \mu_{j}\right)^{y_{j}} e^{-\nu_{j} \mu_{j}}}{\Gamma\left(y_{j}+1\right)}
$$

where $\mu_{j}=\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}+\right.$ offset $\left._{j}\right)$ and $\nu_{j}$ is an unobserved parameter with a $\operatorname{Gamma}(1 / \alpha, \alpha)$ density:

$$
g(\nu)=\frac{\nu^{(1-\alpha) / \alpha} e^{-\nu / \alpha}}{\alpha^{1 / \alpha} \Gamma(1 / \alpha)}
$$

This gamma distribution has mean 1 and variance $\alpha$, where $\alpha$ is our ancillary parameter.
The unconditional likelihood for the $j$ th observation is therefore

$$
f\left(y_{j}\right)=\int_{0}^{\infty} f\left(y_{j} \mid \nu\right) g(\nu) d \nu=\frac{\Gamma\left(m+y_{j}\right)}{\Gamma\left(y_{j}+1\right) \Gamma(m)} p_{j}^{m}\left(1-p_{j}\right)^{y_{j}}
$$

where $p_{j}=1 /\left(1+\alpha \mu_{j}\right)$ and $m=1 / \alpha$. Solutions for $\alpha$ are handled by searching for $\ln \alpha$ because $\alpha$ must be greater than zero.

The log likelihood (with weights $w_{j}$ and offsets) is given by

$$
\begin{aligned}
& m=1 / \alpha \quad p_{j}=1 /\left(1+\alpha \mu_{j}\right) \quad \mu_{j}=\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}+\text { offset }_{j}\right) \\
& \ln L=\sum_{j=1}^{n} w_{j}\left[\ln \left\{\Gamma\left(m+y_{j}\right)\right\}-\ln \left\{\Gamma\left(y_{j}+1\right)\right\}\right. \\
& \left.\quad-\ln \{\Gamma(m)\}+m \ln \left(p_{j}\right)+y_{j} \ln \left(1-p_{j}\right)\right]
\end{aligned}
$$

For gnbreg, $\alpha$ can vary across the observations according to the parameterization $\ln \alpha_{j}=\mathbf{z}_{j} \gamma$.

## Constant-dispersion model

The constant-dispersion model assumes that $y_{j}$ is conditionally distributed as Poisson $\left(\mu_{j}^{*}\right)$, where $\mu_{j}^{*} \sim \operatorname{Gamma}\left(\mu_{j} / \delta, \delta\right)$ for some dispersion parameter $\delta$ (by contrast, the mean-dispersion model assumes that $\mu_{j}^{*} \sim \operatorname{Gamma}\left(1 / \alpha, \alpha \mu_{j}\right)$ ). The log likelihood is given by

$$
\begin{gathered}
m_{j}=\mu_{j} / \delta \quad p=1 /(1+\delta) \\
\ln L=\sum_{j=1}^{n} w_{j}\left[\ln \left\{\Gamma\left(m_{j}+y_{j}\right)\right\}-\ln \left\{\Gamma\left(y_{j}+1\right)\right\}\right. \\
\left.-\ln \left\{\Gamma\left(m_{j}\right)\right\}+m_{j} \ln (p)+y_{j} \ln (1-p)\right]
\end{gathered}
$$

with everything else defined as before in the calculations for the mean-dispersion model.
nbreg and gnbreg support the Huber/White/sandwich estimator of the variance and its clustered version using vce (robust) and vce(cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.

These commands also support estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

## References

Cameron, A. C., and P. K. Trivedi. 2013. Regression Analysis of Count Data. 2nd ed. New York: Cambridge University Press.
——. 2022. Microeconometrics Using Stata. 2nd ed. College Station, TX: Stata Press.
Cummings, T. H., J. W. Hardin, A. C. McLain, J. R. Hussey, K. J. Bennett, and G. M. Wingood. 2015. Modeling heaped count data. Stata Journal 15: 457-479.
Deb, P., E. C. Norton, and W. G. Manning. 2017. Health Econometrics Using Stata. College Station, TX: Stata Press.
Deb, P., and P. K. Trivedi. 2006. Maximum simulated likelihood estimation of a negative binomial regression model with multinomial endogenous treatment. Stata Journal 6: 246-255.

Gutierrez, R. G., S. L. Carter, and D. M. Drukker. 2001. sg160: On boundary-value likelihood-ratio tests. Stata Technical Bulletin 60: 15-18. Reprinted in Stata Technical Bulletin Reprints, vol. 10, pp. 269-273. College Station, TX: Stata Press.

Harris, T., J. M. Hilbe, and J. W. Hardin. 2014. Modeling count data with generalized distributions. Stata Journal 14: 562-579.

Hilbe, J. M. 2011. Negative Binomial Regression. 2nd ed. Cambridge: Cambridge University Press.
——. 2014. Modeling Count Data. New York: Cambridge University Press.
Johnson, N. L., A. W. Kemp, and S. Kotz. 2005. Univariate Discrete Distributions. 3rd ed. New York: Wiley.
Long, J. S. 1997. Regression Models for Categorical and Limited Dependent Variables. Thousand Oaks, CA: Sage.
Long, J. S., and J. Freese. 2001. Predicted probabilities for count models. Stata Journal 1: 51-57.
__. 2014. Regression Models for Categorical Dependent Variables Using Stata. 3rd ed. College Station, TX: Stata Press.
Miranda, A., and S. Rabe-Hesketh. 2006. Maximum likelihood estimation of endogenous switching and sample selection models for binary, ordinal, and count variables. Stata Journal 6: 285-308.

Rodríguez, G. 1993. sbe10: An improvement to poisson. Stata Technical Bulletin 11: 11-14. Reprinted in Stata Technical Bulletin Reprints, vol. 2, pp. 94-98. College Station, TX: Stata Press.
Xu, X., and J. W. Hardin. 2016. Regression models for bivariate count outcomes. Stata Journal 16: 301-315.

## Also see

[R] nbreg postestimation - Postestimation tools for nbreg and gnbreg
[ R$]$ glm - Generalized linear models
[R] npregress kernel - Nonparametric kernel regression
[R] npregress series - Nonparametric series regression
[R] poisson - Poisson regression
[R] tnbreg - Truncated negative binomial regression
[R] zinb - Zero-inflated negative binomial regression
[BAYES] bayes: gnbreg - Bayesian generalized negative binomial regression
[BAYES] bayes: nbreg - Bayesian negative binomial regression
[FMM] fmm: nbreg — Finite mixtures of negative binomial regression models
[ME] menbreg - Multilevel mixed-effects negative binomial regression
[MI] Estimation - Estimation commands for use with mi estimate
[SVY] svy estimation - Estimation commands for survey data
[XT] xtnbreg - Fixed-effects, random-effects, \& population-averaged negative binomial models
[U] 20 Estimation and postestimation commands

## Title

nbreg postestimation - Postestimation tools for nbreg and gnbreg

Postestimation commands
Remarks and examples Also see
predict margins
Methods and formulas Reference

## Postestimation commands

The following postestimation commands are available after nbreg and gnbreg:
Command Description

| contrast | contrasts and ANOVA-style joint tests of estimates |
| :---: | :---: |
| estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estat (svy) | postestimation statistics for survey data |
| estimates | cataloging estimation results |
| etable | table of estimation results |
| * forecast | dynamic forecasts and simulations |
| * hausman | Hausman's specification test |
| lincom | point estimates, standard errors, testing, and inference for linear combinations of coefficients |
| linktest | link test for model specification |
| * 1 rtest | likelihood-ratio test |
| margins | marginal means, predictive margins, marginal effects, and average marginal effects |
| marginsplot | graph the results from margins (profile plots, interaction plots, etc.) |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients |
| predict | number of events, incidence rates, probabilities, etc. |
| predictnl | point estimates, standard errors, testing, and inference for generalized predictions |
| pwcompare | pairwise comparisons of estimates |
| suest | seemingly unrelated estimation |
| test | Wald tests of simple and composite linear hypotheses |
| testnl | Wald tests of nonlinear hypotheses |

[^35]
## predict

## Description for predict

predict creates a new variable containing predictions such as numbers of events, incidence rates, probabilities, linear predictions, standard errors, and predicted values.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

```
predict [type] newvar [if][in] [, statistic nooffset]
predict [type]stub* [if] [in], scores
```

statistic Description

| Main |  |
| :--- | :--- |
| n | number of events; the default |
| ir | incidence rate $($ equivalent to predict $\ldots$, n nooffset $)$ |
| $\operatorname{pr}(n)$ | probability $\operatorname{Pr}\left(y_{j}=n\right)$ |
| $\operatorname{pr}(a, b)$ | probability $\operatorname{Pr}\left(a \leq y_{j} \leq b\right)$ |
| xb | linear prediction |
| stdp | standard error of the linear prediction |

In addition, relevant only after gnbreg are the following: statistic Description

Main
alpha predicted values of $\alpha_{j}$
Inalpha predicted values of $\ln \alpha_{j}$
stdplna standard error of predicted $\ln \alpha_{j}$

These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

## Options for predict

$\qquad$ Main
n , the default, calculates the predicted number of events, which is $\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}\right)$ if neither offset (varname ${ }_{o}$ ) nor exposure (varname $e_{e}$ ) was specified when the model was fit; $\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}+\right.$ offset $\left._{j}\right)$ if offset () was specified; or $\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}\right) \times$ exposure $_{j}$ if exposure() was specified.
ir calculates the incidence rate $\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}\right)$, which is the predicted number of events when exposure is 1 . This is equivalent to specifying both the n and the nooffset options.
$\operatorname{pr}(n)$ calculates the probability $\operatorname{Pr}\left(y_{j}=n\right)$, where $n$ is a nonnegative integer that may be specified as a number or a variable.
$\operatorname{pr}(a, b)$ calculates the probability $\operatorname{Pr}\left(a \leq y_{j} \leq b\right)$, where $a$ and $b$ are nonnegative integers that may be specified as numbers or variables;
$b$ missing ( $b \geq$.) means $+\infty$;
$\operatorname{pr}(20,$.$) calculates \operatorname{Pr}\left(y_{j} \geq 20\right)$;
$\operatorname{pr}(20, b)$ calculates $\operatorname{Pr}\left(y_{j} \geq 20\right)$ in observations for which $b \geq$. and calculates
$\operatorname{Pr}\left(20 \leq y_{j} \leq b\right)$ elsewhere.
$\operatorname{pr}(., b)$ produces a syntax error. A missing value in an observation of the variable $a$ causes a missing value in that observation for $\operatorname{pr}(a, b)$.
xb calculates the linear prediction, which is $\mathbf{x}_{j} \boldsymbol{\beta}$ if neither offset() nor exposure() was specified; $\mathbf{x}_{j} \boldsymbol{\beta}+$ offset $_{j}$ if offset() was specified; or $\mathbf{x}_{j} \boldsymbol{\beta}+\ln$ (exposure $_{j}$ ) if exposure() was specified; see nooffset below.
stdp calculates the standard error of the linear prediction.
alpha, lnalpha, and stdplna are relevant after gnbreg estimation only; they produce the predicted values of $\alpha_{j}, \ln \alpha_{j}$, and the standard error of the predicted $\ln \alpha_{j}$, respectively.
nooffset is relevant only if you specified offset() or exposure() when you fit the model. It modifies the calculations made by predict so that they ignore the offset or exposure variable; the linear prediction is treated as $\mathbf{x}_{j} \boldsymbol{\beta}$ rather than as $\mathbf{x}_{j} \boldsymbol{\beta}+$ offset $_{j}$ or $\mathbf{x}_{j} \boldsymbol{\beta}+\ln \left(\right.$ exposure $\left._{j}\right)$. Specifying predict..., nooffset is equivalent to specifying predict..., ir.
scores calculates equation-level score variables.
The first new variable will contain $\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)$.
The second new variable will contain $\partial \ln L / \partial\left(\ln \alpha_{j}\right)$ for dispersion(mean) and gnbreg.
The second new variable will contain $\partial \ln L / \partial(\ln \delta)$ for dispersion(constant).

## margins

## Description for margins

margins estimates margins of response for numbers of events, incidence rates, probabilities, linear predictions, and predicted values.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...) ...] [options]
```

statistic

| n | number of events; the default |
| :--- | :--- |
| ir | incidence rate $($ equivalent to predict $\ldots, \mathrm{n}$ nooffset $)$ |
| $\operatorname{pr}(n)$ | probability $\operatorname{Pr}\left(y_{j}=n\right)$ |
| $\operatorname{pr}(a, b)$ | probability $\operatorname{Pr}\left(a \leq y_{j} \leq b\right)$ |
| xb | linear prediction |
| stdp | not allowed with margins |

In addition, relevant only after gnbreg are the following:
statistic
Description
alpha predicted values of $\alpha_{j}$
lnalpha predicted values of $\ln \alpha_{j}$
stdplna not allowed with margins

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see $[R]$ margins.

## Remarks and examples

After nbreg and gnbreg, predict returns the expected number of deaths per cohort and the probability of observing the number of deaths recorded or fewer.
. use https://www.stata-press.com/data/r18/rod93
. nbreg deaths i.cohort, nolog
Negative binomial regression $\quad$ Number of obs $=\quad 21$ LR chi2(2) $=0.14$
Dispersion: mean Prob > chi2 $=0.9307$
Log likelihood = -108.48841

| deaths | Coefficient | Std. err | z | $P>\|z\|$ | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| cohort |  |  |  |  |  |  |
| 1960-1967 | . 0591305 | . 2978419 | 0.20 | 0.843 | -. 5246289 | . 64289 |
| 1968-1976 | -. 0538792 | . 2981621 | -0.18 | 0.857 | -. 6382662 | . 5305077 |
| _cons | 4.435906 | . 2107213 | 21.05 | 0.000 | 4.0229 | 4.848912 |
| /lnalpha | -1.207379 | . 3108622 |  |  | -1.816657 | -. 5980999 |
| alpha | . 29898 | . 0929416 |  |  | . 1625683 | . 5498555 |

. predict count
(option $\mathbf{n}$ assumed; predicted number of events)
. predict $\mathrm{p}, \mathrm{pr}$ (0, deaths)
. summarize deaths count p

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| deaths | 21 | 84.66667 | 48.84192 | 10 | 197 |
| count | 21 | 84.66667 | 4.00773 | 80 | 89.57143 |
| $p$ | 21 | .4991542 | .2743702 | .0070255 | .9801285 |

The expected number of deaths ranges from 80 to 90 . The probability $\operatorname{Pr}\left(y_{i} \leq\right.$ deaths $)$ ranges from 0.007 to 0.98 .

The estimated expected and observed mean number of deaths, 84.67 , happen to be the same in our example because our model included only a categorical predictor. In general, in the presence of other continuous predictors, the two estimates may not always be the same.

## Methods and formulas

In the following, we use the same notation as in [R] nbreg.
Methods and formulas are presented under the following headings:

[^36]
## Mean-dispersion model

The equation-level scores are given by

$$
\begin{aligned}
\operatorname{score}(\mathbf{x} \boldsymbol{\beta})_{j} & =p_{j}\left(y_{j}-\mu_{j}\right) \\
\operatorname{score}(\tau)_{j} & =-m\left\{\frac{\alpha_{j}\left(\mu_{j}-y_{j}\right)}{1+\alpha_{j} \mu_{j}}-\ln \left(1+\alpha_{j} \mu_{j}\right)+\psi\left(y_{j}+m\right)-\psi(m)\right\}
\end{aligned}
$$

where $\tau_{j}=\ln \alpha_{j}$ and $\psi(z)$ is the digamma function.

## Constant-dispersion model

The equation-level scores are given by

$$
\begin{aligned}
\operatorname{score}(\mathbf{x} \boldsymbol{\beta})_{j} & =m_{j}\left\{\psi\left(y_{j}+m_{j}\right)-\psi\left(m_{j}\right)+\ln (p)\right\} \\
\operatorname{score}(\tau)_{j} & =y_{j}-\left(y_{j}+m_{j}\right)(1-p)-\operatorname{score}(\mathbf{x} \boldsymbol{\beta})_{j}
\end{aligned}
$$

where $\tau_{j}=\ln \delta_{j}$.

## Reference

Manjón, M., and O. Martínez. 2014. The chi-squared goodness-of-fit test for count-data models. Stata Journal 14: 798-816.

## Also see

[R] nbreg - Negative binomial regression
[U] 20 Estimation and postestimation commands

## Title

nestreg - Nested model statistics

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Acknowledgment |
| References | Also see |  |  |

## Description

nestreg fits nested models by sequentially adding blocks of variables and then reports comparison tests between the nested models.

## Quick start

Fit nested (hierarchical) models sequentially, including covariates $x 1$ and $x 2$ first and then adding $x 3$ and x 4

```
nestreg: regress y (x1 x2) (x3 x4)
```

Also fit third model including indicators for categorical variable a

```
nestreg: regress y (x1 x2) (x3 x4) (i.a)
```

Report table of likelihood-ratio tests instead of Wald tests comparing models
nestreg, lrtable: regress y (x1 x2) (x3 x4) (i.a)
Fit nested models and adjust for complex survey design using svyset data

```
nestreg: svy: regress y (x1 x2) (x3 x4) (i.a)
```

Note: In the above examples, regress could be replaced with any estimation command allowing the nestreg prefix.

## Menu

Statistics $>$ Other $>$ Nested model statistics

## Syntax

Standard estimation command syntax
nestreg [, options]: command_name depvar (varlist) [(varlist)...]
[if $][$ in $][$ weight $][$, command_options $]$
Survey estimation command syntax

```
nestreg [, options]: svy [vcetype] [, svy_options]: command_name depvar
    (varlist) [(varlist) ...] [if] [in] [, command_options]
```

options
Description
Reporting
waldtable
lrtable
quietly
store (stub)
report Wald test results; the default
report likelihood-ratio test results
suppress any output from command_name
store nested estimation results in _est_stub\#
by is allowed; see [U] 11.1.10 Prefix commands.
Weights are allowed if command_name allows them; see [U] 11.1.6 weight.
A varlist in parentheses indicates that this list of variables is to be considered as a block. Each variable in a varlist not bound in parentheses will be treated as its own block.
All postestimation commands behave as they would after command_name without the nestreg prefix; see the postestimation manual entry for command_name.

## Options

$\qquad$ Reporting
waldtable specifies that the table of Wald test results be reported. waldtable is the default. lrtable specifies that the table of likelihood-ratio tests be reported. This option is not allowed if pweights, the vce(robust) option, or the vce(cluster clustvar) option is specified. Irtable is also not allowed with the svy prefix.
quietly suppresses the display of any output from command_name.
store (stub) specifies that each model fit by nestreg be stored under the name _est_stub\#, where \# is the nesting order from first to last.

## Remarks and examples

Remarks are presented under the following headings:
Estimation commands
Wald tests
Likelihood-ratio tests
Programming for nestreg

## Estimation commands

nestreg removes collinear predictors and observations with missing values from the estimation sample before calling command_name.

The following Stata commands are supported by nestreg:
betareg, clogit, cloglog, glm, intreg, logistic, logit, nbreg, ologit, oprobit, poisson, probit, qreg, regress, scobit, stcox, stcrreg, stintreg, streg, and tobit

You do not supply a depvar for stcox, stintreg, stcrreg, or streg; otherwise, depvar is required. You must supply two depvars for intreg.

## Wald tests

Use nestreg to test the significance of blocks of predictors, building the regression model one block at a time. Using the data from example 1 of $[R]$ test, we wish to test the significance of the following predictors of birthrate: median age (medage), median age squared (c.medage\#c.medage), and indicators of the census region (i.region).


Block 3: 2.region 3.region 4.region

| Source | SS | df | MS | Number of obs | = | 50 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | F (5, 44) | = | 100.63 |
| Model | 38803.4208 | 5 | 7760.68416 | Prob > F | = | 0.0000 |
| Residual | 3393.39921 | 44 | 77.1227094 | R -squared | = | 0.9196 |
|  |  |  |  | Adj R-squared |  | 0.9104 |
| Total | 42196.82 | 49 | 861.159592 | Root MSE |  | 8.782 |


| brate | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| :---: | ---: | :--- | :---: | :---: | :---: | :---: |
| medage | -109.0958 | 13.52452 | -8.07 | 0.000 | -136.3527 | -81.83892 |
| c.medage\# |  |  |  |  |  |  |
| c.medage | 1.635209 | .2290536 | 7.14 | 0.000 | 1.173582 | 2.096836 |
| region |  |  |  |  |  |  |
| N Cntrl | 15.00283 | 4.252067 | 3.53 | 0.001 | 6.433353 | 23.57231 |
| South | 7.366445 | 3.953335 | 1.86 | 0.069 | -.6009775 | 15.33387 |
| West | 21.39679 | 4.650601 | 4.60 | 0.000 | 12.02412 | 30.76946 |
| _cons | 1947.611 | 199.8405 | 9.75 | 0.000 | 1544.859 | 2350.363 |


| Block | F | Block <br> df | Residual <br> df | $\operatorname{Pr}>\mathrm{F}$ | R 2 | Change <br> in R2 |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| 1 | 164.72 | 1 | 48 | 0.0000 | 0.7743 |  |
| 2 | 35.25 | 1 | 47 | 0.0000 | 0.8711 | 0.0967 |
| 3 | 8.85 | 3 | 44 | 0.0001 | 0.9196 | 0.0485 |

This single call to nestreg ran regress three times, adding a block of predictors to the model for each run as in
. regress brate medage

| Source | SS | df | MS |
| ---: | ---: | ---: | ---: |
| Model | 32675.1044 | 1 | 32675.1044 |
| Residual | 9521.71561 | 48 | 198.369075 |
| Total | 42196.82 | 49 | 861.159592 |


| Number of obs | $=$ | 50 |
| :--- | :--- | ---: |
| $\mathrm{~F}(1,48)$ | $=$ | 164.72 |
| Prob > F | $=$ | 0.0000 |
| R-squared | $=$ | 0.7743 |
| Adj R-squared | $=$ | 0.7696 |
| Root MSE | $=$ | 14.084 |


| brate | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| medage | -15.24893 | 1.188141 | -12.83 | 0.000 | -17.63785 | -12.86002 |
| _cons | 618.3935 | 35.15416 | 17.59 | 0.000 | 547.7113 | 689.0756 |

. regress brate medage c.medage\#c.medage


| Source | SS | df | MS | Number of obs F(5, 44) |  | 50 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Model | 38803.4208 | 5 | 7760.68416 | Prob > F |  | 0.0000 |
| Residual | 3393.39921 | 44 | 77.1227094 |  |  | 0.9196 |
| Total | 42196.82 | 49 | $861.159592$ | Adj R-squared Root MSE |  | 0.9104 8.782 |
| brate | Coefficient | Std. err. | t | $P>\|t\|$ | [95\% conf. interval] |  |
| medage | -109.0958 | 13.52452 | -8.07 | 0.000 | -136.3527 | -81.83892 |
| c.medage\# <br> c.medage | 1.635209 | . 2290536 | 7.14 | 0.000 | 1.173582 | 2.096836 |
| region <br> N Cntrl | 15.00283 | 4.252067 | 3.53 | 0.001 | 6.433353 | 23.57231 |
| South | 7.366445 | 3.953335 | 1.86 | 0.069 | -. 6009775 | 15.33387 |
| West | 21.39679 | 4.650601 | 4.60 | 0.000 | 12.02412 | 30.76946 |
| _cons | 1947.611 | 199.8405 | 9.75 | 0.000 | 1544.859 | 2350.363 |

nestreg collected the $F$ statistic for the corresponding block of predictors and the model $R^{2}$ statistic from each model fit.

The $F$ statistic for the first block, 164.72, is for a test of the joint significance of the first block of variables; it is simply the $F$ statistic from the regression of brate on medage. The $F$ statistic for the second block, 35.25, is for a test of the joint significance of the second block of variables in a regression of both the first and second blocks of variables. In our example, it is an $F$ test of $c . m e d a g e \# c . m e d a g e ~ i n ~ t h e ~ r e g r e s s i o n ~ o f ~ b r a t e ~ o n ~ m e d a g e ~ a n d ~ c . m e d a g e \# c . m e d a g e . ~ S i m i l a r l y, ~$ the third block's $F$ statistic of 8.85 corresponds to a joint test of the indicators for the N Cntrl, South, and West regions in the final regression.

## Likelihood-ratio tests

The nestreg command provides a simple syntax for performing likelihood-ratio tests for nested model specifications; also see lrtest. Using the data from example 1 of [R] Irtest, we wish to jointly test the significance of the following predictors of low birthweight: age, lwt, ptl, and ht.
. use https://www.stata-press.com/data/r18/lbw
(Hosmer \& Lemeshow data)
. nestreg, lr: logistic low (i.race smoke ui) (age lwt ptl ht)
note: 1.race omitted because of estimability.
Block 1: 2.race 3.race smoke ui
Logistic regression
Log likelihood = -107.93404

Note: _cons estimates baseline odds.
Block 2: age lwt ptl ht
Logistic regression $\quad$ Number of obs $=189$
LR chi2(8) $=33.22$
Prob > chi2 $=0.0001$
Log likelihood $=-100.724$
Pseudo R2 $=0.1416$

| low | Odds ratio | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :--- | :--- | :--- | :--- | :--- |
| race |  |  |  |  |  |  |
| Black | 3.534767 | 1.860737 | 2.40 | 0.016 | 1.259736 | 9.918406 |
| Other | 2.368079 | 1.039949 | 1.96 | 0.050 | 1.001356 | 5.600207 |
|  |  |  |  |  |  |  |
| smoke | 2.517698 | 1.00916 | 2.30 | 0.021 | 1.147676 | 5.523162 |
| ui | 2.1351 | .9808153 | 1.65 | 0.099 | .8677528 | 5.2534 |
| age | .9732636 | .0354759 | -0.74 | 0.457 | .9061578 | 1.045339 |
| lwt | .9849634 | .0068217 | -2.19 | 0.029 | .9716834 | .9984249 |
| ptl | 1.719161 | .5952579 | 1.56 | 0.118 | .8721455 | 3.388787 |
| ht | 6.249602 | 4.322408 | 2.65 | 0.008 | 1.611152 | 24.24199 |
| _cons | 1.586014 | 1.910496 | 0.38 | 0.702 | .1496092 | 16.8134 |

Note: _cons estimates baseline odds.

| Block | LL | LR | df | Pr $>$ LR | AIC | BIC |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| 1 | -107.934 | 18.80 | 4 | 0.0009 | 225.8681 | 242.0768 |
| 2 | -100.724 | 14.42 | 4 | 0.0061 | 219.448 | 248.6237 |

The estimation results from the full model are left in e(), so we can later use estat and other postestimation commands.

```
. estat gof
Goodness-of-fit test after logistic model
Variable: low
    Number of observations = 189
Number of covariate patterns = 182
    Pearson chi2(173) = 179.24
                        Prob > chi2 = 0.3567
```


## Programming for nestreg

If you want your community-contributed command (command_name) to work with nestreg, it must follow standard Stata syntax and allow the if qualifier. Furthermore, command_name must have sw or swml as a program property; see [P] program properties. If command_name has swml as a property, command_name must store the log-likelihood value in e(ll) and the model degrees of freedom in $\mathrm{e}\left(\mathrm{df} \_\mathrm{m}\right)$.

## Stored results

nestreg stores the following in r() :
Matrices
$r$ (wald) matrix corresponding to the Wald table
$r(l r) \quad$ matrix corresponding to the likelihood-ratio table

## Acknowledgment

We thank Paul H. Bern of Syracuse University for developing the hierarchical regression command that inspired nestreg.

## References

Acock, A. C. 2023. A Gentle Introduction to Stata. Rev. 6th ed. College Station, TX: Stata Press.
Lindsey, C., and S. J. Sheather. 2015. Best subsets variable selection in nonnormal regression models. Stata Journal 15: 1046-1059.

## Also see

[P] program properties - Properties of user-defined programs

## Title

net - Install and manage community-contributed additions from the Internet

> Description Syntax Options Remarks and examples
> Also see

## Description

net downloads and installs additions to Stata. The additions can be obtained from the Internet or from physical media. The additions can be ado-files (new commands), help files, or even datasets. Collections of files that may be installed as a group are bound together into a package.
ado manages the packages you have installed by using net. The ado command lets you list and uninstall previously installed packages.

You can also access the net and ado features by selecting Help > SJ and community-contributed features; this is the recommended method to find and install additions to Stata.

## Syntax

Set current location for net
net from directory_or_url

Change to a different net directory net cd path_or_url

Change to a different net site

```
net link linkname
```

Search for installed packages

```
net search (see [R] net search)
```

Report current net location
net

Describe a package

```
net describe pkgname [, from(directory_or_url)]
```

Set location where packages will be installed net set ado dirname

Set location where ancillary files will be installed

```
net set other dirname
```

Report net 'from', 'ado', and 'other' settings

```
net query
```

Install ado-files and help files from a package

```
net install pkgname [, all replace force from(directory_or_url)]
```

Install ancillary files from a package

```
net get pkgname [, all replace force from(directory_or_url)]
```

Shortcut to access Stata Journal (SJ) net site
net sj vol-issue [insert]

List installed packages

```
ado [,\underline{find(string) from(dirname)]}
ado dir [pkgid] [, \underline{find(string) from(dirname)]}
```

Describe installed packages

```
ado describe [pkgid] [, \underline{find(string) from(dirname)]}
```

Update installed packages

```
ado update (see [R] ado update)
```

Uninstall an installed package

```
ado uninstall pkgid [, \underline{from(dirname)]}
```

where
pkgname is name of a package
pkgid is name of a package
or a number in square brackets: [\#]
dirname is a directory name
or PLUS (default)
or PERSONAL
or SITE

## Options

all is used with net install and net get. Typing it with either one makes the command equivalent to typing net install followed by net get.
replace is for use with net install and net get. It specifies that the downloaded files replace existing files if any of the files already exists.
force specifies that the downloaded files replace existing files if any of the files already exists, even if Stata thinks all the files are the same. force implies replace.
find (string) is for use with ado, ado dir, and ado describe. It specifies that the descriptions of the packages installed on your computer be searched, and that the package descriptions containing string be listed.
from(dirname), when used with ado, specifies where the packages are installed. The default is from(PLUS). PLUS is a code word that Stata understands to correspond to a particular directory on your computer that was set at installation time. On Windows computers, PLUS probably means the directory $\mathrm{c}: \backslash \mathrm{ado} \backslash \mathrm{plus}$, but it might mean something else. You can find out what it means by typing sysdir, but doing so is irrelevant if you use the defaults.
from (directory_or_url), when used with net, specifies the directory or URL where installable packages may be found. The directory or URL is the same as the one that would have been specified with net from.

## Remarks and examples

For an introduction to using net and ado, see [U] 29 Using the Internet to keep up to date. The purpose of this documentation is

- to briefly, but accurately, describe net and ado and all their features and
- to provide documentation to those who wish to set up their own sites to distribute additions to Stata.

Remarks are presented under the following headings:

[^37]
## Definition of a package

A package is a collection of files-typically, .ado and .sthlp files-that together provide a new feature in Stata. Packages contain additions that you wish had been part of Stata at the outset. For instance, the package named zz49 might add the xyz command to Stata. At a minimum, such a package would contain xyz.ado, the code to implement the new command, and xyz.sthlp, the system help to describe it. We write such additions, and so do other users.

One source of these additions is the Stata Journal, a printed and electronic journal with corresponding software. If you want the journal, you must subscribe, but the software is available for free from our website.

## The purpose of the net and ado commands

The net command makes it easy to distribute and install packages. The goal is to get you quickly to a package-description page that summarizes the addition, for example,

```
. net describe rte_stat, from(http://www.wemakeitupaswego.edu/faculty/sgazer/)
package rte_stat from http://www.wemakeitupaswego.edu/faculty/sgazer/
TITLE
    rte_stat. The robust-to-everything statistic; update.
DESCRIPTION/AUTHOR(S)
    S. Gazer, Dept. of Applied Theoretical Mathematics, WMIUAWG Univ.
    Aleph-0 100% confidence intervals proved too conservative for some
    applications; Aleph-1 confidence intervals have been substituted.
    The new robust-to-everything supplants the previous robust-to-
    everything-conceivable statistic. See "Inference in the absence
    of data" (forthcoming). After installation, see help rte.
INSTALLATION FILES (type net install rte_stat)
    rte.ado
    rte.sthlp
    nullset.ado
    random.ado
```

If you decide that the addition might prove useful, net makes the installation easy:

```
. net install rte_stat
checking rte_stat consistency and verifying not already installed...
installing into c:\ado\plus\ ...
installation complete.
```

The ado command helps you manage packages installed with net. Perhaps you remember that you installed a package that calculates the robust-to-everything statistic, but you cannot remember the command's name. You could use ado to search what you have previously installed for the rte command,
[1] package tomata from https://fmwww.bc.edu/RePEc/bocode/t
'TOMATA': module to make vectors in Mata of variables in Stata
(output omitted)
[15] package rte_stat from http://www.wemakeitupaswego.edu/faculty/sgazer
rte_stat. The robust-to-everything statistic; update.
(output omitted)
[21] package st0119 from https://www.stata-journal.com/software/sj7-1
SJ7-1 st0119. Rasch analysis
or you might type

```
. ado, find("robust-to-everything")
[15] package rte_stat from http://www.wemakeitupaswego.edu/faculty/sgazer
    rte_stat. The robust-to-everything statistic; update.
```

Perhaps you decide that rte, despite the author's claims, is not worth the disk space it occupies. You can use ado to erase it:

```
. ado uninstall rte_stat
package rte_stat from http://www.wemakeitupaswego.edu/faculty/sgazer
    rte_stat. The robust-to-everything statistic; update.
(package uninstalled)
```

ado uninstall is easier than erasing the files by hand because ado uninstall erases every file associated with the package, and, moreover, ado knows where on your computer rte_stat is installed; you would have to hunt for these files.

## Content pages

There are two types of pages displayed by net: content pages and package-description pages. When you type net from, net cd, net link, or net without arguments, Stata goes to the specified place and displays the content page:

```
. net from https://Www.stata.com
```

```
https://www.stata.com/
```

StataCorp

```
Welcome to StataCorp.
Below we provide links to sites providing additions to Stata, including
the Stata Journal and Statalist. These are NOT THE OFFICIAL UPDATES;
you fetch and install the official updates by typing -update-.
PLACES you could -net link- to:
    sj The Stata Journal
DIRECTORIES you could -net cd- to:
    users materials written by various people, including StataCorp
        employees
    meetings software packages from Stata Users Group meetings
    links links to other locations providing additions to Stata
```

A content page tells you about other content pages and package-description pages. The example above lists other content pages only. Below, we follow one of the links for the Stata Journal:

```
. net link sj
```

https://www.stata-journal.com/
The Stata Journal

```
The Stata Journal is a refereed, quarterly journal containing articles
of interest to Stata users. For more details and subscription information,
visit the Stata Journal website at https://www.stata-journal.com.
PLACES you could -net link- to:
    stata StataCorp website
DIRECTORIES you could -net cd- to:
    production Files for authors of the Stata Journal
    software Software associated with Stata Journal articles
```

. net cd software
https://www.stata-journal.com/software/
The Stata Journal
PLACES you could -net link- to:
stata StataCorp website
stb Historical materials published in the Stata Technical
Bulletin
DIRECTORIES you could -net cd- to:
(output omitted)
sj7-1 volume 7, issue 1
(output omitted)
sj1-1 volume 1, issue 1
. net cd sj7-1
https://www.stata-journal.com/software/sj7-1/
Stata Journal volume 7, issue 1
DIRECTORIES you could -net cd- to:
.. Other Stata Journals
PACKAGES you could -net describe-:
dm0027 File filtering in Stata: handling complex data
formats and navigating log files efficiently
st0119 Rasch analysis
st0120 Multivariable regression spline models
st0121 mhbounds - Sensitivity Analysis for Average
Treatment Effects
dm0027, st0119, ..., st0121 are links to package-description pages.

1. When you type net from, you follow that with a location to display the location's content page.
a. The location could be a URL, such as https://www.stata.com. The content page at that location would then be listed.
b. The location could be e: on a Windows computer or a mounted volume on a Mac computer. The content page on that source would be listed. That would work if you had special media obtained from StataCorp or special media prepared by another user.
c. The location could even be a directory on your computer, but that would work only if that directory contained the right kind of files.
2. Once you have specified a location, typing net cd will take you into subdirectories of that location, if there are any. Typing
```
. net from https://www.stata-journal.com
. net cd software
```

is equivalent to typing
. net from https://www.stata-journal.com/software
Typing net cd displays the content page from that location.
3. Typing net without arguments redisplays the current content page, which is the content page last displayed.
4. net link is similar to net cd in that the result is to change the location, but rather than changing to subdirectories of the current location, net link jumps to another location:

```
. net from https://www.stata-journal.com
```

https://www.stata-journal.com/
The Stata Journal

```
The Stata Journal is a refereed, quarterly journal containing articles
of interest to Stata users. For more details and subscription information,
visit the Stata Journal website at
https://www.stata-journal.com.
PLACES you could -net link- to:
    stata StataCorp website
DIRECTORIES you could -net cd- to:
    production Files for authors of the Stata Journal
    software Software associated with Stata Journal articles
```

Typing net link stata would jump to https://www.stata.com:

```
. net link stata
```

```
https://www.stata.com/
StataCorp
```

Welcome to StataCorp.
(output omitted)

## Package-description pages

Package-description pages describe what could be installed:

```
. net from https://www.stata-journal.com/software/sj7-1
```

```
https://www.stata-journal.com/software/sj7-1/
    (output omitted)
. net describe st0119
```

package st0119 from https://www.stata-journal.com/software/sj7-1
TITLE
SJ7-1 st0119. Rasch analysis
DESCRIPTION/AUTHOR (S)
Rasch analysis
by Jean-Benoit Hardouin, University of Nantes, France
Support: jean-benoit.hardouin@univ-nantes.fr
After installation, type help gammasym, gausshermite,
geekel2d, raschtest, and raschtestv7
INSTALLATION FILES (type net install st0119)
st0119/raschtest.ado
st0119/raschtest.hlp
st0119/raschtestv7.ado
st0119/raschtestv7.hlp
st0119/gammasym.ado
st0119/gammasym.hlp
st0119/gausshermite.ado
st0119/gausshermite.hlp
st0119/geekel2d.ado
st0119/geekel2d.hlp
ANCILLARY FILES (type net get st0119)
st0119/data.dta
st0119/outrasch.do

A package-description page describes the package and tells you how to install the component files. Package-description pages potentially describe two types of files:

1. Installation files: files that you type net install to install and that are required to make the addition work.
2. Ancillary files: additional files that you might want to install-you type net get to install thembut that you can ignore. Ancillary files are typically datasets that are useful for demonstration purposes. Ancillary files are not really installed in the sense of being copied to an official place for use by Stata itself. They are merely copied into the current directory so that you may use them if you wish.
You install the official files by typing net install followed by the package name. For example, to install st0119, you would type
```
. net install st0119
checking st0119 consistency and verifying not already installed...
installing into c:\ado\plus\ ...
installation complete.
```

You get the ancillary files-if there are any and if you want them-by typing net get followed by the package name:

```
. net get st0119
checking st0119 consistency and verifying not already installed...
copying into current directory...
    copying data.dta
    copying outrasch.do
ancillary files successfully copied.
```

Most users ignore the ancillary files.
Once you have installed a package-by typing net install-use ado to redisplay the packagedescription page whenever you wish:

```
. ado describe st0119
```

[1] package st0119 from https://www.stata-journal.com/software/sj7-1
TITLE
SJ7-1 st0119. Rasch analysis
DESCRIPTION/AUTHOR (S)
Rasch analysis
by Jean-Benoit Hardouin, University of Nantes, France
Support: jean-benoit.hardouin@univ-nantes.fr
After installation, type help gammasym, gausshermite,
geekel2d, raschtest, and raschtestv7
INSTALLATION FILES
r/raschtest.ado
r/raschtest.hlp
r/raschtestv7.ado
r/raschtestv7.hlp
g/gammasym.ado
g/gammasym.hlp
g/gausshermite.ado
g/gausshermite.hlp
g/geekel2d.ado
g/geekel2d.hlp
INSTALLED ONT
24 Feb 2023

The package-description page shown by ado includes the location from which we got the package and when we installed it. It does not mention the ancillary files that were originally part of this package because they are not tracked by ado.

## Where packages are installed

Packages should be installed in PLUS or SITE, which are code words that Stata understands and that correspond to some real directories on your computer. Typing sysdir will tell you where these are, if you care.

```
. sysdir
    STATA: C:\Program Files\Stata18\
        BASE: C:\Program Files\Stata18\ado\base\
        SITE: C:\Program Files\Stata18\ado\site\
        PLUS: c:\ado\plus\
PERSONAL: c:\ado\personal\
OLDPLACE: c:\ado\
```

If you type sysdir, you may obtain different results.

By default, net installs in the PLUS directory, and ado tells you about what is installed there. If you are on a multiple-user system, you may wish to install some packages in the SITE directory. This way, they will be available to other Stata users. To do that, before using net install, type

```
. net set ado SITE
```

and when reviewing what is installed or removing packages, redirect ado to that directory:

```
. ado ..., from(SITE)
```

In both cases, you type SITE because Stata will understand that SITE means the site ado-directory as defined by sysdir. To install into SITE, you must have write access to that directory.

If you reset where net installs and then, in the same session, wish to install into your private ado-directory, type

```
. net set ado PLUS
```

That is how things were originally. If you are confused as to where you are, type net query.

## A summary of the net command

The net command displays content pages and package-description pages. Such pages are provided over the Internet, and most users get them there. We recommend that you start at https://www.stata.com and work out from there. We also recommend using net search to find packages of interest to you; see $[R]$ net search.
net from moves you to a location and displays the content page.
net cd and net link change from your current location to other locations. net cd enters subdirectories of the original location. net link jumps from one location to another, depending on the code on the content page.
net describe lists a package-description page. Packages are named, and you type net describe pkgname.
net install installs a package into your copy of Stata. net get copies any additional files (ancillary files) to your current directory.
net sj simplify loading files from the Stata Journal.
net sj vol-issue
is a synonym for typing
net from https://www.stata-journal.com/software/sjvol-issue
whereas
net sj vol-issue insert
is a synonym for typing

```
net from https://www.stata-journal.com/software/sjvol-issue
net describe insert
```

net set controls where net installs files. By default, net installs in the PLUS directory; see [P] sysdir. net set ado SITE would cause subsequent net commands to install in the SITE directory. net set other sets where ancillary files, such as .dta files, are installed. The default is the current directory.

## A summary of the ado command

The ado command lists the package descriptions of previously installed packages.
Typing ado without arguments is the same as typing ado dir. Both list the names and titles of the packages you have installed.
ado describe lists full package-description pages.
ado uninstall removes packages from your computer.
Because you can install packages from a variety of sources, the package names may not always be unique. Thus the packages installed on your computer are numbered sequentially, and you may refer to them by name or by number. For instance, say that you wanted to get rid of the robust-to-everything statistic command you installed. Type

```
. ado, find("robust-to-everything")
[15] package rte_stat from http://www.wemakeitupaswego.edu/faculty/sgazer
    rte_stat. The robust-to-everything statistic; update.
```

You could then type

```
. ado uninstall rte_stat
```

or

```
. ado uninstall [15]
```

Typing ado uninstall rte_stat would work only if the name rte_stat were unique; otherwise, ado would refuse, and you would have to type the number.

The find() option is allowed with ado dir and ado describe. It searches the package description for the word or phrase you specify, ignoring case (alpha matches Alpha). The complete package description is searched, including the author's name and the name of the files. Thus if rte was the name of a command that you wanted to eliminate, but you could not remember the name of the package, you could type

```
. ado, find(rte)
```

[15] package rte_stat from http://www.wemakeitupaswego.edu/faculty/sgazer
rte_stat. The robust-to-everything statistic; update.

## Relationship of net and ado to the point-and-click interface

Users may instead select Help > SJ and community-contributed features. There are advantages and disadvantages:

1. Flipping through content and package-description pages is easier; it is much like a browser. See [GS] 19 Updating and extending Stata-Internet functionality (GSM, GSU, or GSW).
2. When browsing a product-description page, note that the .sthlp files are highlighted. You may click on .sthlp files to review them before installing the package.
3. You may not redirect from where ado searches for files.

## Creating your own site

Users who wish to share content with the Stata community often do so via the Statistical Software Components (SSC) Archive. See [R] ssc and also see http://repec.org/bocode/s/sscsubmit.html.

The rest of this entry concerns how to create your own site to distribute additions to Stata. The idea is that you have written additions for use with Stata—say, xyz.ado and xyz.sthlp—and you wish to put them out so that coworkers or researchers at other institutions can easily install them. Or, perhaps you just have a dataset that you and others want to share.

In any case, all you need is a webpage. You place the files that you want to distribute on your webpage (or in a subdirectory), and you add two more files - a content file and a package-description file-and you are done.

## Format of content and package-description files

The content file describes the content page. It must be named stata.toc:

| OFF <br> (to make site unavailable temporarily) <br> * lines starting with $*$ are comments; they are ignored <br> * blank lines are ignored, too <br> * v indicates version-specify v 3 , which is the current version of .toc files <br> v 3 <br> * d lines display description text <br> * the first d line is the title, and the remaining ones are text <br> * blank d lines display a blank line <br> d title <br> d text <br> d text <br> d <br> ... <br> * 1 lines display links <br> 1 word-to-show path-or-url [description] <br> 1 word-to-show path-or-url [description] <br> ... <br> * t lines display other directories within the site <br> t path [description] <br> t path [description] <br> ... <br> * p lines display packages <br> p pkgname [description] <br> p pkgname [description] |  |
| :---: | :---: |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Package files describe packages and are named pkgname.pkg:

* lines starting with $*$ are comments; they are ignored
* blank lines are ignored, too
* v indicates version-specify v 3 , which is the current version of. toc files
v 3
* d lines display package description text
* the first d line is the title, and the remaining ones are text
* blank d lines display a blank line
d title
d text
d Distribution-Date: date
d text
d
* $f$ identifies the component files
f [path/]filename [description]
f [path/]filename [description]
* e line is optional; it means stop reading
e

Note the Distribution-Date description line. This line is optional but recommended. Stata can look for updates to community-contributed programs with the ado update command if the package files from which those programs were installed contain a Distribution-Date description line.

## Example 1

Say that we want the user to see the following:

```
. net from http://www.university.edu/~me
```

http://www.university.edu/~me
Chris Farrar, Uni University
PACKAGES you could -net describe-:
xyz interval-truncated survival
. net describe xyz
package xyz from http://www.university.edu/~me
TITLE
xyz. interval-truncated survival.
DESCRIPTION/AUTHOR(S)
C. Farrar, Uni University.
INSTALLATION FILES (type net install xyz)
xyz.ado
xyz.sthlp
ANCILLARY FILES
(type net get xyz)
sample.dta

The files needed to do this would be

```
v 3
d Chris Farrar, Uni University
p xyz interval-truncated survival
```

```
v 3
    d xyz. interval-truncated survival.
    d C. Farrar, Uni University.
    f xyz.ado
    f xyz.sthlp
    f sample.dta
```

On his homepage, Chris would place the following files:

| stata.toc | (shown above) |
| :--- | :--- |
| xyz.pkg | (shown above) |
| xyz.ado | file to be delivered (for use by net install) |
| xyz.sthlp | file to be delivered (for use by net install) |
| sample.dta | file to be delivered (for use by net get) |

Chris does nothing to distinguish ancillary files from installation files.

## Example 2

S. Gazer wants to create a more complex site:
. net from http://www.wemakeitupaswego.edu/faculty/sgazer
http://www. wemakeitupaswego.edu/faculty/sgazer
Data-free inference materials
S. Gazer, Department of Applied Theoretical Mathematics

Also see my homepage for the preprint of "Irrefutable inference".
PLACES you could -net link- to:
stata StataCorp website
DIRECTORIES you could -net cd- to:
ir irrefutable inference programs (work in progress)
PACKAGES you could -net describe-:
rtec Robust-to-everything-conceivable statistic
rte Robust-to-everything statistic

```
. net describe rte
```

package rte from http://www.wemakeitupaswego.edu/faculty/sgazer/

## TITLE

rte. The robust-to-everything statistic; update.

## DESCRIPTION/AUTHOR (S)

## S. Gazer, Dept. of Applied Theoretical Mathematics, WMIUAWG Univ.

 Aleph-0 100\% confidence intervals proved too conservative for some applications; Aleph-1 confidence intervals have been substituted. The new robust-to-everything supplants the previous robust-to-everything-conceivable statistic. See "Inference in the absence of data" (forthcoming). After installation, see help rte.Distribution-Date: 20190320
Support: email sgazer@wemakeitupaswego.edu

```
INSTALLATION FILES
(type net install rte_stat)
```

rte.ado
rte.sthlp
nullset.ado
random.ado
ANCILLARY FILES
(type net get rte_stat)
empty.dta

The files needed to do this would be

```
v 3
d Data-free inference materials
d S. Gazer, Department of Applied Theoretical Mathematics
d
d Also see my homepage for the preprint of "Irrefutable inference".
l stata https://www.stata.com
t ir irrefutable inference programs (work in progress)
p rtec Robust-to-everything-conceivable statistic
p rte Robust-to-everything statistic
```

```
v 3
d rte. The robust-to-everything statistic; update.
d {bf:S. Gazer, Dept. of Applied Theoretical Mathematics, WMIUAWG Univ.}
d Aleph-0 100% confidence intervals proved too conservative for some
d applications; Aleph-1 confidence intervals have been substituted.
d The new robust-to-everything supplants the previous robust-to-
d everything-conceivable statistic. See "Inference in the absence
d of data" (forthcoming). After installation, see help {bf:rte}.
d
d Distribution-Date: 20190320
d
d Support: email sgazer@wemakeitupaswego.edu
f rte.ado
f rte.sthlp
f nullset.ado
f random.ado
f empty.dta
```

On his homepage, Mr. Gazer would place the following files:

```
stata.toc (shown above)
rte.pkg (shown above)
rte.ado (file to be delivered)
rte.sthlp (file to be delivered)
nullset.ado (file to be delivered)
random.ado (file to be delivered)
empty.dta (file to be delivered)
rtec.pkg the other package referred to in stata.toc
rtec.ado the corresponding files to be delivered
rtec.sthlp
ir/stata.toc the contents file for when the user types net cd ir
ir/... whatever other .pkg files are referred to
ir/... whatever other files are to be delivered
```

If Mr. Gazer later updated the rte package, he could change the Distribution-Date description line in his package. Then, if someone who had previously installed the rte packaged wanted to obtain the latest version, that person could use the ado update command; see [R] ado update.

For complex sites, a different structure may prove more convenient:

```
stata.toc (shown above)
rte.pkg (shown above)
rtec.pkg
rte/
rte/rte.ado
rte/rte.sthlp
rte/nullset.ado
rte/random.ado
rte/empty.dta
rtec/
rtec/...
ir/stata.toc
ir/*.pkg
ir/*/...
```

(shown above)
(shown above)
the other package referred to in stata.toc
directory containing re files to be delivered:
(file to be delivered)
(file to be delivered)
(file to be delivered)
(file to be delivered)
(file to be delivered)
directory containing rtec files to be delivered:
(files to be delivered)
the contents file for when the user types net cd ir
whatever other package files are referred to
whatever other files are to be delivered

If you prefer this structure, it is simply a matter of changing the bottom of the rte.pkg from

```
f rte.ado
f rte.sthlp
f nullset.ado
f random.ado
f empty.dta
```

to

```
f rte/rte.ado
f rte/rte.sthlp
f rte/nullset.ado
f rte/random.ado
f rte/empty.dta
```

In writing paths and files, the directory separator forward slash (/) is used, regardless of operating system, because this is what the Internet uses.

It does not matter whether the files you put out are in Windows, Mac, or Unix format (how lines end is recorded differently). When Stata reads the files over the Internet, it will figure out the file format on its own and will automatically translate the files to what is appropriate for the receiver.

## Additional package directives

F filename is similar to $f$ filename, except that, when the file is installed, it will always be copied to the system directories (and not the current directory).
With $f$ filename, the file is installed into a directory according to the file's suffix. For instance, xyz.ado would be installed in the system directories, whereas xyz.dta would be installed in the current directory.

Coding F xyz.ado would have the same result as coding $f$ xyz.ado.
Coding F xyz.dta, however, would state that xyz.dta is to be installed in the system directories.
g platformname filename is also a variation on f filename. It specifies that the file be installed only if the user's operating system is of type platformname; otherwise, the file is ignored. The platform names are WIN64 (64-bit x86-64) for Windows; MACARM64 (Apple Silicon, GUI) and OSX. ARM64 (Apple Silicon, console), MACINTEL64 (64-bit Intel, GUI) and OSX.X8664 (64-bit Intel, console) for Mac; and LINUX64 (64-bit x86-64) and LINUX64P (64-bit x86-64, libpng v1.6) for Linux. For LINUX64P, the LINUX64-specific version of a file will be installed if no LINUX64P-specific one exists. LINUX64 will not install a LINUX64P-specific version of a file because of possible changes in system libraries.
G platformname filename is a variation on F filename. The file, if not ignored, is to be installed in the system directories.
g platformname filename1 filename2 is a more detailed version of g platformname filename. In this case, filename 1 is the name of the file on the server (the file to be copied), and filename 2 is to be the name of the file on the user's system; for example, you might code

```
g WIN64 mydll.forwin mydll.plugin
g LINUX64 mydll.forlinux mydll.plugin
```

When you specify one filename, the result is the same as specifying two identical filenames.
G platformname filename1 filename 2 is the install-in-system-directories version of g platformname filename1 filename2.
h filename asserts that filename must be loaded, or this package is not to be installed; for example, you might code

```
g WIN64 mydll.forwin mydll.plugin
g LINUX64 mydll.forlinux mydll.plugin
h mydll.plugin
```

if you were offering the plugin mydll.plugin for Windows and Linux only.

## SMCL in content and package-description files

The text listed on the second and subsequent d lines in both stata.toc and pkgname.pkg may contain SMCL as long as you include v 3; see [P] smcl.

Thus, in rte.pkg, S. Gazer coded the third line as

```
d {bf:S. Gazer, Dept. of Applied Theoretical Mathematics, WMIUAWG Univ.}
```


## Error-free file delivery

Most people transport files over the Internet and never worry about the file being corrupted in the process because corruption rarely occurs. If, however, the files must be delivered perfectly or not at all, you can include checksum files in the directory.

For instance, say that big.dta is included in your package and that it must be sent perfectly. First, use Stata to make the checksum file for big.dta
. checksum big.dta, save
That command creates a small file called big.sum; see [D] checksum. Then, copy both big.dta and big.sum to your homepage.

If you do this, be cautious. If you put big.dta and big. sum on your homepage and then later change big.dta without changing big.sum, people will think that there are transmission errors when they try to download big.dta.

## Also see

[R] ado update - Update community-contributed packages
$[R]$ net search - Search the Internet for installable packages
[R] netio - Control Internet connections
[R] search - Search Stata documentation and other resources
[R] sj — Stata Journal installation instructions
[R] ssc - Install and uninstall packages from SSC
[R] update - Check for official updates
[D] checksum - Calculate checksum of file
[P] smcl - Stata Markup and Control Language
[GSM] 19 Updating and extending Stata-Internet functionality
[GSU] 19 Updating and extending Stata-Internet functionality
[GSW] 19 Updating and extending Stata-Internet functionality
[U] 29 Using the Internet to keep up to date

## Title

net search - Search the Internet for installable packages

| Description | Quick start | Syntax | Options |
| :--- | :--- | :--- | :--- |
| Remarks and examples | Also see |  |  |

## Description

net search searches the Internet for community-contributed additions to Stata, including, but not limited to, community-contributed additions published in the Stata Journal (SJ). net search lists the available additions that contain the specified keywords.

The community-contributed materials found are available for immediate download by using the net command or by clicking on the link.

In addition to typing net search, you may select Help > Search... and choose Search net resources. This is the recommended way to search for community-contributed additions to Stata.

## Quick start

Search community-contributed commands for all terms word1, word2, and word3 net search word1 word2 word3

Same as above, but search for any term
net search word1 word2 word3, or
Search Internet sources other than the Stata Journal

```
net search word1 word2 word3, nosj
```


## Syntax

```
net search word [word ...] [, options]
```

options Description
or list packages that contain any of the keywords; default is all
nosj search non-SJ sources
tocpkg search both tables of contents and packages; the default
toc search tables of contents only
pkg search packages only
everywhere search packages for match
filenames search filenames associated with package for match errnone make return code 111 instead of 0 when no matches found

## Options

or is relevant only when multiple keywords are specified. By default, net search lists only packages that include all the keywords. or changes the command to list packages that contain any of the keywords.
nosj specifies that net search not list matches that were published in the SJ.
tocpkg, toc, and pkg determine what is searched. tocpkg is the default, meaning that both tables of contents (tocs) and packages (pkgs) are searched. toc restricts the search to tables of contents. pkg restricts the search to packages.
everywhere and filenames determine where in packages net search looks for keywords. The default is everywhere. filenames restricts net search to search for matches only in the filenames associated with a package. Specifying everywhere implies pkg.
errnone is a programmer's option that causes the return code to be 111 instead of 0 when no matches are found.

## Remarks and examples

net search searches the Internet for community-contributed additions to Stata. If you want to search the Stata documentation for a particular topic, command, or author, see [R] search. net search word $[$ word ...] (without options) is equivalent to typing search word [word ...], net.

Remarks are presented under the following headings:

Topic searches<br>Author searches<br>Command searches<br>Where does net search look?<br>How does net search work?

## Topic searches

## Example: Find what is available about random effects

. net search random effect
Comments:

- It is best to search using the singular form of a word. net search random effect will find both "random effect" and "random effects".
- net search random effect will also find "random-effect" because net search performs a string search and not a word search.
- net search random effect lists all packages containing the words "random" and "effect", not necessarily used together.
- If you wanted all packages containing the word "random" or the word "effect", you would type net search random effect, or.


## Author searches

## Example: Find what is available by author Jeroen Weesie

. net search weesie
Comments:

- You could type net search jeroen weesie, but that might list fewer results because sometimes the last name is used without the first.
- You could type net search Weesie, but it would not matter. Capitalization is ignored in the search.


## Example: Find what is available by Jeroen Weesie, excluding SJ materials

```
. net search weesie, nosj
```

- The SJ tends to dominate search results because so much has been published in it. If you know that what you are looking for is not in the SJ, specifying the nosj option will narrow the search.
- net search weesie lists everything that net search weesie, nosj lists, and more. If you just type net search weesie, look down the list. SJ materials are listed first, and non-SJ materials are listed last.


## Command searches

## Example: Find the community-contributed command kursus

```
. net search kursus, file
```

- You could just type net search kursus, and that will list everything net search kursus, file lists, and more. Because you know kursus is a command, however, there must be a kursus.ado file associated with the package. Typing net search kursus, file narrows the search.
- You could also type net search kursus.ado, file to narrow the search even more.


## Where does net search look?

net search looks everywhere, not just at https://www.stata.com.
net search begins by looking at https://www.stata.com, but then follows every link, which takes it to other places, and then follows every link again, which takes it to even more places, and so on.

Authors: Please let us know if you have a site that we should include in our search by sending an email to webmaster@stata.com. We will then link to your site from ours to ensure that net search finds your materials. That is not strictly necessary, however, as long as your site is directly or indirectly linked from some site that is linked to ours.

## How does net search work?



Our website maintains a database of Stata resources. When you use net search, it contacts https://www.stata.com with your request, https://www.stata.com searches its database, and Stata returns the results to you.

Another part of the system is called the crawler, which searches the web for new Stata resources to add to the net search database and verifies that the resources already found are still available. When a new resource becomes available, the crawler takes about two days to add it to the database, and, similarly, if a resource disappears, the crawler takes roughly two days to remove it from the database.

## Also see

[R] ado update - Update community-contributed packages
$[R]$ net - Install and manage community-contributed additions from the Internet
[R] search - Search Stata documentation and other resources
[R] sj - Stata Journal installation instructions
[R] ssc - Install and uninstall packages from SSC
$[\mathrm{R}]$ update - Check for official updates

## Title

netio - Control Internet connections
Description Syntax Options Remarks and examples Also see

## Description

Some commands (for example, net and update) are designed specifically for use over the Internet. Many other Stata commands that read a file (for example, copy, type, and use) can also read directly from a URL. All of these commands will usually work without your ever needing to concern yourself with the set commands discussed here. These set commands provide control over network system parameters.

If you experience problems when using Stata's network features, ask your system administrator if your site uses a proxy. A proxy is a server between your computer and the rest of the Internet, and your computer may need to communicate with other computers on the Internet through this proxy. If your site uses a proxy, your system administrator can provide you with its host name and the port your computer can use to communicate with it. If your site's proxy requires you to $\log$ in to it before it will respond, your system administrator will provide you with a user ID and password.
set httpproxyhost sets the name of the host to be used as a proxy server. set httpproxyport sets the port number. set httpproxy turns on or off the use of a proxy server, leaving the proxy host name and port intact, even when not in use.

Under the Mac and Windows operating systems, when you set httpproxy on, Stata will attempt to obtain the values of httpproxyhost and httpproxyport from the operating system if they have not been previously set. set httpproxy on, init attempts to obtain these values from the operating system, even if they have been previously set.

If the proxy requires authorization (user ID and password), set authorization on via set httpproxyauth on. The proxy user and proxy password must also be set to the appropriate user ID and password by using set httpproxyuser and set httpproxypw.

Stata remembers the various proxy settings between sessions and does not need a permanently option.

## Syntax

Turn on or off the use of a proxy server
set httpproxy $\{$ on $\mid$ off $\}[$, init]
Set proxy host name
set httpproxyhost ["]name["]
Set the proxy port number
set httpproxyport \#

Turn on or off proxy authorization
set httpproxyauth $\{$ on | off $\}$
Set proxy authorization user ID
set httpproxyuser ["]name["]
Set proxy authorization password
set httpproxypw ["] password["]

## Options

init specifies that set httpproxy on attempts to initialize httpproxyhost and httpproxyport from the operating system (Mac and Windows only).

## Remarks and examples

If you receive an error message, see https://www.stata.com/support/faqs/web/ for the latest information.

## Also see

[R] query - Display system parameters
[P] creturn - Return c-class values
[U] 29 Using the Internet to keep up to date

## Title

nl - Nonlinear least-squares estimation
Description
Options
Acknowledgments

Quick start<br>Remarks and examples<br>References

Menu
Stored results Also see

Syntax
Methods and formulas

## Description

nl fits an arbitrary nonlinear regression function by least squares. With the interactive version of the command, you enter the function directly on the command line or in the dialog box by using a substitutable expression. If you have a function that you use regularly, you can write a substitutable expression program and use the second syntax to avoid having to reenter the function every time. The function evaluator program version gives you the most flexibility in exchange for increased complexity; with this version, your program is given a vector of parameters and a variable list, and your program computes the regression function.

When you write a substitutable expression program or function evaluator program, the first two letters of the name must be nl. sexp_prog and func_prog refer to the name of the program without the first two letters. For example, if you wrote a function evaluator program named nlregss, you would type nl regss © ... to estimate the parameters.

## Quick start

Linear model of y with parameters b 0 for the constant and b 1 for x 1
$\mathrm{nl}(\mathrm{y}=\{\mathrm{b} 0\}+\{\mathrm{b} 1\} * x 1)$
Add the variables() option to allow for missing values of y and x 1
$\mathrm{nl}(\mathrm{y}=\{\mathrm{b} 0\}+\{\mathrm{b} 1\} * \mathrm{x} 1)$, variables ( y x1)
Same as above, but specify starting values
$\mathrm{nl}(\mathrm{y}=\{\mathrm{b} 0=.5\}+\{\mathrm{b} 1=2\} * \mathrm{x} 1)$, variables $(\mathrm{y} \mathrm{x1})$
Add variables $x 2$ and $x 3$ and parameters $b 2$ and $b 3$
$\mathrm{nl}(\mathrm{y}=\{\mathrm{b} 0\}+\{\mathrm{b} 1\} * x 1+\{b 2\} * x 2+\{b 3\} * x 3)$, variables $(\mathrm{y} x 1 \mathrm{x} 2 \mathrm{x} 3)$
Same as above, but use $\{x b:\}$ to specify a linear combination of variables
$\mathrm{nl}(\mathrm{y}=\{\mathrm{b} 0=.5\}+\{\mathrm{xb}: \mathrm{x} 1 \mathrm{x} 2 \mathrm{x} 3\})$, variables ( y x1 x 2 x 3 )
An exponential model
$\mathrm{nl}\left(\mathrm{y}=\{\mathrm{b} 0\}+\{\mathrm{b} 1\} *\{\mathrm{~b} 2\}^{\wedge} \mathrm{x} 1\right)$, variables $(\mathrm{y} \mathrm{x} 1)$
Same as above, but use nl's built-in function exp3 to specify the model

```
nl exp3: y x1, variables(y x1)
```


## Menu

Statistics $>$ Linear models and related $>$ Nonlinear least-squares estimation

## Syntax

Interactive version
$\mathrm{nl}($ depvar $=\langle\operatorname{sexp}\rangle)[$ if $][$ in $][$ weight $][$, options $]$

Programmed substitutable expression version
nl sexp_prog : depvar $[$ varlist $][$ if $][$ in $][$ weight $][$, options $]$

Function evaluator program version
nl func_prog @ depvar [varlist] [if] [in][weight],
$\{\underline{\text { parameters }(\text { namelist }) \mid \underline{\text { nparameters }}(\#)\}[\text { options }] ~}$
where
depvar is the dependent variable;
$<\operatorname{sexp}>$ is a substitutable expression;
sexp_prog is a substitutable expression program; and func_prog is a function evaluator program.

## options

Model
variables (varlist) initial (initial_values)

* parameters (namelist)
* nparameters (\#)
sexp_options
func_options
Model 2

```
lnlsq(#)
```


## noconstant

hasconstant (name)
SE/Robust
vce (vcetype)

Reporting
level(\#)
leave
title(string)
title2 (string)
display_options
Optimization
optimization_options
eps (\#)
delta(\#)
coeflegend

Description
variables in model
initial values for parameters
parameters in model (function evaluator program version only)
number of parameters in model
(function evaluator program version only)
options for substitutable expression program
options for function evaluator program
use $\log$ least-squares where $\ln ($ depvar $-\#)$ is assumed to be normally distributed
the model has no constant term; seldom used use name as constant term; seldom used
vcetype may be gnr, robust, cluster clustvar, bootstrap, jacknife, hac kernel, hc2, or hc3
set confidence level; default is level (95)
create variables containing derivative of $E(y)$
display string as title above the table of parameter estimates display string as subtitle
control column formats and line width
control the optimization process; seldom used specify \# for convergence criterion; default is eps (1e-5)
specify \# for computing derivatives; default is delta (4e-7)
display legend instead of statistics
*For function evaluator program version, you must specify parameters (namelist) or nparameters(\#), or both. bootstrap, by, collect, jackknife, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands. Weights are not allowed with the bootstrap prefix; see $[R]$ bootstrap. aweights are not allowed with the jackknife prefix; see $[R]$ jackknife. vce(), leave, and weights are not allowed with the svy prefix; see [SVY] svy. aweights, fweights, and iweights are allowed; see [U] 11.1.6 weight.
coeflegend does not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

$\qquad$ Model
variables (varlist) specifies the variables in the model. nl ignores observations for which any of these variables have missing values. If you do not specify variables (), then nl issues an error message with return code 480 if the estimation sample contains any missing values.
initial(initial_values) specifies the initial values to begin the estimation. You can specify a $1 \times k$ matrix, where $k$ is the number of parameters in the model, or you can specify a parameter name, its initial value, another parameter name, its initial value, and so on. For example, to initialize alpha to 1.23 and delta to 4.57 , you would type

```
nl ... , initial(alpha 1.23 delta 4.57) ...
```

Initial values declared using this option override any that are declared within substitutable expressions. If you specify a parameter that does not appear in your model, nl exits with error code 480. If you specify a matrix, the values must be in the same order that the parameters are declared in your model. nl ignores the row and column names of the matrix.
parameters (namelist) specifies the names of the parameters in the model. The names of the parameters must adhere to the naming conventions of Stata's variables; see [U] $\mathbf{1 1 . 3}$ Naming conventions. If you specify both parameters() and nparameters(), the number of names in the former must match the number specified in the latter; if not, nl issues an error message with return code 198.
nparameters (\#) specifies the number of parameters in the model. If you do not specify names with the parameters() option, nl names them $\mathrm{b} 1, \mathrm{~b} 2, \ldots, \mathrm{~b} \#$. If you specify both parameters() and nparameters(), the number of names in the former must match the number specified in the latter; if not, nl issues an error message with return code 198.
sexp_options refer to any options allowed by your sexp_prog.
func_options refer to any options allowed by your func_prog.

## Model 2

lnlsq(\#) fits the model by using log least-squares, which we define as least squares with shifted lognormal errors. In other words, $\ln ($ depvar - \#) is assumed to be normally distributed. Sums of squares and deviance are adjusted to the same scale as depvar.
noconstant indicates that the function does not include a constant term. This option is generally not needed, even if there is no constant term in the model, unless the coefficient of variation (over observations) of the partial derivative of the function with respect to a parameter is less than eps () and that parameter is not a constant term.
hasconstant (name) indicates that parameter name be treated as the constant term in the model and that $n l$ should not use its default algorithm to find a constant term. As with noconstant, this option is seldom used.

## SE/Robust

vce(vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (gnr), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
vce (gnr), the default, uses the conventionally derived variance estimator for nonlinear models fit using Gauss-Newton regression.
nl also allows the following:
vce (hac kernel [\#]) specifies that a heteroskedasticity- and autocorrelation-consistent (HAC) variance estimate be used. HAC refers to the general form for combining weighted matrices to form the variance estimate. There are three kernels available for nl :
\# specifies the number of lags. If \# is not specified, $N-2$ is assumed. vce (hac kernel [\#]) is not allowed if weights are specified.
vce (hc2) and vce (hc3) specify alternative bias corrections for the robust variance calculation. vce(hc2) and vce(hc3) may not be specified with the svy prefix. By default, vce (robust) uses $\widehat{\sigma}_{j}^{2}=\{n /(n-k)\} u_{j}^{2}$ as an estimate of the variance of the $j$ th observation, where $u_{j}$ is the calculated residual and $n /(n-k)$ is included to improve the overall estimate's small-sample properties.
vce (hc2) instead uses $u_{j}^{2} /\left(1-h_{j j}\right)$ as the observation's variance estimate, where $h_{j j}$ is the $j$ th diagonal element of the hat (projection) matrix. This produces an unbiased estimate of the covariance matrix if the model is homoskedastic. vce(hc2) tends to produce slightly more conservative confidence intervals than vce (robust).
vce(hc3) uses $u_{j}^{2} /\left(1-h_{j j}\right)^{2}$ as suggested by Davidson and MacKinnon (1993 and 2004), who report that this often produces better results when the model is heteroskedastic. vce (hc3) produces confidence intervals that tend to be even more conservative.
See, in particular, Davidson and MacKinnon (2004, 239), who advocate the use of vce (hc2) or vce (hc3) instead of the plain robust estimator for nonlinear least squares.

Reporting
level(\#); see [R] Estimation options.
leave leaves behind after estimation a set of new variables with the same names as the estimated parameters containing the derivatives of $E(y)$ with respect to the parameters. If the dataset contains an existing variable with the same name as a parameter, then using leave causes nl to issue an error message with return code 110 .
leave may not be specified with vce(cluster clustvar) or the svy prefix.
title(string) specifies an optional title that will be displayed just above the table of parameter estimates.
title2(string) specifies an optional subtitle that will be displayed between the title specified in title() and the table of parameter estimates. If title2() is specified but title() is not, title2() has the same effect as title().
display_options: cformat (\%fmt), pformat (\%fmt), sformat (\%fmt), and nolstretch; see [R] Estimation options.
$\sqrt{\text { Optimization }}$
optimization_options: iterate (\#), [no]log, trace. iterate() specifies the maximum number of iterations, $\log /$ nolog specifies whether to show the iteration $\log$ (see set iterlog in [R] set iter), and trace specifies that the iteration log should include the current parameter vector. These options are seldom used.
eps (\#) specifies the convergence criterion for successive parameter estimates and for the residual sum of squares. The default is eps $(1 e-5)$.
delta(\#) specifies the relative change in a parameter to be used in computing the numeric derivatives. The derivative for parameter $\beta_{i}$ is computed as $\left\{f\left(X, \beta_{1}, \beta_{2}, \ldots, \beta_{i}+d, \beta_{i+1}, \ldots\right)-\right.$ $\left.f\left(X, \beta_{1}, \beta_{2}, \ldots, \beta_{i}, \beta_{i+1}, \ldots\right)\right\} / d$, where $d$ is $\delta\left(\beta_{i}+\delta\right)$. The default is delta $(4 \mathrm{e}-7)$.

The following options are available with nl but are not shown in the dialog box:
coeflegend; see [R] Estimation options.

## Remarks and examples

Remarks are presented under the following headings:

Substitutable expressions<br>Substitutable expression programs<br>Built-in functions<br>Lognormal errors<br>Other uses<br>Weights<br>Potential errors<br>General comments on fitting nonlinear models<br>Function evaluator programs

nl fits an arbitrary nonlinear function by least squares. The interactive version allows you to enter the function directly on the command line or dialog box using substitutable expressions. You can write a substitutable expression program for functions that you fit frequently to save yourself time. Finally, function evaluator programs give you the most flexibility in defining your nonlinear function, though they are more complicated to use.

The next section explains the substitutable expressions that are used to define the regression function, and the section thereafter explains how to write substitutable expression program files so that you do not need to type in commonly used functions over and over. Later sections highlight other features of nl .

The final section discusses function evaluator programs. If you find substitutable expressions adequate to define your nonlinear function, then you can skip that section entirely. Function evaluator programs are generally needed only for complicated problems, such as multistep estimators. The program receives a vector of parameters at which it is to compute the function and a variable into which the results are to be placed.

## Substitutable expressions

You define the nonlinear function to be fit by nl by using a substitutable expression. Substitutable expressions are just like any other mathematical expressions involving scalars and variables, such as those you would use with Stata's generate command, except that the parameters to be estimated are bound in braces. See [U] 13.2 Operators and [U] 13.3 Functions for more information on expressions.

For example, suppose that you wish to fit the function

$$
y_{i}=\beta_{0}\left(1-e^{-\beta_{1} x_{i}}\right)+\epsilon_{i}
$$

where $\beta_{0}$ and $\beta_{1}$ are the parameters to be estimated and $\epsilon_{i}$ is an error term. You would simply type

$$
\text { . } \mathrm{nl}(\mathrm{y}=\{\mathrm{b} 0\} *(1-\exp (-1 *\{\mathrm{~b} 1\} * x)))
$$

You must enclose the entire equation in parentheses. Because b0 and b1 are enclosed in braces, nl knows that they are parameters in the model. nl will initialize b0 and b1 to zero by default. To request that nl initialize b 0 to 1 and b 1 to 0.25 , you would type

$$
\text { . } \mathrm{nl}(\mathrm{y}=\{\mathrm{b} 0=1\} *(1-\exp (-1 *\{\mathrm{~b} 1=0.25\} * x)))
$$

That is, inside the braces denoting a parameter, you put the parameter name followed by an equal sign and the initial value. If a parameter appears in your function multiple times, you need only specify an initial value only once (or never, if you wish to set the initial value to zero). If you do specify more than one initial value for the same parameter, nl will use the last value given. Parameter names must follow the same conventions as variable names. See [U] 11.3 Naming conventions.

Frequently, even nonlinear functions contain linear combinations of variables. As an example, suppose that you wish to fit the function

$$
y_{i}=\beta_{0}\left\{1-e^{-\left(\beta_{1} x_{1 i}+\beta_{2} x_{2 i}+\beta_{3} x_{3 i}\right)}\right\}+\epsilon_{i}
$$

nl allows you to declare a linear combination of variables by using the shorthand notation

```
. nl (y = {b0=1}*(1 - exp(-1*{xb: x1 x2 x3})))
```

In the syntax $\{\mathrm{xb}$ : x 1 x 2 x 3$\}$, you are telling nl that you are declaring a linear combination named xb that is a function of three variables, $\mathrm{x} 1, \mathrm{x} 2$, and $\mathrm{x} 3 . \mathrm{nl}$ will create three parameters, named $\mathrm{xb} \_\mathrm{x} 1, \mathrm{xb} \_\mathrm{x} 2$, and $\mathrm{xb} \_\mathrm{x} 3$, and initialize them to zero. Instead of typing the previous command, you could have typed

$$
\text { . } \mathrm{nl}\left(\mathrm{y}=\{\mathrm{b} 0=1\} *\left(1-\exp \left(-1 *\left(\left\{\mathrm{xb} \_\mathrm{x} 1\right\} * \mathrm{x} 1+\left\{\mathrm{xb} \_\mathrm{x} 2\right\} * \mathrm{x} 2+\left\{\mathrm{xb} \_\mathrm{x} 3\right\} * \mathrm{x} 3\right)\right)\right)\right)
$$

and yielded the same result. You can refer to the parameters created by $n l$ in the linear combination later in the function, though you must declare the linear combination first if you intend to do that. When creating linear combinations, nl ensures that the parameter names it chooses are unique and have not yet been used in the function.

In general, there are three rules to follow when defining substitutable expressions:

1. Parameters of the model are bound in braces: $\{b 0\}$, \{param\}, etc.
2. Initial values for parameters are given by including an equal sign and the initial value inside the braces: $\{b 0=1\}$, $\{$ param $=3.571\}$, etc.
3. Linear combinations of variables can be included using the notation \{eqname:varlist\}, for example, $\{\mathrm{xb}: \mathrm{mpg}$ price weight\}, \{score: wxz , etc. Parameters of linear combinations are initialized to zero.

If you specify initial values by using the initial() option, they override whatever initial values are given within the substitutable expression. Substitutable expressions are so named because, once values are assigned to the parameters, the resulting expression can be handled by generate and replace.

## > Example 1

We wish to fit the CES production function

$$
\begin{equation*}
\ln Q_{i}=\beta_{0}-\frac{1}{\rho} \ln \left\{\delta K_{i}^{-\rho}+(1-\delta) L_{i}^{-\rho}\right\}+\epsilon_{i} \tag{1}
\end{equation*}
$$

where $\ln Q_{i}$ is the $\log$ of output for firm $i ; K_{i}$ and $L_{i}$ are firm $i$ 's capital and labor usage, respectively; and $\epsilon_{i}$ is a regression error term. Because $\rho$ appears in the denominator of a fraction, zero is not a feasible initial value; for a CES production function, $\rho=1$ is a reasonable choice. Setting $\delta=0.5$ implies that labor and capital have equal impacts on output, which is also a reasonable choice for an initial value. We type
. use https://www.stata-press.com/data/r18/production
. nl (lnoutput $=\{b 0\}-1 /\{r h o=1\} * \ln \left(\{d e l t a=0.5\} * c a p i t a l^{\wedge}(-1 *\{r h o\})+\right.$
> (1 - \{delta\})*labor^(-1*\{rho\})))
Iteration 0: Residual $\mathrm{SS}=29.38631$
Iteration 1: Residual SS = 29.36637
Iteration 2: Residual SS = 29.36583
Iteration 3: Residual $\mathrm{SS}=29.36581$
Iteration 4: Residual $\mathrm{SS}=29.36581$
Iteration 5: Residual $\mathrm{SS}=29.36581$
Iteration 6: Residual $\mathrm{SS}=29.36581$
Iteration 7: Residual $\mathrm{SS}=29.36581$

| Source | SS | df | MS |
| ---: | ---: | ---: | :---: |
| Model | 91.144992 | 2 | 45.5724962 |
| Residual | 29.365806 | 97 | .302740263 |
| Total | 120.5108 | 99 | 1.21728079 |


| Number of obs | $=$ | 100 |
| :--- | ---: | ---: |
| R-squared | $=$ | 0.7563 |
| Adj R-squared | $=$ | 0.7513 |
| Root MSE | $=$ | .5502184 |
| Res. dev. | $=$ | 161.2538 |


| lnoutput | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | :---: | ---: |
| /b0 | 3.792158 | .099682 | 38.04 | 0.000 | 3.594316 | 3.989999 |
| /rho | 1.386993 | .472584 | 2.93 | 0.004 | .4490443 | 2.324941 |
| /delta | .4823616 | .0519791 | 9.28 | 0.000 | .3791975 | .5855258 |

Note: Parameter b0 is used as a constant term during estimation.
nl will attempt to find a constant term in the model and, if one is found, mention it at the bottom of the output. nl found b 0 to be a constant because the partial derivative $\partial \ln Q_{i} / \partial \mathrm{b} 0$ has a coefficient of variation less than eps () in the estimation sample.

The elasticity of substitution for the CES production function is $\sigma=1 /(1+\rho)$; and, having fit the model, we can use nlcom to estimate it:
. nlcom (1/(1 + _b[/rho]))
_nl_1: 1/(1 + _b[/rho])

| lnoutput | Coefficient | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | :---: | ---: |
| _nl_1 | .4189372 | .0829424 | 5.05 | 0.000 | .256373 | .5815014 |

See $[R]$ nlcom and [U] 13.5 Accessing coefficients and standard errors for more information.
nl 's output closely mimics that of regress; see [R] regress for more information. The $R^{2}$, sums of squares, and similar statistics are calculated in the same way that regress calculates them. If no "constant" term is specified, the usual caveats apply to the interpretation of the $R^{2}$ statistic; see the comments and references in Goldstein (1992). Unlike regress, nl does not report a model $F$ statistic, because a test of the joint significance of all the parameters except the constant term may not be relevant in a nonlinear model.

## Substitutable expression programs

If you fit the same model often or if you want to write an estimator that will operate on whatever variables you specify, then you will want to write a substitutable expression program. That program will return a macro containing a substitutable expression that nl can then evaluate, and it may optionally calculate initial values as well. The name of the program must begin with the letters nl.

To illustrate, suppose that you use the CES production function often in your work. Instead of typing in the formula each time, you can write a program like this:

```
program nlces, rclass
    version 18.0 // (or version 18.5 for StataNow)
    syntax varlist(min=3 max=3) [if]
    local logout : word 1 of 'varlist'
    local capital : word 2 of 'varlist'
    local labor : word 3 of 'varlist'
    // Initial value for b0 given delta=0.5 and rho=1
    tempvar y
    generate double 'y' = 'logout' + ln(0.5*'capital'^-1 + 0.5*'labor'^-1)
    summarize 'y' 'if', meanonly
    local bOval = r(mean)
    // Terms for substitutable expression
    local capterm "{delta=0.5}*'capital'^(-1*{rho})"
    local labterm "(1-{delta})*'labor'^(-1*{rho})"
    local term2 "1/{rho=1}*ln('capterm' + 'labterm')"
    // Return substitutable expression and title
    return local eq "'logout' = {b0='b0val'} - 'term2'"
    return local title "CES ftn., ln Q=`logout', K='capital', L=`labor'"
end
```

The program accepts three variables for log output, capital, and labor, and it accepts an if exp qualifier to restrict the estimation sample. All programs that you write to use with nl must accept an if exp qualifier because, when nl calls the program, it passes a binary variable that marks the estimation sample (the variable equals one if the observation is in the sample and zero otherwise). When calculating initial values, you will want to restrict your computations to the estimation sample, and you can do so by using if with any commands that accept if exp qualifiers. Even if your program does not calculate initial values or otherwise use the if qualifier, the syntax statement must still allow it. See [P] syntax for more information on the syntax command and the use of if.

As in the previous example, reasonable initial values for $\delta$ and $\rho$ are 0.5 and 1 , respectively. Conditional on those values, (1) can be rewritten as

$$
\begin{equation*}
\beta_{0}=\ln Q_{i}+\ln \left(0.5 K_{i}^{-1}+0.5 L_{i}^{-1}\right)-\epsilon_{i} \tag{2}
\end{equation*}
$$

so a good initial value for $\beta_{0}$ is the mean of the right-hand side of (2) ignoring $\epsilon_{i}$. Lines 7-10 of the function evaluator program calculate that mean and store it in a local macro. Notice the use of if in the summarize statement so that the mean is calculated only for the estimation sample.

The final part of the program returns two macros. The macro title is optional and defines a short description of the model that will be displayed in the output immediately above the table of parameter estimates. The macro eq is required and defines the substitutable expression that nl will use. If the expression is short, you can define it all at once. However, because the expression used here is somewhat lengthy, defining local macros and then building up the final expression from them is easier.

To verify that there are no errors in your program, you can call it directly and then use return list:

```
. use https://www.stata-press.com/data/r18/production
. nlces lnoutput capital labor
    (output omitted)
. return list
macros:
    r(title) : "CES ftn., ln Q=lnoutput, K=capital, L=labor"
    r(eq) : "lnoutput = {b0=3.711606264663641} - 1/{rho=1}*ln({delt
> a=0.5}*capital^(-1*{rho}) + (1-{delta})*labor^(-1*{rho}))"
```

The macro $r(e q)$ contains the same substitutable expression that we specified at the command line in the preceding example, except for the initial value for b0. In short, an nl substitutable expression program should return in $r$ (eq) the same substitutable expression you would type at the command line. The only difference is that when writing a substitutable expression program, you do not bind the entire expression inside parentheses.

Having written the program, you can use it by typing
. nl ces: lnoutput capital labor
(There is a space between nl and ces.) The output is identical to that shown in example 1, save for the title defined in the function evaluator program that appears immediately above the table of parameter estimates.

## Technical note

You will want to store nlces as an ado-file called nlces.ado. The alternative is to type the code into Stata interactively or to place the code in a do-file. While those alternatives are adequate for occasional use, if you save the program as an ado-file, you can use the function anytime you use Stata without having to redefine the program. When nl attempts to execute nlces, if the program is not in Stata's memory, Stata will search the disk(s) for an ado-file of the same name and, if found, automatically load it. All you have to do is name the file with the .ado suffix and then place it in a directory where Stata will find it. You should put the file in the directory Stata reserves for user-written ado-files, which, depending on your operating system, is c:\ado\personal (Windows), ~/ado/personal (Unix), or ~:ado:personal (Mac). See [U] 17 Ado-files.

Sometimes you may want to pass additional options to the substitutable expression program. You can modify the syntax statement of your program to accept whatever options you wish. Then when you call nl with the syntax
. nl func_prog: varlist, options
any options that are not recognized by nl (see the table of options at the beginning of this entry) are passed on to your function evaluator program. The only other restriction is that your program cannot accept an option named at because nl uses that option with function evaluator programs.

## Built-in functions

Some functions are used so often that $n l$ has them built in so that you do not need to write them yourself. nl automatically chooses initial values for the parameters, though you can use the initial(...) option to override them.

Three alternatives are provided for exponential regression with one asymptote:

$$
\begin{array}{ll}
\exp 3 & y_{i}=\beta_{0}+\beta_{1} \beta_{2}^{x_{i}}+\epsilon_{i} \\
\exp 2 & y_{i}=\beta_{1} \beta_{2}^{x_{i}}+\epsilon_{i} \\
\exp 2 \mathrm{a} & y_{i}=\beta_{1}\left(1-\beta_{2}^{x_{i}}\right)+\epsilon_{i}
\end{array}
$$

For instance, typing nl exp3: ras dvl fits the three-parameter exponential model (parameters $\beta_{0}$, $\beta_{1}$, and $\beta_{2}$ ) using $y_{i}=$ ras and $x_{i}=\operatorname{dvl}$.

Two alternatives are provided for the logistic function (symmetric sigmoid shape; not to be confused with logistic regression):

$$
\begin{array}{ll}
\log 4 & y_{i}=\beta_{0}+\beta_{1} /\left[1+\exp \left\{-\beta_{2}\left(x_{i}-\beta_{3}\right)\right\}\right]+\epsilon_{i} \\
\log 3 & y_{i}=\beta_{1} /\left[1+\exp \left\{-\beta_{2}\left(x_{i}-\beta_{3}\right)\right\}\right]+\epsilon_{i}
\end{array}
$$

Finally, two alternatives are provided for the Gompertz function (asymmetric sigmoid shape):

$$
\begin{array}{ll}
\text { gom4 } & y_{i}=\beta_{0}+\beta_{1} \exp \left[-\exp \left\{-\beta_{2}\left(x_{i}-\beta_{3}\right)\right\}\right]+\epsilon_{i} \\
\text { gom3 } & y_{i}=\beta_{1} \exp \left[-\exp \left\{-\beta_{2}\left(x_{i}-\beta_{3}\right)\right\}\right]+\epsilon_{i}
\end{array}
$$

## Lognormal errors

A nonlinear model with errors that are independent and identically distributed normal may be written as

$$
\begin{equation*}
y_{i}=f\left(\mathbf{x}_{i}, \boldsymbol{\beta}\right)+u_{i}, \quad u_{i} \sim N\left(0, \sigma^{2}\right) \tag{3}
\end{equation*}
$$

for $i=1, \ldots, n$. If the $y_{i}$ are thought to have a $k$-shifted lognormal instead of a normal distributionthat is, $\ln \left(y_{i}-k\right) \sim N\left(\zeta_{i}, \tau^{2}\right)$, and the systematic part $f\left(\mathbf{x}_{i}, \boldsymbol{\beta}\right)$ of the original model is still thought appropriate for $y_{i}$ —the model becomes

$$
\begin{equation*}
\ln \left(y_{i}-k\right)=\zeta_{i}+v_{i}=\ln \left\{f\left(\mathbf{x}_{i}, \boldsymbol{\beta}\right)-k\right\}+v_{i}, \quad v_{i} \sim N\left(0, \tau^{2}\right) \tag{4}
\end{equation*}
$$

This model is fit if lnlsq $(k)$ is specified.
If model (4) is correct, the variance of $\left(y_{i}-k\right)$ is proportional to $\left\{f\left(\mathbf{x}_{i}, \boldsymbol{\beta}\right)-k\right\}^{2}$. Probably the most common case is $k=0$, sometimes called "proportional errors" because the standard error of $y_{i}$ is proportional to its expectation, $f\left(\mathbf{x}_{i}, \boldsymbol{\beta}\right)$. Assuming that the value of $k$ is known, (4) is just another nonlinear model in $\boldsymbol{\beta}$, and it may be fit as usual. However, we may wish to compare the fit of (3) with that of (4) using the residual sum of squares (RSS) or the deviance $D, D=-2 \times \log$-likelihood, from each model. To do so, we must allow for the change in scale introduced by the log transformation.

Assuming, then, the $y_{i}$ to be normally distributed, Atkinson (1985, 85-87, 184), by considering the Jacobian $\prod\left|\partial \ln \left(y_{i}-k\right) / \partial y_{i}\right|$, showed that multiplying both sides of (4) by the geometric mean of $y_{i}-k$, $\dot{y}$, gives residuals on the same scale as those of $y_{i}$. The geometric mean is given by

$$
\dot{y}=e^{n^{-1}} \sum \ln \left(y_{i}-k\right)
$$

which is a constant for a given dataset. The residual deviance for (3) and for (4) may be expressed as

$$
\begin{equation*}
D(\widehat{\boldsymbol{\beta}})=\left\{1+\ln \left(2 \pi \widehat{\sigma}^{2}\right)\right\} n \tag{5}
\end{equation*}
$$

where $\widehat{\boldsymbol{\beta}}$ is the maximum likelihood estimate (MLE) of $\boldsymbol{\beta}$ for each model and $n \widehat{\sigma}^{2}$ is the RSS from (3), or that from (4) multiplied by $\dot{y}^{2}$.

Because (3) and (4) are models with different error structures but the same functional form, the arithmetic difference in their RSS or deviances is not easily tested for statistical significance. However, if the deviance difference is large ( $>4$, say), we would naturally prefer the model with the smaller deviance. Of course, the residuals for each model should be examined for departures from assumptions (nonconstant variance, nonnormality, serial correlations, etc.) in the usual way.

Alternatively, consider modeling

$$
\begin{gather*}
E\left(y_{i}\right)=1 /\left(C+A e^{B x_{i}}\right)  \tag{6}\\
E\left(1 / y_{i}\right)=E\left(y_{i}^{\prime}\right)=C+A e^{B x_{i}} \tag{7}
\end{gather*}
$$

where $C, A$, and $B$ are parameters to be estimated. Using the data $(y, x)=(0.04,5),(0.06,12)$, $(0.08,25),(0.1,35),(0.15,42),(0.2,48),(0.25,60),(0.3,75)$, and $(0.5,120)$ (Danuso 1991), fitting the models yields

| Model | $C$ | $A$ | $B$ | RSS | Deviance |
| :--- | ---: | ---: | ---: | ---: | ---: |
| (6) | 1.781 | 25.74 | -0.03926 | -0.001640 | -51.95 |
| (6) with lnlsq(0) | 1.799 | 25.45 | -0.04051 | -0.001431 | -53.18 |
| (7) | 1.781 | 25.74 | -0.03926 | 8.197 | 24.70 |
| (7) with lnlsq(0) | 1.799 | 27.45 | -0.04051 | 3.651 | 17.42 |

There is little to choose between the two versions of the logistic model (6), whereas for the exponential model (7), the fit using lnlsq(0) is much better (a deviance difference of 7.28). The reciprocal transformation has introduced heteroskedasticity into $y_{i}^{\prime}$, which is countered by the proportional errors property of the lognormal distribution implicit in lnlsq(0). The deviances are not comparable between the logistic and exponential models because the change of scale has not been allowed for, although in principle it could be.

## Other uses

Even if you are fitting linear regression models, you may find that nl can save you some typing. Because you specify the parameters of your model explicitly, you can impose constraints on them directly.
$>$ Example 2
In example 2 of $[R]$ cnsreg, we showed how to fit the model

$$
\begin{aligned}
\operatorname{mpg}= & \beta_{0}+\beta_{1} \text { price }+\beta_{2} \text { weight }+\beta_{3} \text { displ }+\beta_{4} \text { gear_ratio }+\beta_{5} \text { foreign }+ \\
& \beta_{6} \text { length }+u
\end{aligned}
$$

subject to the constraints

$$
\begin{aligned}
& \beta_{1}=\beta_{2}=\beta_{3}=\beta_{6} \\
& \beta_{4}=-\beta_{5}=\beta_{0} / 20
\end{aligned}
$$

An alternative way is to use nl :

```
. use https://www.stata-press.com/data/r18/auto, clear
(1978 automobile data)
. nl (mpg = {b0} + {b1}*price + {b1}*weight + {b1}*displ +
> {b0}/20*gear_ratio - {b0}/20*foreign + {b1}*length)
Iteration 0: Residual SS = 1578.522
Iteration 1: Residual SS = 1578.522
```

| Source | SS | df | MS |  |  |  |
| ---: | ---: | ---: | ---: | :--- | :--- | :--- |
| Model | 34429.478 | 2 | 17214.7389 |  | Number of obs | $=$ |


| mpg | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| /b0 | 26.52229 | 1.375178 | 19.29 | 0.000 | 23.78092 | 29.26365 |
| /b1 | -.000923 | .0001534 | -6.02 | 0.000 | -.0012288 | -.0006172 |

The point estimates and standard errors for $\beta_{0}$ and $\beta_{1}$ are identical to those reported in example 2 of $[\mathrm{R}]$ cnsreg. To get the estimate for $\beta_{4}$, we can use nlcom:
. nlcom _b[/b0]/20
_nl_1: _b[/b0]/20

| mpg | Coefficient | Std. err. | $z$ | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| _nl_1 | 1.326114 | .0687589 | 19.29 | 0.000 | 1.191349 | 1.460879 |

The advantage to using nl is that we do not need to use the constraint command six times.
nl is also a useful tool when doing exploratory data analysis. For example, you may want to run a regression of $y$ on a function of $x$, though you have not decided whether to use $\operatorname{sqrt}(\mathrm{x})$ or $\ln (\mathrm{x})$. You can use nl to run both regressions without having first to generate two new variables:

```
.nl (y = {b0} + {b1}* ln(x))
. nl (y = {b0} + {b1}*sqrt(x))
```

Poi (2008) shows the advantages of using nl when marginal effects of transformed variables are desired as well.

## Weights

Weights are specified in the usual way-analytic and frequency weights as well as iweights are supported; see [U] 20.24 Weighted estimation. Use of analytic weights implies that the $y_{i}$ have different variances. Therefore, model (3) may be rewritten as

$$
\begin{equation*}
y_{i}=f\left(\mathbf{x}_{i}, \boldsymbol{\beta}\right)+u_{i}, \quad u_{i} \sim N\left(0, \sigma^{2} / w_{i}\right) \tag{3a}
\end{equation*}
$$

where $w_{i}$ are (positive) weights, assumed to be known and normalized such that their sum equals the number of observations. The residual deviance for ( $3 a$ ) is

$$
\begin{equation*}
D(\widehat{\boldsymbol{\beta}})=\left\{1+\ln \left(2 \pi \widehat{\sigma}^{2}\right)\right\} n-\sum \ln \left(w_{i}\right) \tag{5a}
\end{equation*}
$$

[compare with (5)], where

$$
n \widehat{\sigma}^{2}=\operatorname{RSS}=\sum w_{i}\left\{y_{i}-f\left(\mathbf{x}_{i}, \widehat{\boldsymbol{\beta}}\right)\right\}^{2}
$$

Defining and fitting a model equivalent to (4) when weights have been specified as in (3a) is not straightforward and has not been attempted. Thus, deviances using and not using the lnlsq() option may not be strictly comparable when analytic weights (other than 0 and 1) are used.

You do not need to modify your substitutable expression in any way to use weights. If, however, you write a substitutable expression program, then you should account for weights when obtaining initial values. When $n l$ calls your program, it passes whatever weight expression (if any) was specified by the user. Here is an outline of a substitutable expression program that accepts weights:

```
program nl name, rclass
    version 18.0 // (or version 18.5 for StataNow)
    syntax varlist [aw fw iw] if
    // Obtain initial values allowing weights
    // Use the syntax ['weight''exp']. For example,
    summarize varname ['weight''exp'] 'if'
    regress depvar varlist ['weight''exp'] 'if'
    // Return substitutable expression
    return local eq "substitutable expression"
    return local title "description of estimator"
end
```

For details on how the syntax command processes weight expressions, see [P] syntax.

## Potential errors

nl is reasonably robust to the inability of your nonlinear function to be evaluated at some parameter values. nl does assume that your function can be evaluated at the initial values of the parameters. If your function cannot be evaluated at the initial values, an error message is issued with return code 480. Recall that if you do not specify an initial value for a parameter, then nl initializes it to zero. Many nonlinear functions cannot be evaluated when some parameters are zero, so in those cases specifying alternative initial values is crucial.

Thereafter, as nl changes the parameter values, it monitors your function for unexpected missing values. If these are detected, nl backs up. That is, nl finds a point between the previous, known-to-be-good parameter vector and the new, known-to-be-bad vector at which the function can be evaluated and continues its iterations from that point.
nl requires that once a parameter vector is found where the predictions can be calculated, small changes to the parameter vector be made to calculate numeric derivatives. If a boundary is encountered at this point, an error message is issued with return code 481.

When specifying lnlsq(), an attempt to take logarithms of $y_{i}-k$ when $y_{i} \leq k$ results in an error message with return code 482.

If iterate() iterations are performed and estimates still have not converged, results are presented with a warning, and the return code is set to 430 .

If you use the programmed substitutable expression version of $n l$ with a function evaluator program, or vice versa, Stata issues an error message. Verify that you are using the syntax appropriate for the program you have.

## General comments on fitting nonlinear models

Achieving convergence is often problematic. For example, a unique minimum of the sum-ofsquares function may not exist. Much literature exists on different algorithms that have been used, on strategies for obtaining good initial parameter values, and on tricks for parameterizing the model to make its behavior as linear-like as possible. Selected references are Kennedy and Gentle (1980, chap. 10) for computational matters and Ross (1990) and Ratkowsky (1983) for all three aspects. Ratkowsky's book is particularly clear and approachable, with useful discussion on the meaning and practical implications of intrinsic and parameter-effects nonlinearity. An excellent text on nonlinear estimation is Gallant (1987). Also see Davidson and MacKinnon (1993 and 2004).

To enhance the success of $n l$, pay attention to the form of the model fit, along the lines of Ratkowsky and Ross. For example, Ratkowsky (1983, 49-59) analyzes three possible three-parameter yield-density models for plant growth:

$$
E\left(y_{i}\right)=\left\{\begin{array}{l}
\left(\alpha+\beta x_{i}\right)^{-1 / \theta} \\
\left(\alpha+\beta x_{i}+\gamma x_{i}^{2}\right)^{-1} \\
\left(\alpha+\beta x_{i}^{\phi}\right)^{-1}
\end{array}\right.
$$

All three models give similar fits. However, he shows that the second formulation is dramatically more linear-like than the other two and therefore has better convergence properties. In addition, the parameter estimates are virtually unbiased and normally distributed, and the asymptotic approximation to the standard errors, correlations, and confidence intervals is much more accurate than for the other models. Even within a given model, the way the parameters are expressed (for example, $\phi^{x_{i}}$ or $e^{\theta x_{i}}$ ) affects the degree of linearity and convergence behavior.

## Function evaluator programs

Occasionally, a nonlinear function may be so complex that writing a substitutable expression for it is impractical. For example, there could be many parameters in the model. Alternatively, if you are implementing a two-step estimator, writing a substitutable expression may be altogether impossible. Function evaluator programs can be used in these situations.
nl will pass to your function evaluator program a list of variables, a weight expression, a variable marking the estimation sample, and a vector of parameters. Your program is to replace the dependent variable, which is the first variable in the variables list, with the values of the nonlinear function evaluated at those parameters. As with substitutable expression programs, the first two letters of the name must be nl.

To focus on the mechanics of the function evaluator program, again let's compare the CES production function to the previous examples. The function evaluator program is

```
program nlces2
    version 18.0 // (or version 18.5 for StataNow)
    syntax varlist(min=3 max=3) if, at(name)
    local logout : word 1 of 'varlist'
    local capital : word 2 of 'varlist'
    local labor : word 3 of 'varlist'
    // Retrieve parameters out of at matrix
    tempname b0 rho delta
    scalar 'b0' = 'at'[1, 1]
    scalar 'rho' = 'at'[1, 2]
    scalar 'delta' = 'at'[1, 3]
    tempvar kterm lterm
    generate double 'kterm' = 'delta'*'capital'^(-1*'rho') 'if'
    generate double 'lterm' = (1-'delta')*'labor'`(-1*'rho') 'if'
    // Fill in dependent variable
    replace 'logout' = 'b0' - 1/'rho'*ln('kterm' + 'lterm') 'if'
end
```

Unlike the previous nlces program, this one is not declared to be r-class. The syntax statement again accepts three variables: one for log output, one for capital, and one for labor. An if exp is again required because $n l$ will pass a binary variable marking the estimation sample. All function evaluator programs must accept an option named at () that takes a name as an argument-that is how nl passes the parameter vector to your program.

The next part of the program retrieves the output, labor, and capital variables from the variables list. It then breaks up the temporary matrix at and retrieves the parameters b0, rho, and delta. Pay careful attention to the order in which the parameters refer to the columns of the at matrix because that will affect the syntax you use with nl . The temporary names you use inside this program are immaterial, however.

The rest of the program computes the nonlinear function, using some temporary variables to hold intermediate results. The final line of the program then replaces the dependent variable with the values of the function. Notice the use of 'if' to restrict attention to the estimation sample. nl makes a copy of your dependent variable so that when the command is finished your data are left unchanged.

To use the program and fit your model, you type

```
. use https://www.stata-press.com/data/r18/production, clear
. nl ces2 @ lnoutput capital labor, parameters(b0 rho delta)
> initial(b0 0 rho 1 delta 0.5)
```

The output is again identical to that shown in example 1. The order in which the parameters were specified in the parameters () option is the same in which they are retrieved from the at matrix in the program. To initialize them, you simply list the parameter name, a space, the initial value, and so on.

If you use the nparameters() option instead of the parameters () option, the parameters are named $\mathrm{b} 1, \mathrm{~b} 2, \ldots, \mathrm{~b} k$, where $k$ is the number of parameters. Thus, you could have typed

```
. nl ces2 @ lnoutput capital labor, nparameters(3) initial(b1 0 b2 1 b3 0.5)
```

With that syntax, the parameters called b0, rho, and delta in the program will be labeled b1, b2, and b3, respectively. In programming situations or if there are many parameters, instead of listing the parameter names and initial values in the initial() option, you may find it more convenient to pass a column vector. In those cases, you could type

```
. matrix myvals = (0, 1, 0.5)
. nl ces2 @ lnoutput capital labor, nparameters(3) initial(myvals)
```

In summary, a function evaluator program receives a list of variables, the first of which is the dependent variable that you are to replace with the values of your nonlinear function. Additionally, it must accept an if exp, as well as an option named at that will contain the vector of parameters at which nl wants the function evaluated. You are then free to do whatever is necessary to evaluate your function and replace the dependent variable.

If you wish to use weights, your function evaluator program's syntax statement must accept them. If your program consists only of, for example, generate statements, you need not do anything with the weights passed to your program. However, if in calculating the nonlinear function you use commands such as summarize or regress, then you will want to use the weights with those commands.

As with substitutable expression programs, nl will pass to it any options specified that nl does not accept, providing you with a way to pass more information to your function.

## - Technical note

Before version 9 of Stata, the nl command used a different syntax, which required you to write an nlfcn program, and it did not have a syntax for interactive use other than the seven functions that were built-in. The old syntax of nl still works, and you can still use those nlfcn programs. If nl does not see a colon, an at sign, or a set of parentheses surrounding the equation in your command, it assumes that the old syntax is being used.

The current version of nl uses scalars and matrices to store intermediate calculations instead of local and global macros as the old version did, so the current version produces more accurate results. In practice, however, any discrepancies are likely to be small.

## Stored results

nl stores the following in e() :
Scalars
e(N)
e(k)
e(k_eq_model)
e(df_m)
e(df_r)
$e\left(d f \_t\right)$
e(mss)
e(rss)
e(tss)
e(mms)
e(msr)
e(11)
e(r2)
e(r2_a)
e(rmse)
e(dev)
e(N_clust)
e(lnlsq)
e(log_t)
e(gm_2)
e(cj)
e(delta)
e(rank)
e(ic)
e(converged)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(wtype)
e(wexp)
e(title)
e(title_2)
e(clustvar)
e(hac_kernel)
e(hac_lag)
e(vce)
e(vcetype)
e(type)
e(sexp)
e(params)
e(funcprog)
e(rhs)
e(properties)
e(predict)
e(marginsok)
e(marginsnotok)
Matrices
e(b)
e(init)
e(V)
e(V_modelbased)
number of observations
number of parameters
number of equations in overall model test; always 0
model degrees of freedom
residual degrees of freedom
total degrees of freedom
model sum of squares
residual sum of squares
total sum of squares
model mean square
residual mean square
log likelihood assuming i.i.d. normal errors
$R^{2}$
adjusted $R^{2}$
root mean squared error
residual deviance
number of clusters
value of $\operatorname{lnl} s q$ if specified
1 if lnlsq specified, 0 otherwise
square of geometric mean of $(y-k)$ if $\ln l$ sq, 1 otherwise
position of constant in $e(b)$ or 0 if no constant
relative change used to compute derivatives
rank of $e(V)$
number of iterations
1 if converged, 0 otherwise
nl
command as typed
name of dependent variable
weight type
weight expression
title in estimation output
secondary title in estimation output
name of cluster variable
HAC kernel
HAC lag
vcetype specified in vce()
title used to label Std. err.
1 = interactively entered expression
$2=$ substitutable expression program
3 = function evaluator program
substitutable expression
names of parameters
function evaluator program
contents of variables()
b V
program used to implement predict
predictions allowed by margins
predictions disallowed by margins
coefficient vector
initial values vector
variance-covariance matrix of the estimators
model-based variance

Functions
e(sample) marks estimation sample

In addition to the above, the following is stored in r() :
Matrices
$r$ (table) matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

The derivation here is based on Davidson and MacKinnon (2004, chap. 6). Let $\boldsymbol{\beta}$ denote the $k \times 1$ vector of parameters, and write the regression function using matrix notation as $\mathbf{y}=\mathbf{f}(\mathbf{x}, \boldsymbol{\beta})+\mathbf{u}$ so that the objective function can be written as

$$
\operatorname{SSR}(\boldsymbol{\beta})=\{\mathbf{y}-\mathbf{f}(\mathbf{x}, \boldsymbol{\beta})\}^{\prime} \mathbf{D}\{\mathbf{y}-\mathbf{f}(\mathbf{x}, \boldsymbol{\beta})\}
$$

The $\mathbf{D}$ matrix contains the weights and is defined in $[R]$ regress; if no weights are specified, then $\mathbf{D}$ is the $N \times N$ identity matrix. Taking a second-order Taylor series expansion centered at $\boldsymbol{\beta}_{0}$ yields

$$
\begin{equation*}
\operatorname{SSR}(\boldsymbol{\beta}) \approx \operatorname{sSR}\left(\boldsymbol{\beta}_{0}\right)+\mathbf{g}^{\prime}\left(\boldsymbol{\beta}_{0}\right)\left(\boldsymbol{\beta}-\boldsymbol{\beta}_{0}\right)+\frac{1}{2}\left(\boldsymbol{\beta}-\boldsymbol{\beta}_{0}\right)^{\prime} \mathbf{H}\left(\boldsymbol{\beta}_{0}\right)\left(\boldsymbol{\beta}-\boldsymbol{\beta}_{0}\right) \tag{8}
\end{equation*}
$$

where $\mathbf{g}\left(\boldsymbol{\beta}_{0}\right)$ denotes the $k \times 1$ gradient of $\operatorname{SSR}(\boldsymbol{\beta})$ evaluated at $\boldsymbol{\beta}_{0}$ and $\mathbf{H}\left(\boldsymbol{\beta}_{0}\right)$ denotes the $k \times k$ Hessian of $\operatorname{SSR}(\boldsymbol{\beta})$ evaluated at $\boldsymbol{\beta}_{0}$. Letting $\mathbf{X}$ denote the $N \times k$ matrix of derivatives of $\mathbf{f}(\mathbf{x}, \boldsymbol{\beta})$ with respect to $\boldsymbol{\beta}$, the gradient $\mathbf{g}(\boldsymbol{\beta})$ is

$$
\begin{equation*}
\mathbf{g}(\boldsymbol{\beta})=-2 \mathbf{X}^{\prime} \mathbf{D u} \tag{9}
\end{equation*}
$$

$\mathbf{X}$ and $\mathbf{u}$ are obviously functions of $\boldsymbol{\beta}$, though for notational simplicity that dependence is not shown explicitly. The $(m, n)$ element of the Hessian can be written as

$$
\begin{equation*}
H_{m n}(\boldsymbol{\beta})=-2 \sum_{i=1}^{i=N} d_{i i}\left[\frac{\partial^{2} f_{i}}{\partial \beta_{m} \partial \beta_{n}} u_{i}-X_{i m} X_{i n}\right] \tag{10}
\end{equation*}
$$

where $d_{i i}$ is the $i$ th diagonal element of $\mathbf{D}$. As discussed in Davidson and MacKinnon (2004, chap. 6), the first term inside the brackets of (10) has expectation zero, so the Hessian can be approximated as

$$
\begin{equation*}
\mathbf{H}(\boldsymbol{\beta})=2 \mathbf{X}^{\prime} \mathbf{D} \mathbf{X} \tag{11}
\end{equation*}
$$

Differentiating the Taylor series expansion of $\operatorname{SSR}(\boldsymbol{\beta})$ shown in (8) yields the first-order condition for a minimum

$$
\mathbf{g}\left(\boldsymbol{\beta}_{0}\right)+\mathbf{H}\left(\boldsymbol{\beta}_{0}\right)\left(\boldsymbol{\beta}-\boldsymbol{\beta}_{0}\right)=\mathbf{0}
$$

which suggests the iterative procedure

$$
\begin{equation*}
\boldsymbol{\beta}_{j+1}=\boldsymbol{\beta}_{j}-\alpha \mathbf{H}^{-1}\left(\boldsymbol{\beta}_{j}\right) \mathbf{g}\left(\boldsymbol{\beta}_{j}\right) \tag{12}
\end{equation*}
$$

where $\alpha$ is a "step size" parameter chosen at each iteration to improve convergence. Using (9) and (11), we can write (12) as

$$
\begin{equation*}
\boldsymbol{\beta}_{j+1}=\boldsymbol{\beta}_{j}+\alpha\left(\mathbf{X}^{\prime} \mathbf{D} \mathbf{X}\right)^{-1} \mathbf{X}^{\prime} \mathbf{D} \mathbf{u} \tag{13}
\end{equation*}
$$

where $\mathbf{X}$ and $\mathbf{u}$ are evaluated at $\boldsymbol{\beta}_{j}$. Apart from the scalar $\alpha$, the second term on the right-hand side of (13) can be computed via a (weighted) regression of the columns of $\mathbf{X}$ on the errors. nl computes the derivatives numerically and then calls regress. At each iteration, $\alpha$ is set to one, and a candidate value $\boldsymbol{\beta}_{j+1}^{*}$ is computed by (13). If $\operatorname{SSR}\left(\boldsymbol{\beta}_{j+1}^{*}\right)<\operatorname{SSR}\left(\boldsymbol{\beta}_{j}\right)$, then $\boldsymbol{\beta}_{j+1}=\boldsymbol{\beta}_{j+1}^{*}$ and the iteration is complete. Otherwise, $\alpha$ is halved, a new $\boldsymbol{\beta}_{j+1}^{*}$ is calculated, and the process is repeated. Convergence is declared when $\alpha\left|\beta_{j+1, m}\right| \leq \epsilon\left(\left|\beta_{j m}\right|+\tau\right)$ for all $m=1, \ldots, k$. nl uses $\tau=10^{-3}$ and, by default, $\epsilon=10^{-5}$, though you can specify an alternative value of $\epsilon$ with the eps() option.

As derived, for example, in Davidson and MacKinnon (2004, chap. 6), an expedient way to obtain the covariance matrix is to compute $\mathbf{u}$ and the columns of $\mathbf{X}$ at the final estimate $\widehat{\boldsymbol{\beta}}$ and then regress that $\mathbf{u}$ on $\mathbf{X}$. The covariance matrix of the estimated parameters of that regression serves as an estimate of $\operatorname{Var}(\widehat{\boldsymbol{\beta}})$. If that regression employs a robust covariance matrix estimator, then the covariance matrix for the parameters of the nonlinear regression will also be robust.

All other statistics are calculated analogously to those in linear regression, except that the nonlinear function $f\left(\mathbf{x}_{i}, \boldsymbol{\beta}\right)$ plays the role of the linear function $\mathbf{x}_{i}^{\prime} \boldsymbol{\beta}$. See $[\mathrm{R}]$ regress.

This command supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

## Acknowledgments

The original version of nl was written by Patrick Royston of the MRC Clinical Trials Unit, London, and coauthor of the Stata Press book Flexible Parametric Survival Analysis Using Stata: Beyond the Cox Model. Francesco Danuso's menu-driven nonlinear regression program (1991) provided the inspiration.

## References

Atkinson, A. C. 1985. Plots, Transformations, and Regression: An Introduction to Graphical Methods of Diagnostic Regression Analysis. Oxford: Oxford University Press.

Canette, I. 2011. A tip to debug your nl/nlsur function evaluator program. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2011/12/05/a-tip-to-debug-your-nlnlsur-function-evaluator-program/.
Danuso, F. 1991. sg1: Nonlinear regression command. Stata Technical Bulletin 1: 17-19. Reprinted in Stata Technical Bulletin Reprints, vol. 1, pp. 96-98. College Station, TX: Stata Press.
Davidson, R., and J. G. MacKinnon. 1993. Estimation and Inference in Econometrics. New York: Oxford University Press.
——. 2004. Econometric Theory and Methods. New York: Oxford University Press.
Gallant, A. R. 1987. Nonlinear Statistical Models. New York: Wiley.
Goldstein, R. 1992. srd7: Adjusted summary statistics for logarithmic regressions. Stata Technical Bulletin 5: 17-21. Reprinted in Stata Technical Bulletin Reprints, vol. 1, pp. 178-183. College Station, TX: Stata Press.
Kennedy, W. J., Jr., and J. E. Gentle. 1980. Statistical Computing. New York: Dekker.
Poi, B. P. 2008. Stata tip 58: nl is not just for nonlinear models. Stata Journal 8: 139-141.
Ratkowsky, D. A. 1983. Nonlinear Regression Modeling: A Unified Practical Approach. New York: Dekker.
Ross, G. J. S. 1987. MLP User Manual, Release 3.08. Oxford: Numerical Algorithms Group.
__. 1990. Nonlinear Estimation. New York: Springer.

## Also see

[R] nl postestimation - Postestimation tools for nl
[R] gmm - Generalized method of moments estimation
[R] ml - Maximum likelihood estimation
[R] mlexp - Maximum likelihood estimation of user-specified expressions
[R] nlcom - Nonlinear combinations of parameters
[R] nlsur - Estimation of nonlinear systems of equations
$[R]$ regress - Linear regression
[ME] menl - Nonlinear mixed-effects regression
[SVY] svy estimation - Estimation commands for survey data
[U] 20 Estimation and postestimation commands

## Title

nl postestimation - Postestimation tools for nl
Postestimation commands predict margins Remarks and examples

## Postestimation commands

The following postestimation commands are available after nl:

| Command | Description |
| :--- | :--- |
| estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian in- <br> formation criteria (AIC, CAIC, AICc, and BIC) |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estat (svy) | postestimation statistics for survey data <br> cataloging estimation results |
| estimates | table of estimation results |
| etable | dynamic forecasts and simulations |
| * hausman | Hausman's specification test |
| lincom | point estimates, standard errors, testing, and inference for linear combinations of |
| *lrtest | coefficients |
| $\dagger$ margins | marginal means, predictive margins, marginal effects, and average marginal effects <br> marginsplot <br> graph the results from margins (profile plots, interaction plots, etc.) |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations <br> of coefficients |
| predict | fitted values, residuals, etc. <br> predictnl |
| point estimates, standard errors, testing, and inference for generalized predictions |  |

[^38]
## predict

## Description for predict

predict creates a new variable containing predictions such as fitted values, residuals, probabilities, and expected values.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

```
predict [type] newvar [if] [in] [, statistic]
predict [type] stub* [if] [in], scores
```

where $k$ is the number of parameters in the model.

## statistic Description

Main

| $\underline{\text { yhat }}$ | fitted values; the default |
| :--- | :--- |
| residuals | residuals |
| $\operatorname{pr}(a, b)$ | $\operatorname{Pr}\left(y_{j} \mid a<y_{j}<b\right)$ |
| $\mathrm{e}(a, b)$ | $E\left(y_{j} \mid a<y_{j}<b\right)$ |
| ystar $(a, b)$ | $E\left(y_{j}^{*}\right), y_{j}^{*}=\max \left\{a, \min \left(y_{j}, b\right)\right\}$ |

These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

## Options for predict

$\qquad$ Main
yhat, the default, calculates the fitted values.
residuals calculates the residuals.
$\operatorname{pr}(a, b)$ calculates $\operatorname{Pr}\left(a<\mathbf{x}_{j} \mathbf{b}+u_{j}<b\right)$, the probability that $y_{j} \mid \mathbf{x}_{j}$ would be observed in the interval $(a, b)$.
$a$ and $b$ may be specified as numbers or variable names; $l b$ and $u b$ are variable names;
$\operatorname{pr}(20,30)$ calculates $\operatorname{Pr}\left(20<\mathbf{x}_{j} \mathbf{b}+u_{j}<30\right)$;
$\operatorname{pr}(l b, u b)$ calculates $\operatorname{Pr}\left(l b<\mathbf{x}_{j} \mathbf{b}+u_{j}<u b\right)$; and
$\operatorname{pr}(20, u b)$ calculates $\operatorname{Pr}\left(20<\mathbf{x}_{j} \mathbf{b}+u_{j}<u b\right)$.
$a$ missing $(a \geq$.$) means -\infty ; \operatorname{pr}(., 30)$ calculates $\operatorname{Pr}\left(-\infty<\mathbf{x}_{j} \mathbf{b}+u_{j}<30\right)$;
$\operatorname{pr}(l b, 30)$ calculates $\operatorname{Pr}\left(-\infty<\mathbf{x}_{j} \mathbf{b}+u_{j}<30\right)$ in observations for which $l b \geq$.
and calculates $\operatorname{Pr}\left(l b<\mathbf{x}_{j} \mathbf{b}+u_{j}<30\right)$ elsewhere.
$b$ missing $(b \geq$.$) means +\infty$; $\operatorname{pr}(20,$.$) calculates \operatorname{Pr}\left(+\infty>\mathbf{x}_{j} \mathbf{b}+u_{j}>20\right)$; $\operatorname{pr}(20, u b)$ calculates $\operatorname{Pr}\left(+\infty>\mathbf{x}_{j} \mathbf{b}+u_{j}>20\right)$ in observations for which $u b \geq$. and calculates $\operatorname{Pr}\left(20<\mathbf{x}_{j} \mathbf{b}+u_{j}<u b\right)$ elsewhere.
$\mathrm{e}(a, b)$ calculates $E\left(\mathbf{x}_{j} \mathbf{b}+u_{j} \mid a<\mathbf{x}_{j} \mathbf{b}+u_{j}<b\right)$, the expected value of $y_{j} \mid \mathbf{x}_{j}$ conditional on $y_{j} \mid \mathbf{x}_{j}$ being in the interval $(a, b)$, meaning that $y_{j} \mid \mathbf{x}_{j}$ is truncated. $a$ and $b$ are specified as they are for pr() .
ystar $(a, b)$ calculates $E\left(y_{j}^{*}\right)$, where $y_{j}^{*}=a$ if $\mathbf{x}_{j} \mathbf{b}+u_{j} \leq a, y_{j}^{*}=b$ if $\mathbf{x}_{j} \mathbf{b}+u_{j} \geq b$, and $y_{j}^{*}=\mathbf{x}_{j} \mathbf{b}+u_{j}$ otherwise, meaning that $y_{j}^{*}$ is censored. $a$ and $b$ are specified as they are for pr() . scores calculates the scores. The $j$ th new variable created will contain the score for the $j$ th parameter in $e(b)$.

## margins

## Description for margins

margins estimates margins of response for fitted values.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [options]
```

statistic Description

| $\underline{\text { yhat }}$ | fitted values; the default |
| :--- | :--- |
| $\operatorname{pr}(a, b)$ | not allowed with margins |
| $\mathrm{e}(a, b)$ | not allowed with margins |
| ystar $(a, b)$ | not allowed with margins |
| $\underline{\underline{r}}$ esiduals | not allowed with margins |

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

## Remarks and examples

## $>$ Example 1

Obtaining predictions after fitting a nonlinear regression model with nl is no more difficult than obtaining predictions after fitting a linear regression model with regress. Here we fit a model of mpg on weight, allowing for a nonlinear relationship:

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. nl (mpg = {b0} + {b1}*weight`{gamma=-.5}), variables(weight) nolog
```

| Source | SS | df | MS |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Model | 1646.4376 | 2 | 823.218806 | squared | 0.6738 |
| Residual | 797.02185 | 71 | 11.2256598 | Adj R-squared | 0.6646 |
|  |  |  |  | Root MSE | 3.350472 |
| Total | 2443.4595 | 73 | 33.4720474 | Res. dev. | 385.8874 |


| mpg | Coefficient | Std. err. | t | $\mathrm{P}>\mid \mathrm{tI}$ | [95\% conf. interval] |  |
| ---: | ---: | :--- | :---: | :---: | :---: | :---: |
| /b0 | -18.17583 | 60.61762 | -0.30 | 0.765 | -139.0439 | 102.6923 |
| /b1 | 1377.267 | 5292.379 | 0.26 | 0.795 | -9175.436 | 11929.97 |
| /gamma | -.4460916 | .6763641 | -0.66 | 0.512 | -1.794723 | .9025401 |

```
Note: Parameter b0 is used as a constant term during estimation.
```

Now, we obtain the predicted values of mpg and plot them in a graph along with the observed values:

```
. predict mpghat
(option yhat assumed; fitted values)
. scatter mpg weight || line mpghat weight, sort
```



Suppose we wanted to know how sensitive mpg is to changes in weight for cars that weigh 3,000 pounds. We can use margins to find out:

```
. margins, eyex(weight) at(weight = 3000)
Conditional marginal effects Number of obs = 74
Model VCE: GNR
Expression: Fitted values, predict()
ey/ex wrt: weight
At: weight = 3000
```

|  | Delta-method |  |  |  |  |  |
| :--- | ---: | ---: | :---: | ---: | ---: | ---: |
|  | ey/ex | std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| weight | -.8408119 | .0804339 | -10.45 | 0.000 | -.9984594 | -.6831644 |

With the eyex () option, margins reports elasticities. These results show that if we increase weight by $1 \%$, then mpg decreases by about $0.84 \%$.

## - Technical note

Observant readers will notice that margins issued a warning message stating that it could not perform its usual check for estimable functions. In the case of nl , as long as you do not specify the predict() option of margins or specify the default predict (yhat), you can safely ignore that message. The predicted values that nl produces are suitable for use with margins. However, if you specify any predict () options other than yhat, then the output from margins after using nl will not be correct.

## Also see

[R] nl - Nonlinear least-squares estimation
[U] 20 Estimation and postestimation commands

## Title

nlcom - Nonlinear combinations of parameters

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

nlcom computes point estimates, standard errors, test statistics, significance levels, and confidence intervals for (possibly) nonlinear combinations of parameter estimates after any Stata estimation command, including survey estimation. Results are displayed in the usual table format used for displaying estimation results. Standard error calculations are based on the "delta method", an approximation appropriate in large samples.

## Quick start

Estimate the ratio of the coefficient of $x 2$ to the coefficient of x1
nlcom _b[x2]/_b[x1]
Also estimate the ratio of the coefficient of $x 3$ to coefficient of $x 1$
nlcom (_b[x2]/_b[x1]) (_b[x3]/_b[x1])
Add labels to the ratios
nlcom (r21:_b[x2]/_b[x1]) (r31:_b[x3]/_b[x1])
Same as above, but post estimates and use the test command to test that both ratios are equal to 1 nlcom (r21:_b[x2]/_b[x1]) (r31:_b[x3]/_b[x1]), post test $(r 21=1)(r 31=1)$

Estimate the ratio of the coefficients of factor indicators 2.a and 3.a
nlcom _b[2.a]/_b[3.a]
Estimate the ratio of the coefficients of x 1 in the equations for y 1 and y 2 in a multiequation model nlcom _b[y1:x1]/_b[y2:x1]

## Menu

Statistics $>$ Postestimation

## Syntax

Nonlinear combination of parameters-one expression
nlcom [name: $] \exp [$, options $]$

Nonlinear combinations of parameters-more than one expression
nlcom ([name: ]exp) [([name: ]exp) ...] [, options $]$
options
Description

| level(\#) | set confidence level; default is level(95) |
| :--- | :--- |
| maximum number of iterations |  |
| iterate(\#) | control column formats and line width |
| display_options | post estimation results |
| post | report exponentiated estimates; column title is "exp(b)" <br> eform <br> eform (string) |
| report exponentiated estimates; column title is "string" |  |
| noheader (\#) suppress output header <br> use $t$ distribution with \# degrees of freedom for computing $p$-values <br> and confidence intervals |  |

collect is allowed; see [U] 11.1.10 Prefix commands.
Only one of eform or eform() is allowed.
noheader and $\mathrm{df}(\#)$ do not appear in the dialog box.
The second syntax means that if more than one expression is specified, each must be surrounded by parentheses. The optional name is any valid Stata name and labels the transformations.
$\exp$ is a possibly nonlinear expression containing

$$
\begin{aligned}
& -\mathrm{b}[\text { coef }] \\
& \mathrm{-b}[\text { eqno: coef }] \\
& {[\text { eqno }] \text { coef }} \\
& {[\text { eqno }] \text { b }[\text { coef }]}
\end{aligned}
$$

eqno is

## \#\#

name
coef identifies a coefficient in the model. coef is typically a variable name, a level indicator, an interaction indicator, or an interaction involving continuous variables. Level indicators identify one level of a factor variable and interaction indicators identify one combination of levels of an interaction; see [U] 11.4.3 Factor variables. coef may contain time-series operators; see [U] 11.4.4 Time-series varlists.

Distinguish between [], which are to be typed, and [ ], which indicate optional arguments.

## Options

level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals.
iterate(\#) specifies the maximum number of iterations used to find the optimal step size in calculating numerical derivatives of the transformation(s) with respect to the original parameters. By default, the maximum number of iterations is 100 , but convergence is usually achieved after only a few iterations. You should rarely have to use this option.
display_options: cformat ( $\%$ fmt $)$, pformat ( $\% f m t$ ), sformat ( $\% f m t$ ), and nolstretch; see [R] Estimation options.
post causes nlcom to behave like a Stata estimation (eclass) command. When post is specified, nlcom will post the vector of transformed parameters and its estimated variance-covariance matrix to $e()$. This option, in essence, makes the transformation permanent. Thus you could, after posting, treat the transformed estimation results in the same way as you would treat results from other Stata estimation commands. For example, after posting, you could redisplay the results by typing nlcom without any arguments, or use test to perform simultaneous tests of hypotheses on linear combinations of the transformed parameters; see [R] test.
Specifying post clears out the previous estimation results, which can be recovered only by refitting the original model or by storing the estimation results before running nlcom and then restoring them; see $[R]$ estimates store.
eform and eform(string) specify that the table of estimates be displayed in exponentiated form: for each estimate, $e^{b}$ rather than $b$ is displayed, where $b$ is the nonlinear combination of parameters. eform labels the exponentiated results $\exp (\mathrm{b})$, whereas eform (string) uses string to label them. Standard errors and confidence intervals are also transformed; see [R] eform_option.

The following options are available with nlcom but are not shown in the dialog box:
noheader suppresses the output header.
$\mathrm{df}(\#)$ specifies that the $t$ distribution with \# degrees of freedom be used for computing $p$-values and confidence intervals.

## Remarks and examples

Remarks are presented under the following headings:
Introduction
Basics
Using the post option
Reparameterizing ML estimates for univariate data
nlcom versus eform

## Introduction

nlcom and predictnl both use the delta method. They take nonlinear transformations of the estimated parameter vector from some fitted model and apply the delta method to calculate the variance, standard error, Wald test statistic, etc., of the transformations. nlcom is designed for functions of the parameters, and predictnl is designed for functions of the parameters and of the data, that is, for predictions.
nlcom generalizes lincom (see $[\mathrm{R}]$ lincom) in two ways. First, nlcom allows the transformations to be nonlinear. Second, nlcom can be used to simultaneously estimate many transformations (whether linear or nonlinear) and to obtain the estimated variance-covariance matrix of these transformations.

## Basics

In $[\mathrm{R}]$ lincom, the following regression was performed:
. use https://www.stata-press.com/data/r18/regress
. regress y x1 x2 x3

| Source | SS | df | MS |
| ---: | ---: | ---: | ---: |
| Model | 3259.3561 | 3 | 1086.45203 |
| Residual | 1627.56282 | 144 | 11.3025196 |
| Total | 4886.91892 | 147 | 33.2443464 |


| Number of obs | $=$ | 148 |
| :--- | :--- | ---: |
| $\mathrm{~F}(3,144)$ | $=$ | 96.12 |
| Prob $>\mathrm{F}$ | $=$ | 0.0000 |
| R-squared | $=$ | 0.6670 |
| Adj R-squared | $=$ | 0.6600 |
| Root MSE | $=$ | 3.3619 |


| y | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| x1 | 1.457113 | 1.07461 | 1.36 | 0.177 | -.666934 | 3.581161 |
| x 2 | 2.221682 | .8610358 | 2.58 | 0.011 | .5197797 | 3.923583 |
| x 3 | -.006139 | .0005543 | -11.08 | 0.000 | -.0072345 | -.0050435 |
| _cons | 36.10135 | 4.382693 | 8.24 | 0.000 | 27.43863 | 44.76407 |

Then, lincom was used to estimate the difference between the coefficients of x 1 and x 2 :

- lincom _b[x2] - _b[x1]
(1) $-\mathrm{x} 1+\mathrm{x} 2=0$

| $y$ | Coefficient | Std. err. | $t$ | P>\|t| | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | :---: | :---: |
| $(1)$ | .7645682 | .9950282 | 0.77 | 0.444 | -1.20218 | 2.731316 |

It was noted, however, that nonlinear expressions are not allowed with lincom:

```
. lincom _b[x2]/_b[x1]
not possible with test
r(131);
```

Nonlinear transformations are instead estimated using nlcom:
. nlcom _b[x2]/_b[x1]
_nl_1: _b[x2]/_b[x1]

| y | Coefficient | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | ---: | ---: |
| _nl_1 | 1.524714 | .9812848 | 1.55 | 0.120 | -.3985686 | 3.447997 |

- Technical note

The notation _b [name] is the standard way in Stata to refer to regression coefficients; see [U] 13.5 Accessing coefficients and standard errors. Some commands, such as lincom and test, allow you to drop the $\_\mathrm{b}[]$ and just refer to the coefficients by name. nlcom, however, requires the full specification _b [name $]$.

Returning to our linear regression example, nlcom also allows simultaneous estimation of more than one combination:

```
. nlcom (_b[x2]/_b[x1]) (_b[x3]/_b[x1]) (_b[x3]/_b[x2])
_nl_1: _b[x2]/_b[x1]
_nl_2: _b[x3]/_b[x1]
_nl_3: _b[x3]/_b[x2]
```

| y | Coefficient | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| _nl_1 | 1.524714 | .9812848 | 1.55 | 0.120 | -.3985686 | 3.447997 |
| _nl_2 | -.0042131 | .0033483 | -1.26 | 0.208 | -.0107756 | .0023494 |
| _nl_3 | -.0027632 | .0010695 | -2.58 | 0.010 | -.0048594 | -.000667 |

We can also label the transformations to produce more informative names in the estimation table:

```
. nlcom (ratio21:_b[x2]/_b[x1]) (ratio31:_b[x3]/_b[x1]) (ratio32:_b[x3]/_b[x2])
```

    ratio21: _b[x2]/_b[x1]
    ratio31: _b[x3]/_b[x1]
    ratio32: _b[x3]/_b[x2]
    | y | Coefficient | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | $[95 \%$ conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| ratio21 | 1.524714 | .9812848 | 1.55 | 0.120 | -.3985686 | 3.447997 |
| ratio31 | -.0042131 | .0033483 | -1.26 | 0.208 | -.0107756 | .0023494 |
| ratio32 | -.0027632 | .0010695 | -2.58 | 0.010 | -.0048594 | -.000667 |

nlcom stores the vector of estimated combinations and its estimated variance-covariance matrix in r() .

```
. matrix list r(b)
r(b)[1,3]
    ratio21 ratio31 ratio32
c1 1.5247143 -.00421315 -.00276324
. matrix list r(V)
symmetric r(V)[3,3]
                ratio21 ratio31 ratio32
ratio21 . }9629198
ratio31 -.00287781 . 00001121
ratio32 -.00014234 2.137e-06 1.144e-06
```


## Using the post option

When used with the post option, nlcom stores the estimation vector and variance-covariance matrix in e() , making the transformation permanent:

```
. quietly nlcom (ratio21:_b[x2]/_b[x1]) (ratio31:_b[x3]/_b[x1])
> (ratio32:_b[x3]/_b[x2]), post
. matrix list e(b)
```

$e(b)[1,3]$
ratio21 ratio31 ratio32
y1 $1.5247143-.00421315-.00276324$
. matrix list e(V)
symmetric e(V) $[3,3]$
ratio21 ratio31 ratio32
ratio21 . 96291982
ratio31 -. 00287781 . 00001121
ratio32 -. $00014234 \quad 2.137 \mathrm{e}-06 \quad 1.144 \mathrm{e}-06$

After posting, we can proceed as if we had just run a Stata estimation (eclass) command. For instance, we can replay the results,
. nlcom

| y | Coefficient | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | $[95 \%$ conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| ratio21 | 1.524714 | .9812848 | 1.55 | 0.120 | -.3985686 | 3.447997 |
| ratio31 | -.0042131 | .0033483 | -1.26 | 0.208 | -.0107756 | .0023494 |
| ratio32 | -.0027632 | .0010695 | -2.58 | 0.010 | -.0048594 | -.000667 |

or perform other postestimation tasks in the transformed metric, this time making reference to the new "coefficients":

```
. display _b[ratio31]
-.00421315
. estat vce, correlation
Correlation matrix of coefficients of nlcom model
\begin{tabular}{c|rrr}
\(e(V)\) & ratio21 & ratio31 & ratio32 \\
\hline ratio21 & 1.0000 & & \\
ratio31 & -0.8759 & 1.0000 & \\
ratio32 & -0.1356 & 0.5969 & 1.0000
\end{tabular}
```

```
. test _b[ratio21] = 1
```

. test _b[ratio21] = 1
( 1) ratio21 = 1
( 1) ratio21 = 1
chi2( 1) = 0.29
chi2( 1) = 0.29
Prob > chi2 = 0.5928

```
            Prob > chi2 = 0.5928
```

We see that testing $\_\mathrm{b}[$ ratio21] $=1$ in the transformed metric is equivalent to testing using testnl _b $[\mathrm{x} 2] / / \mathrm{b}[\mathrm{x} 1]=1$ in the original metric:

```
. quietly regress y x1 x2 x3
. testnl _b[x2]/_b[x1] = 1
    (1) _bb[x2]/_b[x1] = 1
        chi2(1) = 0.29
        Prob > chi2 = 0.5928
```

We needed to refit the regression model to recover the original parameter estimates.

## - Technical note

In a previous technical note, we mentioned that commands such as lincom and test permit reference to name instead of _b [name]. This is not the case when lincom and test are used after nlcom, post. In the above, we used

```
. test _b[ratio21] = 1
```

rather than

```
. test ratio21 = 1
```

which would have returned an error. Consider this a limitation of Stata. For the shorthand notation to work, you need a variable named name in the data. In nlcom, however, name is just a coefficient label that does not necessarily correspond to any variable in the data.

## Reparameterizing ML estimates for univariate data

When run using only a response and no covariates, Stata's maximum likelihood (ML) estimation commands will produce ML estimates of the parameters of some assumed univariate distribution for the response. The parameterization, however, is usually not one we are used to dealing with in a nonregression setting. In such cases, nlcom can be used to transform the estimation results from a regression model to those from a maximum likelihood estimation of the parameters of a univariate probability distribution in a more familiar metric.

## > Example 1

Consider the following univariate data on $Y=\#$ of traffic accidents at a certain intersection in a given year:
. use https://www.stata-press.com/data/r18/trafint
. summarize accidents

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| accidents | 12 | 13.83333 | 14.47778 | 0 | 41 |

A quick glance of the output from summarize leads us to quickly reject the assumption that $Y$ is distributed as Poisson because the estimated variance of $Y$ is much greater than the estimated mean of $Y$.

Instead, we choose to model the data as univariate negative binomial, of which a common parameterization is

$$
\operatorname{Pr}(Y=y)=\frac{\Gamma(r+y)}{\Gamma(r) \Gamma(y+1)} p^{r}(1-p)^{y} \quad 0 \leq p \leq 1, \quad r>0, \quad y=0,1, \ldots
$$

with

$$
E(Y)=\frac{r(1-p)}{p} \quad \operatorname{Var}(Y)=\frac{r(1-p)}{p^{2}}
$$

There exist no closed-form solutions for the maximum likelihood estimates of $p$ and $r$, yet they may be estimated by the iterative method of Newton-Raphson. One way to get these estimates would be to write our own Newton-Raphson program for the negative binomial. Another way would be to write our own ML evaluator; see [R] ml.

The easiest solution, however, would be to use Stata's existing negative binomial ML regression command, nbreg. The only problem with this solution is that nbreg estimates a different parameterization of the negative binomial, but we can worry about that later.

```
. nbreg accidents
Fitting Poisson model:
Iteration 0: Log likelihood = -105.05361
Iteration 1: Log likelihood = -105.05361
Fitting constant-only model:
Iteration 0: Log likelihood = -43.948619
Iteration 1: Log likelihood = -43.891483
Iteration 2: Log likelihood = -43.89144
Iteration 3: Log likelihood = -43.89144
Fitting full model:
Iteration 0: Log likelihood = -43.89144
Iteration 1: Log likelihood = -43.89144
```

| Negative binomial regression | Number of obs $=$ | 12 |
| :--- | :--- | ---: |
| Dispersion: mean | LR chi2 $(0)$ | $=0.00$ |
| Log likelihood $=-43.89144$ | Prob $>$ chi2 | $=$ |
|  | Pseudo R2 | $=0.0000$ |


| accidents | Coefficient | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | :---: | :---: |
| _cons | 2.627081 | .3192233 | 8.23 | 0.000 | 2.001415 | 3.252747 |
| /lnalpha | .1402425 | .4187147 |  |  | -.6804233 | .9609083 |
| alpha | 1.150553 | .4817534 |  |  | .5064026 | 2.61407 |

LR test of alpha=0: chibar2 $(01)=122.32 \quad$ Prob $>=$ chibar2 $=0.000$
. nbreg, coeflegend
Negative binomial regression $\quad$ Number of obs $=\quad 12$
Dispersion: mean
Log likelihood $=-43.89144$
LR chi2 (0) $=0.00$
$\begin{array}{ll}\text { Prob > chi2 } & =0.0 \\ \text { Pseudo R2 } & =0.0000\end{array}$

| accidents | Coefficient | Legend |
| ---: | ---: | :--- |
| _cons | 2.627081 | _b[_cons] |
| /lnalpha | .1402425 | _b[/lnalpha] |
| alpha | 1.150553 |  |

LR test of alpha=0: chibar2(01) $=122.32$
Prob >= chibar2 $=0.000$
From this output, we see that, when used with univariate data, nbreg estimates a regression intercept, $\beta_{0}$, and the logarithm of some parameter $\alpha$. This parameterization is useful in regression models: $\beta_{0}$ is the intercept meant to be augmented with other terms of the linear predictor, and $\alpha$ is an overdispersion parameter used for comparison with the Poisson regression model.

However, we need to transform $\left(\beta_{0}, \ln \alpha\right)$ to $(p, r)$. Examining Methods and formulas of $[\mathrm{R}]$ nbreg reveals the transformation as

$$
p=\left\{1+\alpha \exp \left(\beta_{0}\right)\right\}^{-1} \quad r=\alpha^{-1}
$$

which we apply using nlcom:

```
. nlcom (p:1/(1 + exp(_b[/lnalpha] + _b[_cons])))
> (r:exp(-_b[/lnalpha]))
    p: 1/(1 + exp(_b[/lnalpha] + _b[_cons]))
    r: exp(-_b[/lnalpha])
```

| accidents | Coefficient | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | :---: | ---: |
| p | .0591157 | .0292857 | 2.02 | 0.044 | .0017168 | .1165146 |
| r | .8691474 | .3639248 | 2.39 | 0.017 | .1558679 | 1.582427 |

Given the invariance of maximum likelihood estimators and the properties of the delta method, the above parameter estimates, standard errors, etc., are precisely those we would have obtained had we instead performed the Newton-Raphson optimization in the $(p, r)$ metric.

## Technical note

Note how we referred to the estimate of $\ln \alpha$ above as _b [/lnalpha]. This is not entirely evident from the output of nbreg, which is why we redisplayed the results using the coeflegend option so that we would know how to refer to the coefficients; [U] 13.5 Accessing coefficients and standard errors.

## nlcom versus eform

Many Stata estimation commands allow you to display exponentiated regression coefficients, some by default, some optionally. Known as "eform" in Stata terminology, this reparameterization serves many uses: it gives odds ratios for logistic models, hazard ratios in survival models, incidence-rate ratios in Poisson models, and relative-risk ratios in multinomial logit models, to name a few.

For example, consider the following estimation taken directly from the technical note in [R] poisson:

```
. use https://www.stata-press.com/data/r18/airline
. generate lnN = ln(n)
. poisson injuries XYZowned lnN
Iteration 0: Log likelihood = -22.333875
Iteration 1: Log likelihood = -22.332276
Iteration 2: Log likelihood = -22.332276
Poisson regression Number of obs = 9
```

LR chi2(2) $=19.15$
Prob > chi2 $=0.0001$
Pseudo R2 $=0.3001$

| injuries | Coefficient | Std. err. | $z$ | $\mathrm{P}>\|\mathrm{z}\|$ | $[95 \%$ conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | ---: | ---: |
| XYZowned | .6840667 | .3895877 | 1.76 | 0.079 | -.0795111 | 1.447645 |
| $\operatorname{lnN}$ | 1.424169 | .3725155 | 3.82 | 0.000 | .6940517 | 2.154285 |
| _cons | 4.863891 | .7090501 | 6.86 | 0.000 | 3.474178 | 6.253603 |

When we replay results and specify the irr (incidence-rate ratios) option,

| Poisson regre Log likelihood | n |  | Poisson regression |  | Number of obs LR chi2(2) <br> Prob > chi2 <br> Pseudo R2 | $\begin{array}{lr} = & 9 \\ = & 19.15 \\ = & 0.0001 \\ =0.3001 \end{array}$ |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| injuries | IRR | Std. err. | z | $P>\|z\|$ | [95\% conf. | interval] |
| XYZowned | 1.981921 | . 7721322 | 1.76 | 0.079 | . 9235678 | 4.253085 |
| lnN | 4.154402 | 1.547579 | 3.82 | 0.000 | 2.00181 | 8.621728 |
| _cons | 129.5272 | 91.84126 | 6.86 | 0.000 | 32.2713 | 519.8828 |

Note: _cons estimates baseline incidence rate.
we obtain the exponentiated regression coefficients and their estimated standard errors.
Contrast this with what we obtain if we exponentiate the coefficients manually by using nlcom:

```
. nlcom (E_XYZowned:exp(_b[XYZowned])) (E_lnN:exp(_b[lnN]))
    E_XYZowned: exp(_b[XYZowned])
            E_lnN: exp(_b[lnN])
\begin{tabular}{r|rccccc}
\hline injuries & Coefficient & Std. err. & \(z\) & \(\mathrm{P}>|\mathrm{z}|\) & {\([95 \%\) conf. interval] } \\
\hline E_XYZowned & 1.981921 & .7721322 & 2.57 & 0.010 & .4685701 & 3.495273 \\
E_lnN & 4.154402 & 1.547579 & 2.68 & 0.007 & 1.121203 & 7.187602 \\
\hline
\end{tabular}
```

There are three things to note when comparing an estimation command specified with an eform_option with the nlcom command specified with coefficients that are exponentiated manually:

1. The exponentiated coefficients and standard errors are identical. This is certainly good news.
2. The Wald test statistic ( $\mathbf{z}$ ) and level of significance are different. When using eform or any eform_option with an estimation command, the Wald test does not change from what you would have obtained without the eform option. You can see this by comparing both versions of the poisson output given previously.
When you use eform, Stata knows that what is usually desired is a test of

$$
H_{0}: \exp (\beta)=1
$$

and not the uninformative-by-comparison

$$
H_{0}: \exp (\beta)=0
$$

The test of $H_{0}: \exp (\beta)=1$ is asymptotically equivalent to a test of $H_{0}: \beta=0$, the Wald test in the original metric, but the latter has better small-sample properties. Thus if you specify eform, you get a test of $H_{0}: \beta=0$.
nlcom, however, is general. It does not attempt to infer the test of greatest interest for a given transformation, and so a test of

$$
H_{0}: \text { transformed coefficient }=0
$$

is always given, regardless of the transformation.
3. You may be surprised to see that, even though the coefficients and standard errors are identical, the confidence intervals (both 95\%) are different.
eform confidence intervals reported by an estimation command are standard confidence intervals with the endpoints transformed. For example, the confidence interval for the coefficient on $\operatorname{lnN}$ is $[0.694,2.154]$, whereas the confidence interval for the incidence-rate ratio due to $\operatorname{lnN}$ is $[\exp (0.694), \exp (2.154)]=[2.002,8.619]$, which, except for some roundoff error, is what we see from the output of poisson, irr. For exponentiated coefficients, confidence intervals based on transform-the-endpoints methodology generally have better small-sample properties than their asymptotically equivalent counterparts.

The transform-the-endpoints method, however, gives valid coverage only when the transformation is monotonic. nlcom uses a more general and asymptotically equivalent method for calculating confidence intervals, as described in Methods and formulas.

Notice that nlcom has an eform option that behaves the same way as the eform option of any estimation command. Thus, we could have obtained the exponentiated coefficients and confidence intervals after poisson by specifying the eform option with nlcom; that is, we would type

| . nlcom (E_XYZowned:_b[XYZowned]) (E_lnN:_b[lnN]), eform |
| :--- |
| E_XYZowned: _b[XYZowned] |
| E_InN: _b[lnN] |
| injuries |

We obtain the same results as we did using poisson with the irr option. The interpretation of the coefficients and confidence intervals follows the discussion above.

## Stored results

nlcom stores the following in $r()$ :
Scalars
$r(N) \quad$ number of observations
$r\left(d f \_r\right) \quad$ residual degrees of freedom
Matrices
$r(b) \quad$ vector of transformed coefficients
$r(V) \quad$ estimated variance-covariance matrix of the transformed coefficients
$r$ (table) matrix containing the estimates with their standard errors, test statistics, $p$-values, and confidence intervals

If post is specified, nlcom also stores the following in e() :
Scalars
e(N)
e(df_r)
e(N_strata)
e(N_psu)
$e$ (rank)
Macros
e(cmd) nlcom
$e$ (predict) program used to implement predict
e(properties) b V

Matrices
e(b) vector of transformed coefficients
$e(V) \quad$ estimated variance-covariance matrix of the transformed coefficients
e(V_srs) simple-random-sampling-without-replacement (co)variance $\widehat{V}_{\text {srswor }}$, if svy
e(V_srswr) simple-random-sampling-with-replacement (co)variance $\widehat{V}_{\text {srswr }}$, if svy and fpc()
$e$ (V_msp) misspecification (co)variance $\widehat{V}_{\mathrm{msp}}$, if svy and available
Functions
e(sample) marks estimation sample

## Methods and formulas

Given a $1 \times k$ vector of parameter estimates, $\widehat{\boldsymbol{\theta}}=\left(\widehat{\theta}_{1}, \ldots, \widehat{\theta}_{k}\right)$, consider the estimated $p$-dimensional transformation

$$
g(\widehat{\boldsymbol{\theta}})=\left[g_{1}(\widehat{\boldsymbol{\theta}}), g_{2}(\widehat{\boldsymbol{\theta}}), \ldots, g_{p}(\widehat{\boldsymbol{\theta}})\right]
$$

The estimated variance-covariance of $g(\widehat{\boldsymbol{\theta}})$ is given by

$$
\widehat{\operatorname{Var}}\{g(\widehat{\boldsymbol{\theta}})\}=\mathbf{G V G}^{\prime}
$$

where $\mathbf{G}$ is the $p \times k$ matrix of derivatives for which

$$
\mathbf{G}_{i j}=\left.\frac{\partial g_{i}(\boldsymbol{\theta})}{\partial \theta_{j}}\right|_{\boldsymbol{\theta}=\widehat{\boldsymbol{\theta}}} \quad i=1, \ldots, p \quad j=1, \ldots, k
$$

and $\mathbf{V}$ is the estimated variance-covariance matrix of $\widehat{\boldsymbol{\theta}}$. Standard errors are obtained as the square roots of the variances.

The Wald test statistic for testing

$$
H_{0}: g_{i}(\boldsymbol{\theta})=0
$$

versus the two-sided alternative is given by

$$
Z_{i}=\frac{g_{i}(\widehat{\boldsymbol{\theta}})}{\left[\widehat{\operatorname{Var}}_{i i}\{g(\widehat{\boldsymbol{\theta}})\}\right]^{1 / 2}}
$$

When the variance-covariance matrix of $\widehat{\boldsymbol{\theta}}$ is an asymptotic covariance matrix, $Z_{i}$ is approximately distributed as Gaussian. For linear regression, $Z_{i}$ is taken to be approximately distributed as $t_{1, r}$ where $r$ is the residual degrees of freedom from the original fitted model.

A $(1-\alpha) \times 100 \%$ confidence interval for $g_{i}(\boldsymbol{\theta})$ is given by

$$
g_{i}(\widehat{\boldsymbol{\theta}}) \pm z_{\alpha / 2}\left[\widehat{\operatorname{Var}}_{i i}\{g(\widehat{\boldsymbol{\theta}})\}\right]^{1 / 2}
$$

for those cases where $Z_{i}$ is Gaussian and

$$
g_{i}(\widehat{\boldsymbol{\theta}}) \pm t_{\alpha / 2, r}\left[\widehat{\operatorname{Var}}_{i i}\{g(\widehat{\boldsymbol{\theta}})\}\right]^{1 / 2}
$$

for those cases where $Z_{i}$ is $t$ distributed. $z_{p}$ is the $1-p$ quantile of the standard normal distribution, and $t_{p, r}$ is the $1-p$ quantile of the $t$ distribution with $r$ degrees of freedom.

## References

Feiveson, A. H. 1999. FAQ: What is the delta method and how is it used to estimate the standard error of a transformed parameter? https://www.stata.com/support/faqs/stat/deltam.html.

Oehlert, G. W. 1992. A note on the delta method. American Statistician 46: 27-29. https://doi.org/10.2307/2684406.
Phillips, P. C. B., and J. Y. Park. 1988. On the formulation of Wald tests of nonlinear restrictions. Econometrica 56: 1065-1083. https://doi.org/10.2307/1911359.

## Also see

[ $R$ ] lincom - Linear combinations of parameters
[R] predictnl - Obtain nonlinear predictions, standard errors, etc., after estimation
[R] test - Test linear hypotheses after estimation
[R] testnl - Test nonlinear hypotheses after estimation
[SVY] svy postestimation - Postestimation tools for svy
[U] 20 Estimation and postestimation commands

## Title

nlsur - Estimation of nonlinear systems of equations

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

nlsur fits a system of nonlinear equations by feasible generalized nonlinear least squares (FGNLS). With the interactive version of the command, you enter the system of equations on the command line or in the dialog box by using substitutable expressions. If you have a system that you use regularly, you can write a substitutable expression program and use the second syntax to avoid having to reenter the system every time. The function evaluator program version gives you the most flexibility in exchange for increased complexity; with this version, your program is given a vector of parameters and a variable list, and your program computes the system of equations.

When you write a substitutable expression program or a function evaluator program, the first five letters of the name must be nlsur. sexp_prog and func_prog refer to the name of the program without the first five letters. For example, if you wrote a function evaluator program named nlsurregss, you would type nlsur regss @ ... to estimate the parameters.

## Quick start

Two-parameter exponential model regressing y1 on x using the default FGNLS estimator nlsur $\left(y 1=\{b 1\} *\{b 2\}^{\wedge} x\right)$

Add the variables() option to allow for missing values of $y 1$ and $x$
nlsur ( $\mathrm{y} 1=\{\mathrm{b} 1\} *\{\mathrm{~b} 2\}^{\wedge} \mathrm{x}$ ), variables ( y 1 x )
Two-equation, two-parameter exponential model

```
nlsur (y1 = {b1}*{b2}^x) (y2 = {g1}*{g2}^x), variables(y1 y2 x)
```

Same as above, but use the iterative FGNLS estimator

$$
\text { nlsur }\left(y 1=\{b 1\} *\{b 2\}^{\wedge} x\right)\left(y 2=\{g 1\} *\{g 2\}^{\wedge} x\right), \operatorname{variables}(y 1 y 2 x) \text { ifgnls }
$$

Specify starting values for parameters b1 and g2

```
nlsur \(\left(y 1=\{b 1=.5\} *\{b 2\}^{\wedge} x\right)\left(y 2=\{g 1\} *\{g 2=1\}^{\wedge} x\right)\), variables(y1 y2 x)
```

Same as above

```
nlsur (y1 = {b1}*{b2}^x) (y2 = {g1}*{g2}^x), variables(y1 y2 x) ///
    initial(b1 . }5\mathrm{ g2 1)
```

Same as above, but specify starting values using the matrix i

```
matrix \(\mathrm{i}=(.5,0,0,1)\)
nlsur ( \(\mathrm{y} 1=\{\mathrm{b} 1\} *\{\mathrm{~b} 2\}^{\wedge} \mathrm{x}\) ) ( \(\mathrm{y} 2=\{\mathrm{g} 1\} *\{\mathrm{~g} 2\}^{\wedge} \mathrm{x}\) ), variables(y1 y2 x) ///
        initial(i)
```


## Menu

Statistics $>$ Linear models and related $>$ Multiple-equation models $>$ Nonlinear seemingly unrelated regression

## Syntax

Interactive version
nlsur $\left(\right.$ depvar_1 $=\left\langle\operatorname{sexp} \_>\right)\left(\right.$depvar_2 $\left.=\left\langle\operatorname{sexp} \_2\right\rangle\right) \ldots[$ if $][$ in $][$ weight $][$, options $]$
Programmed substitutable expression version
nlsur sexp_prog : depvar_1 depvar_2 ... [varlist] [if] [in] [weight] [, options]

Function evaluator program version
nlsur func_prog @ depvar_1 depvar_2 ... [varlist] [if] [in] [weight], nequations(\#) $\{\underline{\text { parameters(namelist) } \mid \underline{\text { nparameters(\#) }}\}[\text { options] }] ~}$
where
depvar $-j$ is the dependent variable for equation $j$;
$<\operatorname{sexp}>_{-j}$ is the substitutable expression for equation $j$; sexp_prog is a substitutable expression program; and func_prog is a function evaluator program.
options
Model
fgnls
ifgnls
nls
variables (varlist)
initial (initial_values)

* nequations (\#)
* parameters (namelist)
* nparameters (\#)
sexp_options
func_options
SE/Robust
vce (vcetype)

Reporting
level (\#)
title(string)
title2 (string)
display_options
Optimization
optimization_options
coeflegend

Description
use two-step FGNLS estimator; the default
use iterative FGNLS estimator
use NLS estimator
variables in model
initial values for parameters
number of equations in model (function evaluator program version only)
parameters in model (function evaluator program version only)
number of parameters in model
(function evaluator program version only)
options for substitutable expression program
options for function evaluator program
vcetype may be gnr, robust, cluster clustvar, bootstrap, or jackknife
set confidence level; default is level (95)
display string as title above the table of parameter estimates display string as subtitle
control columns and column formats and line width
control the optimization process; seldom used
display legend instead of statistics
*You must specify nequations(\#) and one of parameters (namelist) or nparameters(\#) or both. bootstrap, by, collect, jackknife, rolling, and statsby are allowed; see [U] 11.1.10 Prefix commands.
Weights are not allowed with the bootstrap prefix; see $[R]$ bootstrap.
aweights are not allowed with the jackknife prefix; see $[R]$ jackknife.
aweights, fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
coeflegend does not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

$\qquad$ Model
fgnls requests the two-step FGNLS estimator; this is the default.
ifgnls requests the iterative FGNLS estimator. For the nonlinear systems estimator, this is equivalent to maximum likelihood estimation.
nls requests the nonlinear least-squares (NLS) estimator.
variables (varlist) specifies the variables in the system. nlsur ignores observations for which any of these variables has missing values. If you do not specify variables(), nlsur issues an error message if the estimation sample contains any missing values.
initial(initial_values) specifies the initial values to begin the estimation. You can specify a $1 \times k$ matrix, where $k$ is the total number of parameters in the system, or you can specify a parameter name, its initial value, another parameter name, its initial value, and so on. For example, to initialize alpha to 1.23 and delta to 4.57, you would type

```
. nlsur ..., initial(alpha 1.23 delta 4.57) ...
```

Initial values declared using this option override any that are declared within substitutable expressions. If you specify a matrix, the values must be in the same order in which the parameters are declared in your model. nlsur ignores the row and column names of the matrix.
nequations(\#) specifies the number of equations in the system.
parameters (namelist) specifies the names of the parameters in the system. The names of the parameters must adhere to the naming conventions of Stata's variables; see [U] 11.3 Naming conventions. If you specify both parameters() and nparameters(), the number of names in the former must match the number specified in the latter.
nparameters(\#) specifies the number of parameters in the system. If you do not specify names with the parameters () option, nlsur names them $\mathrm{b} 1, \mathrm{~b} 2, \ldots, \mathrm{~b} \#$. If you specify both parameters () and nparameters (), the number of names in the former must match the number specified in the latter.
sexp_options refer to any options allowed by your sexp_prog.
func_options refer to any options allowed by your func_prog.

SE/Robust
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (gnr), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
vce (gnr), the default, uses the conventionally derived variance estimator for nonlinear models fit using Gauss-Newton regression.

## Reporting

level(\#); see [R] Estimation options.
title(string) specifies an optional title that will be displayed just above the table of parameter estimates.
title2(string) specifies an optional subtitle that will be displayed between the title specified in title() and the table of parameter estimates. If title2() is specified but title() is not, title2() has the same effect as title().
display_options: noci, nopvalues, cformat (\%fmt), pformat (\%fmt), sformat (\%fmt), and nolstretch; see [R] Estimation options.
optimization_options: iterate (\#), [no]log, trace, eps (\#), ifgnlsiterate(\#), ifgnlseps(\#), delta(\#), noconstants, and hasconstants (namelist).
iterate () specifies the maximum number of iterations to use for NLS at each round of FGNLS estimation. This option is different from ifgnlsiterate(), which controls the maximum rounds of FGNLS estimation to use when the ifgnls option is specified.
log and nolog specify whether to show the iteration $\log$; see set iterlog in [R] set iter.
trace specifies that the iteration $\log$ should include the current parameter vector.
eps (\#) specifies the convergence criterion for successive parameter estimates and for the residual sum of squares (RSS). The default is eps (1e-5) (0.00001). eps () also specifies the convergence criterion for successive parameter estimates between rounds of iterative FGNLS estimation when ifgnls is specified.
ifgnlsiterate (\#) specifies the maximum number of FGNLS iterations to perform. The default is the number set using set maxiter, which is 300 by default. To use this option, you must also specify the ifgnls option.
ifgnlseps (\#) specifies the convergence criterion for successive estimates of the error covariance matrix during iterative FGNLS estimation. The default is ifgnlseps (1e-10). To use this option, you must also specify the ifgnls option.
delta(\#) specifies the relative change in a parameter, $\delta$, to be used in computing the numeric derivatives. The derivative for parameter $\beta_{i}$ is computed as

$$
\left\{f_{i}\left(\mathbf{x}_{i}, \beta_{1}, \beta_{2}, \ldots, \beta_{i}+d, \beta_{i+1}, \ldots\right)-f_{i}\left(\mathbf{x}_{i}, \beta_{1}, \beta_{2}, \ldots, \beta_{i}, \beta_{i+1}, \ldots\right)\right\} / d
$$

where $d=\delta\left(\left|\beta_{i}\right|+\delta\right)$. The default is delta (4e-7).
noconstants indicates that none of the equations in the system includes constant terms. This option is generally not needed, even if there are no constant terms in the system; though in rare cases without this option, nlsur may claim that there is one or more constant terms even if there are none.
hasconstants (namelist) indicates the parameters that are to be treated as constant terms in the system of equations. The number of elements of namelist must equal the number of equations in the system. The $i$ th entry of namelist specifies the constant term in the $i$ th equation. If an equation does not include a constant term, specify a period (.) instead of a parameter name. This option is seldom needed with the interactive and programmed substitutable expression versions, because in those cases nlsur can almost always find the constant terms automatically.

The following options are available with nlsur but are not shown in the dialog box:
coeflegend; see [R] Estimation options.

## Remarks and examples

Remarks are presented under the following headings:

## Introduction

Substitutable expression programs
Function evaluator programs

## Introduction

nlsur fits a system of nonlinear equations by FGNLS. It can be viewed as a nonlinear variant of Zellner's seemingly unrelated regression model (Zellner 1962; Zellner and Huang 1962; Zellner 1963) and is therefore commonly called nonlinear SUR or nonlinear SURE. The model is also discussed in textbooks such as Davidson and MacKinnon (1993) and Greene (2012, 305-306). Formally, the model fit by nlsur is

$$
\begin{aligned}
y_{i 1} & =f_{1}\left(\mathbf{x}_{i}, \boldsymbol{\beta}\right)+u_{i 1} \\
y_{i 2} & =f_{2}\left(\mathbf{x}_{i}, \boldsymbol{\beta}\right)+u_{i 2} \\
\vdots & =\vdots \\
y_{i M} & =f_{M}\left(\mathbf{x}_{i}, \boldsymbol{\beta}\right)+u_{i M}
\end{aligned}
$$

for $i=1, \ldots, N$ observations and $m=1, \ldots, M$ equations. The errors for the $i$ th observation, $u_{i 1}, u_{i 2}, \ldots, u_{i M}$, may be correlated, so fitting the $m$ equations jointly may lead to more efficient estimates. Moreover, fitting the equations jointly allows us to impose cross-equation restrictions on the parameters. Not all elements of the parameter vector $\boldsymbol{\beta}$ and data vector $\mathbf{x}_{i}$ must appear in all the equations, though each element of $\boldsymbol{\beta}$ must appear in at least one equation for $\boldsymbol{\beta}$ to be identified. For this model, iterative FGNLS estimation is equivalent to maximum likelihood estimation with multivariate normal disturbances.

The syntax you use with nlsur closely mirrors that used with nl. In particular, you use substitutable expressions with the interactive and programmed substitutable expression versions to define the functions in your system. See $[R] \mathbf{n l}$ for more information on substitutable expressions. Here we reiterate the three rules that you must follow:

1. Parameters of the model are bound in braces: $\{b 0\}$, \{param\}, etc.
2. Initial values for parameters are given by including an equal sign and the initial value inside the braces: $\{b 0=1\}$, $\{$ param $=3.571\}$, etc. If you do not specify an initial value, that parameter is initialized to zero. The initial() option overrides initial values in substitutable expressions.
3. Linear combinations of variables can be included using the notation \{eqname:varlist\}, for example, $\{\mathrm{xb}$ : mpg price weight\}, \{score: wxz\}, etc. Parameters of linear combinations are initialized to zero.
> Example 1: Interactive version using two-step FGNLS estimator
We have data from an experiment in which two closely related types of bacteria were placed in a Petri dish, and the number of each type of bacteria were recorded every hour. We suspect a two-parameter exponential growth model can be used to model each type of bacteria, but because they shared the same dish, we want to allow for correlation in the error terms. We want to fit the system of equations

$$
\begin{aligned}
& p_{1}=\beta_{1} \beta_{2}{ }^{t}+u_{1} \\
& p_{2}=\gamma_{1}{\gamma_{2}}^{t}+u_{2}
\end{aligned}
$$

where $p_{1}$ and $p_{2}$ are the two populations and $t$ is time, and we want to allow for nonzero correlation between $u_{1}$ and $u_{2}$. We type

```
. use https://www.stata-press.com/data/r18/petridish
. nlsur (p1 = {b1}*{b2}^t) (p2 = {g1}*{g2}^t)
(obs = 25)
Calculating NLS estimates:
Iteration 0: Residual SS = 335.5286
Iteration 1: Residual SS = 333.8583
Iteration 2: Residual SS = 219.9233
Iteration 3: Residual SS = 127.9355
Iteration 4: Residual SS = 14.86765
Iteration 5: Residual SS = 8.628459
Iteration 6: Residual SS = 8.281268
Iteration 7: Residual SS = 8.28098
Iteration 8: Residual SS = 8.280979
Iteration 9: Residual SS = 8.280979
Calculating FGNLS estimates:
Iteration 0: Scaled RSS = 49.99892
Iteration 1: Scaled RSS = 49.99892
Iteration 2: Scaled RSS = 49.99892
FGNLS regression
```

|  | Equation | Obs | Parms | RMSE | R-sq | Constant |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: |
| 1 | p1 | 25 | 2 | .4337019 | $0.9734 *$ | (none) |
| 2 | p2 | 25 | 2 | .3783479 | $0.9776 *$ | (none) |

* Uncentered R-sq

|  | Coefficient | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| /b1 | .3926631 | .064203 | 6.12 | 0.000 | .2668275 | .5184987 |
| /b2 | 1.119593 | .0088999 | 125.80 | 0.000 | 1.102149 | 1.137036 |
| /g1 | .5090441 | .0669495 | 7.60 | 0.000 | .3778256 | .6402626 |
| /g2 | 1.102315 | .0072183 | 152.71 | 0.000 | 1.088167 | 1.116463 |

The header of the output contains a summary of each equation, including the number of observations and parameters and the root mean squared error of the residuals. nlsur checks to see whether each equation contains a constant term, and if an equation does contain a constant term, an $R^{2}$ statistic is presented. If an equation does not have a constant term, an uncentered $R^{2}$ is instead reported. The $R^{2}$ statistic for each equation measures the percentage of variance explained by the nonlinear function and may be useful for descriptive purposes, though it does not have the same formal interpretation in the context of FGNLS as it does with NLS estimation. As we would expect, $\beta_{2}$ and $\gamma_{2}$ are both greater than one, indicating the two bacterial populations increased in size over time.

The model we fit in the next three examples is in fact linear in the parameters, so it could be fit using the sureg command. However, we will fit the model using nlsur so that we can focus on the mechanics of using the command. Moreover, using nlsur will obviate the need to generate several variables as well as the need to use the constraint command to impose parameter restrictions.

## Example 2: Interactive version using iterative FGNLS estimator-the translog production function

Greene (1997, sec. 15.6) discusses the transcendental logarithmic (translog) cost function and provides cost and input price data for capital, labor, energy, and materials for the US economy. One way to fit the translog production function to these data is to fit the system of three equations

$$
\begin{aligned}
s_{k} & =\beta_{k}+\delta_{k k} \ln \left(\frac{p_{k}}{p_{m}}\right)+\delta_{k l} \ln \left(\frac{p_{l}}{p_{m}}\right)+\delta_{k e} \ln \left(\frac{p_{e}}{p_{m}}\right)+u_{1} \\
s_{l} & =\beta_{l}+\delta_{k l} \ln \left(\frac{p_{k}}{p_{m}}\right)+\delta_{l l} \ln \left(\frac{p_{l}}{p_{m}}\right)+\delta_{l e} \ln \left(\frac{p_{e}}{p_{m}}\right)+u_{2} \\
s_{e} & =\beta_{e}+\delta_{k e} \ln \left(\frac{p_{k}}{p_{m}}\right)+\delta_{l e} \ln \left(\frac{p_{l}}{p_{m}}\right)+\delta_{e e} \ln \left(\frac{p_{e}}{p_{m}}\right)+u_{3}
\end{aligned}
$$

where $s_{k}$ is capital's cost share, $s_{l}$ is labor's cost share, and $s_{e}$ is energy's cost share; $p_{k}, p_{l}, p_{e}$, and $p_{m}$ are the prices of capital, labor, energy, and materials, respectively; the $u$ 's are regression error terms; and the $\beta$ 's and $\delta$ 's are parameters to be estimated. There are three cross-equation restrictions on the parameters: $\delta_{k l}, \delta_{k e}$, and $\delta_{l e}$ each appear in two equations. To fit this model by using the iterative FGNLS estimator, we type

```
. use https://www.stata-press.com/data/r18/mfgcost
(Manufacturing cost)
. nlsur (s_k = {bk} + {dkk}*ln(pk/pm) + {dkl}*ln(pl/pm) +{dke}*ln(pe/pm))
> (s_l = {bl} + {dkl}*ln (pk/pm) + {dll}*ln (pl/pm) + {dle}*ln (pe/pm))
> (s_e = {be} + {dke}*ln(pk/pm) + {dle}*ln(pl/pm) + {dee}*ln(pe/pm)),
> ifgnls
(obs = 25)
```

Calculating NLS estimates:
Iteration 0: Residual $\mathrm{SS}=.0009989$
Iteration 1: Residual $\mathrm{SS}=.0009989$
Calculating FGNLS estimates:
Iteration 0: Scaled RSS $=65.45197$
Iteration 1: Scaled RSS $=65.45197$
(output omitted)
FGNLS iteration 10:
Iteration 0: Scaled RSS $=\quad 75$
Iteration 1: Scaled RSS $=\quad 75$
Parameter change $\quad=4.08 \mathrm{e}-06$
Covariance matrix change $=6.26 \mathrm{e}-10$
FGNLS regression

|  | Equation | Obs | Parms | RMSE | R-sq | Constant |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: |
| 1 | s_k | 25 | 4 | .0031722 | 0.4776 | bk |
| 2 | s_1 | 25 | 4 | .0053963 | 0.8171 | bl |
| 3 | s_e | 25 | 4 | .00177 | 0.6615 | be |


|  | Coefficient | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| /bk | .0568925 | .0013454 | 42.29 | 0.000 | .0542556 | .0595294 |
| /dkk | .0294833 | .0057956 | 5.09 | 0.000 | .0181241 | .0408425 |
| /dkl | -.0000471 | .0038478 | -0.01 | 0.990 | -.0075887 | .0074945 |
| /dke | -.0106749 | .0033882 | -3.15 | 0.002 | -.0173157 | -.0040341 |
| /bl | .253438 | .0020945 | 121.00 | 0.000 | .2493329 | .2575432 |
| /dll | .0754327 | .0067572 | 11.16 | 0.000 | .0621889 | .0886766 |
| /dle | -.004756 | .002344 | -2.03 | 0.042 | -.0093501 | -.0001619 |
| /be | .0444099 | .0008533 | 52.04 | 0.000 | .0427374 | .0460823 |
| /dee | .0183415 | .0049858 | 3.68 | 0.000 | .0085694 | .0281135 |

We draw your attention to the iteration log at the top of the output. When iterative FGNLS estimation is used, the final scaled RSS will equal the product of the number of observations in the estimation sample and the number of equations; see Methods and formulas for details. Because the RSS is
scaled by the error covariance matrix during each round of FGNLS estimation, the scaled RSS is not comparable from one FGNLS iteration to the next.

## Technical note

You may have noticed that we mentioned having data for four factors of production, yet we fit only three share equations. Because the four shares sum to one, we must drop one of the equations to avoid having a singular error covariance matrix. The iterative FGNLS estimator is equivalent to maximum likelihood estimation, and thus it is invariant to which one of the four equations we choose to drop. The (linearly restricted) parameters of the fourth equation can be obtained using the lincom command. Nonlinear functions of the parameters, such as the elasticities of substitution, can be computed using nlcom.

## Substitutable expression programs

If you fit the same model repeatedly or you want to share code with colleagues, you can write a substitutable expression program to define your system of equations and avoid having to retype the system every time. The first five letters of the program's name must be nlsur, and the program must set the r-class macro $r$ ( $n \_e q$ ) to the number of equations in your system. The first equation's substitutable expression must be returned in $r$ (eq_1), the second equation's in $r$ (eq_2), and so on. You may optionally set $r$ (title) to label your output; that has the same effect as specifying the title() option.

## > Example 3: Programmed substitutable expression version

We return to our translog cost function, for which a substitutable expression program is

```
program nlsurtranslog, rclass
    version 18.0 // (or version 18.5 for StataNow)
    syntax varlist(min=7 max=7) [if]
    tokenize 'varlist'
    args sk sl se pk pl pe pm
    local pkpm ln('pk'/'pm')
    local plpm ln('pl'/'pm')
    local pepm ln('pe'/'pm')
    return scalar n_eq = 3
    return local eq_1 "'sk'= {bk} + {dkk}*'pkpm' + {dkl}*'plpm' + {dke}*'pepm'"
    return local eq_2 "'sl'= {bl} + {dkl}*'pkpm' + {dll}*'plpm' + {dle}*'pepm'"
    return local eq_3 "'se'= {be} + {dke}*'pkpm' + {dle}*'plpm' + {dee}*'pepm'"
    return local title "4-factor translog cost function"
end
```

We made our program accept seven variables, for the three dependent variables $s_{k}, s_{l}$, and $s_{e}$, and the four factor prices $p_{k}, p_{l}, p_{m}$, and $p_{e}$. The tokenize command assigns to macros ' 1 ', ' 2 ', $\ldots$, ' 7 ' the seven variables stored in 'varlist', and the args command transfers those numbered macros to macros 'sk', 'sl', ..., 'pm'. Because we knew our substitutable expressions were going to be somewhat long, we created local macros to hold the log price ratios. These are simply macros that hold strings such as $\ln (\mathrm{pk} / \mathrm{pm})$, not variables, and they will save us some repetitious typing when we define our substitutable expressions. Our program returns the number of equations in $r$ ( $n \_e q$ ), and we defined our substitutable expressions in eq_1, eq_2, and eq_3. We do not bind the expressions in parentheses as we do with the interactive version of nlsur. Finally, we put a title in $r$ (title) to label our output.

Our syntax command also accepts an if clause, and that is how nlsur indicates the estimation sample to our program. In this application, we can safely ignore it because our program does not compute initial values. However, had we used commands such as summarize or regress to obtain initial values, then we would need to restrict those commands to analyze only the estimation sample. In those cases, typically, you simply need to include 'if' with the commands you are using. For example, instead of the command

```
summarize 'depvar', meanonly
```

you would use

```
summarize 'depvar' 'if', meanonly
```

We can check our program by typing

```
. nlsurtranslog s_k s_l s_e pk pl pe pm
. return list
scalars:
    r(n_eq) = 3
macros:
r(title) : "4-factor translog cost function"
    r(eq_3) : "s_e= {be} + {dke}*ln(pk/pm) + {dle}*ln(pl/pm) + {.."
    r(eq_2) : "s_l= {bl} + {dkl}*ln(pk/pm) + {dll}*ln(pl/pm) + {.."
    r(eq_1) : "s_k= {bk} + {dkk}*ln(pk/pm) + {dkl}*ln(pl/pm) + {.."
```

Now that we know that our program works, we fit our model by typing

```
. nlsur translog: s_k s_l s_e pk pl pe pm, ifgnls
(obs = 25)
Calculating NLS estimates:
Iteration 0: Residual SS = . 0009989
Iteration 1: Residual SS = .0009989
Calculating FGNLS estimates:
Iteration 0: Scaled RSS = 65.45197
Iteration 1: Scaled RSS = 65.45197
FGNLS iteration 2:
Iteration 0: Scaled RSS = 73.28311
Iteration 1: Scaled RSS = 73.28311
Iteration 2: Scaled RSS = 73.28311
Parameter change = 6.54e-03
Covariance matrix change = 1.00e-06
    (output omitted)
FGNLS iteration 10:
Iteration 0: Scaled RSS = 75
Iteration 1: Scaled RSS = 75
Parameter change =4.08e-06
Covariance matrix change = 6.26e-10
FGNLS regression
```

|  | Equation | Obs | Parms | RMSE | R-sq | Constant |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: |
| 1 | s_k | 25 | 4 | .0031722 | 0.4776 | bk |
| 2 | s_1 | 25 | 4 | .0053963 | 0.8171 | bl |
| 3 | s_e | 25 | 4 | .00177 | 0.6615 | be |

4-factor translog cost function

|  | Coefficient | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| /bk | .0568925 | .0013454 | 42.29 | 0.000 | .0542556 | .0595294 |
| /dkk | .0294833 | .0057956 | 5.09 | 0.000 | .0181241 | .0408425 |
| /dkl | -.0000471 | .0038478 | -0.01 | 0.990 | -.0075887 | .0074945 |
| /dke | -.0106749 | .0033882 | -3.15 | 0.002 | -.0173157 | -.0040341 |
| /bl | .253438 | .0020945 | 121.00 | 0.000 | .2493329 | .2575432 |
| /dll | .0754327 | .0067572 | 11.16 | 0.000 | .0621889 | .0886766 |
| /dle | -.004756 | .002344 | -2.03 | 0.042 | -.0093501 | -.0001619 |
| /be | .0444099 | .0008533 | 52.04 | 0.000 | .0427374 | .0460823 |
| /dee | .0183415 | .0049858 | 3.68 | 0.000 | .0085694 | .0281135 |

Because we set $r$ (title) in our substitutable expression program, the coefficient table has a title attached to it. The estimates are identical to those we obtained in example 2.

## Technical note

nlsur accepts frequency and analytic weights as well as pweights (sampling weights) and iweights (importance weights). You do not need to modify your substitutable expressions in any way to perform weighted estimation, though you must make two changes to your substitutable expression program. The general outline of a sexp_prog program is

```
program nlsur name, rclass
    version 18.0 // (or version 18.5 for StataNow)
    syntax varlist [fw aw pw iw] [if]
    // Obtain initial values incorporating weights. For example,
    summarize varname ['weight''exp'] 'if'
    // Return n_eqn and substitutable expressions
    return scalar n_eq = #
    return local eq_1 = ...
end
```

First, we wrote the syntax statement to accept a weight expression. Here we allow all four types of weights, but if you know that your estimator is valid, say, for only frequency weights, then you should modify the syntax line to accept only fweights. Second, if your program computes starting values, then any commands you use must incorporate the weights passed to the program; you do that by including ['weight' 'exp'] when calling those commands.

## Function evaluator programs

Although substitutable expressions are extremely flexible, there are some problems for which the nonlinear system cannot be defined using them. You can use the function evaluator program version of nlsur in these cases. We present two examples, a simple one to illustrate the mechanics of function evaluator programs and a more complicated one to illustrate the power of nlsur.

## > Example 4: Function evaluator program version

Here we write a function evaluator program to fit the translog cost function used in examples 2 and 3. The function evaluator program is

```
program nlsurtranslog2
    version 18.0 // (or version 18.5 for StataNow)
    syntax varlist(min=7 max=7) [if], at(name)
    tokenize 'varlist'
    args sk sl se pk pl pe pm
    tempname bk dkk dkl dke bl dll dle be dee
    scalar 'bk' = 'at'[1,1]
    scalar 'dkk' = 'at'[1,2]
    scalar 'dkl' = 'at'[1,3]
    scalar 'dke' = 'at'[1,4]
    scalar 'bl' = 'at'[1,5]
    scalar 'dll' = 'at'[1,6]
    scalar 'dle' = 'at'[1,7]
    scalar 'be' = 'at'[1,8]
    scalar 'dee' = 'at'[1,9]
    local pkpm ln('pk'/'pm')
    local plpm ln('pl'/'pm')
    local pepm ln('pe'/'pm')
    quietly {
        replace 'sk' = 'bk' + 'dkk'*'pkpm' + 'dkl'*'plpm' + ///
        'dke'*'pepm' 'if'
    replace 'sl' = 'bl' + 'dkl'*'pkpm' + 'dll'*'plpm' + ///
        'dle'*'pepm' 'if'
        replace 'se' = 'be' + 'dke'*'pkpm' + 'dle'*'plpm' + ///
        'dee'*'pepm' 'if'
    }
end
```

Unlike the substitutable expression program we wrote in example 3, nlsurtranslog2 is not declared as r-class because we will not be returning any stored results. We are again expecting seven variables: three shares and four factor prices, and nlsur will again mark the estimation sample with an if expression.

Our function evaluator program also accepts an option named at (), which will receive a parameter vector at which we are to evaluate the system of equations. All function evaluator programs must accept this option. Our model has nine parameters to estimate, and we created nine temporary scalars to hold the elements of the 'at' matrix.

Because our model has three equations, the first three variables passed to our program are the dependent variables that we are to fill in with the function values. We replaced only the observations in our estimation sample by including the 'if' qualifier in the replace statements. Here we could have ignored the 'if' qualifier because nlsur will skip over observations not in the estimation sample and we did not perform any computations requiring knowledge of the estimation sample. However, including the 'if' is good practice and may result in a slight speed improvement if the functions of your model are complicated and the estimation sample is much smaller than the dataset in memory.

We could have avoided creating temporary scalars to hold our individual parameters by writing the replace statements as, for example,

```
replace 'sk' = 'at'[1,1] + 'at'[1,2]*'pkpm' + 'at'[1,3]*'plpm' + 'at'[1,4]*'pepm' 'if'
```

You can use whichever method you find more appealing, though giving the parameters descriptive names reduces the chance for mistakes and makes debugging easier.

To fit our model by using the function evaluator program version of nlsur, we type

```
. nlsur translog2 @ s_k s_l s_e pk pl pe pm, ifgnls nequations(3)
> parameters(bk dkk dkl dke bl dll dle be dee)
> hasconstants(bk bl be)
(obs = 25)
```

Calculating NLS estimates:
Iteration 0: Residual $\mathrm{SS}=.0009989$
Iteration 1: Residual $\mathrm{SS}=.0009989$
Calculating FGNLS estimates:
Iteration 0: Scaled RSS $=65.45197$
Iteration 1: Scaled RSS $=65.45197$
FGNLS iteration 2:
Iteration 0: Scaled RSS $=73.28311$
Iteration 1: Scaled RSS $=73.28311$
Iteration 2: Scaled RSS $=73.28311$
Parameter change $\quad=6.54 \mathrm{e}-03$
Covariance matrix change $=1.00 \mathrm{e}-06$
FGNLS iteration 3:
Iteration 0: Scaled RSS $=74.7113$
Iteration 1: Scaled RSS $=74.7113$
Parameter change $\quad=2.58 \mathrm{e}-03$
Covariance matrix change $=3.96 \mathrm{e}-07$
FGNLS iteration 4:
Iteration 0: Scaled RSS $=74.95356$
Iteration 1: Scaled RSS $=74.95356$
Parameter change $\quad=1.02 \mathrm{e}-03$
Covariance matrix change $=1.57 \mathrm{e}-07$
FGNLS iteration 5:
Iteration 0: Scaled RSS $=74.99261$
Iteration 1: Scaled RSS $=74.99261$
Parameter change $\quad=4.07 \mathrm{e}-04$
Covariance matrix change $=6.25 \mathrm{e}-08$
FGNLS iteration 6:
Iteration 0: Scaled RSS $=74.99883$
Iteration 1: Scaled RSS $=74.99883$
Parameter change $\quad=1.62 \mathrm{e}-04$
Covariance matrix change $=2.49 \mathrm{e}-08$
FGNLS iteration 7:
Iteration 0: Scaled RSS $=74.99981$
Iteration 1: Scaled RSS $=74.99981$
Iteration 2: Scaled RSS $=74.99981$
Parameter change $\quad=6.45 \mathrm{e}-05$
Covariance matrix change $=9.91 \mathrm{e}-09$
FGNLS iteration 8:
Iteration 0: Scaled RSS $=74.99997$
Iteration 1: Scaled RSS $=74.99997$
Iteration 2: Scaled RSS $=74.99997$
Iteration 3: Scaled RSS $=74.99997$
Parameter change $\quad=2.57 \mathrm{e}-05$
Covariance matrix change $=3.95 \mathrm{e}-09$
FGNLS iteration 9:
Iteration 0: Scaled RSS $=\quad 75$
Iteration 1: Scaled RSS $=\quad 75$
Iteration 2: Scaled RSS $=\quad 75$
Parameter change $\quad=1.02 \mathrm{e}-05$
Covariance matrix change $=1.57 \mathrm{e}-09$
FGNLS iteration 10:
Iteration 0: Scaled RSS $=\quad 75$

| Iteration 1: Scaled RSS | $=r$ |
| :--- | :--- |
| Parameter change | $=4.08 \mathrm{e}-06$ |
| Covariance matrix change | $=6.26 \mathrm{e}-10$ |
| FGNLS regression |  |


|  | Equation | Obs | Parms | RMSE | R-sq | Constant |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: |
| 1 | s_k | 25 | . | .0031722 | 0.4776 | bk |
| 2 | s_l | 25 | . | .0053963 | 0.8171 | bl |
| 3 | s_e | 25 | . | .00177 | 0.6615 | be |


|  | Coefficient | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| /bk | .0568925 | .0013454 | 42.29 | 0.000 | .0542556 | .0595294 |
| /dkk | .0294833 | .0057956 | 5.09 | 0.000 | .0181241 | .0408425 |
| /dkl | -.0000471 | .0038478 | -0.01 | 0.990 | -.0075887 | .0074945 |
| /dke | -.0106749 | .0033882 | -3.15 | 0.002 | -.0173157 | -.0040341 |
| /bl | .253438 | .0020945 | 121.00 | 0.000 | .2493329 | .2575432 |
| /dll | .0754327 | .0067572 | 11.16 | 0.000 | .0621889 | .0886766 |
| /dle | -.004756 | .002344 | -2.03 | 0.042 | -.0093501 | -.0001619 |
| /be | .0444099 | .0008533 | 52.04 | 0.000 | .0427374 | .0460823 |
| /dee | .0183415 | .0049858 | 3.68 | 0.000 | .0085694 | .0281135 |

When we use the function evaluator program version, nlsur requires us to specify the number of equations in nequations (), and it requires us to specify either the names for each of our parameters or the number of parameters in the model. Here we used the parameters () option to name our parameters; the order in which we specified them in this option is the same as the order in which we extracted them from the 'at' matrix in our program. Had we instead specified nparameters (9), our parameters would have been labeled /b1, /b2, ..., /b9 in the output.
nlsur has no way of telling how many parameters appear in each equation, so the Parms column in the header contains missing values. Moreover, the function evaluator program version of nlsur does not attempt to identify constant terms, so we used the hasconstant option to tell nlsur which parameter in each equation is a constant term.

The estimates are identical to those we obtained in examples 2 and 3 .

## Technical note

As with substitutable expression programs, if you intend to do weighted estimation with a function evaluator program, you must modify your func_prog program's syntax statement to accept weights. Moreover, if you use any statistical commands when computing your nonlinear functions, then you must include the weight expression with those commands.

## > Example 5: Fitting the basic AIDS model using nlsur

Deaton and Muellbauer (1980) introduce the almost ideal demand system (AIDS), and Poi (2012) presents a set of commands and several extensions for fitting the AIDS automatically. Here we show how to fit the basic AIDS model, which is a common example of a nonlinear system of equations, by manually using nlsur. The dataset food.dta contains household expenditures, expenditure shares, and $\log$ prices for four broad food groups. For a four-good demand system, we need to fit the following system of three equations:

$$
\begin{aligned}
& w_{1}=\alpha_{1}+\gamma_{11} \ln p_{1}+\gamma_{12} \ln p_{2}+\gamma_{13} \ln p_{3}+\beta_{1} \ln \left\{\frac{m}{P(\mathbf{p})}\right\}+u_{1} \\
& w_{2}=\alpha_{2}+\gamma_{12} \ln p_{1}+\gamma_{22} \ln p_{2}+\gamma_{23} \ln p_{3}+\beta_{2} \ln \left\{\frac{m}{P(\mathbf{p})}\right\}+u_{2} \\
& w_{3}=\alpha_{3}+\gamma_{13} \ln p_{1}+\gamma_{23} \ln p_{2}+\gamma_{33} \ln p_{3}+\beta_{3} \ln \left\{\frac{m}{P(\mathbf{p})}\right\}+u_{3}
\end{aligned}
$$

where $w_{k}$ denotes a household's fraction of expenditures on good $k, \ln p_{k}$ denotes the logarithm of the price paid for good $k, m$ denotes a household's total expenditure on all four goods, the $u$ 's are regression error terms, and

$$
\ln P(\mathbf{p})=\alpha_{0}+\sum_{i=1}^{4} \alpha_{i} \ln p_{i}+\frac{1}{2} \sum_{i=1}^{4} \sum_{j=1}^{4} \gamma_{i j} \ln p_{i} \ln p_{j}
$$

The parameters for the fourth good's share equation can be recovered from the following constraints that are imposed by economic theory:

$$
\sum_{i=1}^{4} \alpha_{i}=1 \quad \sum_{i=1}^{4} \beta_{i}=0 \quad \gamma_{i j}=\gamma_{j i} \quad \text { and } \quad \sum_{i=1}^{4} \gamma_{i j}=0 \text { for all } j
$$

Our model has a total of 12 unrestricted parameters. We will not estimate $\alpha_{0}$ directly. Instead, we will set it equal to 5; see Deaton and Muellbauer (1980) for a discussion of why treating $\alpha_{0}$ as fixed is acceptable.

Our function evaluator program is

```
program nlsuraids
    version 18.0 // (or version 18.5 for StataNow)
    syntax varlist(min=8 max=8) if, at(name)
    tokenize 'varlist'
    args w1 w2 w3 lnp1 lnp2 lnp3 lnp4 lnm
    tempname a1 a2 a3 a4
    scalar 'a1' = 'at'[1,1]
    scalar 'a2' = 'at'[1,2]
    scalar 'a3' = 'at'[1,3]
    scalar 'a4' = 1 - 'a1' - 'a2' - 'a3'
    tempname b1 b2 b3
    scalar 'b1' = 'at'[1,4]
    scalar 'b2' = 'at'[1,5]
    scalar 'b3' = 'at'[1,6]
    tempname g11 g12 g13 g14
    tempname g21 g22 g23 g24
    tempname g31 g32 g33 g34
    tempname g41 g42 g43 g44
    scalar 'g11' = 'at'[1,7]
    scalar 'g12' = 'at'[1,8]
    scalar 'g13' = 'at'[1,9]
    scalar 'g14' = -'g11'-'g12'-'g13'
    scalar 'g21' = 'g12'
    scalar 'g22' = 'at'[1,10]
    scalar 'g23' = 'at'[1,11]
    scalar 'g24' = -'g21'-'g22'-'g23'
    scalar 'g31' = 'g13'
    scalar 'g32' = 'g23'
    scalar 'g33' = 'at'[1,12]
    scalar 'g34' = -'g31'-'g32'-'g33'
    scalar 'g41' = 'g14'
    scalar 'g42' = 'g24'
    scalar 'g43' = 'g34'
    scalar 'g44' = -'g41'-'g42'-'g43'
    quietly {
        tempvar lnpindex
        gen double 'lnpindex' = 5 + 'a1'*'lnp1' + 'a2'*'lnp2' + ///
                        'a3'*'lnp3' + 'a4'*'lnp4'
            forvalues i = 1/4 {
            forvalues j = 1/4 {
                    replace 'lnpindex' = 'lnpindex' + ///
                                    0.5*'g'i''j''*'lnp'i''*'lnp'j'',
            }
        }
        replace 'w1' = 'a1' + 'g11'*'lnp1' + 'g12'*'lnp2' + ///
                                'g13'*'lnp3' + 'g14'*'lnp4' + ///
                                'b1'*('lnm' - 'lnpindex')
        replace 'w2' = 'a2' + 'g21'*'lnp1' + 'g22'*'lnp2' + ///
        'g23'*'lnp3' + 'g24'*'lnp4' + ///
        'b2'*('lnm' - 'lnpindex')
        replace 'w3' = 'a3' + 'g31'*'lnp1' + 'g32'*'lnp2' + ///
        'g33'*'lnp3' + 'g34'*'lnp4' + ///
                        'b3'*('lnm' - 'lnpindex')
}
end
```

The syntax statement accepts eight variables: three expenditure share variables, all four log-price variables, and a variable for $\log$ expenditures $(\ln m)$. Most of the code simply extracts the parameters
from the 'at' matrix. Although we are estimating only 12 parameters, to calculate the price index term and the expenditure share equations, we need the restricted parameters as well. Notice how we impose the constraints on the parameters. We then created a temporary variable to hold $\ln P(\mathbf{p})$, and we filled the three dependent variables with the predicted expenditure shares.

To fit our model, we type

```
. use https://www.stata-press.com/data/r18/food
(Four food groups)
. nlsur aids @ w1 w2 w3 lnp1 lnp2 lnp3 lnp4 lnexp,
> parameters(a1 a2 a3 b1 b2 b3
> g11 g12 g13 g22 g32 g33)
> neq(3) ifgnls
(obs = 4,048)
Calculating NLS estimates:
Iteration 0: Residual SS = 126.9713
Iteration 1: Residual SS = 125.669
Iteration 2: Residual SS = 125.669
Iteration 3: Residual SS = 125.669
Iteration 4: Residual SS = 125.669
Calculating FGNLS estimates:
Iteration 0: Scaled RSS = 12080.14
Iteration 1: Scaled RSS = 12080.14
Iteration 2: Scaled RSS = 12080.14
Iteration 3: Scaled RSS = 12080.14
FGNLS iteration 2:
Iteration 0: Scaled RSS = 12143.99
Iteration 1: Scaled RSS = 12143.99
Iteration 2: Scaled RSS = 12143.99
Parameter change = 1.97e-04
Covariance matrix change = 2.94e-06
FGNLS iteration 3:
Iteration 0: Scaled RSS = 12144
Iteration 1: Scaled RSS = 12144
Parameter change = 2.18e-06
Covariance matrix change = 3.47e-08
FGNLS regression
```

|  | Equation | Obs | Parms | RMSE | R-sq | Constant |
| :--- | ---: | ---: | :---: | :---: | :---: | :---: |
| 1 | w1 | 4,048 | . | .1333175 | $0.9017 *$ | (none) |
| 2 | w2 | 4,048 | . | .1024166 | $0.8480 *$ | (none) |
| 3 | w3 | 4,048 | . | .053777 | $0.7906 *$ | (none) |

* Uncentered R-sq

|  | Coefficient | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| /a1 | .3163958 | .0073871 | 42.83 | 0.000 | .3019175 | .3308742 |
| /a2 | .2712501 | .0056938 | 47.64 | 0.000 | .2600904 | .2824097 |
| /a3 | .1039898 | .0029004 | 35.85 | 0.000 | .0983051 | .1096746 |
| /b1 | .0161044 | .0034153 | 4.72 | 0.000 | .0094105 | .0227983 |
| /b2 | -.0260771 | .002623 | -9.94 | 0.000 | -.0312181 | -.0209361 |
| /b3 | .0014538 | .0013776 | 1.06 | 0.291 | -.0012463 | .004154 |
| /g11 | .1215838 | .0057186 | 21.26 | 0.000 | .1103756 | .1327921 |
| /g12 | -.0522943 | .0039305 | -13.30 | 0.000 | -.0599979 | -.0445908 |
| /g13 | -.0351292 | .0021788 | -16.12 | 0.000 | -.0393996 | -.0308588 |
| /g22 | .0644298 | .0044587 | 14.45 | 0.000 | .0556909 | .0731687 |
| /g32 | -.0011786 | .0019767 | -0.60 | 0.551 | -.0050528 | .0026957 |
| /g33 | .0424381 | .0017589 | 24.13 | 0.000 | .0389909 | .0458854 |

To get the restricted parameters for the fourth share equation, we can use lincom. For example, to obtain $\alpha_{4}$, we type

|  | Coefficient | Std. err. | z | $P>\|z\|$ | [95\% conf | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| (1) | . 3083643 | . 0052611 | 58.61 | 0.000 | . 2980528 | . 3186758 |

For more information on lincom, see [R] lincom.

## Stored results

nlsur stores the following in e():
Scalars

```
    e(N)
    e(k)
    e(k_#)
    e(k_eq)
    e(k_eq_model)
    e(n_eq)
    e(mss_#)
    e(rss_#)
    e(rmse_#)
    e(r2_#)
    e(ll)
    e(N_clust)
    e(rank)
    e(converged)
```

Macros
$e$ (cmd)
e(cmdline)
e(method)
e(depvar)
e(depvar_\#)
e(wtype)
e(wexp)
e(title)
e(title_2)
e(clustvar)
e(vce)
e(vcetype)
e(type)
e(sexpprog)
e(sexp_\#)
e(params)
e(params_\#)
e(funcprog)
e(rhs)
e(constants)
e(properties)
e(predict)
number of observations
number of parameters
number of parameters for equation \#
number of equation names in $e(b)$
number of equations in overall model test
number of equations
model sum of squares for equation \#
RSS for equation \#
root mean squared error for equation \#
$R^{2}$ for equation \#
Gaussian log likelihood (iflgs version only)
number of clusters
rank of $e(V)$
1 if converged, 0 otherwise
nlsur
command as typed
fgnls, ifgnls, or nls
names of dependent variables
dependent variable for equation \#
weight type
weight expression
title in estimation output
secondary title in estimation output
name of cluster variable
vcetype specified in vce()
title used to label Std. err.
1 = interactively entered expression
$2=$ substitutable expression program
$3=$ function evaluator program
substitutable expression program
substitutable expression for equation \#
names of all parameters
parameters in equation \#
function evaluator program
contents of variables()
identifies constant terms
b V
program used to implement predict

Matrices
e(b) coefficient vector
e(init) initial values vector
e(Sigma) error covariance matrix ( $\widehat{\boldsymbol{\Sigma}}$ )
e(V) variance-covariance matrix of the estimators
Functions
e(sample) marks estimation sample
In addition to the above, the following is stored in r() :
Matrices
$r$ (table) matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

Write the system of equations for the $i$ th observation as

$$
\begin{equation*}
\mathbf{y}_{i}=\mathbf{f}\left(\mathbf{x}_{i}, \boldsymbol{\beta}\right)+\mathbf{u}_{i} \tag{1}
\end{equation*}
$$

where $\mathbf{y}_{i}$ and $\mathbf{u}_{i}$ are $1 \times M$ vectors, for $i=1, \ldots, N ; \mathbf{f}$ is a function that returns a $1 \times M$ vector; $\mathbf{x}_{i}$ represents all the exogenous variables in the system; and $\boldsymbol{\beta}$ is a $1 \times k$ vector of parameters. The generalized nonlinear least-squares system estimator is defined as

$$
\widehat{\boldsymbol{\beta}} \equiv \operatorname{argmin}_{\boldsymbol{\beta}} \sum_{i=1}^{N}\left\{\mathbf{y}_{i}-\mathbf{f}\left(\mathbf{x}_{i}, \boldsymbol{\beta}\right)\right\} \boldsymbol{\Sigma}^{-1}\left\{\mathbf{y}_{i}-\mathbf{f}\left(\mathbf{x}_{i}, \boldsymbol{\beta}\right)\right\}^{\prime}
$$

where $\boldsymbol{\Sigma}=E\left(\mathbf{u}_{i}^{\prime} \mathbf{u}_{i}\right)$ is an $M \times M$ positive-definite weight matrix. Let $\mathbf{T}$ be the Cholesky decomposition of $\boldsymbol{\Sigma}^{-1}$; that is, $\mathbf{T T}^{\prime}=\boldsymbol{\Sigma}^{-1}$. Postmultiply (1) by $\mathbf{T}$ :

$$
\begin{equation*}
\mathbf{y}_{i} \mathbf{T}=\mathbf{f}\left(\mathbf{x}_{i}, \boldsymbol{\beta}\right) \mathbf{T}+\mathbf{u}_{i} \mathbf{T} \tag{2}
\end{equation*}
$$

Because $E\left(\mathbf{T}^{\prime} \mathbf{u}_{i}^{\prime} \mathbf{u}_{i} \mathbf{T}\right)=\mathbf{I}$, we can "stack" the columns of (2) and write

$$
\begin{align*}
\mathbf{y}_{1} \mathbf{T}_{1} & =\mathbf{f}\left(\mathbf{x}_{1}, \boldsymbol{\beta}\right) \mathbf{T}_{1}+\widetilde{u}_{11} \\
\mathbf{y}_{1} \mathbf{T}_{2} & =\mathbf{f}\left(\mathbf{x}_{1}, \boldsymbol{\beta}\right) \mathbf{T}_{2}+\widetilde{u}_{12} \\
\vdots & =\vdots \\
\mathbf{y}_{1} \mathbf{T}_{M} & =\mathbf{f}\left(\mathbf{x}_{1}, \boldsymbol{\beta}\right) \mathbf{T}_{M}+\widetilde{u}_{1 M} \\
\vdots & =\vdots  \tag{3}\\
\mathbf{y}_{N} \mathbf{T}_{1} & =\mathbf{f}\left(\mathbf{x}_{N}, \boldsymbol{\beta}\right) \mathbf{T}_{1}+\widetilde{u}_{N 1} \\
\mathbf{y}_{N} \mathbf{T}_{2} & =\mathbf{f}\left(\mathbf{x}_{N}, \boldsymbol{\beta}\right) \mathbf{T}_{2}+\widetilde{u}_{N 2} \\
\vdots & =\vdots \\
\mathbf{y}_{N} \mathbf{T}_{M} & =\mathbf{f}\left(\mathbf{x}_{N}, \boldsymbol{\beta}\right) \mathbf{T}_{M}+\widetilde{u}_{N M}
\end{align*}
$$

where $\mathbf{T}_{j}$ denotes the $j$ th column of $\mathbf{T}$. By construction, all $\widetilde{u}_{i j}$ are independently distributed with unit variance. As a result, by transforming the model in (1) to that shown in (3), we have reduced the multivariate generalized nonlinear least-squares system estimator to a univariate nonlinear least-squares problem; and the same parameter estimation technique used by nl can be used here. See [R] nl for the details. Moreover, because the $\widetilde{u}_{i j}$ all have variance 1, the final scaled RSS reported by nlsur is equal to $N M$.

To make the estimator feasible, we require an estimate $\widehat{\boldsymbol{\Sigma}}$ of $\boldsymbol{\Sigma}$. nl sur first sets $\widehat{\boldsymbol{\Sigma}}=\mathbf{I}$. Although not efficient, the resulting estimate, $\widehat{\boldsymbol{\beta}}_{\mathrm{NLS}}$, is consistent. If the nls option is specified, estimation is complete. Otherwise, the residuals

$$
\widehat{\mathbf{u}}_{i}=\mathbf{y}_{i}-\mathbf{f}\left(\mathbf{x}_{i}, \widehat{\boldsymbol{\beta}}_{\mathrm{NLS}}\right)
$$

are calculated and used to compute

$$
\widehat{\boldsymbol{\Sigma}}=\frac{1}{N} \sum_{i=1}^{N} \widehat{\mathbf{u}}_{i}^{\prime} \widehat{\mathbf{u}}_{i}
$$

With $\widehat{\boldsymbol{\Sigma}}$ in hand, a new estimate $\widehat{\boldsymbol{\beta}}$ is then obtained.
If the ifgnls option is specified, the new $\widehat{\boldsymbol{\beta}}$ is used to recompute the residuals and obtain a new estimate of $\widehat{\boldsymbol{\Sigma}}$, from which $\widehat{\boldsymbol{\beta}}$ can then be reestimated. Iterations stop when the relative change in $\widehat{\boldsymbol{\beta}}$ is less than eps (), the relative change in $\widehat{\boldsymbol{\Sigma}}$ is less than ifgnlseps(), or if ifgnlsiterate() iterations have been performed.

If the vce(robust) and vce(cluster clustvar) options were not specified, then

$$
V(\widehat{\boldsymbol{\beta}})=\left(\sum_{i=1}^{N} \mathbf{X}_{i}^{\prime} \widehat{\boldsymbol{\Sigma}}^{-1} \mathbf{X}_{i}\right)^{-1}
$$

where the $M \times k$ matrix $\mathbf{X}_{i}$ has typical element $X_{i s t}$, the derivative of the $s$ th element of $\mathbf{f}$ with respect to the $t$ th element of $\boldsymbol{\beta}$, evaluated at $\mathbf{x}_{i}$ and $\widehat{\boldsymbol{\beta}}$. As a practical matter, once the model is written in the form of (3), the variance-covariance matrix can be calculated via a Gauss-Newton regression; see Davidson and MacKinnon (1993, chap. 6).

If robust is specified, then

$$
V_{R}(\widehat{\boldsymbol{\beta}})=\left(\sum_{i=1}^{N} \mathbf{X}_{i}^{\prime} \widehat{\boldsymbol{\Sigma}}^{-1} \mathbf{X}_{i}\right)^{-1} \sum_{i=1}^{N} \mathbf{X}_{i}^{\prime} \widehat{\boldsymbol{\Sigma}}^{-1} \widehat{\mathbf{u}}_{i}^{\prime} \widehat{\mathbf{u}}_{i} \widehat{\boldsymbol{\Sigma}}^{-1} \mathbf{X}_{i}\left(\sum_{i=1}^{N} \mathbf{X}_{i}^{\prime} \widehat{\boldsymbol{\Sigma}}^{-1} \mathbf{X}_{i}\right)^{-1}
$$

The cluster-robust variance matrix is

$$
V_{C}(\widehat{\boldsymbol{\beta}})=\left(\sum_{i=1}^{N} \mathbf{X}_{i}^{\prime} \hat{\boldsymbol{\Sigma}}^{-1} \mathbf{X}_{i}\right)^{-1} \sum_{c=1}^{N_{C}} \mathbf{w}_{c} \mathbf{w}_{c}^{\prime}\left(\sum_{i=1}^{N} \mathbf{X}_{i}^{\prime} \widehat{\boldsymbol{\Sigma}}^{-1} \mathbf{X}_{i}\right)^{-1}
$$

where $N_{C}$ is the number of clusters and

$$
\mathbf{w}_{c}=\sum_{j \in C_{k}} \mathbf{X}_{j}^{\prime} \widehat{\boldsymbol{\Sigma}}^{-1} \widehat{\mathbf{u}}_{j}^{\prime}
$$

with $C_{k}$ denoting the set of observations in the $k$ th cluster. In evaluating these formulas, we use the value of $\widehat{\boldsymbol{\Sigma}}$ used in calculating the final estimate of $\widehat{\boldsymbol{\beta}}$. That is, we do not recalculate $\widehat{\boldsymbol{\Sigma}}$ after we obtain the final value of $\widehat{\boldsymbol{\beta}}$.

The RSS for the $j$ th equation, $\operatorname{RSS}_{j}$, is

$$
\operatorname{RSS}_{j}=\sum_{i=1}^{N}\left(\widehat{y}_{i j}-y_{i j}\right)^{2}
$$

where $\widehat{y}_{i j}$ is the predicted value of the $i$ th observation on the $j$ th dependent variable; the total sum of squares (TSS) for the $j$ th equation, $\operatorname{TSS}_{j}$, is

$$
\operatorname{TSS}_{j}=\sum_{i=1}^{N}\left(y_{i j}-\bar{y}_{j}\right)^{2}
$$

if there is a constant term in the $j$ th equation, where $\bar{y}_{j}$ is the sample mean of the $j$ th dependent variable, and

$$
\operatorname{TSS}_{j}=\sum_{i=1}^{N} y_{i j}^{2}
$$

if there is no constant term in the $j$ th equation; and the model sum of squares (MSS) for the $j$ th equation, $\mathrm{MSS}_{j}$, is $\mathrm{TSS}_{j}-\mathrm{RSS}_{j}$.

The $R^{2}$ for the $j$ th equation is $\operatorname{MSS}_{j} / \operatorname{TSS}_{j}$. If an equation does not have a constant term, then the reported $R^{2}$ for that equation is "uncentered" and based on the latter definition of $\operatorname{TSS}_{j}$.

Under the assumption that the $\mathbf{u}_{i}$ are independent and identically distributed $N(\mathbf{0}, \widehat{\boldsymbol{\Sigma}})$, the $\log$ likelihood for the model is

$$
\ln L=-\frac{M N}{2}\{1+\ln (2 \pi)\}-\frac{N}{2} \ln |\widehat{\boldsymbol{\Sigma}}|
$$

The log likelihood is reported only when the ifgnls option is specified.

## References

Canette, I. 2011. A tip to debug your $\mathrm{nl} / \mathrm{nlsur}$ function evaluator program. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2011/12/05/a-tip-to-debug-your-nlnlsur-function-evaluator-program/.
Davidson, R., and J. G. MacKinnon. 1993. Estimation and Inference in Econometrics. New York: Oxford University Press.

Deaton, A. S., and J. Muellbauer. 1980. An almost ideal demand system. American Economic Review 70: 312-326.
Greene, W. H. 1997. Econometric Analysis. 3rd ed. Upper Saddle River, NJ: Prentice Hall.
—_. 2012. Econometric Analysis. 7th ed. Upper Saddle River, NJ: Prentice Hall.

Poi, B. P. 2012. Easy demand-system estimation with quaids. Stata Journal 12: 433-446.
Zellner, A. 1962. An efficient method of estimating seemingly unrelated regressions and tests for aggregation bias. Journal of the American Statistical Association 57: 348-368. https://doi.org/10.2307/2281644.
1963. Estimators for seemingly unrelated regression equations: Some exact finite sample results. Journal of the American Statistical Association 58: 977-992. https://doi.org/10.2307/2283326.
Zellner, A., and D. S. Huang. 1962. Further properties of efficient estimators for seemingly unrelated regression equations. International Economic Review 3: 300-313. https://doi.org/10.2307/2525396.

## Also see <br> [R] nlsur postestimation - Postestimation tools for nlsur <br> [R] nl - Nonlinear least-squares estimation <br> [R] demandsys - Estimation of flexible demand systems <br> [R] gmm - Generalized method of moments estimation <br> [R] ml - Maximum likelihood estimation <br> [R] mlexp - Maximum likelihood estimation of user-specified expressions <br> [R] reg3 - Three-stage estimation for systems of simultaneous equations <br> [R] sureg - Zellner's seemingly unrelated regression <br> [U] 20 Estimation and postestimation commands

## Title

nlsur postestimation - Postestimation tools for nlsur
Postestimation commands predict margins Remarks and examples

## Postestimation commands

The following postestimation commands are available after nlsur:

| Command | Description |
| :---: | :---: |
| estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estimates | cataloging estimation results |
| etable | table of estimation results |
| forecast | dynamic forecasts and simulations |
| hausman | Hausman's specification test |
| lincom | point estimates, standard errors, testing, and inference for linear combinations of coefficients |
| lrtest | likelihood-ratio test |
| * margins | marginal means, predictive margins, marginal effects, and average marginal effects |
| marginsplot | graph the results from margins (profile plots, interaction plots, etc.) |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients |
| predict | fitted values, residuals, etc. |
| predictnl | point estimates, standard errors, testing, and inference for generalized predictions |
| test | Wald tests of simple and composite linear hypotheses |
| testnl | Wald tests of nonlinear hypotheses |

[^39]
## predict

## Description for predict

predict creates a new variable containing predictions such as fitted values and residuals.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

$$
\text { predict }[\text { type }] \text { newvar }[\text { if }][\text { in }][\text {, equation(\#eqno) yhat residuals }]
$$

These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

## Options for predict

$\qquad$ Main
equation(\#eqno) specifies to which equation you are referring. equation(\#1) would mean that the calculation is to be made for the first equation, equation(\#2) would mean the second, and so on. If you do not specify equation(), results are the same as if you had specified equation(\#1). yhat, the default, calculates the fitted values for the specified equation.
residuals calculates the residuals for the specified equation.

## margins

## Description for margins

margins estimates margins of response for fitted values.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [options]
```

statistic Description

| yhat | fitted values; the default |
| :--- | :--- |
| $\underline{\underline{r}}$ esiduals | not allowed with margins |

Statistics not allowed with margins are functions of stochastic quantities other than e(b). For the full syntax, see [R] margins.

## Remarks and examples

> Example 1
In example 2 of $[\mathrm{R}]$ nlsur, we fit a four-factor translog cost function to data for the US economy. The own-price elasticity for a factor measures the percentage change in its usage as a result of a $1 \%$ increase in the factor's price, assuming that output is held constant. For the translog production function, the own-price factor elasticities are

$$
\eta_{i}=\frac{\delta_{i i}+s_{i}\left(s_{i}-1\right)}{s_{i}}
$$

Here we compute the elasticity for capital at the sample mean of capital's factor share. First, we use summarize to get the mean of $\mathrm{s} \_\mathrm{k}$ and store that value in a scalar:

```
. use https://www.stata-press.com/data/r18/mfgcost
(Manufacturing cost)
. nlsur (s_k = {bk} + {dkk}*ln(pk/pm) + {dkl}*ln(pl/pm) + {dke}*ln(pe/pm))
> (s_l = {bl} + {dkl}*ln(pk/pm) + {dll}*ln(pl/pm) + {dle}*ln(pe/pm))
> (s_e = {be} + {dke}*ln(pk/pm) + {dle}*ln(pl/pm) + {dee}*ln(pe/pm)),
> ifgnls
    (output omitted)
. summarize s_k
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline s_k & 25 & .053488 & .0044795 & .04602 & .06185
\end{tabular}
. scalar kmean = r(mean)
```

Now, we can use nlcom to calculate the elasticity:

```
. nlcom (([dkk]_cons + kmean*(kmean-1)) / kmean)
    _nl_1: ([dkk]_cons + kmean*(kmean-1)) / kmean
```

|  | Coefficient | Std. err. | $z$ | P>\|z| | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| _nl_1 | -.3952986 | .1083535 | -3.65 | 0.000 | -.6076676 | -.1829295 |

If the price of capital increases by $1 \%$, its usage will decrease by about $0.4 \%$. To maintain its current level of output, a firm would increase its usage of other inputs to compensate for the lower capital usage. The standard error reported by nlcom reflects the sampling variance of the estimated parameter $\widehat{\delta_{k k}}$, but nlcom treats the sample mean of $\mathrm{s} \_\mathrm{k}$ as a fixed parameter that does not contribute to the sampling variance of the estimated elasticity.

## Also see

[R] nlsur - Estimation of nonlinear systems of equations
[U] 20 Estimation and postestimation commands

## Title

npregress intro - Introduction to nonparametric regression
Description Remarks and examples References Also see

## Description

Nonparametric regression models the mean of an outcome given the covariates without making assumptions about its functional form. This makes nonparametric regression estimates robust to functional form misspecification. npregress implements the two most common nonparametric regression estimators: series regression and kernel regression.

Nonparametric series estimation regresses the outcome on a function of the covariates. The function of the covariates is known as a basis function. A basis is a collection of terms that approximates smooth functions arbitrarily well. A basis function includes a subset of these terms. The bases used by npregress series are polynomials, piecewise polynomial splines, and B-splines.

Nonparametric kernel estimation computes a weighted average of the outcome. The weights are functions called kernels, which give rise to the name of the method. npregress kernel performs local-linear and local-constant kernel regression.

Whether we choose to approximate the mean of our outcome using series regression or kernel regression, we obtain estimates that are robust to assumptions about functional form. This robustness comes at a cost; we need many observations and perhaps a long computation time to estimate the elements of the approximating function.

This entry introduces the intuition behind the nonparametric regression estimators implemented in npregress. If you are familiar with these methods, you may want to skip to [ R ] npregress kernel or $[\mathrm{R}]$ npregress series.

## Remarks and examples

Remarks are presented under the following headings:

> Overview
> Nonparametric series regression
> $\quad$ Runge's phenomenon
> $\quad$ Piecewise polynomial splines and B-splines
> Nonparametric kernel regression
> Limitations of nonparametric methods

## Overview

Nonparametric regression is used when we are uncertain about the functional form of the mean of the outcome given the covariates. For example, when we estimate a linear regression, we assume that the functional form for the mean of the outcome is a linear combination of the specified covariates. Both parametric (linear) regression and nonparametric regression provide an estimate of the mean for the different values of the covariates. Consider the simulated data in figure 1 . The mean of the outcome for all values of x is overlaid on these points.

Data and true mean


Figure 1.
Because the mean of the data in figure 1 is not linear in x , using a simple linear regression will not give us a correct picture about the effect of covariate x on the outcome. For example, if we perform a linear regression of the outcome on x for the data, we obtain the plot shown in figure 2.

Linear regression estimates


Figure 2.
The change in the predicted outcome when $x$ changes is positive and constant, yet the true mean is nonlinear. If the assumption about the functional form of the mean is incorrect, the estimates we obtain are inconsistent. If we instead fit the model using npregress and graph the estimates, we obtain figure 3.


Figure 3.
npregress gives us the correct relationship between the outcome and the covariates. The nonparametric regression estimates are consistent as long as the true function is sufficiently smooth. If the linear regression assumptions are true, nonparametric regression is still consistent but less efficient.

Although nonparametric regression is a way to obtain estimates that are robust to functional form misspecification, this robustness comes at a cost. You need many observations and more time to compute the estimates. The cost increases with the number of covariates; this is referred to as the curse of dimensionality.

## Nonparametric series regression

The basis and the basis function are concepts essential to understanding series regression. A basis is a collection of terms that can approximate a smooth function arbitrarily well. A basis function uses a subset of these terms to approximate the mean function. npregress series allows you to use a polynomial basis, a piecewise polynomial spline basis, or a B-spline basis. For each basis, npregress series selects the basis function for you.

We use an example to illustrate the use of a basis and a basis function. Suppose a researcher has data on the outcome $y$ and a covariate $x$. We plot their relationship in the figure 4 below.


Figure 4.
In this case, a regression of $y$ on $x$ will do a good job of approximating the true function. If our data looked like the data in figure 5 , however, a regression of $y$ on $x$ would be inadequate.


Figure 5.
In this case, a regression of $y$ on $x$ and $x^{2}$ is more appropriate.
In each case, we include terms from a polynomial basis. In the first case, we need a constant and the linear term $x$. In the second case, we need a constant, the linear term $x$, and the quadratic term $x^{2}$. A more complex function would require a basis function that includes more terms from the polynomial basis.

If we want to use a polynomial basis function, npregress will select a degree of the polynomial for us. Additional terms reduce bias but increase the variance of the estimator. npregress will select the terms that optimally tradeoff bias and variance. In other words, npregress selects a basis function that includes the terms that minimize the mean squared error. Our example above used a polynomial basis function, but npregress can also select terms from a piecewise polynomial spline or B-spline basis.

## Runge's phenomenon

Polynomials are the most intuitive basis but not the preferred basis for nonparametric series estimation. The reason is that they are poor at interpolating. This problem shows up at the boundaries of the support of the covariates, where, as you increase the order of the polynomial, the polynomial approximation oscillates frequently, even when the true function does not behave this way.

Let us demonstrate. Below is an example for which we model a mean function using a third-order polynomial. We plot the data and the estimate of the mean function:


Figure 6.
Looking at the data, it appears that a higher-order polynomial would be a better fit for the data. Below is the mean function we get using a sixth-order and a tenth-order polynomial:


Figure 7.
The predictions improve at values near the middle of the range of $x$ but become more variable at the edges of the parameter space.

What we illustrated above is referred to as Runge's phenomenon. Increasing the complexity of the polynomial order did not improve our approximation. In fact, as we increased the polynomial order, the behavior at the edges of the parameter space became more variable. The way to address this is to use a basis that does a better job of interpolating: piecewise polynomial splines or B-splines.

## Piecewise polynomial splines and B-splines

Piecewise polynomial splines and B-splines are preferred to a polynomial basis because they are better at approximation. We discuss piecewise polynomial splines to provide intuition for both the piecewise polynomial spline basis and the B -spline basis.

Low-order polynomials do a great job of approximating functions in regions where the true function does not change too much. Splines continuously connect a set of low-order polynomials to create a basis to approximate a smooth function. The graph below illustrates what this definition means. We show in maroon a piecewise polynomial spline estimate of the mean function for the data in the example above.


Figure 8.
To see that splines are better than polynomials, note that the spline approximation of the mean function fits the data well and that there are no regions where the approximation wiggles wildly.

Now, we delve into the definition above. The vertical lines in the graph partition the support of $x$ into subregions. The piecewise polynomial spline basis allows for a different low-order polynomial in each subregion, and it forces the polynomials in neighboring regions to be continuously connected. In figure 8 above, the basis used is a third-order polynomial in each subregion. The graph illustrates that the polynomials are smoothly connected at the subregion boundaries. The subregion boundaries are known as the knot points, or just the knots, because they are where the different polynomials are tied together.

By default, npregress selects the number of knots for you. Alternatively, you may specify the number of knots yourself.

We now look at how the mean function at each region was computed. We show this mathematically and graphically.

Defining the seven knots as $t_{1}, \ldots, t_{7}$, where $t_{1}<t_{2}<\cdots<t_{6}<t_{7}$, the third-order piecewise polynomial spline estimate is given by

$$
\widehat{E}\left(y_{i} \mid x_{i}\right)=\widehat{\beta}_{0}+\widehat{\beta}_{1} x_{i}+\widehat{\beta}_{2} x_{i}^{2}+\widehat{\beta}_{3} x_{i}^{3}+\sum_{j=1}^{7} \beta_{j+3} \max \left(x_{i}-t_{j}, 0\right)^{3}
$$

Thus, for all $x_{i}$ that are less than the smallest knot, $t_{1}$, the mean estimate is given by the third-order polynomial

$$
\widehat{E}\left(y_{i} \mid x_{i} \leq t_{1}\right)=\widehat{\beta}_{0}+\widehat{\beta}_{1} x_{i}+\widehat{\beta}_{2} x_{i}^{2}+\widehat{\beta}_{3} x_{i}^{3}
$$

Here it is graphically:


Figure 9.
Likewise, if $x$ is less than the second knot, $t_{2}$, then the mean estimate for that region is different if $x_{i}>t_{1}$ than if $x_{i} \leq t_{1}$, and is given by

$$
\widehat{E}\left(y_{i} \mid x_{i} \leq t_{2}\right)=\widehat{\beta}_{0}+\widehat{\beta}_{1} x_{i}+\widehat{\beta}_{2} x_{i}^{2}+\widehat{\beta}_{3} x_{i}^{3}+\beta_{4}\left(x_{i}-t_{1}\right)^{3}\left(x_{i}>t_{1}\right)
$$

Here it is graphically:


Figure 10.

As $x$ increases, there are additional contributions from each subregion. If we continue plotting the resulting mean estimates, the following graphs would be what we would obtain:


Figure 11.
This example illustrates how the terms in the spline basis approximate the mean function. Both the graph of the estimated function and the intuition in the example illustrate why the spline basis is better than the polynomial basis.

In the examples above, we used a piecewise polynomial spline basis. Specifically, we used a third-order piecewise polynomial spline basis function to obtain our estimates of the conditional mean. We could have also used second-order or first-order piecewise polynomial splines, where the order of the splines is defined by the order of the polynomial terms in the covariates used in each subregion.

As mentioned before, piecewise polynomial splines are preferred to a polynomial basis because they are better at approximation. However, piecewise polynomial splines also have some issues. In particular, they can be highly collinear and therefore numerically unstable. You can see this in the regions delineated in figure 11 , which are defined by terms of the form $\max \left(x_{i}-t_{j}, 0\right)$ that may overlap.

B-splines avoid this problem, so each term that goes into the conditional mean approximation is orthogonal. It is for this reason that B-splines are the default basis for npregress series. However, the intuition we obtain from piecewise polynomial splines and B-splines is equivalent. In fact, B-spline and piecewise polynomial spline bases can approximate the same functions. For a more detailed explanation of B-splines, see Methods and formulas in [R] npregress series.

In this section, we provided an intuitive and brief introduction to nonparametric series estimation. For detailed introductions to series estimators and the methods implemented by npregress series, see de Boor (2001), Schumaker (2007), Eubank (1999), Schoenberg (1969), Newey (1997), and Chen (2007).

## Nonparametric kernel regression

npregress kernel approximates the mean by using a kernel function. In Overview, we plotted the following data and nonparametric estimate of the mean function:

Data and nonparametric estimate


Figure 12.
We used kernel regression to estimate this mean function. With this method, we estimate the mean of the outcome at different values of the covariate x . In this section, we build our intuition for how kernel regression estimates these means, and we demonstrate this graphically.

Suppose covariate x is discrete. In this case, a consistent estimator of the mean of outcome y given that $\mathrm{x}=a$ is the average of the values of y for which x is equal to a given value $a$. For instance, the sample average of the yearly income for married individuals is a consistent estimator for the population mean yearly income for married individuals.

Now, consider estimating the mean of y given that $\mathrm{x}=a$ when x is continuous and $a$ is a value observed for x . Because x is continuous, the probability of any observed value being exactly equal to $a$ is 0 . Therefore, we cannot compute an average for the values of y for which x is equal to a given value $a$. We use the average of y for the observations in which x is close to $a$ to estimate the mean of y given that $\mathrm{x}=a$. Specifically, we use the observations for which $|\mathrm{x}-a|<h$, where $h$ is small. The parameter $h$ is called a bandwidth. In nonparametric kernel regression, a bandwidth determines the amount of information we use to estimate the conditional mean at each point $a$. We demonstrate how this works graphically.

For the simulated data in our example, we choose $h=0.25$ and $a=-0.19$. The vertical lines in figure 13 delimit the values of x around $a$ for which we are computing the mean of y . The light blue square is our estimate of the conditional mean using the observations between the vertical lines.


Figure 13.
Repeating this estimation when $a=2.66$ produces figure 14 .
Nonparametric estimates at two points


Figure 14.

Doing this estimation for each point in our data produces a nonparametric estimate of the mean for a given value of the covariates (see figure 15).


Figure 15.
The plotted blue squares in figure 15 form what is known as the conditional mean function. Because these are simulated data, we can compare our estimate with the true conditional mean function, a comparison we show in figure 16.


Figure 16.
We see that the estimate is a bit less smooth than the true function. The size of the bandwidth $h$ determines the shape and smoothness of the estimated conditional mean function, because the bandwidth defines how many observations around each point are used. For example, if $h$ is arbitrarily large-say, $h=300$-then we get figure 17 .


Figure 17.
In this case, all observations are used to estimate the conditional mean at each point, and the estimate is therefore a constant. On the other hand, a too-small bandwidth produces a jagged function with high variability, as illustrated in figure 18.


Figure 18.
The optimal bandwidth is somewhere in between. A too-large bandwidth includes too many observations, so the estimate is biased but it has a low variance. A too-small bandwidth includes too few observations, so the estimate has little bias but the variance is large. In other words, the optimal bandwidth trades off bias and variance. In the case of npregress kernel, the bandwidth is chosen to minimize the cost of this tradeoff by using either cross-validation, as suggested by Li and Racine (2004), or an improved Akaike information criterion proposed by Hurvich, Simonoff, and Tsai (1998).

How we average the observations around a point is also important. In the examples above, we gave the same weight to each observation for which $|\mathrm{x}-a|<h$. However, we might weight each observation differently. The weights that observations receive are determined by functions called kernels. We could have used any of the weights in [R] kdensity. For a nice introduction to kernel weighting, see Silverman (1986).

The estimator described above uses only nearby observations and is thus a local estimator. It uses a sample average, which is a regression on a constant, and is thus a locally constant estimator. For these reasons, the estimator described above fits what is known as a local-constant regression.

The generalization that uses the prediction from a local-linear regression on covariates is known as local-linear regression. Local-linear regression estimates the derivative of the conditional mean function in addition to the function itself. Understanding how the conditional mean changes when covariates change is sometimes the research question of interest, for example, how income changes for different levels of taxes. Local-linear regression provides an estimate for these changes for continuous and discrete variables.

See Fan and Gijbels (1996) and Li and Racine (2007) for detailed introductions to the kernel estimators implemented in npregress kernel.

## Limitations of nonparametric methods

As discussed above, series regression and kernel regression approximate an unknown mean function. Series regression uses least squares on the basis function. Kernel regression uses a kernel-weighted average of nearby observations.

Series estimators are considered to be global estimators because they approximate the mean function at each point using the value of one overall approximating function. Kernel regression is considered a local estimator because it only uses nearby observations to approximate the mean for a given covariate pattern.

Although piecewise polynomial splines and B-splines are considered to be global estimators, in fact, they are local estimators. They are local because they fit a polynomial in each region defined by the knots. Like kernel estimators, piecewise polynomial spline and B-spline estimators require that there are enough data in each region. Suppose our data look like the data below.


Figure 19.

Using a method to select knots optimally at percentiles of the data will give us figure 20.


Figure 20.
The vertical line denotes the point at which the knot is placed. The blue line is the B-spline estimate, and the dotted green line is the true mean function. We see that our estimate of the mean function is not good, especially for higher positive values of the covariate. The reason is that data are sparse for these values. An alternative is to place the knots uniformly over the values of x . In this case, our estimate of the mean function improves. However, this does not change the fact that we have regions with insufficient data to make reliable inferences.

Thus, for kernel, piecewise polynomial spline, and B-spline estimators, we must have enough data points for all ranges of the data. In particular, piecewise polynomial spline and B-spline estimates should not be used to predict outside the support of the data.

Another important consideration is model selection. npregress selects the number of terms from a basis for series estimation and the bandwidth for kernel estimation. After model selection, the models are taken as given without accounting for model-selection error. You can find an in-depth discussion and references of some of the issues that arise when performing model selection in [LASSO] Lasso intro.

## References

Cattaneo, M. D., M. Jansson, and X. Ma. 2018. Manipulation testing based on density discontinuity. Stata Journal 18: 234-261.

Chen, X. 2007. Large sample sieve estimation of semi-nonparametric models. In Vol. 6B of Handbook of Econometrics, ed. J. J. Heckman and E. E. Leamer, 5549-5632. Amsterdam: Elsevier. https://doi.org/10.1016/S1573-4412(07)06076-X.
de Boor, C. 2001. A Practical Guide to Splines. Rev. ed. New York: Springer.
Eubank, R. L. 1999. Nonparametric Regression and Spline Smoothing. 2nd ed. New York: Dekker.
Fan, J., and I. Gijbels. 1996. Local Polynomial Modelling and Its Applications. London: Chapman and Hall.
Hansen, B. E. 2009. University of Wisconsin-Madison, ECON 718, NonParametric Econometrics, Spring 2009, course notes. Last visited on 2019/01/15. https://www.ssc.wisc.edu/~bhansen/718/718.htm.
__. 2018. Econometrics. https://www.ssc.wisc.edu/~bhansen/econometrics/Econometrics.pdf.
Hurvich, C. M., J. S. Simonoff, and C.-L. Tsai. 1998. Smoothing parameter selection in nonparametric regression using an improved Akaike information criterion. Journal of the Royal Statistical Society, Series B 60: 271-293. https://doi.org/10.1111/1467-9868.00125.

Li, Q., and J. S. Racine. 2004. Cross-validated local linear nonparametric regression. Statistica Sinica 14: 485-512.
——. 2007. Nonparametric Econometrics: Theory and Practice. Princeton, NJ: Princeton University Press.
Newey, W. K. 1997. Convergence rates and asymptotic normality for series estimators. Journal of Econometrics 79: 147-168. https://doi.org/10.1016/S0304-4076(97)00011-0.
Pinzon, E. 2017. Nonparametric regression: Like parametric regression, but not. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2017/06/27/nonparametric-regression-like-parametric-regression-but-not/.

Schoenberg, I. J., ed. 1969. Approximations with Special Emphasis on Spline Functions. New York: Academic Press.
Schumaker, L. L. 2007. Spline Functions: Basic Theory. 3rd ed. Cambridge: Cambridge University Press.
Silverman, B. W. 1986. Density Estimation for Statistics and Data Analysis. London: Chapman and Hall.

## Also see

[R] npregress kernel - Nonparametric kernel regression
[R] npregress series - Nonparametric series regression
[R] lpoly - Kernel-weighted local polynomial smoothing
[R] kdensity - Univariate kernel density estimation
[ R ] regress - Linear regression

## Title

npregress kernel - Nonparametric kernel regression

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| Acknowledgments | References | Also see |  |

## Description

npregress kernel performs nonparametric local-linear and local-constant kernel regression. Like linear regression, nonparametric regression models the mean of the outcome conditional on the covariates, but unlike linear regression, it makes no assumptions about the functional form of the relationship between the outcome and the covariates. npregress kernel may be used to model the mean of a continuous, count, or binary outcome.

## Quick start

Nonparametric regression of y on x and discrete covariate a using the Epanechnikov kernel for x and the Li -Racine kernel for a

```
npregress kernel y x i.a
```

Same as above, but use 500 replications and compute bootstrap standard errors and percentile confidence intervals
npregress kernel y x i.a, reps (500)
Same as above, but use a Gaussian kernel for x
npregress kernel y x i.a, reps(500) kernel (gaussian)
Same as above, but use the improved AIC to find the optimal bandwidth
npregress kernel y x i.a, reps(500) kernel(gaussian) imaic
Same as above, but additionally specify that only the mean of the outcome be computed
npregress kernel y x i.a, reps(500) kernel (gaussian) imaic noderivatives
Specify h as the vector of bandwidths
npregress kernel y x i.a, bwidth(h)

## Menu

Statistics $>$ Nonparametric analysis $>$ Nonparametric kernel regression

## Syntax

npregress kernel depvar indepvars [if] [in] [, options]
options
Description

## Model

estimator(linear|constant)
kernel (kernel)
dkernel (dkernel)
predict(prspec)
noderivatives
imaic
unidentsample(newvar)
Bandwidth
bwidth (specs)
meanbwidth(specs)
derivbwidth (specs)
SE

* vce (vcetype)
reps (\#)
seed (\#)
bwreplace
Reporting
level(\#)
display_options
citype (citype)

Maximization
maximize_options
coeflegend
use the local-linear or local-constant kernel estimator
kernel density function for continuous covariates
kernel density function for discrete covariates
store predicted values of the mean and derivatives using variable names specified in prspec
suppress derivative computation
use improved AIC instead of cross-validation to compute optimal bandwidth
specify name of variable that marks identification problems
specify kernel bandwidth for all predictions
specify kernel bandwidth for the mean
specify kernel bandwidth for the derivatives
vcetype may be none or bootstrap
equivalent to vce(bootstrap, reps(\#))
set random-number seed to \#; must also specify reps (\#)
vary bandwidth with each bootstrap replication; seldom used
set confidence level; default is level (95)
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
method to compute bootstrap confidence intervals;
default is citype (percentile)
control the maximization process
display legend instead of statistics
indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
bootstrap, by, collect, and jackknife are allowed; see [U] 11.1.10 Prefix commands.

* vce (bootstrap) reports percentile confidence intervals instead of the normal-based confidence intervals reported when vce(bootstrap) is specified with other estimation commands.
coeflegend does not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.


## kernel

Description
epanechnikov
epan2
biweight
cosine
gaussian
parzen
rectangle
triangle

Epanechnikov kernel function; the default alternative Epanechnikov kernel function biweight kernel function cosine trace kernel function Gaussian kernel function Parzen kernel function rectangle kernel function triangle kernel function
dkernel
liracine
cellmean
citype Description
percentile
bc
normal

Description
$\mathrm{Li}-\mathrm{Racine}$ kernel function; the default cell means kernel function
percentile confidence intervals; the default
bias-corrected confidence intervals
normal-based confidence intervals

## Options

estimator (linear |constant) specifies whether the local-constant or local-linear kernel estimator should be used. The default is estimator (linear).
kernel (kernel) specifies the kernel density function for continuous covariates for use in calculating the local-constant or local-linear estimator. The default is kernel(epanechnikov).
dkernel (dkernel) specifies the kernel density function for discrete covariates for use in calculating the local-constant or local-linear estimator. The default is dkernel(liracine); see Methods and formulas for details on the Li-Racine kernel. When dkernel (cellmean) is specified, discrete covariates are weighted by their cell means.
predict(prspec) specifies that npregress kernel store the predicted values for the mean and derivatives of the mean with the specified names. prspec is the following:
predict (varlist|stub* [, replace noderivatives])
The option takes a variable list or a stub. The first variable name corresponds to the predicted outcome mean. The second name corresponds to the derivatives of the mean. There is one derivative for each indepvar.
When replace is used, variables with the names in varlist or stub* are replaced by those in the new computation. If noderivatives is specified, only a variable for the mean is created. This will increase computation speed but will add to the computation burden if you want to obtain marginal effects after estimation.
noderivatives suppresses the computation of the derivatives. In this case, only the mean function is computed.
imaic specifies to use the improved AIC instead of cross-validation to compute optimal bandwidths. unidentsample(newvar) specifies the name of a variable that is 1 if the observation violates the model identification assumptions and is 0 otherwise. By default, this variable is a system variable (_unident_sample).
npregress kernel computes a weighted regression for each observation in our data. An observation violates identification assumptions if the regression cannot be performed at that point. The regression formula, which is discussed in detail in Methods and formulas, is given by

$$
\widehat{\gamma}=\left(\mathbf{Z}^{\prime} \mathbf{W} \mathbf{Z}\right)^{-1} \mathbf{Z}^{\prime} \mathbf{W} \mathbf{y}
$$

npregress kernel verifies that the matrix ( $\mathbf{Z}^{\prime} \mathbf{W Z}$ ) is full rank for each observation to determine identification. Identification problems commonly arise when the bandwidth is too small, resulting in too few observations within a bandwidth. Independent variables that are collinear within the bandwidth can also cause a problem with identification at that point.

Observations that violate identification assumptions are reported as missing for the predicted means and derivatives.

## Bandwidth

bwidth (specs) specifies the half-width of the kernel at each point for the computation of the mean and the derivatives of the mean function. If no bandwidth is specified, one is chosen by minimizing the integrated mean squared error of the prediction.
specs specifies bandwidths for the mean and derivative for each indepvar in one of three ways: by specifying the name of a vector containing the bandwidths (for example, bwidth ( H ), where H is a properly labeled vector); by specifying the equation and coefficient names with the corresponding values (for example, bwidth (Mean:x1=0.5 Effect:x1=0.9) ); or by specifying a list of values for the means, standard errors, and derivatives for indepvars given in the order of the corresponding indepvars and specifying the copy suboption (for example, bwidth(0.5 0.9, copy)).
skip specifies that any parameters found in the specified vector that are not also found in the model be ignored. The default action is to issue an error message.
copy specifies that the list of values or the vector be copied into the bandwidth vector by position rather than by name.
meanbwidth (specs) specifies the half-width of the kernel at each point for the computation of the mean function. If no bandwidth is specified, one is chosen by minimizing the integrated mean squared error of the prediction. For details on how to specify the bandwidth, see the description of bwidth(), above.
derivbwidth (specs) specifies the half-width of the kernel at each point for the computation of the derivatives of the mean. If no bandwidth is specified, one is chosen by minimizing the integrated mean squared error of the prediction. For details on how to specify the bandwidth, see the description of bwidth (), above.

SE
vce (vcetype) specifies the type of standard error reported, which may be either that no standard errors are reported (none; the default) or that bootstrap standard errors are reported (bootstrap); see [R] vce_option.
We recommend that you select the number of replications using reps (\#) instead of specifying vce(bootstrap), which defaults to 50 replications. Be aware that the number of replications needed to produce good estimates of the standard errors varies depending on the problem.

When vce(bootstrap) is specified, npregress kernel reports percentile confidence intervals as recommended by Cattaneo and Jansson (2018) instead of reporting the normal-based confidence intervals that are reported when vce (bootstrap) is specified with other commands. Other types of confidence intervals can be obtained by using the citype (citype) option.
reps(\#) specifies the number of bootstrap replications to be performed. Specifying this option is equivalent to specifying vce(bootstrap, reps(\#)).
seed(\#) sets the random-number seed. You must specify reps(\#) with seed(\#).
bwreplace computes a different bandwidth for each bootstrap replication. The default is to compute the bandwidth once and keep it fixed for each bootstrap replication. This option is seldom used.
$\qquad$ Reporting
level(\#); see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fint), pformat (\% firt), sformat (\% fmt), and nolstretch; see [R] Estimation options.
citype (citype) specifies the type of confidence interval to be computed. By default, bootstrap percentile confidence intervals are reported as recommended by Cattaneo and Jansson (2018). citype may be one of percentile, bc, or normal.

Maximization
maximize_options: iterate(\#), [no]log, trace, showstep, tolerance(\#), ltolerance(\#), from(init_specs); see [R] Maximize. These options are seldom used.

The following option is available with npregress kernel but is not shown in the dialog box: coeflegend; see [R] Estimation options.

## Remarks and examples

This entry assumes that you are already familiar with nonparametric regression. For an introduction to the nonparametric kernel regression methods used in npregress kernel, see [ $R$ ] npregress intro.

Remarks are presented under the following headings:

Overview<br>Estimation and effects<br>Visualizing covariate effects

## Overview

npregress kernel implements local-constant and local-linear regression. The covariates may be continuous or discrete. You can use npregress kernel to nonparametrically estimate a conditional mean. npregress kernel also allows you to estimate covariate effects after estimation and, in models with one covariate, to plot the mean function by using npgraph after estimation.

The word "nonparametric" refers to the fact that the parameter of interest, the mean as a function of the covariates, is given by the unknown function $g\left(\mathbf{x}_{i}\right)$, which is an element of an infinite-dimensional space of functions. In contrast, in a parametric model, the mean for a given value of the covariates, $E\left(y_{i} \mid \mathbf{x}_{i}\right)=f\left(\mathbf{x}_{i}, \boldsymbol{\beta}\right)$, is a known function that is fully characterized by the parameter of interest, $\boldsymbol{\beta}$, which is a finite-dimensional real vector (Shao 2003).

The regression model of outcome $y_{i}$ given the $k$-dimensional vector of covariates $\mathbf{x}_{i}$ is given by

$$
\begin{align*}
y_{i} & =g\left(\mathbf{x}_{i}\right)+\varepsilon_{i}  \tag{1}\\
E\left(\varepsilon_{i} \mid \mathbf{x}_{i}\right) & =0 \tag{2}
\end{align*}
$$

where $\varepsilon_{i}$ is the error term. The covariates may include discrete and continuous variables. Equations (1) and (2) imply that

$$
E\left(y_{i} \mid \mathbf{x}_{i}\right)=g\left(\mathbf{x}_{i}\right)
$$

Once we account for the information in the covariates, the error term provides no information about the mean of our outcome. The conditional mean function is therefore given by $g\left(\mathbf{x}_{i}\right)$. By estimating $E\left(y_{i} \mid \mathbf{x}_{i}=\mathbf{x}\right)$ for all points $\mathbf{x}$ in our data, we obtain an estimate of $E\left(y_{i} \mid \mathbf{x}_{i}\right)$.
npregress kernel, by default, estimates a local-linear regression. Local-linear regression estimates a regression for a subset of observations for each point in our data. See Fan and Gijbels (1996) for a good reference on local-linear regression. Local-linear regression, for each point $\mathbf{x}$, solves the minimization problem given by

$$
\begin{equation*}
\min \gamma \sum_{i=1}^{n}\left\{y_{i}-\gamma_{0}-\gamma_{1}^{\prime}\left(\mathbf{x}_{i}-\mathbf{x}\right)\right\}^{2} K\left(\mathbf{x}_{i}, \mathbf{x}, \mathbf{h}\right) \tag{3}
\end{equation*}
$$

where $\gamma=\left(\gamma_{0}, \gamma_{1}^{\prime}\right)^{\prime}$.
Equation (3) and its solution are similar to parametric ordinary least squares. The slope and the constant in (3), however, have a different interpretation. The constant in (3), $\gamma_{0}$, is the conditional mean at a specific point $\mathbf{x}$. The slope parameter, $\gamma_{1}$, is the derivative of the mean function with respect to $\mathbf{x}$. The solution to this least-squares problem gives us the mean function and its derivative for each one of the elements of $\mathbf{x}$. Repeating this optimization for each point $\mathbf{x}$ gives us the entire mean function and its derivatives.

Another difference between (3) and the minimization problem of parametric ordinary least squares is how the optimization is weighted. The weights are given by the kernel function $K\left(\mathbf{x}_{i}, \mathbf{x}, \mathbf{h}\right)$. The kernel function assigns weights to observations $\mathbf{x}_{i}$ based on how much they differ from $\mathbf{x}$ and based on the bandwidth, $\mathbf{h}$. The smaller $\mathbf{h}$ is, the larger the weight assigned to points between $\mathbf{x}_{i}$ and $\mathbf{x}$.

The bandwidth also determines the bias and variance of the mean function estimator. npregress kernel selects the bandwidth using cross-validation, as suggested by Li and Racine (2004), or if the imaic option is specified, with the improved AIC proposed by Hurvich, Simonoff, and Tsai (1998). Both methods minimize the tradeoff between bias and variance.
npregress kernel computes a conditional mean for each observation in the data and, for each one of these computations, verifies whether identification conditions are fulfilled. The observations for which the regression identification assumptions are not satisfied are dropped from the estimation sample. Additionally, whenever there is a violation of the identification assumption, npregress kernel generates a system variable or a variable with a name provided in noidsample (newvar). This variable is 1 for observations violating the identification assumption and is 0 otherwise. npregress kernel also issues a warning, letting you know the number of observations for which the identification assumption is not satisfied.

## Estimation and effects

The output of npregress kernel reports averages of the mean function and the effects of the mean function. An average effect from nonparametric regression may be either 1) an average marginal effect, in the case of the mean of derivatives for continuous covariates or 2) the mean of contrasts for discrete covariates.

Some well-established literature estimates these average effects directly and uses an optimal bandwidth for this computation; see Powell, Stock, and Stoker (1989) and Powell and Stoker (1996). By taking averages of the local-linear estimates, npregress kernel is more in line with the approach in Li, Lu, and Ullah (2003). Intuitively, choosing the optimal bandwidth for the derivative produces a more efficient estimator than using the bandwidth that is optimal for the function. Both estimators are consistent for the average effect. Cattaneo and Jansson (2018) formally justify the average effect using the function-optimal bandwidth.
npregress kernel also reports an approximation of $n|\mathbf{h}|$ as the expected kernel observations. This statistic rounds the product of the continuous kernel bandwidth values and the number of observations used for estimation. For instance, if the estimation sample was 500 and the bandwidth was 0.246 , the expected kernel observations would be $123(=500 \times 0.246)$. The expected kernel observation number of 123 tells us that, on average, 123 observations are used to compute each one of the 500 regressions performed by npregress kernel.

## > Example 1: Nonparametric regression estimation and graphing

dui.dta contains information about the number of monthly drunk driving citations in a local jurisdiction (citations). Suppose we want to know the effect of increasing fines on the number of citations. Because citations is a count variable, we could consider fitting the model with poisson or nbreg. However, both of these estimators make assumptions about the distribution of the data. If these assumptions are not true, we will obtain inconsistent estimates.

By using npregress kernel, we do not have to make any assumptions about how citations is distributed. We use npregress kernel to estimate the mean of citations as a function of the value of the fines imposed for drunk driving (fines).

```
. use https://www.stata-press.com/data/r18/dui
(Fictional data on monthly drunk driving citations)
. npregress kernel citations fines
Computing mean function
Minimizing cross-validation function:
Iteration 0: Cross-validation criterion = 35.478784
Iteration 1: Cross-validation criterion = 4.0147129
Iteration 2: Cross-validation criterion = 4.0104176
Iteration 3: Cross-validation criterion = 4.0104176
Iteration 4: Cross-validation criterion = 4.0104176
Iteration 5: Cross-validation criterion = 4.0104176
Iteration 6: Cross-validation criterion = 4.0104006
Computing optimal derivative bandwidth
Iteration 0: Cross-validation criterion = 6.1648059
Iteration 1: Cross-validation criterion = 4.3597488
Iteration 2: Cross-validation criterion = 4.3597488
Iteration 3: Cross-validation criterion = 4.3597488
Iteration 4: Cross-validation criterion = 4.3597488
Iteration 5: Cross-validation criterion = 4.3597488
Iteration 6: Cross-validation criterion = 4.3595842
Iteration 7: Cross-validation criterion = 4.3594713
Iteration 8: Cross-validation criterion = 4.3594713
Bandwidth
```

|  | Mean | Effect |
| ---: | ---: | ---: |
| fines | .5631079 | .924924 |


| Local-linear regression | Number of obs | $=$ | 500 |
| :--- | :--- | :--- | ---: |
| Kernel : epanechnikov | E(Kernel obs) | $=$ | 282 |
| Bandwidth: cross-validation | R-squared | $=$ | 0.4380 |


| citations | Estimate |
| :--- | :--- |
| Mean <br> citations | 22.33999 |
| Effect |  |
| $\quad$ fines | -7.692388 |

Note: Effect estimates are averages of derivatives.
Note: You may compute standard errors using vce(bootstrap) or reps().
The first table displays the bandwidths used to estimate the mean function and the derivative of the mean function. Each of these bandwidths is estimated by minimizing a function that trades off bias and variance; the corresponding iteration logs are displayed also. The expected number of observations used to estimate the mean function at each point is reported in E (Kernel obs) as 282.

Unlike other estimation commands, npregress kernel does not report standard errors, test statistics, and confidence intervals by default. In example 2, we demonstrate how to obtain these statistics and further discuss the output.

## > Example 2: Bootstrapping standard errors

We can estimate the standard errors by using the bootstrap; see Cattaneo and Jansson (2018) for formal results. We use the reps (400) option, which is equivalent to vce (bootstrap, reps (400)) and specifies that 400 bootstrap replications be used instead of the default 50 replications that are used when we specify vce(bootstrap).

Each estimation problem requires a different number of replications to produce good estimates of the standard errors. In example 3, we explain how we decided to use 400 replications. Note that nonparametric estimation and the bootstrap are computationally intensive, so running this example and others that compute bootstrap standard errors will take a while.

```
. npregress kernel citations fines, reps(400) seed(12)
(running npregress on estimation sample)
Bootstrap replications (400): .........10.........20.......................40......
> ....50........60........70.........80..........90.......... 100.......... . 110... 
> ......120.........130.........140.........150..........160.......... . . 170......... .
> .180.........190.........200.........210..........220..........230........... 240.
> ........250.........260..........270..........280...........290........... . 300.......
> ...310.........320..........330. . . . . . . . 340. . . . . . . . 350. . . . . . . . . 360 . . . . . . . . . }3
> 0........380........390.........400 done
Bandwidth
```

|  | Mean | Effect |
| ---: | ---: | ---: |
| fines | .5631079 | .924924 |


| Local-linear regression |  |  | Number of obs |  | 500 |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Kernel : epanechnikov |  |  | E (Kernel obs) |  | = | 282 |
| Bandwidth: cross-validation |  |  | R -squared |  | = | 0.4380 |
| citations | Observed estimate | Bootstrap std. err. | z | $P>\|z\|$ | $\begin{aligned} & \text { Perd } \\ {[95 \%} & \text { con } 1 \end{aligned}$ | tile <br> interval] |
| Mean |  |  |  |  |  |  |
| citations | 22.33999 | . 4588298 | 48.69 | 0.000 | 21.48622 | 23.35956 |
| Effect |  |  |  |  |  |  |
| fines | -7.692388 | . 491884 | -15.64 | 0.000 | -8.693068 | -6.757721 |

Note: Effect estimates are averages of derivatives.
The coefficient table now reports the average of the predicted means and the average of the predicted derivatives of the mean function with bootstrap standard errors. The average of the observation-level predicted (citations) is 22.34 . The average of the observation-level marginal effects is -7.69 , which indicates that increasing fines reduces the mean number of citations.

We use npgraph to graph the estimated conditional mean function.

```
. npgraph
```



The graph shows the negative association between fines and the number of drunk driving citations.
npregress kernel generates system variables for the mean function and the derivative of the mean function. To see the variables that npregress kernel generated for example 1, we type

| Variable name | Storage type | Display format | Value label | Variable label |
| :---: | :---: | :---: | :---: | :---: |
| _Mean_citations double |  |  |  | Mean function |
| _d_Mean_citations_dfines |  |  |  |  |
|  | double | $\% 10.0 \mathrm{~g}$ |  | derivative of fines |

To specify a name for each system variable, we can use the predict() option.


Alternatively, we can use the same stub for all the variable names by typing predict(hatvar*), which would generate variables hatvar1 and hatvar2.

You may add noderivatives to the option, as in predict(hatvar*, noderivatives), to specify that no derivatives be generated. You save memory when you use noderivatives, but you add to the computational burden. As you will see below, an important feature of npregress kernel is the availability of the margins command after estimation. margins must compute the derivatives and their optimal bandwidth.

## > Example 3: Selecting the number of bootstrap replications

We start by fitting the model using 200 bootstrap replications. We want to find the number of replications for which the confidence intervals do not change much.

```
. npregress kernel citations fines, reps(200) seed(12)
(running npregress on estimation sample)
Bootstrap replications (200): .........10..........20..........30............40......
> ....50.........60......... . 70. . . . . . . . 80. . . . . . . . . 90. . . . . . . . . 100. . . . . . . . . 110. . .
> ......120.........130.........140..........150.......... . 160.......... . 170......... . . . 
> .180.........190......... }200\mathrm{ done
Bandwidth
```

|  | Mean | Effect |
| ---: | ---: | ---: |
| fines | .5631079 | .924924 |


| Local-linear regression | Number of obs | $=$ | 500 |
| :--- | :--- | :--- | ---: |
| Kernel : epanechnikov | E(Kernel obs) | $=$ | 282 |
| Bandwidth: cross-validation | R-squared | $=$ | 0.4380 |


| citations | Observed <br> estimate | Bootstrap <br> std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | Percentile <br> [95\% conf. interval] |  |
| :--- | :--- | :--- | :---: | :---: | :---: | :---: |
| Mean <br> citations | 22.33999 | .4769389 | 46.84 | 0.000 | 21.49744 | 23.42156 |
| Effect $\quad$ fines | -7.692388 | .5088819 | -15.12 | 0.000 | -8.742081 | -6.77816 |

Note: Effect estimates are averages of derivatives.
For 200 replications, the confidence interval for the mean ranges from 21.50 to 23.42 . For the effect of fines, this range is -8.74 to -6.78 .

We repeat the estimation using 300 replications and the same seed as in the previous case.

```
. npregress kernel citations fines, reps(300) seed(12)
(running npregress on estimation sample)
Bootstrap replications (300): .........10.........20..........30...........40......
> ....50.........60.........70.........80.........90..........100............ . . 110...
> ......120..........130..........140....... . . .150. . . . . . . .160. . . . . . . . 170. . . . . . . .
> .180........190.........200..........210.......... . 220.......... . 230. . . . . . . . 240.
> ........250.........260..........270..........280. . . . . . . . . 290. . . . . . . . . . }300\mathrm{ done
Bandwidth
```

|  | Mean | Effect |
| ---: | ---: | ---: |
| fines | .5631079 | .924924 |


| Local-linear regression | Number of obs | $=$ | 500 |
| :--- | :--- | :--- | ---: |
| Kernel : epanechnikov | E(Kernel obs) | $=$ | 282 |
| Bandwidth: cross-validation | R-squared | $=$ | 0.4380 |


| citations | Observed estimate | Bootstrap std. err. | z | $P>\|z\|$ | ```Percentile [95% conf. interval]``` |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Mean citations | 22.33999 | . 4570611 | 48.88 | 0.000 | 21.49359 | 23.36299 |
| Effect |  |  |  |  |  |  |
| fines | -7.692388 | . 4981956 | -15.44 | 0.000 | -8.673813 | -6.720508 |

Note: Effect estimates are averages of derivatives.
The confidence interval for the mean ranges from 21.49 to 23.36 . For the effect of fines, this range is -8.67 to -6.72 . There are some differences so we try estimation with 400 replications.

```
. npregress kernel citations fines, reps(400) seed(12)
(running npregress on estimation sample)
Bootstrap replications (400): .........10..........20..........30............40......
> ....50.........60......... . 70.......... 80. . . . . . . . 90. . . . . . . . 100. . . . . . . . . 110. . 
> ......120.........130.........140..........150...........160........... 170. . . . . . . . 
> .180........190.........200.........210..........220.......... 230........... . 240.
> ........250.........260.........270..........280..........290........... . . 300.......
> ...310.........320.........330.........340..........350...........360............ . . 37
> 0.........380.........390......... }400\mathrm{ done
Bandwidth
```

|  | Mean | Effect |
| ---: | ---: | ---: |
| fines | .5631079 | .924924 |


| Local-linear regression | Number of obs | $=$ | 500 |
| :--- | :--- | :--- | ---: |
| Kernel : epanechnikov | E(Kernel obs) | $=$ | 282 |
| Bandwidth: cross-validation | R-squared | $=$ | 0.4380 |


| citations | Observed estimate | Bootstrap std. err. | z | $P>\|z\|$ | $\begin{aligned} & \text { Per } \\ {[95 \%} & \text { con } \end{aligned}$ | $\begin{aligned} & \text { ntile } \\ & \text { interval] } \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Mean |  |  |  |  |  |  |
| citations | 22.33999 | . 4588298 | 48.69 | 0.000 | 21.48622 | 23.35956 |
| Effect |  |  |  |  |  |  |
| fines | -7.692388 | . 491884 | -15.64 | 0.000 | -8.693068 | -6.757721 |

Note: Effect estimates are averages of derivatives.
The confidence interval for the mean ranges from 21.49 to 23.36 . In the case of the effect of fines, these ranges are -8.69 to -6.76 . The changes are small so we decide to use 400 replications.

## Example 4: Estimating the effect of a percentage change in a covariate

Nonparametric estimation and the bootstrap are computationally intensive, so we use only 200 replications here.

We now extend example 2. In addition to fines, we model citations as a function of whether the jurisdiction taxes alcoholic beverages (taxes); whether the city is small, medium, or large (csize); and whether there is a college in the jurisdiction (college).

```
. npregress kernel citations fines i.taxes i.csize i.college, nolog
> reps(200) seed(12)
(running npregress on estimation sample)
Bootstrap replications (200): .........10..........20...........30............40......
> ....50........60........70.........80.........90..........100........... . . 110...
> .....120........130........140.........150..........160..........170...........
> .180........190......... }200\mathrm{ done
Bandwidth
```

|  | Mean | Effect |
| ---: | ---: | ---: |
| fines | .4471373 | .6537197 |
| taxes | .4375656 | .4375656 |
| csize | .3938759 | .3938759 |
| college | .554583 | .554583 |


| Local-linear regression | Number of obs | $=$ | 500 |
| :--- | :--- | :--- | ---: |
| Continuous kernel $:$ epanechnikov | E(Kernel obs) | $=$ | 224 |
| Discrete kernel | : liracine | R-squared | $=$ |
| Bandwidth | : cross-validation |  |  |

Bandwidth : cross-validation

| citations | Observed estimate | Bootstrap std. err. | z | $P>\|z\|$ | $\begin{aligned} & \text { Per } \\ {[95 \%} & \text { con } \end{aligned}$ | $\begin{aligned} & \text { atile } \\ & \text { interval] } \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Mean |  |  |  |  |  |  |
| citations | 22.26306 | . 4642464 | 47.96 | 0.000 | 21.46204 | 23.2516 |
| Effect |  |  |  |  |  |  |
| fines | -7.332833 | . 3316656 | -22.11 | 0.000 | -8.013487 | -6.741899 |
| taxes |  |  |  |  |  |  |
| (Tax |  |  |  |  |  |  |
| No tax) | -4.502718 | . 5012 | -8.98 | 0.000 | -5.437733 | -3.544934 |
| $\begin{gathered} \text { csize } \\ \text { (Medium } \end{gathered}$ |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| Small) | 5.300524 | . 2687413 | 19.72 | 0.000 | 4.758121 | 5.797119 |
| vs |  |  |  |  |  |  |
| Small) | 11.05053 | . 502633 | 21.99 | 0.000 | 10.00169 | 11.94311 |
| $\begin{aligned} & \text { college } \\ & \text { (College } \end{aligned}$ |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| Not coll..) | 5.953188 | . 461057 | 12.91 | 0.000 | 5.086511 | 6.88612 |

Note: Effect estimates are averages of derivatives for continuous covariates and averages of contrasts for factor covariates.

The mean number of citations predicted by the mean estimates is 22.26 . The average marginal effect of fines is -7.33 , slightly less in magnitude than the -7.69 that we estimated in example 2 .

The average marginal effect tells us the result of an infinitesimal change in fines on citations. Instead of talking about infinitesimal changes, we want to know the effect of increasing fines by $15 \%$. We can use margins to estimate the mean number of citations that would occur if fines were increased by $15 \%$.

```
. margins, at(fines=generate(fines*1.15)) reps(200) seed(12)
(running margins on estimation sample)
Bootstrap replications (200): .........10.........20.......................40.....
> ....50........60........70.........80.........90..........100............ 110...
> ......120........130.........140.........150..........160.......... . . 170..........
> .180.........190......... }200\mathrm{ done
Predictive margins Number of obs = 500
    Replications = 200
Expression: Mean function, predict()
At: fines = fines*1.15
```

|  | Observed <br> margin | Bootstrap <br> std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | Percentile <br> [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| _cons | 14.00818 | .866694 | 16.16 | 0.000 | 11.39967 | 15.00145 |

The estimated mean number of citations with the new level of fines is 14.01 , which is smaller than the mean 22.26 that was estimated with the observed fines. We can formally compare this estimate with the mean at the original level of fines. We use the contrast () option with margins to estimate the difference in these means.

```
. margins, at(fines=generate(fines)) at(fines=generate(fines*1.15))
> contrast(atcontrast(r) nowald) reps(200) seed(12)
(running margins on estimation sample)
Bootstrap replications (200): .........10.........20................................
> ....50........60........70.........80.........90..........100........... . . 110...
> ......120.........130.........140.........150..........160.......... . . 170..........
> .180........190......... }200\mathrm{ done
Contrasts of predictive margins
                                    Number of obs = 500
                                    Replications = 200
Expression: Mean function, predict()
1._at: fines = fines
2._at: fines = fines*1.15
```

|  | Observed <br> contrast | Bootstrap <br> std. err. | Percentile <br> [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: |
| $\left(2\right.$ vs 1) $^{\text {at }}$ | -8.254875 | .8058215 | -10.44121 | -7.381583 |

We find that increasing fines by $15 \%$ reduces the average number of drunk driving citations by 8.25 .

## > Example 5: Estimating the effect of a change in level

Now, we estimate the effect of increasing fines from $\$ 10,000$ to $\$ 11,000$ for fixed levels of the other covariates. The other covariate values identify a jurisdiction with a set of characteristics of interest: of medium size, with a college, and taxes alcohol.

First, we use margins to estimate the means for a jurisdiction with the characteristics of interest for the two levels of fines.

| Bootstrap replications (200): .........10......... 20............30......... . . 40 ... |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
| Adjusted predictions |  |  |  |  | Number | obs $=500$ |
|  |  |  |  |  | Replicat | ns $=200$ |
| Expression: Mean function, predict() |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
| taxes $=1$ |  |  |  |  |  |  |
| csize $=2$ |  |  |  |  |  |  |
| college $=1$ |  |  |  |  |  |  |
| 2._at: fines = 11 |  |  |  |  |  |  |
| taxes $=1$ |  |  |  |  |  |  |
| csize $=2$ |  |  |  |  |  |  |
| college $=1$ |  |  |  |  |  |  |
|  | Observed margin | Bootstrap <br> std. err. | z | $P>\|z\|$ | Per [95\% conf | $\begin{aligned} & \text { atile } \\ & \text { interval] } \end{aligned}$ |
| _at |  |  |  |  |  |  |
| 1 | 23.17242 | . 5746008 | 40.33 | 0.000 | 21.95222 | 24.30412 |
| 2 | 15.90157 | . 972558 | 16.35 | 0.000 | 13.87449 | 17.7134 |

For a medium-sized jurisdiction that taxes alcohol and has a college, the estimated mean of citations when fines are $\$ 10,000$ is 23.17 , and the estimated mean of citations when fines are $\$ 11,000$ is 15.90 .

We now use margins to estimate the difference in these means.

```
. margins, at(fines=10 taxes=1 csize=2 college=1)
> at(fines=11 taxes=1 csize=2 college=1)
> contrast(atcontrast(r) nowald) reps(200) seed(12)
(running margins on estimation sample)
Bootstrap replications (200): .........10.........20......................40......
> ....50........60.........70.........80..........90..........100............ 110....
> ......120........130.........140.........150..........160...........170..........
> .180.........190......... }200\mathrm{ done
Contrasts of predictive margins
                                    Number of obs = 500
                                    Replications = 200
Expression: Mean function, predict()
1._at: fines = 10
    taxes = 1
    csize = 2
    college = 1
2._at: fines = 11
    taxes = 1
    csize = 2
    college = 1
\begin{tabular}{c|cccc}
\hline & \begin{tabular}{l} 
Observed \\
contrast
\end{tabular} & \begin{tabular}{l} 
Bootstrap \\
std. err.
\end{tabular} & \begin{tabular}{c} 
Percentile \\
[95\% \\
conf. interval]
\end{tabular} \\
\hline\(\left(2\right.\) vs 1) \(^{\text {at }}\) & -7.270858 & 1.003861 & -9.096777 & -5.162513 \\
\hline
\end{tabular}
```

In these jurisdictions, increasing fines from $\$ 10,000$ to $\$ 11,000$ reduces the average number of citations by 7.27.

## Example 6: Population-averaged covariate effects

In example 5, we estimated the means for two values of fines for a medium-sized jurisdiction with a college and taxes on alcohol. We specified values for each covariate in our model. In this example, we will now estimate population-averaged means instead of means at specific levels of all covariates.

We first estimate the means for two levels of fines. We do not specify values for csize, college, or taxes, so the estimated means are unconditional on these covariates. We use margins to estimate means of citations when fines are $\$ 10,000$ and when fines are $\$ 11,000$ :

```
. margins, at(fines=10) at(fines=11) reps(200) seed(12)
(running margins on estimation sample)
Bootstrap replications (200): .........10..........20..........30............40......
> ....50.........60..........70........ . . 80. . . . . . . . 90. . . . . . . . . 100. . . . . . . . . 110. . .
> ......120.........130.........140..........150.......... . 160.......... . 170..........
> .180........190......... }200\mathrm{ done
Predictive margins Number of obs = 500
                                    Replications = 200
Expression: Mean function, predict()
1._at: fines = 10
2._at: fines = 11
```

|  | Observed <br> margin | Bootstrap <br> std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | Percentile <br> [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| at |  |  |  |  |  |  |
| 1 | 20.50161 | .3281821 | 62.47 | 0.000 | 19.90257 | 21.08954 |
| 2 | 14.97432 | .3815647 | 39.24 | 0.000 | 14.14858 | 15.59955 |

The estimated mean of citations when fines are $\$ 10,000$ is 20.50 , and the estimated mean of citations when fines are $\$ 11,000$ is 14.97 . We now use margins to estimate the difference in these means:

```
. margins, at(fines=10) at(fines=11)
> contrast(atcontrast(r) nowald) reps(200) seed(12)
(running margins on estimation sample)
Bootstrap replications (200): .........10.........20................................
> ....50........60.........70..........80..........90...........100........... . . 110...
> ......120.........1130.........140.......... . 150...........160....... . . . 170. . . . . . . .
> .180........190......... }200\mathrm{ done
Contrasts of predictive margins
                                    Number of obs = 500
Expression: Mean function, predict()
1._at: fines = 10
2._at: fines = 11
\begin{tabular}{c|cccc}
\hline & \begin{tabular}{l} 
Observed \\
contrast
\end{tabular} & \begin{tabular}{l} 
Bootstrap \\
std. err.
\end{tabular} & \begin{tabular}{c} 
Percentile \\
[95\% \\
conf. interval]
\end{tabular} \\
\hline (2 vs 1) & -5.527288 & .3529352 & -6.277903 & -4.925523 \\
\hline
\end{tabular}
```

When fines increase from $\$ 10,000$ to $\$ 11,000$, the mean number of citations is estimated to decrease by 5.53 .

Next, we consider the effect of taxing alcoholic beverages. We first estimate the population-averaged number of citations with and without such taxes.


The estimated mean number of citations is 25.47 when there are no alcohol taxes and 20.97 when there are alcohol taxes. We again use margins to estimate the difference in these means.


Expression: Mean function, predict()

|  | df | chi2 | $\mathrm{P}>\operatorname{chi2}$ |
| ---: | :---: | :---: | :---: |
| taxes | 1 | 80.71 | 0.0000 |


|  | Observed <br> contrast | Bootstrap <br> std. err. | Percentile <br> [95\% conf. interval] |
| :---: | :---: | :---: | :---: |
| (Tax vs No tax) | -4.502719 | .5011999 | -5.437733 |

The mean number of citations is estimated to decrease by 4.50 when alcohol sales are taxed.

## Visualizing covariate effects

## > Example 7: Using margins to visualize the mean function and covariate effects

We can also estimate the mean function for the jurisdiction with characteristics of interest over a range of observed fines. We simply add a range of fines to our margins specification from example 4.

```
. margins, at(fines=(8(0.5)12) taxes=1 csize=2 college=1) reps(200) seed(12)
(output omitted)
```

We graph these results using marginsplot.

```
. marginsplot
    (output omitted)
```

Adjusted predictions with 95\% Cls


We estimated the mean when fines are $\$ 8,000, \$ 8,500$, and so on. From these estimated means, we can estimate the effect of a $\$ 500$ increase for each of these levels of fines.

We simply reissue our margins command and specify a reverse adjacent contrast that subtracts the current level from the next level for each level of fines.

```
. margins, at(fines=(8(0.5)12) taxes=1 csize=2 college=1)
> contrast(atcontrast(ar)) reps(200) seed(12)
    (output omitted)
```

We again graph the results, adding a reference line at 0 that designates no change in citations:

```
. marginsplot, yline(0)
    (output omitted)
```

Contrasts of adjusted predictions with 95\% Cls


For each level of fines between $\$ 8,500$ and $\$ 11,500$, the effect of a $\$ 500$ increase reduces the mean number of drunk driving incidents. Between $\$ 11,500$ and $\$ 12,000$, the difference of a $\$ 500$ increase is not statistically different from 0 .

It would be easy to construct a similar graph for the population-averaged effects in example 6. Simply omit the terms that set the other covariates at fixed values.

## Stored results

npregress kernel stores the following in e():
Scalars
$e(N) \quad$ number of observations
$e($ mean ) mean of mean function
e(r2)
$e(n h)$
e(converged_effect)
$R^{2}$
expected kernel observations
1 if effect optimization converged, 0 otherwise
$e$ (converged_mean) 1 if mean optimization converged, 0 otherwise
$e$ (converged) $\quad 1$ if effect and mean optimization converged, 0 otherwise
Macros
$e(\mathrm{cmd}) \quad$ npregress
e(cmdline)
e(depvar)
e(estimator)
e(kname)
e(dkname)
e(bselector)
command as typed
name of dependent variable
linear or constant
name of continuous kernel
name of discrete kernel
criterion function for bandwidth selection
$e$ (title) title in estimation output
e (vce) vcetype specified in vce()
e(properties) b (or b V if reps() specified)
$e($ datasignaturevars) variables used in calculation of checksum
e(datasignature)
e(estat_cmd)
e(predict)
e(marginsok)
the checksum
program used to implement estat
program used to implement predict
predictions allowed by margins
$e$ (marginsprop) signals to the margins command

Matrices
e(b)
coefficient vector
e(bwidth)
bandwidth for all predictions
e(derivbwidth)
e(meanbwidth)
bandwidth for the derivative
e(ilog_mean)
e(ilog_effect)
bandwidth for the mean
iteration $\log$ for mean (up to 20 iterations)
iteration $\log$ for effects (up to 20 iterations)
Functions
e(sample) marks estimation sample
In addition to the above, the following is stored in r() :

Matrices $r$ (table)
matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

The regression model of outcome $y_{i}$ given the $k$-dimensional vector of covariates $\mathbf{x}_{i}$ was defined in (1) and (2) of Remarks and examples and repeated here:

$$
\begin{align*}
y_{i} & =g\left(\mathbf{x}_{i}\right)+\varepsilon_{i}  \tag{1}\\
E\left(\varepsilon_{i} \mid \mathbf{x}_{i}\right) & =0 \tag{2}
\end{align*}
$$

where $\varepsilon_{i}$ is the error term. The covariates may include discrete and continuous variables. Equations (1) and (2) imply that

$$
E\left(y_{i} \mid \mathbf{x}_{i}\right)=g\left(\mathbf{x}_{i}\right)
$$

npregress kernel, by default, estimates a local-linear regression; see Fan and Gijbels (1996) for a good reference on local-linear regression. As we discussed in Remarks and examples, local-linear regression estimates a regression for a subset of observations for each point in our data and solves the minimization problem given by

$$
\begin{equation*}
\min \gamma \sum_{i=1}^{n}\left\{y_{i}-\gamma_{0}-\gamma_{1}^{\prime}\left(\mathbf{x}_{i}-\mathbf{x}\right)\right\}^{2} K\left(\mathbf{x}_{i}, \mathbf{x}, \mathbf{h}\right) \tag{3}
\end{equation*}
$$

where $\gamma=\left(\gamma_{0}, \gamma_{1}^{\prime}\right)^{\prime}$ and $K\left(\mathbf{x}_{i}, \mathbf{x}, \mathbf{h}\right)$ is the product of the kernels for each covariate.

$$
K\left(\mathbf{x}_{i}, \mathbf{x}, \mathbf{h}\right)=\prod_{j=1}^{k} K_{j}\left(x_{i j}, x_{j}, h_{j}\right)
$$

The kernel for a continuous covariate is of the form

$$
K_{j}\left(x_{i j}, x_{j}, h_{j}\right)=k_{j}\left(\frac{x_{i j}-x_{j}}{h_{j}}\right)
$$

where $k_{j}(\cdot)$ is one of the kernels listed in $[\mathrm{R}]$ kdensity. For discrete covariates, npregress kernel uses the $\mathrm{Li}-\mathrm{Racine}$ kernel given by

$$
K_{j}\left(x_{i j}, x_{j}, h_{j}\right)=\left\{\begin{array}{l}
1 \text { if } x_{i j}=x_{j} \\
h_{j} \text { otherwise }
\end{array}\right.
$$

By estimating $E\left(y_{i} \mid \mathbf{x}_{i}=\mathbf{x}\right)$ for all points $\mathbf{x}$ in our data, we obtain an estimate of $E\left(y_{i} \mid \mathbf{x}_{i}\right)$. For a given $\mathbf{x}$, the solution to the minimization problem in (3) is given by

$$
\widehat{\gamma}=\left(\mathbf{Z}^{\prime} \mathbf{W} \mathbf{Z}\right)^{-1} \mathbf{Z}^{\prime} \mathbf{W} \mathbf{y}
$$

where $\widehat{\gamma}=\left(\widehat{\gamma}_{0}, \widehat{\gamma}_{1}^{\prime}\right)^{\prime}, \mathbf{Z}$ is an $n \times(k+1)$ matrix with an $i$ th row given by $\left\{1,\left(\mathbf{x}_{i}-\mathbf{x}\right)^{\prime}\right\}^{\prime}, \mathbf{W}$ is an $n \times n$ diagonal matrix with an $i$ th diagonal given by $K\left(\mathbf{x}_{i}, \mathbf{x}, \mathbf{h}\right)$, and $\mathbf{y}$ is the $n \times 1$ outcome vector. $\widehat{\gamma}_{0}$ is an estimate of $g(\mathbf{x})$, whereas $\widehat{\gamma}_{1}$ is an estimate of the derivative of $g(\mathbf{x})$ with respect to $\mathbf{x}$. When the matrix $\left(\mathbf{Z}^{\prime} \mathbf{W} \mathbf{Z}\right)$ is not full rank, the parameter $\gamma$ is not identified. The observations for which this is true are dropped from the estimation sample.

The local-constant estimator of $g(\mathbf{x})$ is a special case of (3) with $\gamma_{1}=0$. In this case, the solution to the optimization problem is given by

$$
\frac{\sum_{i=1}^{n} y_{i} K\left(\mathbf{x}_{i}, \mathbf{x}, \mathbf{h}\right)}{\sum_{i=1}^{n} K\left(\mathbf{x}_{i}, \mathbf{x}, \mathbf{h}\right)}
$$

This is also known as the Nadaraya-Watson kernel estimator, for Nadaraya (1965) and Watson (1964).
npregress kernel and margins, when used after npregress kernel, use a bootstrap estimate of the standard errors for all the estimated effects and report percentile confidence intervals by default. Cattaneo and Jansson (2018) formally justify this use of the bootstrap and provide a definitive reference for semiparametric estimation and inference using kernel-based estimators. Their work demonstrates that the percentile bootstrap provides better coverage than a normal-based confidence interval for statistics based on kernel estimates. See Methods and formulas in $[R]$ bootstrap for confidence interval formulas.

The rate of convergence of nonparametric regression estimates is given by the product of the sample size and the bandwidths $\sqrt{n|\mathbf{h}|}$, where $|\mathbf{h}|$ is the product of the bandwidths for each covariate. As the sample size increases, the bandwidth decreases. Thus, the rate of convergence of the estimator is slower than the parametric rate $\sqrt{n}$. Another way of thinking about $n|\mathbf{h}|$ is that, because we are not using all our observations to estimate the mean at each point, we require more data to get more reliable estimates; the convergence rate is thus slower. The rate of convergence also decreases as the number of covariates increases, because $|\mathbf{h}|$ decreases. This is referred to as the curse of dimensionality; see Li and Racine (2007, chap. 2) and Stinchcombe and Drukker (2013) for details.

The convergence rate for the derivative of the mean function is different from the convergence rate of the mean function. Therefore, the bandwidth and bandwidth computation for the derivative are different. npregress kernel computes the bandwidth for the derivative function by using cross-validation, as suggested by Henderson et al. (2015).

## Acknowledgments

We thank Matias Cattaneo of the Department of Economics at the University of Michigan and Qi Li of the Department of Economics at Texas A\&M University for many helpful conversations. We thank Xinrong Li of the Department of Economics at the Central University of Finance and Economics in China for research assistance during her summer internship at StataCorp during the summer of 2008.

## References

Cattaneo, M. D., and M. Jansson. 2018. Kernel-based semiparametric estimators: Small bandwidth asymptotics and bootstrap consistency. Econometrica 86: 955-995. https://doi.org/10.3982/ECTA12701.

Chetverikov, D., D. Kim, and D. Wilhelm. 2018. Nonparametric instrumental-variable estimation. Stata Journal 18: 937-950.

Cox, N. J. 2005. Speaking Stata: Smoothing in various directions. Stata Journal 5: 574-593.
——. 2007. Kernel estimation as a basic tool for geomorphological data analysis. Earth Surface Processes and Landforms 32: 1902-1912. https://doi.org/10.1002/esp.1518.

Eubank, R. L. 1999. Nonparametric Regression and Spline Smoothing. 2nd ed. New York: Dekker.
Fan, J., and I. Gijbels. 1996. Local Polynomial Modelling and Its Applications. London: Chapman and Hall.
Gutierrez, R. G., J. M. Linhart, and J. S. Pitblado. 2003. From the help desk: Local polynomial regression and Stata plugins. Stata Journal 3: 412-419.

Hansen, B. E. 2014. Nonparametric sieve regression: Least squares, averaging least squares, and crossvalidation. In The Oxford Handbook of Applied Nonparametric and Semiparametric Econometrics and Statistics, ed. J. S. Racine, L. Su, and A. Ullah, 215-248. New York: Oxford University Press. https://doi.org/10.1093/oxfordhb/9780199857944.013.008.

Henderson, D. J., Q. Li, C. F. Parmeter, and S. Yao. 2015. Gradient-based smoothing parameter selection for nonparametric regression estimation. Journal of Econometrics 184: 233-241. https://doi.org/10.1016/j.jeconom.2014.09.007.
Hurvich, C. M., J. S. Simonoff, and C.-L. Tsai. 1998. Smoothing parameter selection in nonparametric regression using an improved Akaike information criterion. Journal of the Royal Statistical Society, Series B 60: 271-293. https://doi.org/10.1111/1467-9868.00125.
Li, Q., X. Lu, and A. Ullah. 2003. Multivariate local polynomial regression for estimating average derivatives. Journal of Nonparametric Statistics 15: 607-624. https://doi.org/10.1080/10485250310001605450.
Li, Q., and J. S. Racine. 2004. Cross-validated local linear nonparametric regression. Statistica Sinica 14: 485-512.
——. 2007. Nonparametric Econometrics: Theory and Practice. Princeton, NJ: Princeton University Press.
Nadaraya, E. A. 1965. On nonparametric estimates of density functions and regression curves. Theory of Probability and Its Applications 10: 186-190. https://doi.org/10.1137/1110024.

Pinzon, E. 2017. Nonparametric regression: Like parametric regression, but not. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2017/06/27/nonparametric-regression-like-parametric-regression-but-not/.
Powell, J. L., J. H. Stock, and T. M. Stoker. 1989. Semiparametric estimation of index coefficients. Econometrica 57: 1403-1430. https://doi.org/10.2307/1913713.

Powell, J. L., and T. M. Stoker. 1996. Optimal bandwidth choice for density-weighted averages. Journal of Econometrics 75: 291-316. https://doi.org/10.1016/0304-4076(95)01761-5.
Rios-Avila, F. 2020. Smooth varying-coefficient models in Stata. Stata Journal 20: 647-679.
Shao, J. 2003. Mathematical Statistics. 2nd ed. New York: Springer.
Sheather, S. J., and M. C. Jones. 1991. A reliable data-based bandwidth selection method for kernel density estimation. Journal of the Royal Statistical Society, Series B 53: 683-690. https://doi.org/10.1111/j.2517-6161.1991.tb01857.x.
Stinchcombe, M. B., and D. M. Drukker. 2013. Regression efficacy and the curse of dimensionality. In Recent Advances and Future Directions in Causality, Prediction, and Specification Analysis: Essays in Honor of Halbert L. White Jr, ed. X. Chen and N. R. Swanson, 527-549. New York: Springer.

Verardi, V., and N. Debarsy. 2012. Robinson's square root of $N$ consistent semiparametric regression estimator in Stata. Stata Journal 12: 726-735.

Watson, G. S. 1964. Smooth regression analysis. Sankhyā, Series A 26: 359-372.

## Also see

$[R]$ npregress kernel postestimation - Postestimation tools for npregress kernel
$[R]$ npregress intro - Introduction to nonparametric regression
$[R]$ kdensity - Univariate kernel density estimation
[R] lpoly - Kernel-weighted local polynomial smoothing
[U] 20 Estimation and postestimation commands

## Title

npregress kernel postestimation - Postestimation tools for npregress kernel

| Postestimation commands | predict | margins |
| :--- | :--- | :--- |
| npgraph | Remarks and examples | Methods and formulas |
| References | Also see |  |

## Postestimation commands

The following postestimation command is of special interest after npregress kernel:

| Command | Description |
| :--- | :--- |
| npgraph | plot of conditional means |

The following standard postestimation commands are also available:
Command Description

| estat summarize | summary statistics for the estimation sample |
| :--- | :--- |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estimates | cataloging estimation results <br> table of estimation results |
| etable | point estimates, standard errors, testing, and inference for linear combinations of <br> coefficients |
| lincom | marginal means, predictive margins, marginal effects, and average marginal effects |
| margins | graph the results from margins (profile plots, interaction plots, etc.) <br> point estimates, standard errors, testing, and inference for nonlinear combinations <br> of coefficients |
| nlcom | conditional means and residuals <br> preint estimates, standard errors, testing, and inference for generalized predictions |
| predictnl | Wald tests of simple and composite linear hypotheses |
| test | Wald tests of nonlinear hypotheses |
| testnl |  |

## predict

## Description for predict

predict creates a new variable containing predictions such as conditional mean of the outcome, residuals, or derivatives of the mean function.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

```
predict [type] newvar [if] [in] [, statistic]
predict [type] {stub*|newvarlist} [if] [in], derivatives
```

statistic Description

Main mean conditional mean of the outcome; the default residuals residuals

These statistics are calculated only for the estimation sample.

## Options for predict

Main mean, the default, calculates the conditional mean of the outcome variable. residuals calculates the residuals. derivatives calculates the derivatives of the conditional mean.

## margins

## Description for margins

margins estimates margins of the conditional mean.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [options]
```


## statistic Description

Main

| mean | conditional mean of the outcome; the default |
| :--- | :--- |
| residuals | not allowed with margins |
| derivatives | not allowed with margins |

options
Description
SE
nose
vce (bootstrap)
reps (\#)
seed (\#)
Reporting
citype
percentile bc normal
citype (citype) method to compute bootstrap confidence intervals; default is citype (percentile)

## Description

do not estimate standard errors; the default
estimate bootstrap standard errors
equivalent to vce(bootstrap, reps(\#))
set random-number seed to \#; must also specify reps (\#)
percentile confidence intervals; the default
bias-corrected confidence intervals
normal-based confidence intervals

Statistics not allowed with margins are functions of stochastic quantities other than e(b). For the full syntax, see $[\mathrm{R}]$ margins.

## Options for margins

$\square$ SE
nose suppresses calculation of the VCE and standard errors. This is the default.
vce(bootstrap) specifies that bootstrap standard errors are reported; see [R] vce_option.
We recommend that you select the number of replications using reps (\#) instead of specifying vce(bootstrap), which defaults to 50 replications. Be aware that the number of replications needed to produce good estimates of the standard errors varies depending on the problem.
reps(\#) specifies the number of bootstrap replications to be performed. Specifying this option is equivalent to specifying vce(bootstrap, reps(\#)).
seed (\#) sets the random-number seed. You must specify reps (\#) with seed(\#).
$\qquad$ Reporting
citype (citype) specifies the type of confidence interval to be computed. By default, bootstrap percentile confidence intervals are reported as recommended by Cattaneo and Jansson (2018). citype may be one of percentile, bc, or normal.

## npgraph

## Description for npgraph

npgraph plots the conditional mean estimated by npregress kernel overlayed on a scatterplot of the data. npgraph is available only after fitting models with one covariate.

## Syntax for npgraph

npgraph [if] [in] [, options]
options
Description
Plot
marker_options
marker_label_options
noscatter
Smoothed line lineopts(cline_options)

Add plots addplot (plot) add other plots to the generated graph

Y axis, X axis, Titles, Legend, Overall twoway_options
change look of markers (color, size, etc.)
add marker labels; change look or position
suppress scatterplot
affect rendition of the smoothed line
any options other than by () documented in [G-3] twoway_options

## Options for npgraph

$\qquad$ Plot
marker_options affect the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G-3] marker_options.
marker_label_options specify if and how the markers are to be labeled; see [G-3] marker_label_options. noscatter suppresses superimposing a scatterplot of the observed data over the smooth. This option is useful when the number of resulting points would be so large as to clutter the graph.

Smoothed line
lineopts (cline_options) affects the rendition of the smoothed line; see [G-3] cline_options.
$\qquad$ Add plots
addplot (plot) provides a way to add other plots to the generated graph; see [G-3] addplot_option.

Y axis, X axis, Titles, Legend, Overall
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

## Remarks and examples

For examples of margins after npregress kernel, see example 4, example 5, and example 6 in [ R$]$ npregress kernel.

For examples of marginsplot, see example 7 in [R] npregress kernel.
For an example of npgraph, see example 2 in [R] npregress kernel.

## Methods and formulas

The formulas used by predict and margins for the conditional mean function and the mean marginal effect of a covariate are given in Methods and formulas of [R] npregress kernel.

## References

Cattaneo, M. D., and M. Jansson. 2018. Kernel-based semiparametric estimators: Small bandwidth asymptotics and bootstrap consistency. Econometrica 86: 955-995. https://doi.org/10.3982/ECTA12701.

MacDonald, K. 2018. Exploring results of nonparametric regression models. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2018/06/18/exploring-results-of-nonparametric-regression-models/.

[^40]
## Title

npregress series - Nonparametric series regression

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

npregress series performs nonparametric series estimation using a B-spline, piecewise polynomial spline, or polynomial basis. Like linear regression, nonparametric regression models the mean of the outcome conditional on the covariates, but unlike linear regression, it makes no assumptions about the functional form of the relationship between the outcome and the covariates. npregress series may be used to model the mean of a continuous, count, or binary outcome.

## Quick start

Nonparametric regression of y on x and discrete covariate a using the default B-spline basis
npregress series y x i.a
Same as above, but use a polynomial basis
npregress series y x i.a, polynomial
Same as above, but use a piecewise polynomial spline basis
npregress series y x i.a, spline
Same as above, but use AIC to find the optimal basis function
npregress series y x i.a, criterion(aic) spline
Interpolate using three knots
npregress series y x i.a, knots(3)
Specify values of knots in matrix K
npregress series y x i.a, knotsmat(K)

## Menu

Statistics $>$ Nonparametric analysis $>$ Nonparametric series regression

## Syntax

npregress series depvar indepvars series $[$ if $]$ [in] [weight $]$ [, options]
indepvars $_{\text {series }}$ is the list of independent variables for which a basis function will be formed.
options
Model
bspline
bspline(\#)
spline
spline(\#)
polynomial
polynomial(\#)
asis(varlist)
nointeract (seriesvarlist)
criterion (crittype)
knots(\#)
knotsmat (matname)
distinct(\#)
basis(stub [, replace])
rescale(stub [, replace]) SE
vce (vcetype)
Reporting
level(\#)
aequations
display_options

Maximization
maximize_options
coeflegend

Description
use a third-order B-spline basis; the default use a B-spline basis of order \#
use a third-order piecewise polynomial spline basis
use a piecewise polynomial spline basis of order \#
use a polynomial basis
use a polynomial basis of order \#
include varlist in model as specified; do not use in basis use seriesvarlist in basis without interactions
criterion to use; crittype may be cv, gcv, aic, bic, or mallows use a piecewise polynomial spline or B-spline basis function with \# knots
use knots in matrix matname for piecewise polynomial spline or B-spline estimation
minimum number of distinct values allowed in continuous covariates; default is distinct (10)
store elements of piecewise polynomial spline or B-spline basis function using stub
store rescaled values of covariates using stub
vcetype may be robust, ols, or bootstrap
set confidence level; default is level(95)
display auxiliary regression coefficients
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
control the maximization process
display legend instead of statistics
indepvars and varlist may contain factor variables; see [U] 11.4.3 Factor variables.
bootstrap, by, collect, and jackknife are allowed; see [U] 11.1.10 Prefix commands.
Weights are not allowed with the bootstrap prefix; see [R] bootstrap.
fweights and iweights are allowed; see [U] 11.1.6 weight.
coeflegend does not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

Model
bspline specifies that a third-order B-spline be selected. It is the default basis.
bspline(\#) specifies that a B-spline of order \# be used as the basis. The order may be 1,2 , or 3 . spline specifies that a third-order piecewise polynomial spline be selected as the basis.
spline(\#) specifies that a piecewise polynomial spline of order \# be used as the basis. The order may be 1,2 , or 3 .
polynomial specifies that a polynomial be selected as the basis.
polynomial(\#) specifies that a polynomial of order \# be used as the basis. The order may be an integer between 1 and 16.
asis(varlist) specifies that variables in varlist be included as independent variables in the model without any transformation. No B-spline, piecewise polynomial spline, or polynomial basis function will be formed from these variables. Variables in varlist may not be specified in indepvars ${ }_{\text {series }}$.
nointeract (seriesvarlist) specifies that the terms in the basis function formed from variables in seriesvarlist not be interacted with the terms of the basis function formed from other variables in indepvars $_{\text {series }}$. Covariates specified in seriesvarlist must be in indepvars series .
criterion(crittype) specifies that crittype be used to select the optimal number of terms in the basis function. crittype may be one of the following: cv (cross-validation), gcv (generalized crossvalidation), aic (Akaike's information criterion), bic (Schwarz's Bayesian information criterion), or mallows (Mallows's $C_{p}$ ). The default is criterion(cv).
knots(\#) specifies that a piecewise polynomial spline or B-spline basis function with \# knots be used. The minimum number of knots must be an integer greater than or equal to 1 . The maximum number of knots is either 4,096 or two-thirds of the sample size, whichever is smaller.
knotsmat (matname) specifies that the knots for each continuous covariate be the values in each row of matname. The number of knots should be the same for each covariate, and there must be as many rows as there are continuous covariates. If rows of matname are not labeled with varnames, then rows are assumed to be in the order of indepvars ${ }_{\text {series }}$.
distinct(\#) specifies the minimum number of distinct values allowed in continuous variables. By default, continuous variables that enter the basis through either indepvars ${ }_{\text {series }}$ or seriesvarlist are required to have at least 10 distinct values. Continuous variables with few distinct values provide little information for determining an appropriate basis function and may produce unreliable estimates.
basis (stub [, replace]) specifies that the elements of the basis function generated by npregress series be stored with the specified names.

The option argument $s t u b$ is the prefix used to generate enumerated variables for each element of the basis.

When replace is used, existing variables named with stub are replaced by those from the new computation.
rescale (stub [, replace]) specifies that the rescaled covariates used to generate the basis function be stored with the specified names.
The option argument stub is the prefix used to generate enumerated variable names for the covariates.
When replace is used, existing covariates named with stub are replaced by those from the new computation.
vce(vcetype) specifies the type of standard error reported, which includes types that are robust to some kinds of misspecification (robust), that assume homoskedasticity (ols), and that use bootstrap methods (bootstrap); see [R] vce_option.

## Reporting

level(\#); see [R] Estimation options.
aequations specifies that the auxiliary regression coefficients be reported. By default, only the average marginal effects of the covariates on the outcome are reported.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fint), pformat (\%fmt), sformat (\% fmt), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: iterate(\#), [no]log, tolerance(\#); see [R] Maximize. These options are seldom used.

The following option is available with npregress series but is not shown in the dialog box: coeflegend; see [R] Estimation options.

## Remarks and examples

This entry assumes that you are already familiar with nonparametric regression. Below, we discuss nonparametric series estimation; see [R] npregress intro for an overview of nonparametric regression and the models fit by npregress series and npregress kernel.

Remarks are presented under the following headings:
Overview
Estimation and effects

## Overview

npregress series implements nonparametric series estimation using a B-spline, piecewise polynomial spline, or polynomial basis. The covariates may be continuous or discrete. npregress series allows you to estimate covariate effects and other counterfactuals related to the unknown mean function after estimation.

The word "nonparametric" refers to the fact that the parameter of interest-the mean as a function of the covariates-is given by the unknown function $g\left(\mathbf{x}_{i}\right)$, which is an element of an infinite-dimensional space of functions. In contrast, in a parametric model, the mean for a given value of the covariates, $E\left(y_{i} \mid \mathbf{x}_{i}\right)=f\left(\mathbf{x}_{i}, \boldsymbol{\beta}\right)$, is a known function that is fully characterized by the parameter of interest, $\boldsymbol{\beta}$, which is a finite-dimensional real vector (Shao 2003).

The nonparametric regression model of outcome $y_{i}$ given the $k$-dimensional vector of covariates $\mathbf{x}_{i}$ is given by

$$
\begin{align*}
y_{i} & =g\left(\mathbf{x}_{i}\right)+\varepsilon_{i}  \tag{1}\\
E\left(\varepsilon_{i} \mid \mathbf{x}_{i}\right) & =0 \tag{2}
\end{align*}
$$

where $\varepsilon_{i}$ is the error term. Equations (1) and (2) imply that

$$
E\left(y_{i} \mid \mathbf{x}_{i}\right)=g\left(\mathbf{x}_{i}\right)
$$

Once we account for the information in the covariates, the error term provides no information about the mean of our outcome. The conditional mean function is therefore given by $g\left(\mathbf{x}_{i}\right)$.

The mean estimate we obtain using nonparametric series estimation has the same form of the mean function estimate we obtain using linear regression. The regressors, however, are not variables in the data but functions of the variables. An example would be a $k$ th-order polynomial. Suppose we have one covariate. The elements of the polynomial in this case would be $\left(x_{i}, x_{i}^{2}, \ldots, x_{i}^{k}\right)$. If we define $\mathbf{z}_{i}$ as a vector with elements $\left(x_{i}, x_{i}^{2}, \ldots, x_{i}^{k}\right)$, we may write the estimate of the mean function as

$$
\mathbf{z}_{i}^{\prime} \widehat{\boldsymbol{\beta}}
$$

where $\widehat{\boldsymbol{\beta}}$ has the form of an ordinary least-squares estimate.
npregress series allows us to specify other functional forms for $\mathbf{z}_{i}$ depending on the basis we select: B-spline, piecewise polynomial spline, or polynomial. See Methods and formulas for the formulas for each basis.

Although the estimate of the mean function has the form of a linear regression, the individual coefficients are not easily interpretable. For instance, in our $k$ th-order polynomial example, if $x_{i}$ is continuous, the marginal effect of $x_{i}$ is not a single coefficient but rather is a function of $k$ elements of $\boldsymbol{\beta}$ and the covariate $x_{i}$.

In the example above, we had only one covariate, $x_{i}$. If we have more than one covariate, we approximate the mean function by using interactions of the terms in the basis function for each covariate. For instance, a polynomial of $x_{i}$ and $w_{i}$ would have terms $\left(x_{i}, w_{i}, x_{i} w_{i}, x_{i}^{2}, w_{i}^{2}, \ldots, w_{i}^{k} x_{i}^{k}\right)$. As the number of covariates increases, the number of terms in the basis function increases exponentially. This is referred to in the literature as the curse of dimensionality.
npregress series allows us to reduce the dimensionality by using the nointeract () option to request that some covariates not be interacted with others. For the example above, this is equivalent to specifying a model of the form

$$
\begin{equation*}
y_{i}=g_{1}\left(x_{i}\right)+g_{2}\left(w_{i}\right)+\varepsilon_{i} \tag{3}
\end{equation*}
$$

In (3), $g_{1}\left(x_{i}\right)$ and $g_{2}\left(w_{i}\right)$ are unknown functions, but there are no interactions between $x_{i}$ and $w_{i}$. This ameliorates the curse of dimensionality but imposes more structure to the model.

You may also want to reduce the curse of dimensionality by requesting a parametric component, by using the asis() option, to fit models like this:

$$
\begin{equation*}
y_{i}=g\left(x_{i}\right)+w_{i} \boldsymbol{\beta}+\varepsilon_{i} \tag{4}
\end{equation*}
$$

In (4), $g\left(x_{i}\right)$ is unknown but we assume that $w_{i}$ enters the model linearly.
As mentioned above, the regression coefficients are not easily interpretable. We can, however, estimate marginal effects, as reported in the npregress series output, and use margins to answer specific questions about the effects of covariates on the conditional mean, $g\left(\mathbf{x}_{i}\right)$. We demonstrate this in the examples below.

For detailed introductions to series estimators and the methods implemented by npregress series, see de Boor (2001), Schumaker (2007), Eubank (1999), Schoenberg (1969), Newey (1997), and Chen (2007).

## Estimation and effects

## > Example 1: Nonparametric series regression estimation

dui.dta contains information about the number of monthly drunk driving citations in a local jurisdiction (citations). Suppose we want to know the effect of increasing fines on the number of citations. Because citations is a count variable, we could consider fitting the model with poisson or nbreg. However, both of these estimators make assumptions about the distribution of the data. If these assumptions are not true, we will obtain inconsistent estimates.

By using npregress series, we do not have to make any assumptions about how citations is distributed. We use npregress series to estimate the average marginal effect of drunk driving penalties (fines) on citations.

```
. use https://www.stata-press.com/data/r18/dui
(Fictional data on monthly drunk driving citations)
. npregress series citations fines
Computing approximating function
Minimizing cross-validation criterion
Iteration 0: Cross-validation criterion = 55.15697
Iteration 1: Cross-validation criterion = 55.11413
Computing average derivatives
Cubic B-spline estimation Number of obs = 500
Criterion: cross-validation Number of knots = 3
\begin{tabular}{r|rrrrrr}
\hline citations & Effect & \begin{tabular}{c} 
Robust \\
std. err.
\end{tabular} & \(z\) & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline fines & -8.020769 & .464836 & -17.26 & 0.000 & -8.931831 & -7.109707 \\
\hline
\end{tabular}
```

Note: Effect estimates are averages of derivatives.
The iteration log first tells us that the approximating function is being computed. At this stage, the number of knots of the cubic B-spline is selected using cross-validation. Three knots were selected.

After the approximating function is computed, average marginal effects are computed. This second step is computationally expensive. The computation time increases with the number of elements in the basis function, which in turn increases with the complexity of the mean function we are trying to compute.

The table reports that the average marginal effect of fines on the mean number of citations is -8.02 . Increasing fines, on average, reduces the number of citations.
npregress series generates a system variable for each element of the basis function. Additionally, variables are generated with the rescaled values of the continuous covariates used to construct the basis function. To see the variables that npregress series generated for example 1, we type

| Variable name | Storage type | Display format | Value <br> label | Variable label |
| :---: | :---: | :---: | :---: | :---: |
| __x1rs | double | \%10.0g |  | fines rescaled to [0,1] |
| _x1__b1 | double | \%10.0g |  | Basis term 1 for fines |
| _x1__b2 | double | \%10.0g |  | Basis term 2 for fines |
| _x1__b3 | double | \%10.0g |  | Basis term 3 for fines |
| _x1__b4 | double | \%10.0g |  | Basis term 4 for fines |
| _x1__b5 | double | \%10.0g |  | Basis term 5 for fines |
| _x1__b6 | double | \%10.0g |  | Basis term 6 for fines |
| _x1__b7 | double | \%10.0g |  | Basis term 7 for fines |

To specify a name for each of the elements of the basis function, we can use the basis() option with a stub.

```
. npregress series citations fines, basis(basis)
    (output omitted)
```

We get the following set of names for the elements of the basis function:

| Variable name | Storage type | Display format | Value <br> label | Variable label |
| :---: | :---: | :---: | :---: | :---: |
| basis1 | double | \%10.0g |  | Basis term 1 for fines |
| basis2 | double | \%10.0g |  | Basis term 2 for fines |
| basis3 | double | \%10.0g |  | Basis term 3 for fines |
| basis4 | double | \%10.0g |  | Basis term 4 for fines |
| basis5 | double | \%10.0g |  | Basis term 5 for fines |
| basis6 | double | \%10.0g |  | Basis term 6 for fines |
| basis7 | double | \%10.0g |  | Basis term 7 for fines |

We may also modify the name of the rescaled variable by using the rescale() option.

```
. npregress series citations fines, rescale(rescaled)
    (output omitted)
```

This will give us

| variable name | storage type | display <br> format | value <br> label | variable label |
| :---: | :---: | :---: | :---: | :---: |
| rescaled1 | double | $\% 10.0 \mathrm{~g}$ |  | fines rescaled to [ 0,1$]$ |

## > Example 2: Estimation with more than one regressor

We now extend example 1. In addition to fines, we model citations as a function of whether the jurisdiction is small, medium, or large (csize) and whether there is a college in the jurisdiction (college).

| Computing approximating function |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Minimizing cross-validation criterion |  |  |  |  |  |  |
| Iteration 0: Cross-validation criterion $=30.26251$ |  |  |  |  |  |  |
| Computing average derivatives |  |  |  |  |  |  |
| Cubic B-spline estimation Criterion: cross-validation |  |  | Number of obs Number of knots |  |  | $\begin{array}{r} 500 \\ 1 \end{array}$ |
|  |  |  |  |  |  |  |
| citations | Robust |  |  |  |  |  |
| fines | -7.787386 | . 2917941 | -26.69 | 0.000 | -8.359292 | -7.215481 |
| $\begin{gathered} \text { csize } \\ \text { (Medium } \end{gathered}$ |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| Small) | 4.732592 | . 5087968 | 9.30 | 0.000 | 3.735368 | 5.729815 |
| (Large |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| Small) | 10.91757 | . 5350892 | 20.40 | 0.000 | 9.868813 | 11.96632 |
| $\begin{aligned} & \text { college } \\ & \text { (College } \end{aligned}$ |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| Not coll..) | 6.514286 | . 5958949 | 10.93 | 0.000 | 5.346353 | 7.682218 |

Note: Effect estimates are averages of derivatives for continuous covariates and averages of contrasts for factor covariates.

The average marginal effect of fines is -7.79 , slightly less in magnitude than the -8.02 that we estimated in example 1. The output also shows effects for the variables csize and college. In these categorical variables, the effects are differences instead of derivatives. For example, if every jurisdiction in the population were a college town, we would expect 6.51 more citations than if none were college towns.
> Example 3: Expected citations for different levels of fines
The npregress series command reported that the average marginal effect of fines on number of citations is negative. We can use margins to further explore the relationship between level of fines and expected number of citations. What would we expect if all jurisdictions set fines to $\$ 8,000$ ? What if they all set fines to $\$ 9,000$ ? $\$ 10,000$ ? $\$ 11,000$ ? We use the at (fines= (8 91011 ) ) option with margins to estimate these expected values.

| Predictive ma Model VCE: R |  |  |  |  | Number of obs $=500$ |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Expression: Mean function, predict() |  |  |  |  |  |  |
| 1._at: fines = 8 |  |  |  |  |  |  |
| 2._at: fines $=9$ |  |  |  |  |  |  |
| 3._at: fines $=10$ |  |  |  |  |  |  |
| 4._at: fines = 11 |  |  |  |  |  |  |
| Delta-method |  |  |  |  |  |  |
|  | Margin | std. er | z | $P>\|z\|$ | [95\% con | interval] |
| _at |  |  |  |  |  |  |
| 1 | 49.58234 | 1.47392 | 33.64 | 0.000 | 46.69351 | 52.47117 |
| 2 | 28.35154 | . 5730302 | 49.48 | 0.000 | 27.22842 | 29.47466 |
| 3 | 20.40163 | . 3320855 | 61.43 | 0.000 | 19.75075 | 21.0525 |
| 4 | 14.78085 | . 4297201 | 34.40 | 0.000 | 13.93862 | 15.62309 |

There appears to be a dramatic drop in the expected number of citations as fines increase from $\$ 8,000$ to $\$ 9,000$. We can visualize these results if we type marginsplot.

Predictive margins with $95 \%$ Cls


Figure 1.
Are there significant differences in the expected number of citations as we increase fines in increments of $\$ 1,000$ ? If we use the reverse-adjacent contrast operator, ar., with margins, we can estimate these differences and perform tests.


When fines are increased from $\$ 8,000$ to $\$ 9,000$, we expect a decrease of 21.23 in the number of citations. Smaller but still statistically significant decreases in the number of citations are expected as fines are increased from $\$ 9,000$ to $\$ 10,000$ and from $\$ 10,000$ to $\$ 11,000$.

## Example 4: Estimating the effect for different levels of jurisdiction size

Now, we estimate the effect of increasing fines for different jurisdiction sizes.

```
. margins csize, dydx(fines)
Average marginal effects Number of obs = 500
Model VCE: Robust
Expression: Mean function, predict()
dy/dx wrt: fines
```

|  | Delta-method |  |  |  |  |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: | ---: |
|  | dy/dxstd.err.$\quad \mathbf{z}$ | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |  |  |
| fines |  |  |  |  |  |  |
| csize |  |  |  |  |  |  |
| Small | -5.992484 | .4491224 | -13.34 | 0.000 | -6.872747 | -5.11222 |
| Medium | -7.740284 | .4366709 | -17.73 | 0.000 | -8.596144 | -6.884425 |
| Large | -10.20492 | .564166 | -18.09 | 0.000 | -11.31067 | -9.099178 |

If all jurisdictions were small but other characteristics were as they are observed, then we expect that the marginal effect of fines would be -5.99 . We see that the effect is more extreme as the size of the jurisdiction increases. If all jurisdictions were large, we expect that the average marginal effect of fines would be -10.20 .

We can further explore the effects of fines for different jurisdiction sizes by estimating the expected number of citations with fines at specific levels.

```
. margins csize, at(fines=(8(1)11))
    (output omitted)
```

To visualize the effect, we type marginsplot.


Figure 2.
For each jurisdiction size, we see that on average higher fines result in fewer citations. We also see that the effect of changing fine levels is nonlinear and differs across the counterfactual jurisdiction size. For instance, as fines increase from $\$ 8,000$ to $\$ 9,000$, the expected number of citations decreases faster for small jurisdictions than for medium ones.

## Stored results

npregress series stores the following in e():
Scalars

```
e(N)
e(r2)
e(r2_a)
e(converged)
e(order)
e(rank)
        number of observations
    R
    adjusted R }\mp@subsup{}{}{2
    1 if converged, 0 otherwise
    order of basis function
    rank of e(V)
```

Macros

```
        e(cmd)}\mp@code{e(cmdline)
        e(cmd)}\mp@code{e(cmdline)
        e(cmd)}\mp@code{e(cmdline)
        e(cmd)}\mp@code{e(cmdline)
        e(cmd)}\mp@code{e(cmdline)
        e(cmd)}\mp@code{e(cmdline)
        e(cmd)}\mp@code{e(cmdline)
        e(cmd)}\mp@code{e(cmdline)
        e(cmd)}\mp@code{e(cmdline)
        e(cmd)}\mp@code{e(cmdline)
        e(cmd)}\mp@code{e(cmdline)
        e(cmd)}\mp@code{e(cmdline)
        e(cmd)}\mp@code{e(cmdline)
        e(cmd)}\mp@code{e(cmdline)
        e(cmd)}\mp@code{e(cmdline)
        e(cmd)}\mp@code{e(cmdline)
        e(cmd)}\mp@code{e(cmdline)
```

    e(marginsok) predictions allowed by margins
    Macros
npregress
command as typed
name of dependent variable
bsplines, splines, or polynomials
weight type
weight expression
title in estimation output
vcetype specified in vce()
title used to label Std. err.
number of knots selected
variables used in calculation of checksum
the checksum
program used to implement estat
program used to implement predict
b V
signals to the margins command
predictions disallowed by margins

Matrices
e(b)
coefficient vector
$e(V) \quad$ variance-covariance matrix of estimators
e(V_modelbased)
model-based variance
e(ilog)
Functions
e(sample) marks estimation sample
In addition to the above, the following is stored in r() :

Matrices $r$ (table) matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

Methods and formulas are presented under the following headings:

Overview<br>Polynomials<br>Piecewise polynomial splines<br>$B$-splines<br>Model selection<br>Cross-validation<br>Generalized cross-validation<br>Mallows's $C_{p}$<br>AIC and BIC

## Overview

The regression model of outcome $y_{i}$ given the $k$-dimensional vector of covariates $\mathbf{x}_{i}$ was defined in (1) and (2) of Remarks and examples and is repeated here:

$$
\begin{align*}
y_{i} & =g\left(\mathbf{x}_{i}\right)+\varepsilon_{i}  \tag{1}\\
E\left(\varepsilon_{i} \mid \mathbf{x}_{i}\right) & =0 \tag{2}
\end{align*}
$$

where $\varepsilon_{i}$ is the error term. The covariates may include discrete and continuous variables. Equations (1) and (2) imply that

$$
E\left(y_{i} \mid \mathbf{x}_{i}\right)=g\left(\mathbf{x}_{i}\right)
$$

As discussed in Remarks and examples, series estimators have the form of ordinary least squares. Thus, we can write the estimate of the mean function as

$$
\begin{equation*}
\widehat{E}\left(y_{i} \mid \mathbf{x}_{i}\right)=\mathbf{z}\left(\mathbf{x}_{i}\right) \widehat{\boldsymbol{\beta}} \tag{5}
\end{equation*}
$$

where $\mathbf{z}\left(\mathbf{x}_{i}\right)$ is a known $q$-dimensional vector for which every one of the $q$ terms is a function of the $k$-dimensional vector of covariates $\mathbf{x}_{i}$. Let $n$ be the sample size. If we define $\mathbf{Z}$ as the $n \times q$ matrix formed by the $\mathbf{z}\left(\mathbf{x}_{i}\right)$ for each individual $i$, then the $q$-dimensional coefficient vector $\widehat{\boldsymbol{\beta}}$ is the ordinary least-squares vector that comes from regressing the $n \times 1$ outcome vector $\mathbf{y}$ on $\mathbf{Z}$ and has the known form

$$
\begin{equation*}
\widehat{\boldsymbol{\beta}}=\left(\mathbf{Z}^{\prime} \mathbf{Z}\right)^{-1}\left(\mathbf{Z}^{\prime} \mathbf{y}\right) \tag{6}
\end{equation*}
$$

Each one of the series estimators has a different form for $\mathbf{z}\left(\mathbf{x}_{i}\right)$. Below, we define $\mathbf{z}\left(\mathbf{x}_{i}\right)$ for polynomials, piecewise polynomial splines, and B-splines.

## Polynomials

For a polynomial of order 1 with $k$ continuous covariates, $\mathbf{x}_{i} \equiv\left(x_{i 1}, x_{i 2}, \ldots, x_{i k}\right), \mathbf{z}\left(\mathbf{x}_{i}\right)$ is

$$
\begin{equation*}
\mathbf{z}\left(\mathbf{x}_{i}\right)=\left(x_{i 1}, x_{i 2}, \ldots, x_{i k}\right) \tag{P1}
\end{equation*}
$$

For notational convenience, we will refer to the polynomial above as $P 1$, which also corresponds to the name we gave to the equation. More formally, we could have written $\left(x_{i 1}, x_{i 2}, \ldots, x_{i k}\right) \equiv P 1$. We will maintain this notational convention below.

A polynomial of order 2 with $k$ continuous covariates includes $P 1$ and second-order terms:

$$
\begin{equation*}
\mathbf{z}\left(\mathbf{x}_{i}\right)=\left(P 1, x_{i 1}^{2}, x_{i 1} x_{i 2}, \ldots, x_{i 1} x_{i k}, x_{i 2} x_{i 3}, \ldots, x_{i k}^{2}\right) \tag{P2}
\end{equation*}
$$

A third-order polynomial with $k$ continuous covariates includes the terms in $P 2$ (which already includes $P 1$ ) and third-order terms:

$$
\begin{equation*}
\mathbf{z}\left(\mathbf{x}_{i}\right)=\left(P 2, x_{i 1}^{3}, x_{i 1}^{2} x_{i 2}, \ldots, x_{i 1}^{2} x_{i k}, x_{i 1} x_{i 2} x_{i 3}, \ldots, x_{i k}^{3}\right) \tag{P3}
\end{equation*}
$$

This recursive relationship continues. Thus, fourth-order polynomials includes the terms in P3 (which already includes $P 1$ and $P 2$ ) plus fourth-order terms.

For the polynomials above and all series estimators below, discrete covariates enter the model in levels, and each level is interacted with all other covariates in the model.

## Piecewise polynomial splines

Piecewise polynomial splines are formed by a polynomial and functions of the form

$$
\max \left(x_{i k}-t_{1 k}, 0\right)
$$

In the expression above, $t_{1 k}$ is a constant that is called a knot. The subscript of $t_{1 k}$ indicates that it is the first knot of the continuous covariate $\mathbf{x}_{k}$. The $\max (\cdot)$ function is 0 when $x_{i k}<t_{1 k}$ and is $x_{i k}-t_{1 k}$ otherwise. npregress series selects a set of knots for each one of the continuous covariates.

The regressors for npregress series using a piecewise polynomial spline of order 3 with one continuous covariate, $\mathbf{x}_{1}$, and $k$ knots, $t_{11}<t_{21}<\cdots<t_{k 1}$, are given by

$$
\begin{align*}
\mathbf{z}\left(x_{i 1}\right)= & \left\{x_{i 1}, x_{i 1}^{2}, x_{i 1}^{3}, \max \left(x_{i 1}-t_{11}, 0\right)^{3}, \max \left(x_{i 1}-t_{21}, 0\right)^{3}, \ldots\right. \\
& \left.\max \left(x_{i 1}-t_{k 1}, 0\right)^{3}\right\} \tag{S1}
\end{align*}
$$

Equivalently, for another continuous covariate, $\mathbf{x}_{2}$, and $k$ knots, $t_{12}<t_{22}<\cdots<t_{k 2}$, we have

$$
\begin{align*}
\mathbf{z}\left(x_{i 2}\right)= & \left\{x_{i 2}, x_{i 2}^{2}, x_{i 2}^{3}, \max \left(x_{i 2}-t_{12}, 0\right)^{3}, \max \left(x_{i 2}-t_{22}, 0\right)^{3}, \ldots,\right. \\
& \left.\max \left(x_{i 2}-t_{k 2}, 0\right)^{3}\right\} \tag{S2}
\end{align*}
$$

To get $\mathbf{z}\left(x_{i 1}, x_{i 2}\right)$ for $\mathbf{x}_{1}$ and $\mathbf{x}_{2}$ and $k$ knots, we include all terms in $S 1$ and $S 2$ as well as all the terms that result from the interaction of their terms. We write it succinctly as

$$
\begin{equation*}
\mathbf{z}\left(x_{i 1}, x_{i 2}\right)=\{S 1, S 2,(S 1)(S 2)\} \tag{S12}
\end{equation*}
$$

The description above refers to the default third-order piecewise polynomial spline. Below, we describe the cases for piecewise polynomial splines of order 1 and order 2. Going back to the one covariate case, if we want a piecewise polynomial spline of order 1 with $k$ knots, we have

$$
\mathbf{z}\left(x_{i 1}\right)=\left\{x_{i 1}, \max \left(x_{i 1}-t_{11}, 0\right), \max \left(x_{i 1}-t_{21}, 0\right), \ldots, \max \left(x_{i 1}-t_{k 1}, 0\right)\right\}
$$

And for order 2 with $k$ knots and one covariate, we have

$$
\mathbf{z}\left(x_{i 1}\right)=\left\{x_{i 1}, x_{i 1}^{2}, \max \left(x_{i 1}-t_{11}, 0\right)^{2}, \max \left(x_{i 1}-t_{21}, 0\right)^{2}, \ldots, \max \left(x_{i 1}-t_{k 1}, 0\right)^{2}\right\}
$$

If we have more than one covariate, the logic of interacting the expressions for each covariate is the same as the logic we used for the third-order piecewise polynomial spline in (S12).

To construct $\mathbf{z}\left(\mathbf{x}_{i}\right)$, continuous covariates are rescaled to be between 0 and 1 with the expression

$$
\left\{\mathbf{x}_{i}-\min \left(\mathbf{x}_{i}\right)\right\}\left\{\frac{1}{\max \left(\mathbf{x}_{i}\right)-\min \left(\mathbf{x}_{i}\right)}\right\}
$$

This rescaling is used to construct the piecewise polynomial spline and B-spline bases.

## B-splines

To construct a B-spline basis, we need to define knots that are on the interior of the range of the covariates and knots that are at the upper and lower limits of the range or outside the range. The number of knots that are not in the interior differs depending on the order of the B-spline. For a B -spline of order 1 with $k$ interior knots, $t_{1}, t_{2}, \ldots, t_{k}$, we need 4 additional knots. The set of knots for a first-order B-spline is therefore

$$
t_{-1}, t_{0}, t_{1}, \ldots, t_{k}, t_{k+1}, t_{k+2}
$$

We added $t_{-1}, t_{0}, t_{k+1}$, and $t_{k+2}$ to the interior knots. By convention, $t_{-1}=t_{0}$ and $t_{k+1}=t_{k+2}$. For a B-spline of order 2 with $k$ interior knots, we need 6 additional knots. The set of knots is

$$
t_{-2}, t_{-1}, t_{0}, t_{1}, \ldots, t_{k}, t_{k+1}, t_{k+2}, t_{k+3}
$$

For a B-spline of order 3, the set of knots is

$$
t_{-3}, t_{-2}, t_{-1}, t_{0}, t_{1}, \ldots, t_{k}, t_{k+1}, t_{k+2}, t_{k+3}, t_{k+4}
$$

We first define a first-order B-spline for one continuous covariate $\mathbf{x}$ with $k$ interior knots. Let $\mathbf{t}_{j}$ denote an $n \times 1$ vector for which all elements take the value of the $j$ th knot $t_{j}$. The B-spline basis is formed by $k+2$ functions of the form

$$
\begin{align*}
B_{j, 1} & =\frac{\left(\mathbf{x}-t_{j}\right)}{t_{j+1}-t_{j}} \mathbf{1}\left(\mathbf{t}_{j} \leq \mathbf{x}<\mathbf{t}_{j+1}\right)+\frac{\left(t_{j+2}-\mathbf{x}\right)}{t_{j+2}-t_{j+1}} \mathbf{1}\left(\mathbf{t}_{j+1} \leq \mathbf{x}<\mathbf{t}_{j+2}\right)  \tag{7}\\
j & =-1,0,1,2, \ldots, k
\end{align*}
$$

Above, we use the indicator function $\mathbf{1}(\cdot)$, which is 1 when the condition inside the parentheses is satisfied and is 0 otherwise. Also, any term for which $t_{j+1}=t_{j}$ or $t_{j+2}=t_{j+1}$ is considered to be a vector of 0s.

The function $\mathbf{z}(\mathbf{x})$ used to estimate $g(\mathbf{x})$ is given by

$$
\mathbf{z}(\mathbf{x})=\left(B_{-1,1}, B_{0,1}, B_{1,1}, \ldots, B_{k, 1}\right)
$$

We now define a second-order B-spline for one continuous covariate $\mathbf{x}$ with $k$ interior knots. The basis is constructed using the relationship given by

$$
\begin{aligned}
B_{j, 2} & =\frac{\left(\mathbf{x}-t_{j}\right)}{t_{j+2}-t_{j}} B_{j, 1}+\frac{\left(t_{j+3}-\mathbf{x}\right)}{t_{j+3}-t_{j+1}} B_{j+1,1} \\
j & =-2,-1,0,1,2, \ldots, k
\end{aligned}
$$

where $B_{j, 1}$ and $B_{j+1,1}$ come from (7) above. Thus, second-order B-splines are a function of first-order B -splines, and as we will see below, third-order B-splines are a function of second-order B-splines. This recursion continues into higher orders, but npregress series stops at B-splines of order 3.

The function $\mathbf{z}(\mathbf{x})$ for the second-order B-spline is given by

$$
\mathbf{z}(\mathbf{x})=\left(B_{-2,2}, B_{-1,2}, B_{0,2}, B_{1,2}, \ldots, B_{k, 2}\right)
$$

The terms of the basis for a third-order B-spline are given by

$$
\begin{aligned}
B_{j, 3} & =\frac{\left(\mathbf{x}-t_{j}\right)}{t_{j+3}-t_{j}} B_{j, 2}+\frac{\left(t_{j+4}-\mathbf{x}\right)}{t_{j+4}-t_{j+1}} B_{j+1,2} \\
j & =-3,-2,-1,0,1,2, \ldots, k
\end{aligned}
$$

and the function $\mathbf{z}(\mathbf{x})$ for the third-order B -spline is

$$
\begin{equation*}
\mathbf{z}(\mathbf{x})=\left(B_{-3,3}, B_{-2,3}, B_{-1,3}, B_{0,3}, B_{1,3}, \ldots, B_{k, 3}\right) \tag{B1}
\end{equation*}
$$

As was the case with piecewise polynomial splines, when there is more than one covariate, you include all functions of the form $(B 1)$ and their interactions to form expressions like the one in (S12).

## Model selection

Below, we define the criteria used for model selection. In the case of B-splines and piecewise polynomial splines, npregress series selects the number of knots to be used for estimation. In the case of a polynomial basis, npregress series selects the order of the polynomial.

Let us first define the squared residuals, $e_{i}^{2}$, where $e_{i}=y_{i}-\widehat{g}\left(\mathbf{x}_{i}\right)$ and $\widehat{g}(\cdot)$ is the mean function estimate defined in (5). We denote the residuals for the regressions below as $e_{i}\left(\mathbf{t}_{k}\right)$ instead of $e_{i}$ to signal that the estimates we obtain are a function of the set of knots, $\mathbf{t}_{k}$, used. In the case of polynomials, $\mathbf{t}_{k}$ will refer to the degree of the polynomial instead of knots.

## Cross-validation

The cross-validation criterion, $\operatorname{CV}\left(\mathbf{t}_{k}\right)$, is defined by

$$
\begin{equation*}
\mathrm{CV}\left(\mathbf{t}_{k}\right)=\frac{1}{n} \sum_{i=1}^{n} \frac{e_{i}\left(\mathbf{t}_{k}\right)^{2}}{\left(1-h_{i i}\right)^{2}} \tag{8}
\end{equation*}
$$

In (8), $h_{i i}$ are the diagonal elements of the matrix $\mathbf{Z}\left(\mathbf{Z}^{\prime} \mathbf{Z}\right) \mathbf{Z}^{\prime}$, where $Z$ is defined in (6) above and $n$ is the size of the estimation sample.
npregress series computes $\mathrm{CV}\left(\mathbf{t}_{k}\right)$ for different sets of knots, $\mathbf{t}_{1}, \mathbf{t}_{2}, \ldots, \mathbf{t}_{k}, \ldots$, where $\mathbf{t}_{1} \subset \mathbf{t}_{2} \subset \ldots \subset \mathbf{t}_{k} \subset \ldots$, and then selects the model with the smallest value for the cross-validation criterion.

## Generalized cross-validation

The generalized cross-validation criterion, $\operatorname{GCV}\left(\mathbf{t}_{k}\right)$, is given by

$$
\operatorname{GCV}\left(\mathbf{t}_{k}\right)=\frac{1}{n} \sum_{i=1}^{n} \frac{e_{i}\left(\mathbf{t}_{k}\right)^{2}}{\{1-(K / n)\}^{2}}
$$

where $K$ is the number of estimated parameters and the other arguments are equivalent to those defined in (8). As with cross-validation, $\operatorname{GCV}\left(\mathbf{t}_{k}\right)$ is computed for a set of models with an increasing number of nested knots, in the case of piecewise polynomial splines and B-splines, and of polynomial order in the case of polynomials. The minimum of the sequence is the selected model.

## Mallows's $\mathbf{C}_{p}$

$$
\text { Mallows }\left(\mathbf{t}_{k}\right)=\frac{1}{n} \sum_{i=1}^{n} e_{i}\left(\mathbf{t}_{k}\right)^{2}\left(1+\frac{2 K}{n}\right)
$$

As with cross-validation, Mallows $\left(\mathbf{t}_{k}\right)$ is computed for a set of models with an increasing number of nested knots, and the minimum of the sequence is the selected model.

## AIC and BIC

See Methods and formulas in [R] estat ic.

## References

Chen, X. 2007. Large sample sieve estimation of semi-nonparametric models. In Vol. 6B of Handbook of Econometrics, ed. J. J. Heckman and E. E. Leamer, 5549-5632. Amsterdam: Elsevier. https://doi.org/10.1016/S1573-4412(07)06076-X.

Chetverikov, D., D. Kim, and D. Wilhelm. 2018. Nonparametric instrumental-variable estimation. Stata Journal 18: 937-950.
de Boor, C. 2001. A Practical Guide to Splines. Rev. ed. New York: Springer.
Eubank, R. L. 1999. Nonparametric Regression and Spline Smoothing. 2nd ed. New York: Dekker.
Hansen, B. E. 2009. University of Wisconsin-Madison, ECON 718, NonParametric Econometrics, Spring 2009, course notes. Last visited on 2019/01/15. https://www.ssc.wisc.edu/~bhansen/718/718.htm.
——. 2018. Econometrics. https://www.ssc.wisc.edu/~bhansen/econometrics/Econometrics.pdf.
Li, J., Z. Liao, and M. Gao. 2020. Uniform nonparametric inference for time series using Stata. Stata Journal 20 : 706-720.

Li, J., Z. Liao, R. Quaedvlieg, and W. Zhou. 2022. Conditional evaluation of predictive models: The cspa command. Stata Journal 22: 924-940.

Li, Q., and J. S. Racine. 2007. Nonparametric Econometrics: Theory and Practice. Princeton, NJ: Princeton University Press.

Newey, W. K. 1997. Convergence rates and asymptotic normality for series estimators. Journal of Econometrics 79: 147-168. https://doi.org/10.1016/S0304-4076(97)00011-0.
Schoenberg, I. J., ed. 1969. Approximations with Special Emphasis on Spline Functions. New York: Academic Press.
Schumaker, L. L. 2007. Spline Functions: Basic Theory. 3rd ed. Cambridge: Cambridge University Press.
Shao, J. 2003. Mathematical Statistics. 2nd ed. New York: Springer.

## Also see

[R] npregress series postestimation - Postestimation tools for npregress series
$[\mathrm{R}]$ npregress intro - Introduction to nonparametric regression
[R] kdensity - Univariate kernel density estimation
[R] Ipoly - Kernel-weighted local polynomial smoothing
[ $R$ ] makespline - Spline generation
[U] 20 Estimation and postestimation commands

## Title

npregress series postestimation - Postestimation tools for npregress series

| Postestimation commands | predict | margins |
| :--- | :--- | :--- |
| Remarks and examples | Methods and formulas | Reference |
| Also see |  |  |

## Postestimation commands

The following standard postestimation commands are available after npregress series:

| Command | Description |
| :--- | :--- |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) <br> estaloging estimation results |
| etable | table of estimation results <br> point estimates, standard errors, testing, and inference for linear combinations of <br> coefficients |
| lincom | marginal means, predictive margins, marginal effects, and average marginal effects <br> graph the results from margins (profile plots, interaction plots, etc.) |
| marginsplot | point estimates, standard errors, testing, and inference for nonlinear combinations <br> of coefficients |
| nlcom | conditional means and residuals |
| predict | point estimates, standard errors, testing, and inference for generalized predictions <br> predictnl <br> test |
| testnl tests of simple and composite linear hypotheses |  |$\quad$| Wald tests of nonlinear hypotheses |
| :--- |

## predict

## Description for predict

predict creates a new variable containing predictions such as conditional mean of the outcome, residuals, or score of the mean function.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

```
predict [type] newvar [if] [in] [, statistic atsample tolerance(#)]
```

statistic Description
Main

| mean | conditional mean of the outcome; the default |
| :--- | :--- |
| residuals | residuals |
| score | score; equivalent to residuals |

These statistics are available for the estimation sample only.

## Options for predict

$\qquad$ Main
mean, the default, calculates the conditional mean of the outcome variable.
residuals calculates the residuals.
score is a synonym for residuals.
atsample restricts predictions to the range of covariates in the data. If requested predictions extend beyond the range of the data, atsample will compute predictions only for observations within the range of the original data and will exclude those observations that are beyond the range of the data.

By default, predictions will not be computed if any covariate is set to a value outside the range of the data, unless atsample or tolerance() is specified.
tolerance(\#) sets the tolerance for predictions outside the range of the covariates.
By default, predictions will not be computed if any covariate is set to a value outside the range of the data, unless tolerance () or atsample is specified.

## margins

## Description for margins

margins estimates margins of the conditional mean.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [options]
```

statistic Description
Main
mean conditional mean of the outcome; the default
residuals not allowed with margins
score not allowed with margins
options Description
SE
vce(vcetype) vcetype may be delta, unconditional, or bootstrap
reps(\#) equivalent to vce(bootstrap, reps (\#))
seed (\#) set random-number seed to \#; must also specify reps (\#)
nose do not estimate standard errors
Reporting
citype (citype) method to compute bootstrap confidence intervals; default is
citype(percentile)
citype Description
percentile percentile confidence intervals; the default
bc bias-corrected confidence intervals
normal normal-based confidence intervals

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see $[R]$ margins.

## Options for margins

$\qquad$ SE
vce(delta), vce(unconditional), and vce(bootstrap) specify how the VCE and, correspondingly, standard errors are calculated.
vce(delta) is the default. The delta method is applied to the formula for the response and the VCE of the estimation command. This method assumes that values of the covariates used to calculate the response are given or, if all covariates are not fixed using at (), that the data are given.
vce (unconditional) specifies that the covariates that are not fixed be treated in a way that accounts for their having been sampled. The VCE is estimated using the linearization method. This method allows for heteroskedasticity or other violations of distributional assumptions in the same manner as vce (robust), which is the default for npregress series.
vce(bootstrap) specifies that bootstrap standard errors be reported; see [R] vce_option. We recommend that you select the number of replications using reps (\#) instead of specifying vce (bootstrap), which defaults to 50 replications. Be aware that the number of replications needed to produce good estimates of the standard errors varies depending on the problem.
reps(\#) specifies the number of bootstrap replications to be performed. Specifying this option is equivalent to specifying vce(bootstrap, reps(\#)).
seed (\#) sets the random-number seed. You must specify reps(\#) with seed(\#).
nose suppresses calculation of the VCE and standard errors.

[^41]
## Remarks and examples

For examples of margins after npregress series, see example 3 and example 4 in $[R]$ npregress series.

For an example of marginsplot, see example 4 in [R] npregress series.

## Methods and formulas

The formulas used by predict and margins for the conditional mean function and the mean marginal effect of a covariate are given in Methods and formulas of $[R]$ npregress series.

## Reference

Cattaneo, M. D., and M. Jansson. 2018. Kernel-based semiparametric estimators: Small bandwidth asymptotics and bootstrap consistency. Econometrica 86: 955-995. https://doi.org/10.3982/ECTA12701.

## Also see

[R] npregress series - Nonparametric series regression
$[R]$ bootstrap postestimation - Postestimation tools for bootstrap
[U] 20 Estimation and postestimation commands

## Title

nptrend - Tests for trend across ordered groups

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| Acknowledgments | References | Also see |  |

## Description

nptrend performs four different nonparametric tests for trend: the Cochran-Armitage test, the Jonckheere-Terpstra test, the linear-by-linear trend test, and a test using ranks developed by Cuzick.

## Quick start

Cochran-Armitage test for the association of a binary outcome in $y$ and the ordered values of the categories of $x$

```
nptrend y, group(x) carmitage
```

Same as above, but report an exact $p$-value calculated using Monte Carlo permutations nptrend y, group(x) carmitage exact

Same as above, but perform 100,000 Monte Carlo permutations rather than the default of 10,000, and set the random-number seed for reproducibility nptrend y, group(x) carmitage exact(montecarlo, reps(100000) rseed(1234))

Rather than testing the trend in y by the values of x , which are $1,2,3$, test the trend by $1,2,4$ nptrend y, group(x) carmitage scoregroup(1 24 4)

Jonckheere-Terpstra test for trend in y across ordered categories of catvar nptrend y, group(catvar) jterpstra

Same as above, but perform the linear-by-linear trend test
nptrend y, group(catvar) linear
Same as above, but perform Cuzick's test using ranks
nptrend y, group(catvar) cuzick

## Menu

Statistics $>$ Nonparametric analysis $>$ Tests of hypotheses $>$ Trend test across ordered groups

## Syntax

nptrend varname $[$ if $][$ in $][$ weight $], \underline{\operatorname{group}(g r o u p v a r) ~ t e s t o p t ~[o p t i o n s] ~}$
options
Description
Main

* group (groupvar)
scoreresponse (numlist)
scoregroup (numlist)
exact[(exact_specs)]
nolabel
notable
ordered group variable
scores for the response varname
scores for the groups specified by groupvar
report an exact $p$-value
display numerical values of groupvar rather than value labels
do not display the table of mean response scores by group
* testopt
carmitage
jterpstra
linear
cuzick

Description
Cochran-Armitage test for trend
Jonckheere-Terpstra test for trend
linear-by-linear test for trend
use ranks for the response scores in the test due to Cuzick

* group (groupvar) and one of the choices for testopt are required.
collect is allowed; see [U] 11.1.10 Prefix commands.
fweights are allowed, except when exact is specified; see [U] 11.1.6 weight.


## Options

$\qquad$ Main
group (groupvar) defines the ordered groups across which differences in the response varname are to be tested. group() is required.

When option scoregroup() is not specified, the values of groupvar are the group scores. When scoregroup() is specified, groupvar defines the ordered categories, and the scores are the values specified in scoregroup(). When scoregroup() is specified, groupvar must be a positive integer (with gaps allowed).
testopt specifies the test that is reported. It is required and must be one of carmitage, jterpstra, linear, or cuzick.
carmitage specifies that the Cochran-Armitage test for trend be reported. Typically, the response varname is a $0 / 1$ variable, and the values of groupvar represent a set of ordered categories. It gives a test of the trend of the proportions of positive responses in the groups, with the hypothesized trend given by the values of groupvar (or the values of scoregroup() if specified). If the response varname is not $0 / 1$, it can be valued $1 / 2$ and mapped to $0 / 1$ or $1 / 0$ using scoreresponse (0 1) or scoreresponse (10), respectively.
Response and group specifications for carmitage can be flipped. The groupvar variable can be a $0 / 1$ predictor, and the response varname can contain ordered outcomes. If groupvar is not $0 / 1$, it can be valued $1 / 2$ and mapped to $0 / 1$ or $1 / 0$ using scoregroup ( 01 ) or scoregroup ( 1 0 ), respectively.
jterpstra specifies that the Jonckheere-Terpstra test for trend be reported. The response varname contains outcomes, which can be ordered categories or continuous values, and the groupvar variable is an ordered group indicator that is believed to be a predictor of outcome. The Jonckheere-Terpstra statistic is dependent on which is the response variable and which is the group variable. If response and group variables are interchanged, the value of the statistic will be different.

Only the orderings given by the response varname and groupvar are used to calculate the Jonckheere-Terpstra statistic. Different numerical values that give the same orderings produce the same statistic. Hence, this statistic is used when a test for trend dependent only on order is desired. Typically, the options scoreresponse() and scoregroup() are not specified. However, either or both of these options can be used to change the orderings given by varname and groupvar.
linear specifies that the linear-by-linear test for trend be reported. The linear-by-linear statistic is symmetric in the response variable varname and the group indicator groupvar. Interchanging response and group variables produces the same result. Despite the name "linear", nonlinear trends can be tested. The trend, linear or nonlinear, is based on the numerical values of the response and group variables. scoreresponse() and scoregroup() can be specified to test different trends.
cuzick specifies that ranks be used for the responses and that the test for trend across ordered groups developed by Cuzick (1985) be reported. Because ranks are used, only the ordering given by response variable varname matters. The numerical values of groupvar are used in calculation, and different values, even those that give the same ordering, will result in different values of the statistic for Cuzick's test. scoreresponse() can be specified to change the ordering of the responses. scoregroup() can be specified to test different trends by group.
scoreresponse (numlist) specifies scores for the responses in varname. When specified, varname must contain only positive integers (with gaps allowed), and the response score for varname $=i$ is the $i$ th number in numlist. When scoreresponse() is not specified, the response scores are the values of the response varname.
scoregroup (numlist) specifies scores for the groups specified by groupvar. When specified, groupvar must contain only positive integers (with gaps allowed), and the group score for groupvar $=i$ is the $i$ th number in numlist. When scoregroup() is not specified, the group scores are the values of groupvar.
exact and exact (exact_specs) specify that an exact $p$-value be reported.
exact specifies that an exact $p$-value from a Monte Carlo permutation test be reported. exact is a synonym for exact (montecarlo).
exact (montecarlo enumerate [, options]) specifies that an exact $p$-value be reported in addition to the approximate or asymptotic $p$-value. Specifying exact (montecarlo) does a Monte Carlo permutation test. Specifying exact (enumerate) does an enumeration of all possible permutations. Because the number of all possible permutations is typically extremely large, enumeration is feasible only for very small datasets. The number of permutations will be displayed, and you can click on Break to stop the computation. The exact $p$-value is computed by permute.
exact(montecarlo[, options]) allows options show, reps(\#), rseed(\#), saving (filename [, sav_options]), level(\#), dots (\#), nodots, and eps (\#). The show option specifies that the table produced by permute also be displayed. By default, 10,000 Monte Carlo permutations are done. That is, the default is the same as specifying exact(montecarlo, reps(10000)). The default for $\operatorname{dots}()$ is dots(100) when reps() is $\geq 10,000$; otherwise, it is dots (1). See Options in [R] permute.
exact (enumerate [, options]) allows options show, saving (filename [, sav_options]), dots (\#), nodots, and eps (\#). The show option specifies that the table produced by permute also be displayed. The default for dots() is dots (100). See Options in [R] permute.
nolabel causes numerical values of groupvar to be displayed in the table of mean response scores rather than value labels.
notable suppresses the display of the table of mean response scores by group.

## Remarks and examples

Remarks are presented under the following headings:

Introduction<br>Cochran-Armitage test<br>Jonckheere-Terpstra test<br>Linear-by-linear trend test<br>Cuzick's test

## Introduction

nptrend performs four different nonparametric tests for trend: the Cochran-Armitage test, the Jonckheere-Terpstra test, the linear-by-linear trend test, and a test using ranks developed by Cuzick.

Data for these tests consist of two variables. The first variable, specified immediately after nptrend, contains responses. The second variable, specified in group(), identifies groups.

For the Cochran-Armitage test, the response is typically a $0 / 1$ variable, and the values of the group variable are ordered categories. It tests the trend of the proportions of positive responses in the groups. The numerical values of the hypothesized trend are called "scores". For this form of the test, we call the scores that specify the hypothesized trend "group scores" because it is the trend by group being tested. For example, group scores of 1, 2, 3 would test for a linear trend. Group scores of $1,4,9$ would test for a quadratic trend.

The specification of the Cochran-Armitage test can be flipped. That is, the response variable can identify ordered outcome categories, and the group variable can be $0 / 1$. In this case, we call the scores that specify the hypothesized trend "response scores".

The Cochran-Armitage test requires either a $0 / 1$ response variable or a $0 / 1$ group variable. It cannot test responses with more than two levels when there are more than two groups.

The Jonckheere-Terpstra test is typically used when there are more than two groups and more than two response levels. The test assesses whether there is an association between the response scores and the group scores, but only the relative orders of the responses and groups matter. Because there is no need to specify a hypothesized trend, this test is in this sense "more nonparametric" than the other tests for trend reported by nptrend.

The linear-by-linear trend test uses response scores to specify the trend being tested. So it is typically used when the trend of the response scores, rather than just their ordering, is meaningful. The linear-by-linear statistic is asymptotically equivalent to the Pearson correlation coefficient (see [R] correlate).

The statistics for the Cochran-Armitage test (when the response is $0 / 1$ ), the linear-by-linear trend test, and Cuzick's test are based on the numerical values of the group scores. The linear-by-linear test requires numerical response scores as well. Different scores, even those that have the same ordering, produce different values of the statistic. For instance, group scores 1, 2, 3 would give a different result than 1, 2, 4. For the linear-by-linear test, the same holds for response scores.

The options scoreresponse() and scoregroup() are provided as convenience tools to test different response scores and group scores without having to modify the underlying variables. They do, however, require that the underlying variables be valued $1,2,3, \ldots$

If your data are not grouped, you can test for trend with the signtest and spearman commands; see [R] signrank and [R] spearman. With signtest, you can perform the Cox and Stuart test, a sign test applied to differences between equally spaced observations of varname. With spearman, you can perform the Daniels test, a test of zero Spearman correlation between varname and a time index. See Conover (1999, 169-175, 323) for a discussion of these tests.

## Cochran-Armitage test

The Cochran-Armitage test tests the trend in ordered groups of the probability of a positive response. As we stated earlier, response and group specifications can be flipped, and it can assess whether a trend based on ordered responses is associated with membership in one of two groups. In the discussion that follows we will consider only the former case: ordered groups with $0 / 1$ responses.

The Cochran-Armitage test is based on a linear probability model for $\theta_{i}$, the probability of a positive response for subjects in the $i$ th group,

$$
\theta_{i}=\alpha+\beta\left(g_{i}-\bar{g}\right)
$$

where $\alpha$ is the intercept, $\beta$ is the slope, $g_{i}$ are the group scores, and $\bar{g}$ is the mean of the group scores.

The estimate for $\beta$ is the ordinary least-squares (OLS) estimate $\widehat{\beta}$, and this is the statistic for the Cochran-Armitage test.

However, the standard error of the statistic $\widehat{\beta}$ for the Cochran-Armitage test has a different formulation than that of the standard error from OLS regression-although they are the same asymptotically. The standard error se $(\widehat{\beta})$ for the Cochran-Armitage statistic is based on decomposing the Pearson statistic for independence.

The Pearson statistic $X_{\text {Ind }}^{2}$ for independence can be written as

$$
X_{\mathrm{Ind}}^{2}=z^{2}+X_{\mathrm{DL}}^{2}
$$

where $z=\widehat{\beta} / \operatorname{se}(\widehat{\beta})$ is the $z$ score for the Cochran-Armitage test and $X_{\mathrm{DL}}^{2}$ is a statistic for departure from the linear trend. Asymptotically, $z$ has a standard normal distribution, and $X_{\mathrm{DL}}^{2}$ has a $\chi^{2}$ distribution with $I-2$ degrees of freedom, where $I$ is the number of ordered categories. See Cochran-Armitage test for trend in Methods and formulas below.

This is one of the advantages of performing the Cochran-Armitage test. We get not only a test for trend but also a test for departure from this trend. Let's be clear about what we mean when we say "departure from this trend". The Pearson statistic for independence is decomposed into two pieces. Either piece being large means that the test for independence is rejected. The linear piece $z^{2}$ being large means there is a linear trend that rejects independence. The departure piece $X_{\mathrm{DL}}^{2}$ being large means there are differences other than the linear trend that reject independence.

We should also be clear about what we mean when we say "linear trend". Linear trend means whatever trend is given by the group scores $g_{i}$, which do not have to be $1,2,3, \ldots$ They could be, for instance, $1,3,10,12, \ldots$ It is linear in the scores but not necessarily linear by the ordering of the groups.

When the exact option is specified, a $p$-value from the permutation test of the null hypothesis $\beta=0$ is reported.

## > Example 1

We have fictional data from a clinical trial of a new drug for treating migraines. The variable dose contains the dose of the drug given to a subject. The variable relief is $0 / 1$ with 0 indicating no relief of the migraine and 1 indicating partial or total relief. Here is a tabulation of the data:


There appears to be a trend of more subjects reporting relief at the higher doses. Let's calculate the Cochran-Armitage test for trend:

```
. nptrend relief, group(dose) carmitage
Cochran-Armitage test for trend
    Number of observations = 800
            Number of groups = 4
Number of response levels = 2
\begin{tabular}{c|rrr}
\hline Group & Group score & \begin{tabular}{r} 
Mean \\
response \\
score
\end{tabular} & \begin{tabular}{c} 
Number \\
of obs
\end{tabular} \\
\hline dose & & & \\
10 & 10 & .6 & 200 \\
20 & 20 & .54 & 200 \\
30 & 30 & .585 & 200 \\
40 & 40 & .685 & 200 \\
\hline
\end{tabular}
    Statistic = . 003
    Std. err. = .0015476
        z = 1.939
    Prob > |z| = 0.0526
Test of departure from trend:
    chi2(2) = 5.45
    Prob > chi2 = 0.0656
```

The asymptotic $p$-value for the test of a linear trend is 0.0526 . The $p$-value for the test of departure from linearity is 0.0656 .

The $p$-value for the test of a linear trend is close to 0.05 . Rather than report the asymptotic $p$-value, we may want to report an exact $p$-value. We run nptrend again, this time specifying the exact option.

```
. nptrend relief, group(dose) carmitage exact
Permutations (10,000): ........1,000.........2,000.........3,000.........4,000
> ........5,000........6,000.........7,000.........8,000.........9,000........
> ..10,000 done
Cochran-Armitage test for trend
    Number of observations = 800
    Number of groups = 4
Number of response levels = 2
```

| Group | Group score | Mean <br> response <br> score | Number <br> of obs |
| :---: | ---: | ---: | ---: |
| dose |  |  |  |
| 10 | 10 | .6 | 200 |
| 20 | 20 | .54 | 200 |
| 30 | 30 | .585 | 200 |
| 40 | 40 | .685 | 200 |

    Statistic \(=\quad .003\)
    Std. err. = . 0015476
        \(z=1.939\)
        Prob \(>|z|=0.0526\)
        Exact prob \(=0.0576\) ( 10,000 Monte Carlo permutations)
    Test of departure from trend:
chi2(2) $=5.45$
Prob > chi2 $=0.0656$

The exact $p$-value reported is 0.0576 , larger than the asymptotic $p$-value of 0.0526 . The exact $p$-value comes from a permutation test performed by computing 10,000 random permutations. This means that there is some random error associated with the reported exact $p$-value.

To see the Monte Carlo error in the exact $p$-value, we can use the show suboption of exact, which displays the full output of permute. The option we specify is

```
exact(montecarlo, show reps(100000) dots(1000) rseed(1234))
```

We increase the number of random permutations to 100,000 . dots (1000) displays a dot after every 1,000 th permutation rather than the default of every 100 th. rseed (1234) sets the random-number seed so we can reproduce our result. We did not set the random-number seed when we ran nptrend $\ldots$ exact earlier. If we were to run that again, we would get a different value for the exact $p$-value. Here are the results with exact() specified this new way:

```
. nptrend relief, group(dose) carmitage
> exact(montecarlo, show reps(100000) dots(1000) rseed(1234))
Permutations (100,000): .........10,000.........20,000..........30,000..........4
> 0,000........50,000 . . . . . . . 60,000. . . . . . . . 70,000 . . . . . . . . 80,000 . . . . . . . . 90,
> 000........100,000 done
Monte Carlo permutation results Number of observations = 800
Permutation variable: relief Number of permutations = 100,000
```

| T | T(obs) | Test | c | n | p | Monte Carlo error |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  |  |  | SE(p) | [95\% | $\mathrm{CI}(\mathrm{p})]$ |
| _pm_1 | . 003 | lower | 97435 | 100000 | . 9744 | . 0005 | . 9734 | . 9753 |
|  |  | upper | 2959 | 100000 | . 0296 | . 0005 | . 0285 | . 0307 |
|  |  | sided |  |  | . 0592 | . 0007 | . 0577 | . 0606 |

Notes: For lower one-sided test, $c=\#\{T<=T(o b s)\}$ and $p=p \_l o w e r=c / n$.
For upper one-sided test, $c=\#\{T>=T(o b s)\}$ and $p=p \_u p p e r=c / n$.
For two-sided test, $\mathrm{p}=2 * \min (\mathrm{p}$ lower, p _upper) ; SE and CI approximate.
Cochran-Armitage test for trend
Number of observations $=800$
Number of groups = 4
Number of response levels $=\quad 2$

| Group | Group score | Mean <br> response score | Number <br> of obs |
| :---: | :---: | :---: | :---: |
| dose |  |  |  |
| 10 | 10 | . 6 | 200 |
| 20 | 20 | . 54 | 200 |
| 30 | 30 | . 585 | 200 |
| 40 | 40 | . 685 | 200 |
| Statistic = . 003 |  |  |  |
| Std. err. $=.0015476$ |  |  |  |
|  | z $=$ | 939 |  |
| Prob $>\|z\|=0.0526$ |  |  |  |
| Exact prob $=0.0592$ (100,000 Monte Carlo |  |  |  |

Test of departure from trend:

$$
\operatorname{chi2}(2)=5.45
$$

Prob > chi2 = 0.0656
The reported exact $p$-value from this specification is 0.0592 , and we see from the table reported by permute that the Monte Carlo $95 \%$ confidence interval for it is [0.0577, 0.0606]. Clearly, the asymptotic $p$-value 0.0526 is a bit anticonservative. If we were going to publish the result, we might want to run the command again with even more permutations, perhaps $1,000,000$ or more, to narrow the width of the confidence level to less than 0.001 . With only 800 observations in these data, the Monte Carlo permutations are computed quickly.

## - Example 2

We continue with the previous example to illustrate the use of the scoregroup() option. When we tested the linear trend of dose, we got a $p$-value of 0.0656 for the departure from linearity. So maybe the trend is not linear. Let's test a quadratic trend.

To use the scoregroup() option, the variable in group() must be $1,2, \ldots$ The variable dose is $10,20,30,40$. We can convert it to $1,2,3,4$ in several ways. Here we use the egen group() function because this works in general.

```
. egen idose = group(dose)
. tabulate idose
\begin{tabular}{r|rrr} 
group(dose) & Freq. & Percent & Cum. \\
\hline 1 & 200 & 25.00 & 25.00 \\
2 & 200 & 25.00 & 50.00 \\
3 & 200 & 25.00 & 75.00 \\
4 & 200 & 25.00 & 100.00 \\
\hline Total & 800 & 100.00 &
\end{tabular}
```

Now, we specify scoregroup(14916) to test a quadratic trend.


Of course, we could have just typed generate dose $2=$ dose ${ }^{\wedge} 2$ and used this variable for group() to test the quadratic trend.

The $p$-value for the quadratic trend is 0.0206 , and the departure $p$-value is 0.1462 . The $p$-value for the linear trend was 0.0526 . So it appears the relationship between dose and relief is closer to quadratic than linear. But should we be using nptrend to search for the trend with the smallest $p$-value? Surely, logit is the tool for that!

```
. generate dose2 = dose^2
. logit relief dose dose2
Iteration 0: Log likelihood = -537.58798
Iteration 1: Log likelihood = -532.96973
Iteration 2: Log likelihood = -532.96346
Iteration 3: Log likelihood = -532.96346
Logistic regression
```

Log likelihood $=-532.96346$

| Number of obs | $=800$ |
| :--- | ---: | ---: |
| LR chi2(2) | $=9.25$ |
| Prob > chi2 | $=0.0098$ |
| Pseudo R2 | $=0.0086$ |

                            LR chi2(2) \(=9.25\)
                            Prob > chi2 \(=0.0098\)
                            Pseudo R2 \(=0.0086\)
    | relief | Coefficient | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | ---: | ---: |
| dose | -.0720094 | .0367396 | -1.96 | 0.050 | -.1440177 | $-1.19 \mathrm{e}-06$ |
| dose2 | .0017005 | .0007276 | 2.34 | 0.019 | .0002744 | .0031266 |
| _cons | .946503 | .4021702 | 2.35 | 0.019 | .1582638 | 1.734742 |

To be clear, we are not recommending a model such as this for these data. There are only four doses, and this logit model fits three of the four degrees of freedom. The purpose of the CochranArmitage test is to test a single hypothesized trend and report an honest $p$-value for it. If we want to test for a trend but do not have any idea what the trend might be, we may be better off using the Jonckheere-Terpstra test.

## Jonckheere-Terpstra test

The Jonckheere-Terpstra test is useful when it is not clear what the trend might be and we simply want to test for any trend. That is, it tests whether the ordering of the groups is associated with the ordering of the responses. The test is typically used when there are many response levels and any number of groups.

Suppose there are $J \geq 2$ ordered groups. Let $y_{j k}$ be the $k$ th response in the $j$ th group, where $k=1,2, \ldots, n_{j}$. The responses $y_{j k}$ are given by varname (the variable specified immediately after nptrend) or by varname mapped to the values of scoreresponse().

For groups $j$ and $j^{\prime}$, where $j<j^{\prime}$, consider the $n_{j} n_{j^{\prime}}$ pairs

$$
\left(y_{j k}, y_{j^{\prime} k^{\prime}}\right)
$$

Let $C_{j j^{\prime}}$ be the number of pairs such that $y_{j k}<y_{j^{\prime} k^{\prime}}$ and $D_{j j^{\prime}}$ be the number of pairs such that $y_{j k}>y_{j^{\prime} k^{\prime}}$. The Jonckheere-Terpstra statistic is

$$
T=\sum_{j<j^{\prime}}\left(C_{j j^{\prime}}-D_{j j^{\prime}}\right)
$$

In other words, count the number of response pairs, $C_{j j^{\prime}}$, where the group $j$ response is less than the group $j^{\prime}$ response. That is, pairs in which the ordering of the responses is concordant with the ordering of the groups. Subtract from this the number of discordant pairs, $D_{j j^{\prime}}$. Then, sum over the $J(J-1) / 2$ combinations of selecting two of the $J$ groups, where group $j$ is ordered before group $j^{\prime}$.

Clearly, only the ordering of responses matters relative to the ordering of the groups. There is no need to hypothesize what the trend is.

Kendall's $\tau$ (see [R] spearman) also uses only relative differences, and the Jonckheere-Terpstra test can be viewed as a generalization of Kendall's $\tau$ for groups of unequal sizes. See Jonckheere-Terpstra test for trend in Methods and formulas below.

## > Example 3

The following data (Altman 1991, 217) show ocular exposure to ultraviolet radiation for 32 pairs of sunglasses classified into 3 groups according to the amount of visible light transmitted.


Entering these data into Stata, we have

```
. use https://www.stata-press.com/data/r18/sg, clear
(Ultraviolet radiation exposure with sunglasses)
. list, separator(6)
```

|  | group | exposure |
| :---: | :---: | :---: |
| 1. | < 25\% | 1.4 |
| 2. | < 25\% | 1.4 |
| 3. | < 25\% | 1.4 |
| 4. | < 25\% | 1.6 |
| 5. | < 25\% | 2.3 |
| 6. | < 25\% | 2.5 |
| 7. | 25\% to 35\% | . 9 |
| 8. | 25\% to 35\% | 1 |
| (output omitted) |  |  |
| 31. | > 35\% | 8.9 |
| 32. | > 35\% | 13.5 |

We can now use nptrend to report the Jonckheere-Terpstra test results.

| Jonckheere-Terpstra test for trend |  |  |  |
| :---: | :---: | :---: | :---: |
| Number of observations $=32$ |  |  |  |
| Number of groups = |  |  |  |
| Number of response levels $=23$ |  |  |  |
| Group | Group score | Mean <br> response <br> score | Number of obs |
| group |  |  |  |
| < 25\% | 1 | 1.766667 | 6 |
| $25 \%$ to $35 \%$ 2 2.311111 | 2 | 2.311111 | 18 |
| > 35\% | 3 | 4.85 | 8 |
| Statistic = 82 |  |  |  |
| Std. err. $=54.80056$ |  |  |  |
|  | $\mathrm{z}=$ | 496 |  |
| Prob $>\|z\|=0.1346$ |  |  |  |

The approximate $p$-value is 0.1346 . This $p$-value is an approximation to the permutation test $p$-value. That is, it comes from a $z$ statistic that uses the exact variance of the permutation distribution. With only 32 observations, it seems better to simply compute the exact $p$-value using Monte Carlo permutations.

```
. set seed 1234
. nptrend exposure, group(group) jterpstra exact
Permutations (10,000): .........1,000.........2,000..........3,000..........4,000
> . .......5,000........6,000.........7,000..........8,000.........9,000. . . . . .
> ..10,000 done
Jonckheere-Terpstra test for trend
    Number of observations = 32
    Number of groups = 3
Number of response levels = 23
\begin{tabular}{r|rrr}
\hline Group & Group score & \begin{tabular}{r} 
Mean \\
response \\
score
\end{tabular} & \begin{tabular}{r} 
Number \\
of obs
\end{tabular} \\
\hline group & & & \\
\(<25 \%\) \\
\(25 \%\) to \(35 \%\) & 1 & 1.766667 & 6 \\
\(>35 \%\) & 2 & 2.311111 & 18 \\
\hline
\end{tabular}
    Statistic = 82
    Std. err. = 54.80056
    z = 1.496
    Prob > |z| = 0.1346
    Exact prob = 0.1378 (10,000 Monte Carlo permutations)
```

Despite the small number of observations, the Monte Carlo exact $p$-value 0.1378 is close to the approximate $p$-value 0.1346 . Because of the small number of observations, we might be tempted to try the exact (enumerate) option. We do so.

```
. nptrend exposure, group(group) jterpstra exact(enumerate, dots(1000))
(enumerating all 1.42e+12 possible permutations)
Permutations (1,415,721,106,800):
r(1);
```

nptrend calls permute to do the enumeration, and permute tells us there are $1.42 \times 10^{12}$ possible permutations. Because we do not want to wait years for the result, we pressed Break.

Let's randomly cut the dataset in half, from 32 observations to 16 .

```
. set seed 25
. generate r = runiform()
. sort r
. keep in 1/16
(16 observations deleted)
. tabulate group
Transmissio
    n of
    visible
        light Freq. Percent Cum.
\begin{tabular}{r|rrr}
\hline\(<25 \%\) & 2 & 12.50 & 12.50 \\
\(25 \%\) to \(35 \%\) & 9 & 56.25 & 68.75 \\
\(>35 \%\) & 5 & 31.25 & 100.00 \\
\hline Total & 16 & 100.00 &
\end{tabular}
```

Then, let's do the enumeration for these data:

```
. nptrend exposure, group(group) jterpstra exact(enumerate, dots(1000))
(enumerating all 240,240 possible permutations)
Permutations (240,240): .........10,000.........20,000..........30,000.......... 4
> 0,000 . . . . . . .50,000 . . . . . . . . 60,000 . . . . . . . .70,000 . . . . . . . . 80,000 . . . . . . . . 90,
> 000 . . . . . . . 100,000 . . . . . . . . 110,000 . . . . . . . . 120,000 . . . . . . . .130,000 . . . . . . . . . 1
> 40,000 ........150,000 . . . . . . .160,000 . . . . . . .170,000 . . . . . . . 180,000 . . . . . .
> ..190,000.........200,000..........210,000..........220,000.......... 230,000....
> .....240,000. done
Jonckheere-Terpstra test for trend
    Number of observations = 16
    Number of groups = 3
Number of response levels = 15
```

| Group | Group score | Mean <br> response <br> score | Number <br> of obs |
| ---: | ---: | ---: | ---: |
| group |  |  |  |
| < $25 \%$ | 1 | 1.5 | 2 |
| $25 \%$ to $35 \%$ |  |  |  |
| $>35 \%$ | 2 | 2.377778 | 9 |

    Statistic = 49
    Std. err. = 19.57188
        \(z=2.504\)
        Prob > |z| = 0.0123
        Exact prob \(=0.0103\) (enumerated all 240,240 permutations)
    There are only 240,240 possible permutations now, and the computation takes little time. Despite there being only 16 observations, the exact $p$-value (without any error) 0.0103 is close to the approximate $p$-value 0.0123 . Enumeration works quite well for tiny datasets such as this.

## Linear-by-linear trend test

The linear-by-linear trend test is an alternative to the Jonckheere-Terpstra test. The difference is that the linear-by-linear trend test uses response scores to specify the trend being tested. How the trend is hypothesized to vary across groups is specified by the group scores. The linear-by-linear trend test could be described as a parametric specification for trend that is tested nonparametrically!

If scores are not specified in the options, then the values of the response varname are used as response scores, and the values of groupvar are used as group scores.

The linear-by-linear statistic is

$$
T=\frac{1}{N} \sum_{i=1}^{I} \sum_{j=1}^{J} n_{i j}\left(g_{j}-\bar{g}\right)\left(r_{i}-\bar{r}\right)
$$

where $N$ is the total number of observations, $g_{j}$ is the group score for the $j$ th group, $\bar{g}$ is the mean group score over all observations, $r_{i}$ is the response score for the $i$ th ordered response, and $\bar{r}$ is the mean response score over all observations.

The linear-by-linear statistic is equivalent to the Pearson correlation coefficient (see [R] correlate), the difference being that the Pearson correlation coefficient is standardized by the standard deviations of the scores. The $p$-values are slightly different because the $p$-value for the linear-by-linear test is based on its permutation distribution while the $p$-value for the Pearson correlation coefficient assumes normality.

Clearly, the statistic depends on the numerical values of both the response scores and the group scores. Different scores, even ones that give the same ordering, will produce different values of the statistic.

## > Example 4

We again use the sunglasses data from example 3, where we did the Jonckheere-Terpstra test.

```
. use https://www.stata-press.com/data/r18/sg, clear
(Ultraviolet radiation exposure with sunglasses)
```

We calculate the linear-by-linear trend test for these data.

```
. nptrend exposure, group(group) linear
Linear-by-linear test for trend
    Number of observations = 32
            Number of groups = 3
Number of response levels = 23
```

|  |  | Mean <br> response <br> score | Number <br> of obs |
| ---: | ---: | ---: | ---: |
| group | Group score |  |  |
| < 25\% | 1 | 1.766667 | 6 |
| $25 \%$ to $35 \%$ |  |  |  |
| $>35 \%$ | 2 | 2.311111 | 18 |

    Statistic \(=.7035156\)
    Std. err. = . 3063377
        \(z=2.297\)
    Prob \(>|z|=0.0216\)
    The approximate $p$-value is 0.0216 , which is much smaller than the Jonckheere-Terpstra test approximate $p$-value 0.1346 . Such a difference is not surprising. The linear-by-linear trend test uses the actual values of the response in the statistic. The values of the response seem to follow a linear trend by group, and when this is true, one generally expects the $p$-value from the linear-by-linear trend test to be smaller than the $p$-value from the Jonckheere-Terpstra test.

Again, we might want to compute the exact $p$-value using Monte Carlo permutations.

```
. nptrend exposure, group(group) linear
> exact(montecarlo, reps(100000) dots(1000) rseed(1234))
Permutations (100,000): . . . . . . . 10,000..........20,000...........30,000...........4
> 0,000.........50,000 . . . . . . . 60,000 . . . . . . . .70,000 . . . . . . . . 80,000 . . . . . . . . 90,
> 000........100,000 done
Linear-by-linear test for trend
    Number of observations = 32
        Number of groups = 3
Number of response levels = 23
\begin{tabular}{r|rrr}
\hline Group & Group score & \begin{tabular}{r} 
Mean \\
response \\
score
\end{tabular} & \begin{tabular}{r} 
Number \\
of obs
\end{tabular} \\
\hline group & & & \\
\(25 \% \%\) \\
to \(35 \%\) & 1 & 1.766667 & 6 \\
\(>35 \%\) & 2 & 2.311111 & 18 \\
\hline
\end{tabular}
            Statistic = .7035156
            Std. err. = . }306337
            z = 2.297
            Prob > |z| = 0.0216
            Exact prob = 0.0146 (100,000 Monte Carlo permutations)
```

In this case, the exact $p$-value 0.0146 is smaller than the approximate $p$-value 0.0216 .

## Cuzick's test

Cuzick's test (Cuzick 1985) is an extension of the Kruskal-Wallis test (see [R] kwallis) for ordered groups. The response scores are ranked, the sum of the ranks is calculated for each group, and then these sums of ranks are weighted by the deviation of the group scores from their mean.

The statistic for Cuzick's test is

$$
T=\frac{1}{N} \sum_{j=1}^{J}\left(g_{j}-\bar{g}\right) R_{j}
$$

where $N$ is the total number of observations, $g_{j}$ is the group score for the $j$ th group, $\bar{g}$ is the mean group score over all observations, and $R_{j}$ is the sum of the ranks of the response scores in the $j$ th group.

The form of the statistic is similar to that of the statistic for the linear-by-linear test, but instead of using response scores being used directly, the ranks of the response scores are used. So only the ordering of the response scores matters. The numerical values of the group scores, however, are used in the calculation, so different group scores will give different values of the statistic.

## > Example 5

We continue with the sunglasses data from example 4 . We now calculate Cuzick's test, calculating the exact $p$-value as well.

```
. nptrend exposure, group(group) cuzick
> exact(montecarlo, reps(100000) dots(1000) rseed(1234))
Permutations (100,000): . ........10,000..........20,000..........30,000..........4
> 0,000.........50,000 . . . . . . . 60,000. . . . . . . .70,000 . . . . . . . . 80,000 . . . . . . . . 90,
> 000.........100,000 done
Cuzick's test with rank scores
    Number of observations = 32
        Number of groups = 3
Number of response levels = 23
```

| Group | Group score | Mean <br> response <br> score | Number <br> of obs |
| ---: | ---: | ---: | ---: |
| group |  |  |  |
| $<25 \%$ |  |  |  |
| $25 \%$ to $35 \%$ |  |  |  |
| $>35 \%$ | 1 | 1.766667 | 6 |

Statistic $=1.65625$
Std. err. = 1.090461
z = 1.519
Prob $>|z|=0.1288$
Exact prob $=0.1322$ ( 100,000 Monte Carlo permutations)

The approximate $p$-value 0.1288 is close to the exact $p$-value 0.1322 . These results are also close to the results from the Jonckheere-Terpstra test, where the approximate $p$-value was 0.1346 . This is not surprising because using ranks of the responses in Cuzick's test is likely to be similar to the comparisons of orderings in the Jonckheere-Terpstra test. The latter test is better known than Cuzick's test, so for this reason alone, it may be preferable to use the Jonckheere-Terpstra test when reporting results.

## Stored results

nptrend stores the following in r() :

```
Scalars
    r(N) number of observations
    r(n_groups) number of groups
    r(n_rsp_levels) number of response levels
    r(beta) slope from Cochran-Armitage linear probability model
    r(T) statistic
    r(se) standard error of the statistic
    r(z) z statistic
    r(p) two-sided p-value from z statistic
    r(p_l) lower one-sided p-value from z statistic
    r(p_u) upper one-sided p-value from z statistic
    r(chi2_depart) }\quad\mp@subsup{\chi}{}{2}\mathrm{ for departure from Cochran-Armitage linear probability model
    r(df_depart) degrees of freedom for departure }\mp@subsup{\chi}{}{2
    r(p_depart) }\quadp\mathrm{ -value for departure }\mp@subsup{\chi}{}{2
    r(p_exact) two-sided exact p-value
```

```
r(p_l_exact) lower one-sided exact p-value
r(p_u_exact) upper one-sided exact p-value
r(n_perm) number of permutations performed
    r(test)
    r(group)
    r(exact)
    r(rngstate)
    r(table)
```

```
"carmitage", "jterpstra", "linear", or "cuzick"
```

"carmitage", "jterpstra", "linear", or "cuzick"
group variable
group variable
"montecarlo" or "enumerate"
"montecarlo" or "enumerate"
random-number state used for Monte Carlo permutations
random-number state used for Monte Carlo permutations
table of mean response scores by group

```
table of mean response scores by group
```

Macros
Matrices

If exact (..., show) is specified, the stored results from permute are returned as well; see Stored results in $[R]$ permute.

## Methods and formulas

Methods and formulas are presented under the following headings:

Overview<br>Cochran-Armitage test for trend<br>Jonckheere-Terpstra test for trend<br>Linear-by-linear test for trend<br>Cuzick's test with rank scores

## Overview

We will use the same notation for all the tests. Let the set of distinct responses be indexed by $i=1,2, \ldots, I$. Let the groups be indexed by $j=1,2, \ldots, J$.

We call the scores for the responses "response scores" and denote them by $r_{i}$.
For the Cochran-Armitage test (when the response is $0 / 1$ ), linear-by-linear trend test, and Cuzick's test the groups have scores as well. We call these scores "group scores", and denote them by $g_{j}$.

All the tests for trend reported by nptrend are based on tests of independence between the ordering of the responses (or their scores) and the ordering of the groups (or their scores). Because these are tests of independence, there is no need to hypothesize a direction of causal inference. Responses could be outcomes predicted by group membership, or "groups" could be outcomes predicted by what we are calling "responses".

Let $n_{i j}$ be the number of observations in the $i$ th response category and $j$ th group. Let $N=\sum_{i j} n_{i j}$ be the total number of observations. Let $n_{i+}=\sum_{j} n_{i j}$ be the row margins, and let $n_{+j}=\sum_{i} n_{i j}$ be the column margins.

## Cochran-Armitage test for trend

When carmitage is specified, the Cochran-Armitage test is performed (Cochran 1954; Armitage 1955; Agresti 2013). It assesses whether the probability of a positive response is associated with a trend based on ordered groups. Or the specification can be flipped. It can be used to assess whether a trend based on ordered responses is associated with membership in one of two groups.

In these methods, we assume that the response is $0 / 1$-that is, the response is indexed by $i=0,1-$ and that there are $J$ ordered groups with scores $g_{j}$. When the specification is flipped, response and group are interchanged in the equations below.

The Cochran-Armitage test is based on a linear probability model for $\theta_{j}$, the probability of having response $i=1$ for the subjects in the $j$ th group,

$$
\theta_{j}=\alpha+\beta\left(g_{j}-\bar{g}\right)
$$

where $\alpha$ is the intercept, $\beta$ is the slope, $g_{j}$ are the group scores, and $\bar{g}$ is the mean of the group scores.

The equation is solved using OLS,

$$
\widehat{\theta}_{j}=p+\widehat{\beta}\left(g_{j}-\bar{g}\right)
$$

where $p=\left(\sum_{j} n_{1 j}\right) / N$ and $\bar{g}=\left(\sum_{j} n_{+j} g_{j}\right) / N$.
The OLS solution for $\widehat{\beta}$ is

$$
\widehat{\beta}=\frac{\sum_{j} n_{+j}\left(p_{j}-p\right)\left(g_{j}-\bar{g}\right)}{\sum_{j} n_{+j}\left(g_{j}-\bar{g}\right)^{2}}
$$

where $p_{j}=n_{1 j} / n_{+j}$.
The test for independence between groups and responses can be separated into a test for the linear trend and a test for departure from the linear trend. The Pearson statistic for independence

$$
X_{\mathrm{Ind}}^{2}=\frac{1}{p(1-p)} \sum_{j=1}^{J} n_{+j}\left(p_{j}-p\right)^{2}
$$

is decomposed as

$$
X_{\mathrm{Ind}}^{2}=z^{2}+X_{\mathrm{DL}}^{2}
$$

where

$$
z^{2}=\frac{\widehat{\beta}^{2}}{p(1-p)} \sum_{j=1}^{J} n_{+j}\left(g_{j}-\bar{g}\right)^{2}
$$

and

$$
X_{\mathrm{DL}}^{2}=\frac{1}{p(1-p)} \sum_{j=1}^{J} n_{+j}\left(p_{j}-\widehat{\theta}_{j}\right)^{2}
$$

$z$ is the $z$ statistic for the linear trend, and $X_{\mathrm{DL}}^{2}$ is the statistic for departure from a linear trend. Asymptotically, $z$ has a standard normal distribution, and $X_{\mathrm{DL}}^{2}$ has a $\chi^{2}$ distribution with $J-2$ degrees of freedom.

## Jonckheere-Terpstra test for trend

When jterpstra is specified, the Jonckheere-Terpstra test is performed (Terpstra 1952; Jonckheere 1954). The Jonckheere-Terpstra test is a test of whether the hypothesized ordering of the responses is associated with the hypothesized ordering of the groups. Typically, this test is used when there are only a few groups but many response categories or even continuous responses.

Only the ordering of the responses and groups matter. The numerical values of the scores do not matter, but scores can be specified to change the ordering.

Let $y_{j k}$ be the $k$ th response in the $j$ th group, where $k=1,2, \ldots, n_{+j}$. If the response falls in the $i$ th response category, then $y_{j k}=r_{i}$, the response score for the $i$ th category.

For groups $j$ and $j^{\prime}$, where $j<j^{\prime}$, consider the pairs

$$
\left(y_{j k}, y_{j^{\prime} k^{\prime}}\right)
$$

where $k=1,2, \ldots, n_{+j}$ and $k^{\prime}=1,2, \ldots, n_{+j^{\prime}}$. A total of $n_{+j} n_{+j^{\prime}}$ pairs are formed for these two groups.

Let $C_{j j^{\prime}}$ be the number of pairs such that $y_{j k}<y_{j^{\prime} k^{\prime}}$, that is, pairs in which the response ordering is concordant with the group ordering. Let $D_{j j^{\prime}}$ be the number of discordant pairs with $y_{j k}>y_{j^{\prime} k^{\prime}}$. The Jonckheere-Terpstra statistic is

$$
T=\sum_{j=1}^{J-1} \sum_{j^{\prime}=j+1}^{J}\left(C_{j j^{\prime}}-D_{j j^{\prime}}\right)
$$

The Jonckheere-Terpstra statistic is sometimes expressed as sums over $C_{j j^{\prime}}+E_{j j^{\prime}} / 2$, where $E_{j j^{\prime}}$ is the number of tied pairs with $y_{j k}=y_{j^{\prime} k^{\prime}}$. Because $C_{j j^{\prime}}+D_{j j^{\prime}}+E_{j j^{\prime}}=n_{+j} n_{+j^{\prime}}$, the statistic $T$ is twice the statistic based on $C_{j j^{\prime}}+E_{j j^{\prime}} / 2$ minus a constant. The advantage of the formulation using $C_{j j^{\prime}}-D_{j j^{\prime}}$ is that under the null hypothesis of independence between groups and responses, its expectation is zero. When $T$ is positive, the observed trend is a positive trend between the group ordering and the response ordering, and when $T$ is negative, it is a negative trend.

The variance of $T$ can be computed exactly based on the permutation distribution of responses where the table margins $n_{i+}$ and $n_{+j}$ are fixed and each permutation is considered equally likely under the null hypothesis. The formula, however, is complicated when there are ties.

Let $M$ be the number of distinct values in the responses. Let $e_{m}$ be the multiplicity of the $m$ th distinct response value. That is, the $N$ observations $y_{j k}$ consist of $e_{1}$ observations with value $z_{1}, e_{2}$ observations with value $z_{2}$, etc., with $\sum_{m} e_{m}=N$. The variance of $T$ under the null permutation distribution is exactly

$$
\begin{aligned}
\operatorname{Var}(T)= & \frac{1}{18}[N(N-1)(2 N+5)- \\
& -\sum_{j=1}^{J} n_{+j}\left(n_{+j}-1\right)\left(2 n_{+j}+5\right) \\
+ & \frac{1}{2 N(N-1)}\left\{\sum_{j=1}^{J} n_{+j}\left(n_{+j}-1\right)\left(2 e_{m}+5\right)\right] \\
+ & \frac{1}{9 N(N-1)(N-2)}\left\{\sum_{m=1}^{M} e_{m}\left(e_{m}-1\right)\right\} \\
& \times\left\{\sum_{m=1}^{J} n_{+j}\left(n_{+j}-1\right)\left(n_{+j}-2\right)\right\}
\end{aligned}
$$

The approximate $p$-value is based on a normal approximation using the exact variance of $T$.

## Linear-by-linear test for trend

When linear is specified, the linear-by-linear trend test is calculated based on the statistic

$$
T=\frac{1}{N} \sum_{i=1}^{I} \sum_{j=1}^{J} n_{i j}\left(g_{j}-\bar{g}\right)\left(r_{i}-\bar{r}\right)
$$

where $g_{j}$ is the group score for the $j$ th group, $\bar{g}=\left(\sum_{j} n_{+j} g_{j}\right) / N$ is the mean group score over all observations, $r_{i}$ is the response score for the $i$ th ordered response, and $\bar{r}=\left(\sum_{i} n_{i+} r_{i}\right) / N$ is the mean response score over all observations.

Under the null hypothesis of independence between groups and responses, the expectation of $T$ is zero. The variance of $T$ is based on the permutation distribution of responses where the table margins $n_{i+}$ and $n_{+j}$ are fixed and each permutation is considered equally likely under the null hypothesis. The variance of $T$ is exactly

$$
\operatorname{Var}(T)=\frac{1}{N^{2}(N-1)}\left\{\sum_{j=1}^{J} n_{+j}\left(g_{j}-\bar{g}\right)^{2}\right\}\left\{\sum_{i=1}^{I} n_{i+}\left(r_{i}-\bar{r}\right)^{2}\right\}
$$

The approximate $p$-value is based on a normal approximation using the exact variance of $T$.
Note the symmetry between the group and response scores in these equations. In the linear-by-linear test for trend, it is arbitrary as to what is a group with a group score and what is a response with a response score. The two are interchangeable.

## Cuzick's test with rank scores

When cuzick is specified, the test for trend is based on a method in Cuzick (1985), which is described in Altman (1991, 215-217).

Rank all $N$ responses, using average ranks for ties. Let $R_{j}$ be the sum of the ranks in the $j$ th group. The statistic for Cuzick's test $T$ is

$$
T=\frac{1}{N} \sum_{j=1}^{J}\left(g_{j}-\bar{g}\right) R_{j}
$$

where $g_{j}$ is the group score for the $j$ th group and $\bar{g}=\left(\sum_{j} n_{+j} g_{j}\right) / N$.
Under the null hypothesis of independence between groups and responses, the expectation of $T$ is zero. When there are no ties, the variance of $T$ is

$$
\operatorname{Var}(T)=\frac{N+1}{12 N} \sum_{j=1}^{J} n_{+j}\left(g_{j}-\bar{g}\right)^{2}
$$

When there are ties, we again let $e_{m}$ be the multiplicity of the $m$ th distinct response value, $m=1,2, \ldots, M$. Define

$$
a=\frac{1}{N\left(N^{2}-1\right)} \sum_{m=1}^{M} e_{m}\left(e_{m}^{2}-1\right)
$$

The corrected variance of $T$ when there are ties is

$$
\widetilde{\operatorname{Var}}(T)=(1-a) \operatorname{Var}(T)
$$

## Exact p-values

Exact $p$-values are computed by permute. The variable permuted is that specified by group(), except for carmitage when the response variable is $0 / 1$; in this case, the response variable is permuted. For details on the permutation computation, see $[R]$ permute.

## Acknowledgments

The original version of nptrend was written by K. A. Stepniewska, Worldwide Antimalarial Resistance Network, University of Oxford, UK, and D. G. Altman (1948-2018).

## References

Agresti, A. 2013. Categorical Data Analysis. 3rd ed. Hoboken, NJ: Wiley.
Altman, D. G. 1991. Practical Statistics for Medical Research. London: Chapman and Hall/CRC.
Armitage, P. 1955. Tests for linear trends in proportions and frequencies. Biometrics 11: 375-386. https://doi.org/10.2307/3001775.
Cochran, W. G. 1954. Some methods for strengthening the common chi-squared tests. Biometrics 10: 417-451. https://doi.org/10.2307/3001616.

Conover, W. J. 1999. Practical Nonparametric Statistics. 3rd ed. New York: Wiley.
Cuzick, J. 1985. A Wilcoxon-type test for trend. Statistics in Medicine 4: 87-90. https://doi.org/10.1002/sim. 4780040112.
Jonckheere, A. R. 1954. A distribution-free k-sample test against ordered alternatives. Biometrika 41: 133-145. https://doi.org/10.2307/2333011.
Terpstra, T. J. 1952. The asymptotic normality and consistency of Kendall's test against trend, when ties are present in one ranking. Indagationes Mathematicae 14: 327-333.

## Also see

[R] Epitab - Tables for epidemiologists
[R] kwallis - Kruskal-Wallis equality-of-populations rank test
[R] permute - Permutation tests
[R] signrank - Equality tests on matched data
[R] spearman - Spearman's and Kendall's correlations
[R] symmetry - Symmetry and marginal homogeneity tests
[ST] stmc - Calculate rate ratios with the Mantel-Cox method
[ST] stmh - Calculate rate ratios with the Mantel-Haenszel method

## Title

ologit - Ordered logistic regression

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

ologit fits ordered logit models of ordinal variable depvar on the independent variables indepvars. The actual values taken on by the dependent variable are irrelevant, except that larger values are assumed to correspond to "higher" outcomes.

## Quick start

Ordinal logit model of $y$ on $x 1$ and categorical variables $a$ and $b$ ologit y x1 i.a i.b

Same as above, and include interaction between a and b and report results as odds ratios ologit y x1 a\#\#b, or

With bootstrap standard errors
ologit y x1 i.a i.b, vce(bootstrap)
Analysis restricted to cases where catvar $=0$ using svyset data with replicate weights svy bootstrap, subpop(if catvar==0): ologit y x1 i.a i.b

## Menu

Statistics $>$ Ordinal outcomes $>$ Ordered logistic regression

## Syntax

ologit depvar [indepvars] [if] [in] [weight] [, options]
options
Description

Model
offset (varname)
constraints (constraints)
SE/Robust
vce (vcetype)

Reporting
level(\#)
or
nocnsreport
display_options
include varname in model with coefficient constrained to 1
apply specified linear constraints
vcetype may be oim, opg, robust, cluster clustvar, bootstrap, or jackknife
set confidence level; default is level (95)
report odds ratios
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
control the maximization process; seldom used
keep collinear variables
display legend instead of statistics
indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
depvar and indepvars may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bayes, bootstrap, by, fmm, fp, jackknife, mfp, mi estimate, nestreg, rolling, statsby, stepwise, and svy are allowed; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: ologit and [FMM] fmm: ologit.
vce (bootstrap) and vce(jackknife) are not allowed with the mi estimate prefix; see [MI] mi estimate.
Weights are not allowed with the bootstrap prefix; see $[R]$ bootstrap.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

$\qquad$ Model
offset (varname), constraints (constraints); see [R] Estimation options.

SE/Robust
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
level (\#); see [R] Estimation options.
or reports the estimated coefficients transformed to odds ratios, that is, $e^{b}$ rather than $b$. Standard errors and confidence intervals are similarly transformed. This option affects how results are displayed, not how they are estimated or may be specified at estimation or when replaying previously estimated results.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.

[^42]The following options are available with ologit but are not shown in the dialog box: collinear, coeflegend; see [R] Estimation options.

## Remarks and examples

Ordered logit models are used to estimate relationships between an ordinal dependent variable and a set of independent variables. An ordinal variable is a variable that is categorical and ordered, for instance, "poor", "good", and "excellent", which might indicate a person's current health status or the repair record of a car. If there are only two outcomes, see $[R]$ logistic, $[R]$ logit, and $[R]$ probit. This entry is concerned only with more than two outcomes. If the outcomes cannot be ordered (for example, residency in the north, east, south, or west), see [R] mlogit. This entry is concerned only with models in which the outcomes can be ordered.

In ordered logit, an underlying score is estimated as a linear function of the independent variables and a set of cutpoints. The probability of observing outcome $i$ corresponds to the probability that the estimated linear function, plus random error, is within the range of the cutpoints estimated for the outcome:

$$
\operatorname{Pr}\left(\text { outcome }_{j}=i\right)=\operatorname{Pr}\left(\kappa_{i-1}<\beta_{1} x_{1 j}+\beta_{2} x_{2 j}+\cdots+\beta_{k} x_{k j}+u_{j} \leq \kappa_{i}\right)
$$

$u_{j}$ is assumed to be logistically distributed in ordered logit. In either case, we estimate the coefficients $\beta_{1}, \beta_{2}, \ldots, \beta_{k}$ together with the cutpoints $\kappa_{1}, \kappa_{2}, \ldots, \kappa_{k-1}$, where $k$ is the number of possible outcomes. $\kappa_{0}$ is taken as $-\infty$, and $\kappa_{k}$ is taken as $+\infty$. All of this is a direct generalization of the ordinary two-outcome logit model.

## > Example 1

We wish to analyze the 1977 repair records of 66 foreign and domestic cars. The data are a variation of the automobile dataset described in [U] 1.2.2 Example datasets. The 1977 repair records, like those in 1978, take on values "Poor", "Fair", "Average", "Good", and "Excellent". Here is a cross-tabulation of the data:

```
. use https://www.stata-press.com/data/r18/fullauto
(Automobile models)
```

. tabulate rep77 foreign, chi2

| Repair |  |  |  |
| ---: | ---: | ---: | ---: |
| record |  |  |  |
| 1977 | Foreign |  |  |
| Domestic | Foreign | Total |  |
| Poor | 2 | 1 | 3 |
| Fair | 10 | 1 | 11 |
| Average | 20 | 7 | 27 |
| Good | 13 | 7 | 20 |
| Excellent | 0 | 5 | 5 |
| Total | 45 | 21 | 66 |
| Pearson $\operatorname{chi2}(4)=$ 13.8619 $\operatorname{Pr}=0.008$ |  |  |  |

Although it appears that foreign takes on the values Domestic and Foreign, it is actually a numeric variable taking on the values 0 and 1 . Similarly, rep77 takes on the values $1,2,3,4$, and 5, corresponding to Poor, Fair, and so on. The more meaningful words appear because we have attached value labels to the data; see [U] 12.6.3 Value labels.

Because the $\chi^{2}$ value is significant, we could claim that there is a relationship between foreign and rep77. Literally, however, we can only claim that the distributions are different; the $\chi^{2}$ test is not directional. One way to model these data is to model the categorization that took place when the data were created. Cars have a true frequency of repair, which we will assume is given by $S_{j}=\beta$ foreign $_{j}+u_{j}$, and a car is categorized as "poor" if $S_{j} \leq \kappa_{0}$, as "fair" if $\kappa_{0}<S_{j} \leq \kappa_{1}$, and so on:

```
. ologit rep77 foreign
Iteration 0: Log likelihood = -89.895098
Iteration 1: Log likelihood = -85.951765
Iteration 2: Log likelihood = -85.908227
Iteration 3: Log likelihood = -85.908161
Iteration 4: Log likelihood = -85.908161
Ordered logistic regression
```

Log likelihood $=-85.908161$

| rep77 | Coefficient | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | ---: | ---: |
| foreign | 1.455878 | .5308951 | 2.74 | 0.006 | .4153425 | 2.496413 |
| /cut1 | -2.765562 | .5988208 |  |  | -3.939229 | -1.591895 |
| /cut2 | -.9963603 | .3217706 |  |  | -1.627019 | -.3657016 |
| /cut3 | .9426153 | .3136398 |  |  | .3278925 | 1.557338 |
| /cut4 | 3.123351 | .5423257 |  |  | 2.060412 | 4.18629 |

Our model is $S_{j}=1.46$ foreign $_{j}+u_{j}$; the expected value for foreign cars is 1.46 and, for domestic cars, 0 ; foreign cars have better repair records.

The estimated cutpoints tell us how to interpret the score. For a foreign car, the probability of a poor record is the probability that $1.46+u_{j} \leq-2.77$, or equivalently, $u_{j} \leq-4.23$. Making this calculation requires familiarity with the logistic distribution: the probability is $1 /\left(1+e^{4.23}\right)=0.014$. On the other hand, for domestic cars, the probability of a poor record is the probability $u_{j} \leq-2.77$, which is 0.059 .

This, it seems to us, is a far more reasonable prediction than we would have made based on the table alone. The table showed that 2 of 45 domestic cars had poor records, whereas 1 of 21 foreign cars had poor records-corresponding to probabilities $2 / 45=0.044$ and $1 / 21=0.048$. The predictions from our model imposed a smoothness assumption-foreign cars should not, overall, have better repair records without the difference revealing itself in each category. In our data, the fractions of foreign and domestic cars in the poor category are virtually identical only because of the randomness associated with small samples.

Thus if we were asked to predict the true fractions of foreign and domestic cars that would be classified in the various categories, we would choose the numbers implied by the ordered logit model:

|  | tabulate |  | logit |  |
| :--- | :---: | :---: | :---: | :---: |
|  | Domestic | Foreign | Domestic | Foreign |
| Poor | 0.044 | 0.048 | 0.059 | 0.014 |
| Fair | 0.222 | 0.048 | 0.210 | 0.065 |
| Average | 0.444 | 0.333 | 0.450 | 0.295 |
| Good | 0.289 | 0.333 | 0.238 | 0.467 |
| Excellent | 0.000 | 0.238 | 0.043 | 0.159 |

See $[\mathrm{R}]$ ologit postestimation for a more complete explanation of how to generate predictions from an ordered logit model.

## Technical note

Here ordered logit provides an alternative to ordinary two-outcome logistic models with an arbitrary dichotomization, which might otherwise have been tempting. We could, for instance, have summarized these data by converting the five-outcome rep77 variable to a two-outcome variable, combining cars in the average, fair, and poor categories to make one outcome and combining cars in the good and excellent categories to make the second.

Another even less appealing alternative would have been to use ordinary regression, arbitrarily labeling "excellent" as 5, "good" as 4, and so on. The problem is that with different but equally valid labelings (say, 10 for "excellent"), we would obtain different estimates. We would have no way of choosing one metric over another. That assertion is not, however, true of ologit. The actual values used to label the categories make no difference other than through the order they imply.

In fact, our labeling was 5 for "excellent", 4 for "good", and so on. The words "excellent" and "good" appear in our output because we attached a value label to the variables; see [U] 12.6.3 Value labels. If we were to now go back and type replace rep $77=10$ if rep $77==5$, changing all the 5 s to 10 s, we would still obtain the same results when we refit our model.

## $>$ Example 2

In the example above, we used ordered logit as a way to model a table. We are not, however, limited to including only one explanatory variable or to including only categorical variables. We can explore the relationship of rep77 with any of the variables in our data. We might, for instance, model rep77 not only in terms of the origin of manufacture, but also including length (a proxy for size) and mpg:

```
. ologit rep77 foreign length mpg
Iteration 0: Log likelihood = -89.895098
Iteration 1: Log likelihood = -78.775147
Iteration 2: Log likelihood = -78.254294
Iteration 3: Log likelihood = -78.250719
Iteration 4: Log likelihood = -78.250719
Ordered logistic regression Number of obs = 66
LR chi2(3) = 23.29
Prob > chi2 = 0.0000
Pseudo R2 = 0.1295
```

| rep77 | Coefficient | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | ---: | :---: | ---: | ---: | ---: |
| foreign | 2.896807 | .7906411 | 3.66 | 0.000 | 1.347179 | 4.446435 |
| length | .0828275 | .02272 | 3.65 | 0.000 | .0382972 | .1273579 |
| mpg | .2307677 | .0704548 | 3.28 | 0.001 | .0926788 | .3688566 |
| /cut1 | 17.92748 | 5.551191 |  |  | 7.047344 | 28.80761 |
| /cut2 | 19.86506 | 5.59648 |  |  | 8.896161 | 30.83396 |
| /cut3 | 22.10331 | 5.708936 |  |  | 10.914 | 33.29262 |
| /cut4 | 24.69213 | 5.890754 |  |  | 13.14647 | 36.2378 |

foreign still plays a role-and an even larger role than previously. We find that larger cars tend to have better repair records, as do cars with better mileage ratings.

## Stored results

ologit stores the following in e():
Scalars

| e(N) | number of observations |
| :---: | :---: |
| e(N_cd) | number of completely determined observations |
| e(k_cat) | number of categories |
| e(k) | number of parameters |
| e(k_aux) | number of auxiliary parameters |
| e(k_eq) | number of equations in e(b) |
| e(k_eq_model) | number of equations in overall model test |
| e (k_dv) | number of dependent variables |
| $e\left(d f \_m\right)$ | model degrees of freedom |
| e(r2_p) | pseudo- $R^{2}$ |
| e(ll) | log likelihood |
| e(ll_0) | log likelihood, constant-only model |
| e(N_clust) | number of clusters |
| e(chi2) | $\chi^{2}$ |
| e (p) | $p$-value for model test |
| e (rank) | rank of e(V) |
| e(ic) | number of iterations |
| e(rc) | return code |
| e(converged) | 1 if converged, 0 otherwise |
| ros |  |
| e (cmd) | ologit |
| e(cmdline) | command as typed |
| e(depvar) | name of dependent variable |
| e(wtype) | weight type |
| e(wexp) | weight expression |
| e(title) | title in estimation output |

```
    e(offset)
    e(chi2type)
    e(vce)
    e(vcetype)
    e(opt)
    e(which)
    e(ml_method)
    e(user)
    e(technique)
    e(properties)
    e(predict)
    e(marginsdefault)
    e(asbalanced)
    e(asobserved)
Matrices
    e(b)
    e(Cns)
    e(ilog)
    e(gradient)
    e(cat)
    e(V)
    e(V_modelbased)
Functions
```

    e(clustvar) name of cluster variable
    e(sample) marks estimation sample
    In addition to the above, the following is stored in r() :

Matrices
$r($ table ) matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

See Long and Freese (2014, chap. 7) for a discussion of models for ordinal outcomes and examples that use Stata. Cameron and Trivedi (2005, chap. 15) describe multinomial models, including the model fit by ologit. When you have a qualitative dependent variable, several estimation procedures are available. A popular choice is multinomial logistic regression (see [R] mlogit), but if you use this procedure when the response variable is ordinal, you are discarding information because multinomial logit ignores the ordered aspect of the outcome. Ordered logit and probit models provide a means to exploit the ordering information.

There is more than one "ordered logit" model. The model fit by ologit, which we will call the ordered logit model, is also known as the proportional odds model. Another popular choice, not fit by ologit, is known as the stereotype model; see [R] slogit. All ordered logit models have been derived by starting with a binary logit/probit model and generalizing it to allow for more than two outcomes.

The proportional-odds ordered logit model is so called because, if we consider the odds odds $(k)=$ $P(Y \leq k) / P(Y>k)$, then odds $\left(k_{1}\right)$ and odds $\left(k_{2}\right)$ have the same ratio for all independent variable combinations. The model is based on the principle that the only effect of combining adjoining categories in ordered categorical regression problems should be a loss of efficiency in estimating the regression parameters (McCullagh 1980). This model was also described by McKelvey and Zavoina (1975) and, previously, by Aitchison and Silvey (1957) in a different algebraic form. Brant (1990) offers a set of diagnostics for the model.

Peterson and Harrell (1990) suggest a model that allows nonproportional odds for a subset of the explanatory variables. ologit does not allow this, but a model similar to this was implemented by Fu (1998).

The stereotype model rejects the principle on which the ordered logit model is based. Anderson (1984) argues that there are two distinct types of ordered categorical variables: "grouped continuous", such as income, where the "type a" model applies; and "assessed", such as extent of pain relief, where the stereotype model applies. Greenland (1985) independently developed the same model. The stereotype model starts with a multinomial logistic regression model and imposes constraints on this model.

Goodness of fit for ologit can be evaluated by comparing the likelihood value with that obtained by fitting the model with mlogit. Let $\ln L_{1}$ be the log-likelihood value reported by ologit, and let $\ln L_{0}$ be the log-likelihood value reported by mlogit. If there are $p$ independent variables (excluding the constant) and $k$ categories, mlogit will estimate $p(k-1)$ additional parameters. We can then perform a "likelihood-ratio test", that is, calculate $-2\left(\ln L_{1}-\ln L_{0}\right)$, and compare it with $\chi^{2}\{p(k-2)\}$. This test is suggestive only because the ordered logit model is not nested within the multinomial logit model. A large value of $-2\left(\ln L_{1}-\ln L_{0}\right)$ should, however, be taken as evidence of poorness of fit. Marginally large values, on the other hand, should not be taken too seriously.

The coefficients and cutpoints are estimated using maximum likelihood as described in [R] Maximize. In our parameterization, no constant appears, because the effect is absorbed into the cutpoints.
ologit and oprobit begin by tabulating the dependent variable. Category $i=1$ is defined as the minimum value of the variable, $i=2$ as the next ordered value, and so on, for the empirically determined $k$ categories.

The probability of a given observation for ordered logit is

$$
\begin{aligned}
p_{i j}=\operatorname{Pr}\left(y_{j}=i\right) & =\operatorname{Pr}\left(\kappa_{i-1}<\mathbf{x}_{j} \boldsymbol{\beta}+u \leq \kappa_{i}\right) \\
& =\frac{1}{1+\exp \left(-\kappa_{i}+\mathbf{x}_{j} \boldsymbol{\beta}\right)}-\frac{1}{1+\exp \left(-\kappa_{i-1}+\mathbf{x}_{j} \boldsymbol{\beta}\right)}
\end{aligned}
$$

$\kappa_{0}$ is defined as $-\infty$ and $\kappa_{k}$ as $+\infty$.
For ordered probit, the probability of a given observation is

$$
\begin{aligned}
p_{i j}=\operatorname{Pr}\left(y_{j}=i\right) & =\operatorname{Pr}\left(\kappa_{i-1}<\mathbf{x}_{j} \boldsymbol{\beta}+u \leq \kappa_{i}\right) \\
& =\Phi\left(\kappa_{i}-\mathbf{x}_{j} \boldsymbol{\beta}\right)-\Phi\left(\kappa_{i-1}-\mathbf{x}_{j} \boldsymbol{\beta}\right)
\end{aligned}
$$

where $\Phi(\cdot)$ is the standard normal cumulative distribution function.
The log likelihood is

$$
\ln L=\sum_{j=1}^{N} w_{j} \sum_{i=1}^{k} I_{i}\left(y_{j}\right) \ln p_{i j}
$$

where $w_{j}$ is an optional weight and

$$
I_{i}\left(y_{j}\right)=\left\{\begin{array}{l}
1, \text { if } y_{j}=i \\
0, \text { otherwise }
\end{array}\right.
$$

ologit and oprobit support the Huber/White/sandwich estimator of the variance and its clustered version using vce(robust) and vce(cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.

These commands also support estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

## References

Aitchison, J., and S. D. Silvey. 1957. The generalization of probit analysis to the case of multiple responses. Biometrika 44: 131-140. https://doi.org/10.2307/2333245.
Anderson, J. A. 1984. Regression and ordered categorical variables (with discussion). Journal of the Royal Statistical Society, Series B 46: 1-30. https://doi.org/10.1111/j.2517-6161.1984.tb01270.x.
Baetschmann, G., A. Ballantyne, K. E. Staub, and R. Winkelmann. 2020. feologit: A new command for fitting fixed-effects ordered logit models. Stata Journal 20: 253-275.

Bauldry, S., J. Xu, and A. S. Fullerton. 2018. gencrm: A new command for generalized continuation-ratio models. Stata Journal 18: 924-936.
Brant, R. 1990. Assessing proportionality in the proportional odds model for ordinal logistic regression. Biometrics 46: 1171-1178. https://doi.org/10.2307/2532457.
Cameron, A. C., and P. K. Trivedi. 2005. Microeconometrics: Methods and Applications. New York: Cambridge University Press.
Fagerland, M. W. 2014. adjcatlogit, ccrlogit, and ucrlogit: Fitting ordinal logistic regression models. Stata Journal 14: 947-964.

Fu, V. K. 1998. sg88: Estimating generalized ordered logit models. Stata Technical Bulletin 44: 27-30. Reprinted in Stata Technical Bulletin Reprints, vol. 8, pp. 160-164. College Station, TX: Stata Press.
Fullerton, A. S., and J. Xu. 2016. Ordered Regression Models: Parallel, Partial, and Non-Parallel Alternatives. Boca Raton, FL: CRC Press.

Greenland, S. 1985. An application of logistic models to the analysis of ordinal responses. Biometrical Journal 27: 189-197. https://doi.org/10.1002/bimj. 4710270212.
Liu, X. 2016. Applied Ordinal Logistic Regression Using Stata. Thousand Oaks, CA: Sage.
Long, J. S. 1997. Regression Models for Categorical and Limited Dependent Variables. Thousand Oaks, CA: Sage.
Long, J. S., and J. Freese. 2014. Regression Models for Categorical Dependent Variables Using Stata. 3rd ed. College Station, TX: Stata Press.
McCullagh, P. 1977. A logistic model for paired comparisons with ordered categorical data. Biometrika 64: 449-453. https://doi.org/10.2307/2345320.
_-. 1980. Regression models for ordinal data (with discussion). Journal of the Royal Statistical Society, Series B 42: 109-142. https://doi.org/10.1111/j.2517-6161.1980.tb01109.x.
McCullagh, P., and J. A. Nelder. 1989. Generalized Linear Models. 2nd ed. London: Chapman and Hall/CRC.
McKelvey, R. D., and W. Zavoina. 1975. A statistical model for the analysis of ordinal level dependent variables. Journal of Mathematical Sociology 4: 103-120. https://doi.org/10.1080/0022250X.1975.9989847.

Miranda, A., and S. Rabe-Hesketh. 2006. Maximum likelihood estimation of endogenous switching and sample selection models for binary, ordinal, and count variables. Stata Journal 6: 285-308.
Peterson, B., and F. E. Harrell, Jr. 1990. Partial proportional odds models for ordinal response variables. Journal of the Royal Statistical Society, Series C 39: 205-217. https://doi.org/10.2307/2347760.
Smith, E. K., M. G. Lacy, and A. Mayer. 2019. Performance simulations for categorical mediation: Analyzing khb estimates of mediation in ordinal regression models. Stata Journal 19: 913-930.
Williams, R. 2006. Generalized ordered logit/partial proportional odds models for ordinal dependent variables. Stata Journal 6: 58-82.
-_. 2010. Fitting heterogeneous choice models with oglm. Stata Journal 10: 540-567.
Xu, J., and J. S. Long. 2005. Confidence intervals for predicted outcomes in regression models for categorical outcomes. Stata Journal 5: 537-559.

## Also see

[R] ologit postestimation - Postestimation tools for ologit
[R] clogit - Conditional (fixed-effects) logistic regression
$[\mathrm{R}]$ logistic - Logistic regression, reporting odds ratios
[R] logit - Logistic regression, reporting coefficients
[R] mlogit - Multinomial (polytomous) logistic regression
[R] oprobit - Ordered probit regression
[R] slogit - Stereotype logistic regression
[R] ziologit - Zero-inflated ordered logit regression
[BAYES] bayes: ologit - Bayesian ordered logistic regression
[CM] cmrologit - Rank-ordered logit choice model
[CM] cmroprobit - Rank-ordered probit choice model
[FMM] fmm: ologit - Finite mixtures of ordered logistic regression models
[ME] meologit - Multilevel mixed-effects ordered logistic regression
[MI] Estimation - Estimation commands for use with mi estimate
[SVY] svy estimation - Estimation commands for survey data
[XT] xtologit - Random-effects ordered logistic models
[U] 20 Estimation and postestimation commands

## Title

ologit postestimation - Postestimation tools for ologit

| Postestimation commands | predict | margins |
| :--- | :--- | :--- |
| Reference | Also see |  |

## Postestimation commands

The following postestimation commands are available after ologit:

| Command | Description |
| :---: | :---: |
| contrast | contrasts and ANOVA-style joint tests of estimates |
| estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estat (svy) | postestimation statistics for survey data |
| estimates | cataloging estimation results |
| etable | table of estimation results |
| * forecast | dynamic forecasts and simulations |
| * hausman | Hausman's specification test |
| lincom | point estimates, standard errors, testing, and inference for linear combinations of coefficients |
| linktest | link test for model specification |
| * Irtest | likelihood-ratio test |
| margins | marginal means, predictive margins, marginal effects, and average marginal effects |
| marginsplot | graph the results from margins (profile plots, interaction plots, etc.) |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients |
| predict | probabilities, linear predictions and their SEs, etc. |
| predictnl | point estimates, standard errors, testing, and inference for generalized predictions |
| pwcompare | pairwise comparisons of estimates |
| suest | seemingly unrelated estimation |
| test | Wald tests of simple and composite linear hypotheses |
| testnl | Wald tests of nonlinear hypotheses |

[^43]
## predict

## Description for predict

predict creates a new variable containing predictions such as probabilities, linear predictions, and standard errors.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

```
    predict [type] {stub*|newvar | newvarlist } [if ] [in] [, statistic
        outcome(outcome) nooffset]
    predict [type] stub* [if] [in], scores
    statistic Description
```


## Main

pr predicted probabilities; the default
$\mathrm{xb} \quad$ linear prediction
stdp standard error of the linear prediction

You specify one or $k$ new variables with pr , where $k$ is the number of outcomes. If you specify one new variable and you do not specify outcome(), then outcome(\#1) is assumed.
You specify one new variable with xb and stdp.
These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

## Options for predict

Main
pr , the default, computes the predicted probabilities for all outcomes or for a specific outcome. To compute probabilities for all outcomes, you specify $k$ new variables, where $k$ is the number of categories of the dependent variable. Alternatively, you can specify stub*; in which case, pr will store predicted probabilities in variables stub1, stub2, ..., stubk. To compute the probability for a specific outcome, you specify one new variable and, optionally, the outcome value in option outcome(); if you omit outcome(), the first outcome value, outcome(\#1), is assumed.

Say that you fit a model by typing estimation_cmd y x1 x2, and y takes on four values. Then, you could type predict p1 p2 p3 p4 to obtain all four predicted probabilities; alternatively, you could type predict $p *$ to generate the four predicted probabilities. To compute specific probabilities one at a time, you can type predict p 1 , outcome (\#1) (or simply predict p1), predict p 2 , outcome(\#2), and so on. See option outcome() for other ways to refer to outcome values.
xb calculates the linear prediction. You specify one new variable, for example, predict linear, xb . The linear prediction is defined, ignoring the contribution of the estimated cutpoints.
stdp calculates the standard error of the linear prediction. You specify one new variable, for example, predict se, stdp.
outcome (outcome) specifies for which outcome the predicted probabilities are to be calculated. outcome() should contain either one value of the dependent variable or one of $\# 1, \# 2, \ldots$, with \#1 meaning the first category of the dependent variable, \#2 meaning the second category, etc. outcome() is available only with the default pr option.
nooffset is relevant only if you specified offset (varname) for ologit. It modifies the calculations made by predict so that they ignore the offset variable; the linear prediction is treated as $\mathbf{x}_{j} \mathbf{b}$ rather than as $\mathbf{x}_{j} \mathbf{b}+$ offset $_{j}$. nooffset is not allowed with scores.
scores calculates equation-level score variables. The number of score variables created will equal the number of outcomes in the model. If the number of outcomes in the model was $k$, then the first new variable will contain $\partial \ln L / \partial\left(\mathbf{x}_{j} \mathbf{b}\right)$;
the second new variable will contain $\partial \ln L / \partial \kappa_{1}$;
the third new variable will contain $\partial \ln L / \partial \kappa_{2}$;
and the $k$ th new variable will contain $\partial \ln L / \partial \kappa_{k-1}$, where $\kappa_{i}$ refers to the $i$ th cutpoint.

## margins

## Description for margins

margins estimates margins of response for probabilities and linear predictions.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...)...] [options]
```


## statistic

Description

| default | probabilities for each outcome |
| :--- | :--- |
| pr | probability for a specified outcome |
| xb | linear prediction |
| stdp | not allowed with margins |

pr defaults to the first outcome.
Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see $[R]$ margins.

## Remarks and examples

See [U] 20 Estimation and postestimation commands for instructions on obtaining the variancecovariance matrix of the estimators, predicted values, and hypothesis tests. Also see [R] lrtest for performing likelihood-ratio tests.
> Example 1
In example 2 of [R] ologit, we fit the model ologit rep77 foreign length mpg. The predict command can be used to obtain the predicted probabilities.

We type predict followed by the names of the new variables to hold the predicted probabilities, ordering the names from low to high. In our data, the lowest outcome is "poor", and the highest is "excellent". We have five categories, so we must type five names following predict; the choice of names is up to us:

```
. use https://www.stata-press.com/data/r18/fullauto
(Automobile models)
. ologit rep77 foreign length mpg
    (output omitted)
. predict poor fair avg good exc
(option pr assumed; predicted probabilities)
. list exc good make model rep78 if rep77>=., sep(4) divider
```

| exc | good | make | model | rep78 |
| :---: | :---: | :---: | :---: | :---: |
| . 0033341 | . 0393056 | AMC | Spirit |  |
| . 0098392 | . 1070041 | Buick | Opel |  |
| . 0023406 | . 0279497 | Ford | Fiesta | Good |
| . 015697 | . 1594413 | Merc. | Monarch | Average |
| . 065272 | . 4165188 | Peugeot | 604 |  |
| . 005187 | . 059727 | Plym. | Horizon | Average |
| . 0261461 | . 2371826 | Plym. | Sapporo |  |
| . 0294961 | . 2585825 | Pont. | Phoenix |  |

The eight cars listed were introduced after 1977, so they do not have 1977 repair records in our data. We predicted what their 1977 repair records might have been using the fitted model. We see that, based on its characteristics, the Peugeot 604 had about a $41.65+6.53 \approx 48.2 \%$ chance of a good or excellent repair record. The Ford Fiesta, which had only a $3 \%$ chance of a good or excellent repair record, in fact, had a good record when it was introduced in the following year.

## Technical note

For ordered logit, predict, xb produces $S_{j}=x_{1 j} \beta_{1}+x_{2 j} \beta_{2}+\cdots+x_{k j} \beta_{k}$. The ordered-logit predictions are then the probability that $S_{j}+u_{j}$ lies between a pair of cutpoints, $\kappa_{i-1}$ and $\kappa_{i}$. Some handy formulas are

$$
\begin{aligned}
\operatorname{Pr}\left(S_{j}+u_{j}<\kappa\right) & =1 /\left(1+e^{S_{j}-\kappa}\right) \\
\operatorname{Pr}\left(S_{j}+u_{j}>\kappa\right) & =1-1 /\left(1+e^{S_{j}-\kappa}\right) \\
\operatorname{Pr}\left(\kappa_{1}<S_{j}+u_{j}<\kappa_{2}\right) & =1 /\left(1+e^{S_{j}-\kappa_{2}}\right)-1 /\left(1+e^{S_{j}-\kappa_{1}}\right)
\end{aligned}
$$

Rather than using predict directly, we could calculate the predicted probabilities by hand. If we wished to obtain the predicted probability that the repair record is excellent and the probability that it is good, we look back at ologit's output to obtain the cutpoints. We find that "good" corresponds to the interval /cut3 $<S_{j}+u</$ cut4 and "excellent" to the interval $S_{j}+u>/$ cut4:
. predict score, xb

- generate probgood $=1 /(1+\exp ($ score-_b[/cut4] $))-1 /(1+\exp ($ score-_b[/cut3] $))$
. generate probexc = $1-1 /(1+\exp ($ score-_b[/cut4] $))$
The results of our calculation will be the same as those produced in the previous example. We refer to the estimated cutpoints just as we would any coefficient, so _b[/cut3] refers to the value of the /cut3 coefficient; see [U] 13.5 Accessing coefficients and standard errors.


## Reference

Fagerland, M. W., and D. W. Hosmer, Jr. 2017. How to test for goodness of fit in ordinal logistic regression models. Stata Journal 17: 668-686.

## Also see

$[\mathrm{R}]$ ologit - Ordered logistic regression
[U] 20 Estimation and postestimation commands

## Title

oneway - One-way analysis of variance

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

The oneway command reports one-way analysis-of-variance (ANOVA) models and performs multiplecomparison tests.

If you wish to fit more complicated ANOVA layouts or wish to fit analysis-of-covariance (ANCOVA) models, see [R] anova.

See [D] encode for examples of fitting ANOVA models on string variables.
See $[R]$ loneway for an alternative oneway command with slightly different features.

## Quick start

One-way ANOVA model of $y$ for factor a oneway y a

Report the mean and std. dev. of $y$ and number of observations for each level of a oneway y a, tabulate

Report all pairwise comparisons of the means of y across levels of a with $p$-values adjusted using Bonferroni's procedure
oneway y a, bonferroni
Same as above, but adjust $p$-values for multiple comparisons using Scheffé's method oneway y a, scheffe

## Menu

Statistics $>$ Linear models and related $>$ ANOVA/MANOVA $>$ One-way ANOVA

## Syntax

oneway response_var factor_var [if] [in] [weight] [, options]
options
Main
bonferroni
scheffe
sidak
tabulate
[no] means
[no] standard
[no] freq
[no] obs
noanova
nolabel
wrap
missing

Description
by and collect are allowed; see [U] 11.1.10 Prefix commands.
aweights and fweights are allowed; see [U] 11.1.6 weight.

## Options

$\qquad$
bonferroni reports the results of a Bonferroni multiple-comparison test.
scheffe reports the results of a Scheffé multiple-comparison test.
sidak reports the results of a Šidák multiple-comparison test.
tabulate produces a table of summary statistics of the response_var by levels of the factor_var. The table includes the mean, standard deviation, frequency, and, if the data are weighted, the number of observations. Individual elements of the table may be included or suppressed by using the [no]means, [no]standard, [no]freq, and [no]obs options. For example, typing
oneway response factor, tabulate means standard
produces a summary table that contains only the means and standard deviations. You could achieve the same result by typing
oneway response factor, tabulate nofreq
[no]means includes or suppresses only the means from the table produced by the tabulate option.
See tabulate above.
[no]standard includes or suppresses only the standard deviations from the table produced by the tabulate option. See tabulate above.
[no]freq includes or suppresses only the frequencies from the table produced by the tabulate option. See tabulate above.
[no]obs includes or suppresses only the reported number of observations from the table produced by the tabulate option. If the data are not weighted, only the frequency is reported. If the data are weighted, the frequency refers to the sum of the weights. See tabulate above.
noanova suppresses the display of the ANOVA table.
nolabel causes the numeric codes to be displayed rather than the value labels in the ANOVA and multiple-comparison test tables.
wrap requests that Stata not break up wide tables to make them more readable.
missing requests that missing values of factor_var be treated as a category rather than as observations to be omitted from the analysis.

## Remarks and examples

Remarks are presented under the following headings:

Introduction<br>Obtaining observed means<br>Multiple-comparison tests<br>Weighted data<br>Video example

## Introduction

The oneway command reports one-way ANOVA models. To perform a one-way layout of a variable called endog on exog, type oneway endog exog.

## $\rangle$ Example 1

We run an experiment varying the amount of fertilizer used in growing apple trees. We test four concentrations, using each concentration in three groves of 12 trees each. Later in the year, we measure the average weight of the fruit.

If all had gone well, we would have had 3 observations on the average weight for each of the four concentrations. Instead, two of the groves were mistakenly leveled by a confused man on a large bulldozer. We are left with the following dataset:


[^44]. list, abbreviate(10)

|  | treatment | weight |
| ---: | ---: | ---: |
| 1. | 1 | 117.5 |
| 2. | 1 | 113.8 |
| 3. | 1 | 104.4 |
| 4. | 2 | 48.9 |
| 5. | 2 | 50.4 |
| 6. | 2 | 58.9 |
| 7. | 3 | 70.4 |
| 8. | 3 | 86.9 |
| 9. | 4 | 87.7 |
| 10. | 4 | 67.3 |

To obtain the one-way ANOVA results, we type

| Analysis of variance |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Between groups | 5295.54433 | 3 | 1765.18144 | 21.46 | 0.0013 |
| Within groups | 493.591667 | 6 | 82.2652778 |  |  |
| Total | 5789.136 | 9 | 643.237333 |  |  |
| Bartlett's equa | riances test | i2( | $=1.3900$ | Prob>c | i2 $=0.708$ |

We find significant (at better than the $1 \%$ level) differences among the four concentrations.

## - Technical note

Rather than using the oneway command, we could have performed this analysis by using anova. Example 1 in $[R]$ anova repeats this same analysis. You may wish to compare the output.

You will find the oneway command quicker than the anova command, and, as you will learn, oneway allows you to perform multiple-comparison tests. On the other hand, anova will let you generate predictions, examine the covariance matrix of the estimators, and perform more general hypothesis tests.

## - Technical note

Although the output is a usual ANOVA table, let's run through it anyway. The between-group sum of squares for the model is 5295.5 with 3 degrees of freedom, resulting in a mean square of $5295.5 / 3 \approx 1765.2$. The corresponding $F$ statistic is 21.46 and has a significance level of 0.0013 . Thus, the model appears to be significant at the $0.13 \%$ level.

The second line summarizes the within-group (residual) variation. The within-group sum of squares is 493.59 with 6 degrees of freedom, resulting in a mean squared error of 82.27.

The between- and residual-group variations sum to the total sum of squares (TSS), which is reported as 5789.1 in the last line of the table. This is the TSS of weight after removal of the mean. Similarly, the between plus residual degrees of freedom sum to the total degrees of freedom, 9. Remember that there are 10 observations. Subtracting 1 for the mean, we are left with 9 total degrees of freedom.

At the bottom of the table, Bartlett's test for equal variances is reported. The value of the statistic is 1.39 . The corresponding significance level ( $\chi^{2}$ with 3 degrees of freedom) is 0.708 , so we cannot reject the assumption that the variances are homogeneous.

## Obtaining observed means

> Example 2
We typed oneway weight treatment to obtain an ANOVA table of weight of fruit by fertilizer concentration. Although we obtained the table, we obtained no information on which fertilizer seems to work the best. If we add the tabulate option, we obtain that additional information:


We find that the average weight was largest when we used fertilizer concentration 1 .

## Multiple-comparison tests

$>$ Example 3: Bonferroni multiple-comparison test
oneway can also perform multiple-comparison tests using either Bonferroni, Scheffé, or Šidák normalizations. For instance, to obtain the Bonferroni multiple-comparison test, we specify the bonferroni option:

```
. oneway weight treatment, bonferroni
            Analysis of variance
\begin{tabular}{|c|c|c|c|c|c|}
\hline Source & SS & df & MS & F & Prob > F \\
\hline Between groups & 5295.54433 & 3 & 1765.18144 & 21.46 & 0.0013 \\
\hline Within groups & 493.591667 & 6 & 82.2652778 & & \\
\hline Total & 5789.136 & 9 & 643.237333 & & \\
\hline \multicolumn{6}{|l|}{\multirow[t]{2}{*}{Bartlett's equal-variances test: chi2(3) \(=1.3900 \quad\) Prob>chi2 \(=0.708\) Comparison of Average weight in grams by Fertilizer (Bonferroni)}} \\
\hline & & & & & \\
\hline
\end{tabular}
\begin{tabular}{r|rrr}
\begin{tabular}{l} 
Row Mean- \\
Col Mean
\end{tabular} & 1 & 2 & 3 \\
\hline 2 & -59.1667 & & \\
& 0.001 & & \\
3 & -33.25 & 25.9167 & \\
& 0.042 & 0.122 & \\
4 & -34.4 & 24.7667 & -1.15 \\
& 0.036 & 0.146 & 1.000
\end{tabular}
```

The results of the Bonferroni test are presented as a matrix. The first entry, -59.17 , represents the difference between fertilizer concentrations 2 and 1 (labeled "Row Mean - Col Mean" in the upper stub of the table). Remember that in the previous example we requested the tabulate option. Looking back, we find that the means of concentrations 1 and 2 are 111.90 and 52.73 , respectively. Thus, $52.73-111.90=-59.17$.

Underneath that number is reported " 0.001 ". This is the Bonferroni-adjusted significance of the difference. The difference is significant at the $0.1 \%$ level. Looking down the column, we see that concentration 3 is also worse than concentration 1 ( $4.2 \%$ level), as is concentration 4 ( $3.6 \%$ level).

On the basis of this evidence, we would use concentration 1 if we grew apple trees.

## > Example 4: Scheffé multiple-comparison test

We can just as easily obtain the Scheffé-adjusted significance levels. Rather than specifying the bonferroni option, we specify the scheffe option.

We will also add the noanova option to prevent Stata from redisplaying the ANOVA table:

```
. oneway weight treatment, noanova scheffe
            Comparison of Average weight in grams by Fertilizer
                            (Scheffe)
\begin{tabular}{r|rrr}
\begin{tabular}{l} 
Row Mean- \\
Col Mean
\end{tabular} & 1 & 2 & 3 \\
\hline 2 & -59.1667 & & \\
& 0.001 & & \\
3 & -33.25 & 25.9167 & \\
& 0.039 & 0.101 & \\
4 & -34.4 & 24.7667 & -1.15 \\
& 0.034 & 0.118 & 0.999
\end{tabular}
```

The differences are the same as those we obtained in the Bonferroni output, but the significance levels are not. According to the Bonferroni-adjusted numbers, the significance of the difference between fertilizer concentrations 1 and 3 is $4.2 \%$. The Scheffé-adjusted significance level is $3.9 \%$.

We will leave it to you to decide which results are more accurate.

## Example 5: Šidák multiple-comparison test

Let's conclude this example by obtaining the Šidák-adjusted multiple-comparison tests. We do this to illustrate Stata's capabilities to calculate these results, because searching across adjustment methods until you find the results you want is not a valid technique for obtaining significance levels.

```
. oneway weight treatment, noanova sidak
    Comparison of Average weight in grams by Fertilizer
        (Sidak)
\begin{tabular}{r|rrr}
\begin{tabular}{l} 
Row Mean- \\
Col Mean
\end{tabular} & 1 & 2 & 3 \\
\hline 2 & -59.1667 & & \\
& 0.001 & & \\
3 & -33.25 & 25.9167 & \\
& 0.041 & 0.116 & \\
4 & -34.4 & 24.7667 & -1.15 \\
& 0.035 & 0.137 & 1.000
\end{tabular}
```

We find results that are similar to the Bonferroni-adjusted numbers.

Henry Scheffé (1907-1977) was born in New York. He studied mathematics at the University of Wisconsin, gaining a doctorate with a dissertation on differential equations. He taught mathematics at Wisconsin, Oregon State University, and Reed College, but his interests changed to statistics and he joined Wilks at Princeton. After periods at Syracuse, UCLA, and Columbia, Scheffé settled in Berkeley from 1953. His research increasingly focused on linear models and particularly ANOVA, on which he produced a celebrated monograph. His death was the result of a bicycle accident.

## Weighted data

## > Example 6

oneway can work with both weighted and unweighted data. Let's assume that we wish to perform a one-way layout of the deathrate on the four census regions of the United States using state data. Our data contain three variables, drate (the deathrate), region (the region), and pop (the population of the state).

To fit the model, we type oneway drate region [weight=pop], although we typically abbreviate weight as w. We will also add the tabulate option to demonstrate how the table of summary statistics differs for weighted data:


When the data are weighted, the summary table has four columns rather than three. The column labeled "Freq." reports the sum of the weights. The overall frequency tells us that there are approximately 226 million people in the United States, or at least there were in 1980.

The ANOVA table is appropriately weighted. Also see [U] 11.1.6 weight.

## Video example

One-way ANOVA in Stata

## Stored results

oneway stores the following in $r()$ :
Scalars

| $r(N)$ | number of observations | $r\left(d f \_m\right)$ | between-group degrees of freedom |
| :--- | :--- | :--- | :--- |
| $r(F)$ | $F$ statistic | $r(r s s)$ | within-group sum of squares |
| $r\left(d f \_r\right)$ | within-group degrees of freedom | $r(c h i 2 b a r t)$ | Bartlett's $\chi^{2}$ |
| $r(m s s)$ | between-group sum of squares | $r\left(d f \_b a r t\right)$ | Bartlett's degrees of freedom |

## Methods and formulas

Methods and formulas are presented under the following headings:
One-way analysis of variance
Bartlett's test
Multiple-comparison tests

## One-way analysis of variance

The model of one-way ANOVA is

$$
y_{i j}=\mu+\alpha_{i}+\epsilon_{i j}
$$

for levels $i=1, \ldots, k$ and observations $j=1, \ldots, n_{i}$. Define $\bar{y}_{i}$ as the (weighted) mean of $y_{i j}$ over $j$ and $\bar{y}$ as the overall (weighted) mean of $y_{i j}$. Define $w_{i j}$ as the weight associated with $y_{i j}$, which is 1 if the data are unweighted. $w_{i j}$ is normalized to sum to $n=\sum_{i} n_{i}$ if aweights are used and is otherwise not normalized. $w_{i}$ refers to $\sum_{j} w_{i j}$, and $w$ refers to $\sum_{i}^{i} w_{i}$.

The between-group sum of squares is then

$$
S_{1}=\sum_{i} w_{i}\left(\bar{y}_{i}-\bar{y}\right)^{2}
$$

The TSS is

$$
S=\sum_{i} \sum_{j} w_{i j}\left(y_{i j}-\bar{y}\right)^{2}
$$

The within-group sum of squares is given by $S_{e}=S-S_{1}$.
The between-group mean square is $s_{1}^{2}=S_{1} /(k-1)$, and the within-group mean square is $s_{e}^{2}=S_{e} /(w-k)$. The test statistic is $F=s_{1}^{2} / s_{e}^{2}$. See, for instance, Snedecor and Cochran (1989).

## Bartlett's test

Bartlett's test assumes that you have $m$ independent, normal, random samples and tests the hypothesis $\sigma_{1}^{2}=\sigma_{2}^{2}=\cdots=\sigma_{m}^{2}$. The test statistic, $M$, is defined as

$$
M=\frac{(T-m) \ln \widehat{\sigma}^{2}-\sum\left(T_{i}-1\right) \ln \widehat{\sigma}_{i}^{2}}{1+\frac{1}{3(m-1)}\left\{\left(\sum \frac{1}{T_{i}-1}\right)-\frac{1}{T-m}\right\}}
$$

where there are $T$ overall observations, $T_{i}$ observations in the $i$ th group, and

$$
\begin{aligned}
\left(T_{i}-1\right) \widehat{\sigma}_{i}^{2} & =\sum_{j=1}^{T_{i}}\left(y_{i j}-\bar{y}_{i}\right)^{2} \\
(T-m) \widehat{\sigma}^{2} & =\sum_{i=1}^{m}\left(T_{i}-1\right) \widehat{\sigma}_{i}^{2}
\end{aligned}
$$

An approximate test of the homogeneity of variance is based on the statistic $M$ with critical values obtained from the $\chi^{2}$ distribution of $m-1$ degrees of freedom. See Bartlett (1937) or Draper and Smith (1998, 56-57).

## Multiple-comparison tests

Let's begin by reviewing the logic behind these adjustments. The "standard" $t$ statistic for the comparison of two means is

$$
t=\frac{\bar{y}_{i}-\bar{y}_{j}}{s \sqrt{\frac{1}{n_{i}}+\frac{1}{n_{j}}}}
$$

where $s$ is the overall standard deviation, $\bar{y}_{i}$ is the measured average of $y$ in group $i$, and $n_{i}$ is the number of observations in the group. We perform hypothesis tests by calculating this $t$ statistic. We simultaneously choose a critical level, $\alpha$, and look up the $t$ statistic corresponding to that level in a table. We reject the hypothesis if our calculated $t$ exceeds the value we looked up. Alternatively, because we have a computer at our disposal, we calculate the significance level $e$ corresponding to our calculated $t$ statistic, and if $e<\alpha$, we reject the hypothesis.

This logic works well when we are performing one test. Now consider what happens when we perform several separate tests, say, $n$ of them. Let's assume, just for discussion, that we set $\alpha$ equal to 0.05 and that we will perform six tests. For each test, we have a 0.05 probability of falsely rejecting the equality-of-means hypothesis. Overall, then, our chances of falsely rejecting at least one of the hypotheses is $1-(1-0.05)^{6} \approx 0.26$ if the tests are independent.

The idea behind multiple-comparison tests is to control for the fact that we will perform multiple tests and to reduce our overall chances of falsely rejecting each hypothesis to $\alpha$ rather than letting our chances increase with each additional test. (See Miller [1981] and Hochberg and Tamhane [1987] for rather advanced texts on multiple-comparison procedures.)

The Bonferroni adjustment (see Miller [1981]; also see van Belle et al. [2004, 534-537]) does this by (falsely but approximately) asserting that the critical level we should use, $a$, is the true critical level, $\alpha$, divided by the number of tests, $n$; that is, $a=\alpha / n$. For instance, if we are going to perform six tests, each at the 0.05 significance level, we want to adopt a critical level of $0.05 / 6 \approx 0.00833$.

We can just as easily apply this logic to $e$, the significance level associated with our $t$ statistic, as to our critical level $\alpha$. If a comparison has a calculated significance of $e$, then its "real" significance, adjusted for the fact of $n$ comparisons, is $n \times e$. If a comparison has a significance level of, say, 0.012 , and we perform six tests, then its "real" significance is 0.072 . If we adopt a critical level of 0.05 , we cannot reject the hypothesis. If we adopt a critical level of 0.10 , we can reject it.

Of course, this calculation can go above 1 , but that just means that there is no $\alpha<1$ for which we could reject the hypothesis. (This situation arises because of the crude nature of the Bonferroni adjustment.) Stata handles this case by simply calling the significance level 1. Thus, the formula for the Bonferroni significance level is

$$
e_{b}=\min (1, e n)
$$

where $n=k(k-1) / 2$ is the number of comparisons.

The Šidák adjustment (Šidák [1967]; also see Winer, Brown, and Michels [1991, 165-166]) is slightly different and provides a tighter bound. It starts with the assertion that

$$
a=1-(1-\alpha)^{1 / n}
$$

Turning this formula around and substituting calculated significance levels, we obtain

$$
e_{s}=\min \left\{1,1-(1-e)^{n}\right\}
$$

For example, if the calculated significance is 0.012 and we perform six tests, the "real" significance is approximately 0.07 .

The Scheffé test (Scheffé [1953, 1959]; also see Kuehl [2000, 97-98]) differs in derivation, but it attacks the same problem. Let there be $k$ means for which we want to make all the pairwise tests. Two means are declared significantly different if

$$
t \geq \sqrt{(k-1) F(\alpha ; k-1, \nu)}
$$

where $F(\alpha ; k-1, \nu)$ is the $\alpha$-critical value of the $F$ distribution with $k-1$ numerator and $\nu$ denominator degrees of freedom. Scheffé's test has the nicety that it never declares a contrast significant if the overall $F$ test is not significant.

Turning the test around, Stata calculates a significance level

$$
\widehat{e}=F\left(\frac{t^{2}}{k-1}, k-1, \nu\right)
$$

For instance, you have a calculated $t$ statistic of 4.0 with 50 degrees of freedom. The simple $t$ test says that the significance level is 0.00021 . The $F$ test equivalent, 16 with 1 and 50 degrees of freedom, says the same. If you are comparing three means, however, you calculate an $F$ test of 8.0 with 2 and 50 degrees of freedom, which says that the significance level is 0.0010 .

## References

Acock, A. C. 2023. A Gentle Introduction to Stata. Rev. 6th ed. College Station, TX: Stata Press.
Altman, D. G. 1991. Practical Statistics for Medical Research. London: Chapman and Hall/CRC.
Bartlett, M. S. 1937. Properties of sufficiency and statistical tests. Proceedings of the Royal Society, Series A 160: 268-282.

Daniel, C., and E. L. Lehmann. 1979. Henry Scheffé 1907-1977. Annals of Statistics 7: 1149-1161. https://doi.org/10.1214/aos/1176344837.
Draper, N., and H. Smith. 1998. Applied Regression Analysis. 3rd ed. New York: Wiley.
Hochberg, Y., and A. C. Tamhane. 1987. Multiple Comparison Procedures. New York: Wiley.
Kuehl, R. O. 2000. Design of Experiments: Statistical Principles of Research Design and Analysis. 2nd ed. Belmont, CA: Duxbury.
Marchenko, Y. V. 2006. Estimating variance components in Stata. Stata Journal 6: 1-21.
Miller, R. G., Jr. 1981. Simultaneous Statistical Inference. 2nd ed. New York: Springer.
Scheffé, H. 1953. A method for judging all contrasts in the analysis of variance. Biometrika 40: 87-104. https://doi.org/10.2307/2333100.
__. 1959. The Analysis of Variance. New York: Wiley.

Šidák, Z. 1967. Rectangular confidence regions for the means of multivariate normal distributions. Journal of the American Statistical Association 62: 626-633. https://doi.org/10.2307/2283989.

Snedecor, G. W., and W. G. Cochran. 1989. Statistical Methods. 8th ed. Ames, IA: Iowa State University Press.
van Belle, G., L. D. Fisher, P. J. Heagerty, and T. S. Lumley. 2004. Biostatistics: A Methodology for the Health Sciences. 2nd ed. New York: Wiley.

Winer, B. J., D. R. Brown, and K. M. Michels. 1991. Statistical Principles in Experimental Design. 3rd ed. New York: McGraw-Hill.

## Also see

[R] anova - Analysis of variance and covariance
[R] loneway - Large one-way ANOVA, random effects, and reliability [PSS-2] power oneway - Power analysis for one-way analysis of variance

## Title

oprobit - Ordered probit regression

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

oprobit fits ordered probit models of ordinal variable depvar on the independent variables indepvars. The actual values taken on by the dependent variable are irrelevant, except that larger values are assumed to correspond to "higher" outcomes.

## Quick start

Ordinal probit model of y on x 1 and categorical variables a and b oprobit y x1 i.a i.b

Model of y on x 1 and a one-period lagged value of x 1 using tsset data oprobit $y$ x1 L.x1

Same as above, but calculate results for each level of catvar and save statistics to myfile.dta statsby, by(catvar) saving(myfile): oprobit y x1 L.x1

## Menu

Statistics $>$ Ordinal outcomes $>$ Ordered probit regression

## Syntax

oprobit depvar [indepvars] [if] [in] [weight] [, options]
options
Description

Model
offset (varname)
constraints (constraints)
SE/Robust
vce (vcetype)

Reporting
level(\#)
nocnsreport
display_options
include varname in model with coefficient constrained to 1
apply specified linear constraints
vcetype may be oim, opg, robust, cluster clustvar, bootstrap, or jackknife
set confidence level; default is level (95)
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
control the maximization process; seldom used
keep collinear variables
display legend instead of statistics
indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
depvar and indepvars may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bayes, bootstrap, by, collect, fmm, fp, jackknife, mfp, mi estimate, nestreg, rolling, statsby, stepwise, and svy are allowed; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: oprobit and [FMM] fmm: oprobit.
vce (bootstrap) and vce (jackknife) are not allowed with the mi estimate prefix; see [MI] mi estimate.
Weights are not allowed with the bootstrap prefix; see $[R]$ bootstrap.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

$\qquad$

SE/Robust
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.

Reporting
level (\#); see [R] Estimation options.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon (style), cformat (\% fimt), pformat (\%fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate (\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from(init_specs); see [R] Maximize. These options are seldom used.

The following options are available with oprobit but is not shown in the dialog box: collinear, coeflegend; see [R] Estimation options.

## Remarks and examples

An ordered probit model is used to estimate relationships between an ordinal dependent variable and a set of independent variables. An ordinal variable is a variable that is categorical and ordered, for instance, "poor", "good", and "excellent", which might indicate a person's current health status or the repair record of a car. If there are only two outcomes, see $[R]$ logistic, $[R]$ logit, and $[R]$ probit. This entry is concerned only with more than two outcomes. If the outcomes cannot be ordered (for example, residency in the north, east, south, or west), see [R] mlogit. This entry is concerned only with models in which the outcomes can be ordered. See [R] logistic for a list of related estimation commands.

In ordered probit, an underlying score is estimated as a linear function of the independent variables and a set of cutpoints. The probability of observing outcome $i$ corresponds to the probability that the estimated linear function, plus random error, is within the range of the cutpoints estimated for the outcome:

$$
\operatorname{Pr}\left(\text { outcome }_{j}=i\right)=\operatorname{Pr}\left(\kappa_{i-1}<\beta_{1} x_{1 j}+\beta_{2} x_{2 j}+\cdots+\beta_{k} x_{k j}+u_{j} \leq \kappa_{i}\right)
$$

$u_{j}$ is assumed to be normally distributed. In either case, we estimate the coefficients $\beta_{1}, \beta_{2}, \ldots$, $\beta_{k}$ together with the cutpoints $\kappa_{1}, \kappa_{2}, \ldots, \kappa_{I-1}$, where $I$ is the number of possible outcomes. $\kappa_{0}$ is taken as $-\infty$, and $\kappa_{I}$ is taken as $+\infty$. All of this is a direct generalization of the ordinary two-outcome probit model.

## > Example 1

In example 2 of [R] ologit, we use a variation of the automobile dataset (see [U] 1.2.2 Example datasets) to analyze the 1977 repair records of 66 foreign and domestic cars. We use ordered logit to explore the relationship of rep77 in terms of foreign (origin of manufacture), length (a proxy for size), and mpg. Here we fit the same model using ordered probit rather than ordered logit:

```
. use https://www.stata-press.com/data/r18/fullauto
(Automobile models)
```

. oprobit rep77 foreign length mpg
Iteration 0: Log likelihood $=-89.895098$
Iteration 1: Log likelihood $=-78.106316$
Iteration 2: Log likelihood $=-78.020086$
Iteration 3: Log likelihood $=-78.020025$
Iteration 4: Log likelihood $=-78.020025$
Ordered probit regression $\quad \begin{aligned} & \text { Number of obs }=\quad 66\end{aligned}$
Log likelihood $=-78.020025$
LR chi2(3) $=23.75$
Prob > chi2 $=0.0000$
Pseudo R2 $=0.1321$

| rep77 | Coefficient | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | :---: | ---: | ---: | ---: |
| foreign | 1.704861 | .4246796 | 4.01 | 0.000 | .8725037 | 2.537217 |
| length | .0468675 | .012648 | 3.71 | 0.000 | .022078 | .0716571 |
| mpg | .1304559 | .0378628 | 3.45 | 0.001 | .0562463 | .2046656 |
| /cut1 | 10.1589 | 3.076754 |  |  | 4.128577 | 16.18923 |
| /cut2 | 11.21003 | 3.107527 |  |  | 5.119389 | 17.30067 |
| /cut3 | 12.54561 | 3.155233 |  |  | 6.361467 | 18.72975 |
| /cut4 | 13.98059 | 3.218793 |  |  | 7.671874 | 20.28931 |

We find that foreign cars have better repair records, as do larger cars and cars with better mileage ratings.

## Stored results

oprobit stores the following in e():
Scalars
e(N)
e(N_cd)
e(k_cat)
e(k)
e(k_aux)
e(k_eq)
e(k_eq_model)
e(k_dv)
$e\left(d f \_m\right)$
e(r2_p)
e(11)
e(ll_0)
e(N_clust)
e(chi2)
$e(p)$
e(rank)
e(ic)
e(rc)
e(converged)
Macros

```
e(cmd)
    e(cmdline)
    e(depvar)
    e(wtype)
```

number of observations
number of completely determined observations
number of categories
number of parameters
number of auxiliary parameters
number of equations in $e(b)$
number of equations in overall model test
number of dependent variables
model degrees of freedom
pseudo- $R^{2}$
log likelihood
log likelihood, constant-only model
number of clusters
$\chi^{2}$
$p$-value for model test
rank of e(V)
number of iterations
return code
1 if converged, 0 otherwise
oprobit
command as typed
name of dependent variable
weight type

```
e(wexp) weight expression
e(title) title in estimation output
e(clustvar) name of cluster variable
e(offset) linear offset variable
e(chi2type) Wald or LR; type of model }\mp@subsup{\chi}{}{2}\mathrm{ test
e(vce)
e(vcetype)
e(opt)
e(which)
e(ml_method)
e(user)
e(technique)
e(properties)
e(predict)
e(marginsdefault)
e(asbalanced)
e(asobserved)
    e(b)
    e(Cns)
    e(ilog)
    e(gradient)
    e(cat)
    e(V)
    e(V_modelbased)
    e(sample) marks estimation sample
```

Matrices
Functions

In addition to the above, the following is stored in r() :

```
Matrices
```

    \(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values,
    and confidence intervals
    Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

See Methods and formulas of [R] ologit.

## References

Aitchison, J., and S. D. Silvey. 1957. The generalization of probit analysis to the case of multiple responses. Biometrika 44: 131-140. https://doi.org/10.2307/2333245.
Bauldry, S., J. Xu, and A. S. Fullerton. 2018. gencrm: A new command for generalized continuation-ratio models. Stata Journal 18: 924-936.
Cameron, A. C., and P. K. Trivedi. 2005. Microeconometrics: Methods and Applications. New York: Cambridge University Press.
Canette, I. 2013. Fitting ordered probit models with endogenous covariates with Stata's gsem command. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2013/11/07/fitting-ordered-probit-models-with-endogenous-covariates-with-statas-gsem-command/.
Chiburis, R., and M. Lokshin. 2007. Maximum likelihood and two-step estimation of an ordered-probit selection model. Stata Journal 7: 167-182.
De Luca, G., and V. Perotti. 2011. Estimation of ordered response models with sample selection. Stata Journal 11: 213-239.

Drukker, D. M. 2016. An ordered-probit inverse probability weighted (IPW) estimator. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2016/09/13/an-ordered-probit-inverse-probability-weighted-ipw-estimator/.
Huismans, J., J. W. Nijenhuis, and A. Sirchenko. 2022. A mixture of ordered probit models with endogenous switching between two latent classes. Stata Journal 22: 557-596.
Long, J. S. 1997. Regression Models for Categorical and Limited Dependent Variables. Thousand Oaks, CA: Sage.
Long, J. S., and J. Freese. 2014. Regression Models for Categorical Dependent Variables Using Stata. 3rd ed. College Station, TX: Stata Press.
Miranda, A., and S. Rabe-Hesketh. 2006. Maximum likelihood estimation of endogenous switching and sample selection models for binary, ordinal, and count variables. Stata Journal 6: 285-308.
Smith, E. K., M. G. Lacy, and A. Mayer. 2019. Performance simulations for categorical mediation: Analyzing khb estimates of mediation in ordinal regression models. Stata Journal 19: 913-930.
Stewart, M. B. 2004. Semi-nonparametric estimation of extended ordered probit models. Stata Journal 4: 27-39.
Williams, R. 2010. Fitting heterogeneous choice models with oglm. Stata Journal 10: 540-567.
Xu, J., and J. S. Long. 2005. Confidence intervals for predicted outcomes in regression models for categorical outcomes. Stata Journal 5: 537-559.

## Also see

[R] oprobit postestimation - Postestimation tools for oprobit
[R] heckoprobit - Ordered probit model with sample selection
[R] hetoprobit - Heteroskedastic ordered probit regression
[R] logistic - Logistic regression, reporting odds ratios
[R] mlogit — Multinomial (polytomous) logistic regression
[R] mprobit - Multinomial probit regression
$[R]$ ologit - Ordered logistic regression
[R] probit - Probit regression
[R] zioprobit - Zero-inflated ordered probit regression
[BAYES] bayes: oprobit - Bayesian ordered probit regression
[CM] cmroprobit - Rank-ordered probit choice model
[ERM] eoprobit - Extended ordered probit regression
[FMM] fmm: oprobit - Finite mixtures of ordered probit regression models
[ME] meoprobit - Multilevel mixed-effects ordered probit regression
[MI] Estimation - Estimation commands for use with mi estimate
[SVY] svy estimation - Estimation commands for survey data
[XT] xtoprobit — Random-effects ordered probit models
[U] 20 Estimation and postestimation commands

## Title

oprobit postestimation - Postestimation tools for oprobit
Postestimation commands predict margins Remarks and examples

## Postestimation commands

The following postestimation commands are available after oprobit:

| Command | Description |
| :---: | :---: |
| contrast | contrasts and ANOVA-style joint tests of estimates |
| estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estat (svy) | postestimation statistics for survey data |
| estimates | cataloging estimation results |
| etable | table of estimation results |
| * forecast | dynamic forecasts and simulations |
| * hausman | Hausman's specification test |
| lincom | point estimates, standard errors, testing, and inference for linear combinations of coefficients |
| linktest | link test for model specification |
| * Irtest | likelihood-ratio test |
| margins | marginal means, predictive margins, marginal effects, and average marginal effects |
| marginsplot | graph the results from margins (profile plots, interaction plots, etc.) |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients |
| predict | probabilities, linear predictions and their SEs, etc. |
| predictnl | point estimates, standard errors, testing, and inference for generalized predictions |
| pwcompare | pairwise comparisons of estimates |
| suest | seemingly unrelated estimation |
| test | Wald tests of simple and composite linear hypotheses |
| testnl | Wald tests of nonlinear hypotheses |

[^45]
## predict

## Description for predict

predict creates a new variable containing predictions such as probabilities, linear predictions, and standard errors.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

```
predict [type] {stub*|newvar|newvarlist } [if] [in] [, statistic
    outcome(outcome) nooffset]
predict [type] stub* [if] [in], scores
```

statistic Description
Main
$\mathrm{pr} \quad$ predicted probabilities; the default
xb linear prediction
stdp standard error of the linear prediction
You specify one or $k$ new variables with pr , where $k$ is the number of outcomes. If you specify one new variable and you do not specify outcome(), then outcome(\#1) is assumed.
You specify one new variable with xb and stdp.
These statistics are available both in and out of sample; type predict ... if e(sample)... if wanted only for the estimation sample.

## Options for predict

## Main

pr , the default, computes the predicted probabilities for all outcomes or for a specific outcome. To compute probabilities for all outcomes, you specify $k$ new variables, where $k$ is the number of categories of the dependent variable. Alternatively, you can specify stub*; in which case, pr will store predicted probabilities in variables stub1, stub2, ..., stubk. To compute the probability for a specific outcome, you specify one new variable and, optionally, the outcome value in option outcome(); if you omit outcome(), the first outcome value, outcome(\#1), is assumed.

Say that you fit a model by typing estimation_cmd y x1 x2, and y takes on four values. Then, you could type predict p1 p2 p3 p4 to obtain all four predicted probabilities; alternatively, you could type predict $p *$ to generate the four predicted probabilities. To compute specific probabilities one at a time, you can type predict p1, outcome(\#1) (or simply predict p1), predict p2, outcome(\#2), and so on. See option outcome() for other ways to refer to outcome values.
xb calculates the linear prediction. You specify one new variable, for example, predict linear, xb . The linear prediction is defined, ignoring the contribution of the estimated cutpoints.
stdp calculates the standard error of the linear prediction. You specify one new variable, for example, predict se, stdp.
outcome (outcome) specifies for which outcome the predicted probabilities are to be calculated. outcome() should contain either one value of the dependent variable or one of \#1, \#2, $\ldots$, with \#1 meaning the first category of the dependent variable, \#2 meaning the second category, etc. outcome() is available only with the default pr option.
nooffset is relevant only if you specified offset (varname) for oprobit. It modifies the calculations made by predict so that they ignore the offset variable; the linear prediction is treated as $\mathbf{x}_{j} \mathbf{b}$ rather than as $\mathbf{x}_{j} \mathbf{b}+$ offset $_{j}$. nooffset is not allowed with scores.
scores calculates equation-level score variables. The number of score variables created will equal the number of outcomes in the model. If the number of outcomes in the model was $k$, then the first new variable will contain $\partial \ln L / \partial\left(\mathbf{x}_{j} \mathbf{b}\right)$;
the second new variable will contain $\partial \ln L / \partial \kappa_{1}$;
the third new variable will contain $\partial \ln L / \partial \kappa_{2}$;
and the $k$ th new variable will contain $\partial \ln L / \partial \kappa_{k-1}$, where $\kappa_{i}$ refers to the $i$ th cutpoint.

## margins

## Description for margins

margins estimates margins of response for probabilities and linear predictions.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...)...] [options]
```


## statistic Description

default probabilities for each outcome
pr probability for a specified outcome
$\mathrm{xb} \quad$ linear prediction
stdp not allowed with margins
pr defaults to the first outcome.
Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

## Remarks and examples

See [U] 20 Estimation and postestimation commands for instructions on obtaining the variancecovariance matrix of the estimators, predicted values, and hypothesis tests. Also see [R] lrtest for performing likelihood-ratio tests.

## $>$ Example 1

In example 1 of [R] oprobit, we fit the model oprobit rep77 foreign length mpg. The predict command can be used to obtain the predicted probabilities. We type predict followed by the names of the new variables to hold the predicted probabilities, ordering the names from low to high. In our data, the lowest outcome is "poor" and the highest is "excellent". We have five categories, so we must type five names following predict; the choice of names is up to us:

```
. use https://www.stata-press.com/data/r18/fullauto
(Automobile models)
. oprobit rep77 foreign length mpg
    (output omitted)
. predict poor fair avg good exc
(option pr assumed; predicted probabilities)
. list make model exc good if rep77>=., sep(4) divider
```

| make | model | exc | good |
| ---: | ---: | ---: | ---: |
| AMC | Spirit | .0006044 | .0351813 |
| Buick | Opel | .0043803 | .1133763 |
| Ford | Fiesta | .0002927 | .0222789 |
| Merc. | Monarch | .0093209 | .1700846 |
| Peugeot | 604 | .0734199 | .4202766 |
| Plym. | Horizon | .001413 | .0590294 |
| Plym. | Sapporo <br> Pont. | .0197543 | .2466034 |
| Phoenix | .0234156 | .266771 |  |

## - Technical note

For ordered probit, predict, xb produces $S_{j}=x_{1 j} \beta_{1}+x_{2 j} \beta_{2}+\cdots+x_{k j} \beta_{k}$. Ordered probit is identical to ordered logit, except that we use different distribution functions for calculating probabilities. The ordered-probit predictions are then the probability that $S_{j}+u_{j}$ lies between a pair of cutpoints $\kappa_{i-1}$ and $\kappa_{i}$. The formulas for ordered probit are

$$
\begin{aligned}
\operatorname{Pr}\left(S_{j}+u<\kappa\right) & =\Phi\left(\kappa-S_{j}\right) \\
\operatorname{Pr}\left(S_{j}+u>\kappa\right) & =1-\Phi\left(\kappa-S_{j}\right)=\Phi\left(S_{j}-\kappa\right) \\
\operatorname{Pr}\left(\kappa_{1}<S_{j}+u<\kappa_{2}\right) & =\Phi\left(\kappa_{2}-S_{j}\right)-\Phi\left(\kappa_{1}-S_{j}\right)
\end{aligned}
$$

Rather than using predict directly, we could calculate the predicted probabilities by hand.
. predict pscore, xb
. generate probexc = normal(pscore-_b[/cut4])
. generate probgood = normal(_b[/cut4]-pscore) - normal(_b[/cut3]-pscore)

## Also see

[R] oprobit - Ordered probit regression
[U] 20 Estimation and postestimation commands

## Title

orthog - Orthogonalize variables and compute orthogonal polynomials

## Description

Syntax
Remarks and examples
Also see

## Quick start

Options for orthog
Methods and formulas

## Menu

Options for orthpoly References

## Description

orthog orthogonalizes a set of variables, creating a new set of orthogonal variables (all of type double), using a modified Gram-Schmidt procedure (Golub and Van Loan 2013). The order of the variables determines the orthogonalization; hence, the "most important" variables should be listed first.

Execution time is proportional to the square of the number of variables. With many ( $>10$ ) variables, orthog will be fairly slow.
orthpoly computes orthogonal polynomials for one variable.

## Quick start

Generate $o x 1$, ox2, and ox3 containing orthogonalized versions of x 1 , x 2 , and x 3 orthog x1 x2 x3, generate(ox1 ox2 ox3)

Same as above

```
orthog x1 x2 x3, generate(ox*)
```

Generate op1, op2, and op3 containing degree 1, 2, and 3 orthogonal polynomials for x 1 orthpoly x1, generate(op1 op2 op3) degree(3)

Same as above
orthpoly x1, generate(op1-op3) degree(3)
Same as above, and generate matrix op containing coefficients of the orthogonal polynomials orthpoly $x 1$, generate(op1-op3) degree(3) poly(op)

## Menu

## orthog

Data $>$ Create or change data $>$ Other variable-creation commands $>$ Orthogonalize variables
orthpoly
Data $>$ Create or change data $>$ Other variable-creation commands $>$ Orthogonal polynomials

## Syntax

Orthogonalize variables

$$
\text { orthog }[\text { varlist }][\text { if }][\text { in }][\text { weight }], \underline{g}{ }^{\text {generate(newvarlist) })}[\underline{\text { matrix }} \text { (matname) }]
$$

Compute orthogonal polynomial

$$
\begin{aligned}
& \text { orthpoly varname }[\text { if }][\text { in }][\text { weight }], \\
& \qquad\{\underline{\text { generate }(\text { newvarlist }) \mid \underline{\text { poly }}(\text { matname })\}[\underline{\text { degree }}(\#)]}
\end{aligned}
$$

orthpoly requires that generate (newvarlist) or poly (matname), or both, be specified.
varlist may contain time-series operators; see [U] 11.4.4 Time-series varlists.
iweights, aweights, fweights, and pweights are allowed, see [U] 11.1.6 weight.

## Options for orthog

Main
generate (newvarlist) is required. generate() creates new orthogonal variables of type double.
For orthog, newvarlist will contain the orthogonalized varlist. If varlist contains $d$ variables, then so will newvarlist. newvarlist can be specified by giving a list of exactly $d$ new variable names, or it can be abbreviated using the styles newvar1-newvard or newvar*. For these two styles of abbreviation, new variables newvar 1 , newvar $2, \ldots$, newvard are generated.
matrix (matname) creates a $(d+1) \times(d+1)$ matrix containing the matrix $R$ defined by $X=Q R$, where $X$ is the $N \times(d+1)$ matrix representation of varlist plus a column of ones and $Q$ is the $N \times(d+1)$ matrix representation of newvarlist plus a column of ones ( $d=$ number of variables in varlist, and $N=$ number of observations).

## Options for orthpoly

$\qquad$ Main
generate (newvarlist) or poly(), or both, must be specified. generate() creates new orthogonal variables of type double. newvarlist will contain orthogonal polynomials of degree 1, 2, ..., $d$ evaluated at varname, where $d$ is as specified by degree $(d)$. newvarlist can be specified by giving a list of exactly $d$ new variable names, or it can be abbreviated using the styles newvar1newvard or newvar*. For these two styles of abbreviation, new variables newvar1, newvar2, ..., newvard are generated.
poly (matname) creates a $(d+1) \times(d+1)$ matrix called matname containing the coefficients of the orthogonal polynomials. The orthogonal polynomial of degree $i \leq d$ is

```
matname[i,d+1] + matname[i, 1]*varname + matname [i, 2]*varname }\mp@subsup{}{}{2
    +\cdots+ matname[i,i]*varname }\mp@subsup{}{}{i
```

The coefficients corresponding to the constant term are placed in the last column of the matrix. The last row of the matrix is all zeros, except for the last column, which corresponds to the constant term.
degree(\#) specifies the highest-degree polynomial to include. Orthogonal polynomials of degree 1, $2, \ldots, d=\#$ are computed. The default is $d=1$.

## Remarks and examples

Orthogonal variables are useful for two reasons. The first is numerical accuracy for highly collinear variables. Stata's regress and other estimation commands can face much collinearity and still produce accurate results. But, at some point, these commands will drop variables because of collinearity. If you know with certainty that the variables are not perfectly collinear, you may want to retain all their effects in the model. If you use orthog or orthpoly to produce a set of orthogonal variables, all variables will be present in the estimation results.

Users are more likely to find orthogonal variables useful for the second reason: ease of interpreting results. orthog and orthpoly create a set of variables such that the "effects" of all the preceding variables have been removed from each variable. For example, if we issue the command

```
. orthog x1 x2 x3, generate(q1 q2 q3)
```

the effect of the constant is removed from $x 1$ to produce $q 1$; the constant and $x 1$ are removed from x 2 to produce q 2 ; and finally the constant, x 1 , and x 2 are removed from x 3 to produce q 3 . Hence,

$$
\begin{aligned}
& \mathrm{q} 1=r_{01}+r_{11} \mathrm{x} 1 \\
& \mathrm{q} 2=r_{02}+r_{12} \mathrm{x} 1+r_{22} \mathrm{x} 2 \\
& \mathrm{q} 3=r_{03}+r_{13} \mathrm{x} 1+r_{23} \mathrm{x} 2+r_{33} \times 3
\end{aligned}
$$

This effect can be generalized and written in matrix notation as

$$
X=Q R
$$

where $X$ is the $N \times(d+1)$ matrix representation of varlist plus a column of ones, and $Q$ is the $N \times(d+1)$ matrix representation of newvarlist plus a column of ones ( $d=$ number of variables in varlist and $N=$ number of observations). The $(d+1) \times(d+1)$ matrix $R$ is a permuted uppertriangular matrix, that is, $R$ would be upper triangular if the constant were first, but the constant is last, so the first row/column has been permuted with the last row/column. Because Stata's estimation commands list the constant term last, this allows $R$, obtained via the matrix() option, to be used to transform estimation results.
$>$ Example 1: orthog
Consider Stata's auto.dta dataset. Suppose that we postulate a model in which price depends on the car's length, weight, headroom, and trunk size (trunk). These predictors are collinear, but not extremely so-the correlations are not that close to 1 :

regress certainly has no trouble fitting this model:
. regress price length weight headroom trunk

| Source | SS | df | MS |
| ---: | :---: | ---: | ---: |
| Model | 236016580 | 4 | 59004145 |
| Residual | 399048816 | 69 | 5783316.17 |
| Total | 635065396 | 73 | 8699525.97 |


| Number of obs | $=$ | 74 |
| :--- | :--- | ---: |
| $\mathrm{~F}(4,69)$ | $=$ | 10.20 |
| Prob $>$ F | $=$ | 0.0000 |
| R-squared | $=$ | 0.3716 |
| Adj R-squared | $=$ | 0.3352 |
| Root MSE | $=$ | 2404.9 |


| price | Coefficient | Std. err. | t | P>\|t| | [95\% conf. interval] |  |
| ---: | ---: | :--- | :---: | :---: | ---: | ---: |
| length | -101.7092 | 42.12534 | -2.41 | 0.018 | -185.747 | -17.67147 |
| weight | 4.753066 | 1.120054 | 4.24 | 0.000 | 2.518619 | 6.987512 |
| headroom | -711.5679 | 445.0204 | -1.60 | 0.114 | -1599.359 | 176.2236 |
| trunk | 114.0859 | 109.9488 | 1.04 | 0.303 | -105.2559 | 333.4277 |
| _cons | 11488.47 | 4543.902 | 2.53 | 0.014 | 2423.638 | 20553.31 |

However, we may believe a priori that length is the most important predictor, followed by weight, headroom, and trunk. We would like to remove the "effect" of length from all the other predictors, remove weight from headroom and trunk, and remove headroom from trunk. We can do this by running orthog, and then we fit the model again using the orthogonal variables:

```
. orthog length weight headroom trunk, gen(olength oweight oheadroom otrunk)
> matrix(R)
. regress price olength oweight oheadroom otrunk
```

| Source | SS | df | MS |
| ---: | :---: | ---: | ---: |
| Model | 236016580 | 4 | 59004145 |
| Residual | 399048816 | 69 | 5783316.17 |
| Total | 635065396 | 73 | 8699525.97 |


| Number of obs | $=$ | 74 |
| :--- | :--- | ---: |
| F (4, 69) | $=$ | 10.20 |
| Prob > F | $=$ | 0.0000 |
| R-squared | $=$ | 0.3716 |
| Adj R-squared | $=$ | 0.3352 |
| Root MSE | $=$ | 2404.9 |


| price | Coefficient | Std. err. | t | P>\|t| | [95\% conf. interval] |  |
| ---: | ---: | :--- | :---: | :---: | ---: | ---: |
| olength | 1265.049 | 279.5584 | 4.53 | 0.000 | 707.3454 | 1822.753 |
| oweight | 1175.765 | 279.5584 | 4.21 | 0.000 | 618.0617 | 1733.469 |
| oheadroom | -349.9916 | 279.5584 | -1.25 | 0.215 | -907.6955 | 207.7122 |
| otrunk | 290.0776 | 279.5584 | 1.04 | 0.303 | -267.6262 | 847.7815 |
| _cons | 6165.257 | 279.5584 | 22.05 | 0.000 | 5607.553 | 6722.961 |

Using the matrix $R$, we can transform the results obtained using the orthogonal predictors back to the metric of original predictors:

```
. matrix b = e(b)*inv(R)'
. matrix list b
b [1,5]
\begin{tabular}{rrrrrr} 
& length & weight & headroom & trunk & -cons \\
y1 & -101.70924 & 4.7530659 & -711.56789 & 114.08591 & 11488.475
\end{tabular}
```


## Technical note

The matrix $R$ obtained using the matrix() option with orthog can also be used to recover $X$ (the original varlist) from $Q$ (the orthogonalized newvarlist), one variable at a time. Continuing with the previous example, we illustrate how to recover the trunk variable:

```
. matrix C = R[1...,"trunk"]'
. matrix score double rtrunk = C
. compare rtrunk trunk
```

|  |  |  | Count |  |  | Minimum | Difference <br> Average | Maximum |
| :--- | ---: | ---: | ---: | ---: | :---: | :---: | :---: | :---: |
| rtrunk>trunk | 74 | $8.88 \mathrm{e}-15$ | $1.91 \mathrm{e}-14$ | $3.55 \mathrm{e}-14$ |  |  |  |  |
|  |  | 74 | $8.88 \mathrm{e}-15$ | $1.91 \mathrm{e}-14$ |  |  |  |  |
| Jointly defined | 74 |  | $3.55 \mathrm{e}-14$ |  |  |  |  |  |
| Total |  |  |  |  |  |  |  |  |

Here the recovered variable rtrunk is almost exactly the same as the original trunk variable. When you are orthogonalizing many variables, this procedure can be performed to check the numerical soundness of the orthogonalization. Because of the ordering of the orthogonalization procedure, the last variable and the variables near the end of the varlist are the most important ones to check.

The orthpoly command effectively does for polynomial terms what the orthog command does for an arbitrary set of variables.

## > Example 2: orthpoly

Again consider the auto.dta dataset. Suppose that we wish to fit the model

$$
\operatorname{mpg}=\beta_{0}+\beta_{1} \text { weight }+\beta_{2} \text { weight }^{2}+\beta_{3} \text { weight }^{3}+\beta_{4} \text { weight }^{4}+\epsilon
$$

We will first compute the regression with natural polynomials:

```
. generate double w1 = weight
. generate double w2 = w1*w1
. generate double w3 = w2*w1
. generate double w4 = w3*w1
. correlate w1-w4
(obs=74)
```

|  | w1 | w2 | w3 | w4 |
| ---: | ---: | ---: | ---: | ---: |
| w1 | 1.0000 |  |  |  |
| w2 | 0.9915 | 1.0000 |  |  |
| w3 | 0.9665 | 0.9916 | 1.0000 |  |
| w4 | 0.9279 | 0.9679 | 0.9922 | 1.0000 |

. regress mpg w1-w4

| Source | SS | df | MS | Number of obs | = | 74 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | F (4, 69) | = | 36.06 |
| Model | 1652.73666 | 4 | 413.184164 | Prob > F | = | 0.0000 |
| Residual | 790.722803 | 69 | 11.4597508 | R -squared | = | 0.6764 |
|  |  |  |  | Adj R-squared | = | 0.6576 |
| Total | 2443.45946 | 73 | 33.4720474 | Root MSE | = | 3.3852 |


| mpg | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | ---: | ---: |
| w1 | .0289302 | .1161939 | 0.25 | 0.804 | -.2028704 | .2607307 |
| w2 | -.0000229 | .0000566 | -0.40 | 0.687 | -.0001359 | .0000901 |
| w3 | $5.74 \mathrm{e}-09$ | $1.19 \mathrm{e}-08$ | 0.48 | 0.631 | $-1.80 \mathrm{e}-08$ | $2.95 \mathrm{e}-08$ |
| w4 | $-4.86 \mathrm{e}-13$ | $9.14 \mathrm{e}-13$ | -0.53 | 0.596 | $-2.31 \mathrm{e}-12$ | $1.34 \mathrm{e}-12$ |
| _cons | 23.94421 | 86.60667 | 0.28 | 0.783 | -148.8314 | 196.7198 |

Some of the correlations among the powers of weight are very large, but this does not create any problems for regress. However, we may wish to look at the quadratic trend with the constant removed, the cubic trend with the quadratic and constant removed, etc. orthpoly will generate polynomial terms with this property:

```
. orthpoly weight, generate(pw*) deg(4) poly(P)
```

. regress mpg pw1-pw4

| Source | SS | df | MS |
| ---: | :---: | ---: | :---: |
| Model | 1652.73666 | 4 | 413.184164 |
| Residual | 790.722803 | 69 | 11.4597508 |
| Total | 2443.45946 | 73 | 33.4720474 |


| Number of obs | $=$ | 74 |
| :--- | :--- | ---: |
| $\mathrm{~F}(4,69)$ | $=$ | 36.06 |
| Prob $>$ F | $=$ | 0.0000 |
| R-squared | $=$ | 0.6764 |
| Adj R-squared | $=$ | 0.6576 |
| Root MSE | $=$ | 3.3852 |


| mpg | Coefficient | Std. err. | t | P>\|t| | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | ---: | ---: |
| pw1 | -4.638252 | .3935245 | -11.79 | 0.000 | -5.423312 | -3.853192 |
| pw2 | .8263545 | .3935245 | 2.10 | 0.039 | .0412947 | 1.611414 |
| pw3 | -.3068616 | .3935245 | -0.78 | 0.438 | -1.091921 | .4781982 |
| pw4 | -.209457 | .3935245 | -0.53 | 0.596 | -.9945168 | .5756028 |
| _cons | 21.2973 | .3935245 | 54.12 | 0.000 | 20.51224 | 22.08236 |

Compare the $p$-values of the terms in the natural polynomial regression with those in the orthogonal polynomial regression. With orthogonal polynomials, it is easy to see that the pure cubic and quartic trends are not significant and that the constant, linear, and quadratic terms each have $p<0.05$.

The matrix P obtained with the poly () option can be used to transform coefficients for orthogonal polynomials to coefficients for natural polynomials:

```
. orthpoly weight, poly(P) deg(4)
. matrix b = e(b)*P
. matrix list b
b [1,5]
\begin{tabular}{lrrrrr} 
& deg 1 & deg 2 & deg 3 & deg 4 & - cons \\
y 1 & .02893016 & -.00002291 & \(5.745 \mathrm{e}-09\) & \(-4.862 \mathrm{e}-13\) & 23.944212
\end{tabular}
```


## Methods and formulas

orthog's orthogonalization can be written in matrix notation as

$$
X=Q R
$$

where $X$ is the $N \times(d+1)$ matrix representation of varlist plus a column of ones and $Q$ is the $N \times(d+1)$ matrix representation of newvarlist plus a column of ones ( $d=$ number of variables in varlist, and $N=$ number of observations). The $(d+1) \times(d+1)$ matrix $R$ is a permuted uppertriangular matrix; that is, $R$ would be upper triangular if the constant were first, but the constant is last, so the first row/column has been permuted with the last row/column.
$Q$ and $R$ are obtained using a modified Gram-Schmidt procedure; see Golub and Van Loan (2013, 254-255) for details. The traditional Gram-Schmidt procedure is notoriously unsound, but the modified procedure is good. orthog performs two passes of this procedure.
orthpoly uses the Christoffel-Darboux recurrence formula (Abramowitz and Stegun 1964).

Both orthog and orthpoly normalize the orthogonal variables such that

$$
Q^{\prime} W Q=M I
$$

where $W=\operatorname{diag}\left(w_{1}, w_{2}, \ldots, w_{N}\right)$ with weights $w_{1}, w_{2}, \ldots, w_{N}$ (all 1 if weights are not specified), and $M$ is the sum of the weights (the number of observations if weights are not specified).

## References

Abramowitz, M., and I. A. Stegun, ed. 1964. Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables. Washington, DC: National Bureau of Standards.

Golub, G. H., and C. F. Van Loan. 2013. Matrix Computations. 4th ed. Baltimore: Johns Hopkins University Press.

## Also see

$[\mathrm{R}]$ regress - Linear regression

## Title

pcorr - Partial and semipartial correlation coefficients

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Remarks and examples | Stored results | Methods and formulas | Acknowledgment |
| References | Also see |  |  |

## Description

pcorr displays the partial and semipartial correlation coefficients of a specified variable with each variable in a varlist after removing the effects of all other variables in the varlist. The squared correlations and corresponding significance are also reported.

## Quick start

Partial and semipartial correlations of v1 with v2, v3, and v4
pcorr v1 v2 v3 v4
Same as above, but for each level of categorical variable catvar
by catvar: pcorr v1 v2 v3 v4
Partial and semipartial correlations of v5 with v6, v7, and v8, and a one-period lag of v5 using tsset data
pcorr v5 L.v5 v6 v7 v8

## Menu

Statistics $>$ Summaries, tables, and tests $>$ Summary and descriptive statistics $>$ Partial correlations

## Syntax

pcorr varname varlist [if] [in] [weight]
varlist may contain factor variables; see [U] 11.4.3 Factor variables.
varname and varlist may contain time-series operators; see [U] 11.4.4 Time-series varlists.
by and collect are allowed; see [U] 11.1.10 Prefix commands.
aweights and fweights are allowed; see [U] 11.1.6 weight.

## Remarks and examples

Assume that $y$ is determined by $x_{1}, x_{2}, \ldots, x_{k}$. The partial correlation between $y$ and $x_{1}$ is an attempt to estimate the correlation that would be observed between $y$ and $x_{1}$ if the other $x$ 's did not vary. The semipartial correlation, also called part correlation, between $y$ and $x_{1}$ is an attempt to estimate the correlation that would be observed between $y$ and $x_{1}$ after the effects of all other $x$ 's are removed from $x_{1}$ but not from $y$.

Both squared correlations estimate the proportion of the variance of $y$ that is explained by each predictor. The squared semipartial correlation between $y$ and $x_{1}$ represents the proportion of variance in $y$ that is explained by $x_{1}$ only. This squared correlation can also be interpreted as the decrease in the model's $R^{2}$ value that results from removing $x_{1}$ from the full model. Thus, one could use the squared semipartial correlations as criteria for model selection. The squared partial correlation between $y$ and $x_{1}$ represents the proportion of variance in $y$ not associated with any other $x$ 's that is explained by $x_{1}$. Thus, the squared partial correlation gives an estimate of how much of the variance of $y$ not explained by the other $x$ 's is explained by $x_{1}$.

## > Example 1

Using our automobile dataset (described in [U] 1.2.2 Example datasets), we can obtain the simple correlations between price, mpg, weight, and foreign from correlate (see [R] correlate):

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. correlate price mpg weight foreign
(obs=74)
```

|  | price | mpg | weight | foreign |
| ---: | ---: | ---: | ---: | ---: |
| price | 1.0000 |  |  |  |
| mpg | -0.4686 | 1.0000 |  |  |
| weight | 0.5386 | -0.8072 | 1.0000 |  |
| foreign | 0.0487 | 0.3934 | -0.5928 | 1.0000 |

Although correlate gave us the full correlation matrix, our interest is in just the first column. We find, for instance, that the higher the mpg, the lower the price. We obtain the partial and semipartial correlation coefficients by using pcorr:

| Partial and semipartial correlations of price with |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Variable | Partial corr. | Semipartial corr. | Partial <br> corr. ${ }^{2}$ | Semipartial corr.^2 | Significance value |
| mpg | 0.0352 | 0.0249 | 0.0012 | 0.0006 | 0.7693 |
| weight | 0.5488 | 0.4644 | 0.3012 | 0.2157 | 0.0000 |
| foreign | 0.5402 | 0.4541 | 0.2918 | 0.2062 | 0.0000 |

We now find that the partial and semipartial correlations of price with mpg are near 0 . In the simple correlations, we found that price and foreign were virtually uncorrelated. In the partial and semipartial correlations, we find that price and foreign are positively correlated. The nonsignificance of mpg tells us that the amount in which $R^{2}$ decreases by removing mpg from the model is not significant. We find that removing either weight or foreign results in a significant drop in the $R^{2}$ of the model.

## - Technical note

Use caution when interpreting the above results. As we said at the outset, the partial and semipartial correlation coefficients are an attempt to estimate the correlation that would be observed if the effects of all other variables were taken out of both $y$ and $x$ or only $x$. pcorr makes it too easy to ignore the fact that we are fitting a model. In the example above, the model is

$$
\text { price }=\beta_{0}+\beta_{1} \mathrm{mpg}+\beta_{2} \text { weight }+\beta_{3} \text { foreign }+\epsilon
$$

which is, in all honesty, a rather silly model. Even if we accept the implied economic assumptions of the model-that consumers value mpg, weight, and foreign-do we really believe that consumers place equal value on every extra 1,000 pounds of weight? That is, have we correctly parameterized the model? If we have not, then the estimated partial and semipartial correlation coefficients may not represent what they claim to represent. Partial and semipartial correlation coefficients are a reasonable way to summarize data if we are convinced that the underlying model is reasonable. We should not, however, pretend that there is no underlying model and that these correlation coefficients are unaffected by the assumptions and parameterization.

## Stored results

pcorr stores the following in $r()$ :
Scalars
$r(N) \quad$ number of observations
$r(\mathrm{df}) \quad$ degrees of freedom
Matrices
$r$ (p_corr) partial correlation coefficient vector
$r$ (sp_corr) semipartial correlation coefficient vector

## Methods and formulas

Results are obtained by fitting a linear regression of varname on varlist; see [R] regress. The partial correlation coefficient between varname and each variable in varlist is then calculated as

$$
\frac{t}{\sqrt{t^{2}+n-k}}
$$

(Greene 2018, 39), where $t$ is the $t$ statistic, $n$ is the number of observations, and $k$ is the number of independent variables, including the constant but excluding any dropped variables.

The semipartial correlation coefficient between varname and each variable in varlist is calculated as

$$
\operatorname{sign}(t) \sqrt{\frac{t^{2}\left(1-R^{2}\right)}{n-k}}
$$

(Cohen et al. 2003, 89), where $R^{2}$ is the model $R^{2}$ value, and $t, n$, and $k$ are as described above.
The significance is given by $2 \operatorname{Pr}\left(t_{n-k}>|t|\right)$, where $t_{n-k}$ follows a Student's $t$ distribution with $n-k$ degrees of freedom.

## Acknowledgment

The addition of semipartial correlation coefficients to pcorr is based on the pcorr2 command by Richard Williams of the Department of Sociology at the University of Notre Dame.

## References

Cohen, J., P. Cohen, S. G. West, and L. S. Aiken. 2003. Applied Multiple Regression/Correlation Analysis for the Behavioral Sciences. 3rd ed. Hillsdale, NJ: Erlbaum.
Greene, W. H. 2018. Econometric Analysis. 8th ed. New York: Pearson.

## Also see

[R] correlate - Correlations of variables
[R] spearman - Spearman's and Kendall's correlations

## Title

permute - Permutation tests

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

permute performs permutation tests using Monte Carlo permutations or by enumeration of all possible distinct permutations. A single variable is chosen to be permuted, and the permutation distribution is estimated (or in the case of enumeration, fully determined) for specified statistics returned by a Stata command or a user-written program.

## Quick start

Estimate $p$-values for a permutation test of the coefficient of x in a linear regression, permuting values of the outcome y
permute y _b x$]$ : regress $\mathrm{y} x$
Test for $r$ (mystat) returned by program myprog, permuting values of $y$
permute y r(mystat): myprog
Same as above, but increase the number of permutations from the default of 100 to 10,000
permute y r(mystat), reps(10000): myprog
Same as above, but display a dot for every 100 permutations instead of every permutation permute y r(mystat), reps(10000) dots(100): myprog

Same as above, but set the random-number seed for reproducibility, and save the permuted statistics in myfile.dta

```
permute y r(mystat), reps(10000) dots(100) rseed(1) saving(myfile): ///
        myprog
```

Test for $r$ (mystat1) and $r$ (mystat2), naming the statistics stat1 and stat2, respectively permute y stat1=r(mystat1) stat2=r(mystat2), reps(10000) rseed(1): /// myprog

Perform permutations within strata defined by svar
permute y stat=r(mystat), reps(10000) rseed(1) strata(svar): myprog
Enumerate the full permutation distribution and display a dot for every 1,000 permutations
permute $y$ stat=r(mystat), enumerate dots(1000): myprog

## Menu

Statistics $>$ Resampling $>$ Permutation tests

## Syntax

Perform permutation test

```
permute permvar exp_list [, options] : command
```

Report saved results

options
Main
reps(\#)
enumerate
Options
rseed (\#) set random-number seed to \#
strata(varlist)
saving (filename, ...)

Reporting
standardize
level(\#)
title(text)
dots(\#)
nodots
nowarning
noisily
trace
verbose
noheader
nolegend
Advanced
nodrop do not drop observations
reject (exp)
eps(\#)
Description compute all possible distinct permutations
permute within strata mean and variance
set confidence level; default is level (95)
use text as title for permutation results
display dots every \# permutations
suppress permutation dots
do not warn when e(sample) is not set
display any output from command
trace command
display full table legend
suppress table header
suppress table legend
specify criterion for invalid results
numerical tolerance; seldom used
perform \# Monte Carlo permutations; default is reps (100)
save results to filename with options for saving in double precision and saving results to filename every \# permutations
standardize test statistic using permutation distribution
collect is allowed; see [U] 11.1.10 Prefix commands.
weights are allowed in command.

| display_options | Description |
| :--- | :--- |
| standardize | standardize test statistic using permutation distribution <br> mean and variance |
| level(\#) | set confidence level; default is level (95) <br> use text as title for results |
| title(text) | display full table legend <br> verbose |
| suppress table header <br> noheader <br> suppress table legend <br> numend | numerical tolerance; seldom used |


| exp_list contains | (name: elist) elist |
| :---: | :---: |
|  | eexp |
| elist contains | $\begin{aligned} & \text { newvar }=(\text { exp }) \\ & (\text { exp }) \end{aligned}$ |
| eexp is | specname |
|  | [eqno]specname |
| specname is | _b |
|  | _b [] |
|  | _se |
|  | _se [] |
| eqno is | \#\# |
|  | name |

exp is a standard Stata expression; see [U] 13 Functions and expressions.
Distinguish between [], which are to be typed, and [], which indicate optional arguments.

## Options

$\square$ Mai
reps (\#) specifies the number of Monte Carlo permutations to perform. The default is reps (100).
The default of 100 permutations is chosen for convenience. In real-world applications, you will most likely need to use more permutations. permute reports the Monte Carlo error, which you can use to evaluate whether the specified number of permutations provides sufficient precision for the reported $p$-value estimates.
enumerate specifies that all possible distinct permutations be computed. This gives the full permutation distribution and $p$-values with no error. reps() and rseed() cannot be specified with enumerate.

The number of all possible distinct permutations is typically extremely large. Only for some small problems will it be practical to fully enumerate the permutation distribution. If all the values of permvar are unique, the number of possible permutations is $N$ !, where $N$ is the number of observations. When permvar has a lot of repeated values (when, for example, it is a $0 / 1$ variable), the number of possible distinct permutations can be considerably less than $N$ ! and may make enumeration feasible.

Before beginning the enumeration, permute will calculate and display the number of distinct permutations, allowing you to press Break when you see that the number of permutations is so large as to make computation time impossibly long.

Options
rseed(\#) sets the random-number seed. Specifying this option is equivalent to typing the following command prior to calling permute:
. set seed \#
strata(varlist) specifies that the permutations be performed within each stratum defined by the values of varlist.
$\operatorname{saving}$ (filename[, double every (\#) replace]) creates a Stata data file (.dta file) consisting of variables for each statistic in exp_list containing the results for each permutation.
double specifies that the results for each permutation be saved as doubles, meaning 8-byte reals. By default, they are saved as floats, meaning 4-byte reals.
every (\#) specifies that results be written to disk every \#th permutation. every() should be specified only in conjunction with saving() when command takes a long time for each permutation. This will allow recovery of partial results should some other software crash your computer. See [P] postfile.
replace specifies that filename be overwritten if it exists.
standardize specifies that the observed value of each test statistic be standardized. That is, the observed test statistic $T_{\text {obs }}$ is standardized by calculating $\left[T_{\text {obs }}-\operatorname{mean}(T)\right] / \sqrt{\operatorname{Var}(T)}$, where mean $(T)$ and $\operatorname{Var}(T)$ are the mean and variance of the permutation distribution of $T$. Standardized test statistics are useful for seeing roughly how extreme (and in what direction) the observed test statistic is.
level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [R] level.
title (text) specifies a title to be displayed above the table of permutation results.
dots(\#) and nodots specify whether to display permutation dots. By default, one dot character is displayed for each successful permutation. An " $x$ " is displayed if command returns an error or if any value in exp_list is missing. You can also control whether dots are displayed using set dots; see $[R]$ set.
dots(\#) displays dots every \# permutations. dots (0) is a synonym for nodots.
nodots suppresses display of the permutation dots.
nowarning suppresses the printing of a warning message when command does not set e(sample).
noisily requests that any output from command be displayed. This option implies the nodots option.
trace causes a trace of the execution of command to be displayed. This option implies the noisily option.
verbose requests that the full table legend be displayed when multiple coefficients or standard errors are specified using the _b or _se notation.
noheader suppresses display of the table header. This option implies the nolegend option.
nolegend suppresses display of the table legend. The table legend identifies the rows of the table with the expressions they represent.

Advanced
nodrop prevents permute from dropping observations outside the if and in qualifiers. nodrop will also cause permute to ignore the contents of e(sample) if it exists as a result of running command. By default, permute temporarily drops out-of-sample observations.
reject (exp) specifies an expression that indicates when results should be rejected. When exp is true, the resulting values are reset to missing values.
eps (\#) specifies the numerical tolerance for testing $T \leq T_{\text {obs }}$ and $T \geq T_{\text {obs }}$, where $T$ is the test statistic and $T_{\text {obs }}$ is its observed value. These are considered true if, respectively, $T \leq T_{\text {obs }}+\#$ or $T \geq T_{\text {obs }}-\#$. The default is eps (1e-7). You will not have to specify eps() under normal circumstances.

## Remarks and examples

Remarks are presented under the following headings:

Introduction<br>Monte Carlo permutation tests<br>Two-sided $p$-values from permutation tests<br>One-sided permutation test<br>Enumeration<br>Efficiency considerations for Monte Carlo permutations<br>Efficiency considerations for enumeration

## Introduction

Permutation tests are based on the idea of scrambling-that is, permuting-the order of a variable in all possible ways, calculating the value of a test statistic for each permutation, and taking this set of values of the statistic as its distribution.

For instance, consider the correlation of two variables, $\operatorname{corr}(\mathbf{x}, \mathbf{y})$, where $\mathbf{x}=\left(x_{1}, \ldots, x_{n}\right)$ and $\mathbf{y}=\left(y_{1}, \ldots, y_{n}\right)$. We hold the order of $\mathbf{x}$ fixed and permute the order of $\mathbf{y}$ in all possible ways. For each permutation $\mathbf{y}^{*}$, we calculate $T^{*}=\operatorname{corr}\left(\mathbf{x}, \mathbf{y}^{*}\right)$. The set of $T^{*}$ gives the permutation distribution for the correlation. This permutation distribution is the distribution of the correlation under the null hypothesis that the ordering of the elements of $\mathbf{y}$ are independent of the ordering of the elements of $\mathbf{x}$, conditional on the observed values of $\mathbf{x}$ and $\mathbf{y}$.

Aside: Actually, the null hypothesis does not require independence. A weaker assumption of exchangeability is sufficient. If $\mathbf{x}$ and $\mathbf{y}$ are observed values of the random variates $\mathbf{X}=\left(X_{1}, \ldots, X_{n}\right)$ and $\mathbf{Y}=\left(Y_{1}, \ldots, Y_{n}\right)$, then the joint distribution $f(\mathbf{X}, \mathbf{Y})$ is called exchangeable when it is invariant to the orderings of $X_{1}, \ldots, X_{n}$ and $Y_{1}, \ldots, Y_{n}$.

The $p$-value for the permutation test is the proportion of permutations that produce a test statistic $T^{*}$ as extreme or more extreme than the test statistic $T_{\text {obs }}$ computed using the observed data.
permute estimates $p$-values for permutation tests using Monte Carlo permutations or by enumerating all possible permutations when the enumerate option is specified. To do Monte Carlo permutations, you type

The values in the variable permvar are randomly permuted \# times, each time executing command and collecting the associated values from the expressions in exp_list.
command defines the statistical command to be executed. Most Stata commands and user-written programs can be used with permute, as long as they follow standard Stata syntax; see [U] 11 Language syntax. exp_list specifies the statistics to be collected from the execution of command. Despite the fact that permute works with most Stata commands, that does not mean the resulting permutation test is a sensible test. See, for example, Good (2006).

To enumerate all possible distinct permutations, you type
. permute permvar exp_list, enumerate: command
permute may be used for replaying results, but this feature is appropriate only when a dataset generated by permute is currently in memory or is identified by the using specification.

## Monte Carlo permutation tests

We first demonstrate how to apply the permute prefix by testing for a difference in the distribution of a variable across two groups. Here we perform the test using Monte Carlo permutations. In example 3 , we do the same test using complete enumeration.
$>$ Example 1: Wilcoxon rank-sum test
Let's consider calculating the $p$-value for the Wilcoxon rank-sum test performed by ranksum. Suppose that we collected data from some experiment: y is some measure we took on 17 individuals, and group identifies the group to which an individual belongs.

```
. use https://www.stata-press.com/data/r18/permute2
. list, sepby(group)
```

|  | group | y |
| :---: | ---: | ---: |
| 1. | 1 | 6 |
| 2. | 1 | 11 |
| 3. | 1 | 20 |
| 4. | 1 | 2 |
| 5. | 1 | 9 |
| 6. | 1 | 5 |
|  |  |  |
| 7. | 0 | 2 |
| 8. | 0 | 1 |
| 9. | 0 | 6 |
| 10. | 0 | 0 |
| 11. | 0 | 2 |
| 12. | 0 | 3 |
| 13. | 0 | 3 |
| 14. | 0 | 12 |
| 15. | 0 | 4 |
| 16. | 0 | 1 |
| 17. | 0 | 5 |

We analyze the data using ranksum:
. ranksum y, by (group)
Two-sample Wilcoxon rank-sum (Mann-Whitney) test
group

The test gives an approximate $p$-value of 0.0434 and an exact $p$-value of 0.0436 .
Let's try to reproduce these results using permute. The test statistic $T$ for the Wilcoxon rank-sum test is the sum of the ranks for the first group, which is 79 , and is stored as $r$ (sum_obs). We specify reps (10000) to do 10,000 Monte Carlo permutations and dots (100) to display a dot every 100th permutation. We set the random-number seed so that we can duplicate our results.

```
. set seed 1234
. permute group r(sum_obs), reps(10000) dots(100): ranksum y, by(group)
(running ranksum on estimation sample)
warning: ranksum does not set e(sample), so no observations will be excluded
    from the permutations because of missing values or other reasons. To
    exclude observations, press Break, save the data, drop any
    observations that are to be excluded, and rerun permute.
Permutations (10,000): .........1,000.........2,000..........3,000..........4,000
> .........5,000.........6,000.........7,000..........8,000.........9,000........
> ..10,000 done
Monte Carlo permutation results Number of observations = 17
Permutation variable: group Number of permutations = 10,000
    Command: ranksum y, by(group)
        _pm_1: r(sum_obs)
```

| T | T (obs) | Test | c | n | p | Monte Carlo error |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  |  |  | SE (p) | [95\% | $\mathrm{CI}(\mathrm{p})]$ |
| _pm_1 | 79 | lower | 223 | 10000 | . 0223 | . 0015 | . 0195 | . 0254 |
|  |  | upper | 9817 | 10000 | . 9817 | . 0013 | . 9789 | . 9842 |
|  |  | sided |  |  | . 0446 | . 0021 | . 0406 | . 0486 |

```
Notes: For lower one-sided test, c = #{T <= T(obs)} and p = p_lower = c/n.
    For upper one-sided test, c = #{T >= T(obs)} and p = p_upper = c/n.
    For two-sided test, p = 2*min(p_lower, p_upper); SE and CI approximate.
```

The lengthy message about e(sample) is worth noting. If there were missing values in the data, we might want to drop those observations before running permute. To suppress the message in future runs, use the nowarning option.

The two-sided $p$-value obtained by this Monte Carlo procedure is 0.0446 , which is close to the exact $p$-value of 0.0436 computed by ranksum. See the next section for a description of how two-sided
$p$-values are calculated when performing permutation tests. See example 2 for a test that requires a one-sided $p$-value.

Note that we typed
. permute group ...
rather than

```
. permute y ...
```

We permuted the $0 / 1$ variable group, which defines the groups, rather than the outcome variable $y$. For a statistic dependent on only two variables, it obviously does not matter which one we permute in terms of the theory of the test, but it does matter in terms of the efficiency of how permute does the computation. permute uses a different random shuffling algorithm for $0 / 1$ (or dichotomous) variables than it does with other variables. See Efficiency considerations for Monte Carlo permutations below for details.
permute reports standard errors and confidence intervals for $p$-values because, as with any other Monte Carlo procedure, they are approximations to the true exact $p$-values. These statistics are useful to assess the precision of the computed $p$-values. If you need more precision, specify more permutations in the reps() option. See Methods and formulas for a description of how the standard errors and confidence intervals are calculated.

The confidence interval for the Monte Carlo two-sided $p$-value in this example is [0.0406, 0.0486]. If we want to increase the precision of the $p$-value, we could run permute again with more random permutations to narrow the confidence interval. The total number of possible distinct permutations of group, however, is not extremely large, and we can perform the permutation test using enumeration. See example 3, where we do just that.

## Two-sided p -values from permutation tests

In the above example, we used the two-sided $p$-value for our hypothesis testing. For permutation distributions, two-sided $p$-values require some explanation about how they are calculated.
permute calculates the two-sided $p$-value as $p=2 \min \left(p_{\text {lower }}, p_{\text {upper }}\right)$, where $p_{\text {lower }}$ is the lower one-sided $p$-value and $p_{\text {upper }}$ the upper one-sided $p$-value. (More precisely, $p=$ $\min \left[1,2 \min \left(p_{\text {lower }}, p_{\text {upper }}\right)\right]$ is used because obviously $p$-values must be bounded by 1.)

In general, the $p$-value is defined as the probability under the null hypothesis of obtaining a value of the test statistic $T$ equal to or more extreme than the value $T_{\text {obs }}$ that was actually observed. For one-sided $p$-values, what is "more extreme" is clear. For lower one-sided $p$-values, it is the probability that $T \leq T_{\text {obs }}$, and for upper one-sided $p$-values, it is the probability that $T \geq T_{\text {obs }}$. When $T$ has a symmetric distribution, the two-sided $p$-value is typically defined as the probability that $|T| \geq\left|T_{\text {obs }}\right|$. Permutation distributions, however, are not in general symmetric.

Under a permutation-based null hypothesis, the domain of $T$ consists of all the possible permutations of the underlying data used to calculate $T$. The domain is discrete and finite, and hence the permutation distribution of $T$ is discrete and finite. These finite distributions are symmetric only in certain cases. For instance, with our example of the Wilcoxon rank-sum test, if the data consist of untied ranks, the distribution is symmetric. When there are ties in the ranks, however, the distribution is in most cases not symmetric.

When distributions are asymmetric, what values of $T$ are "more extreme" than $T_{\text {obss }}$ ? Suppose $T_{\text {obs }}$ is below the mean of the distribution. Clearly, the lower-tail values $T \leq T_{\text {obs }}$ are more extreme. But what values of $T$ from the upper tail are more extreme?

For asymmetric distributions, the rationale for using $p=2 \min \left(p_{\text {lower }}, p_{\text {upper }}\right)$ for two-sided tests is the following: It takes the smallest one-sided $p$-value and doubles it. Comparing this two-sided $p$-value against a significance level of, say, 0.05 is equivalent to comparing the smallest one-sided $p$-value against a level of 0.025 . It essentially turns the two-sided test into a one-sided test with the significance level cut in half. So this definition conveniently sidesteps the need to define what values of $T$ from the opposite tail from $T_{\text {obs }}$ are more extreme! Also, it is appropriate for both symmetric and asymmetric distributions.

## One-sided permutation test

In some cases, we will want to perform a permutation test based on a one-sided $p$-value.

## > Example 2: Permutation tests with ANOVA

Consider some fictional data from an experimental randomized complete-block design in which there are 5 subjects each receiving 10 different treatments. We want to test whether any of the treatments have an effect different from the effects of the other treatments.

Let's load the data and list the data for the first two subjects:

```
. use https://www.stata-press.com/data/r18/permute1, clear
. sort subject treatment
. list subject treatment y in 1/20, abbrev(10)
```

|  | subject | treatment | y |
| :---: | :---: | :---: | :---: |
| 1. | 1 | 1 | 4.407557 |
| 2. | 1 | 2 | 4.280349 |
| 3. | 1 | 3 | 4.418574 |
| 4. | 1 | 4 | 4.075359 |
| 5. | 1 | 5 | 3.899775 |
| 6. | 1 | 6 | 5.533271 |
| 7. | 1 | 7 | 5.142111 |
| 8. | 1 | 8 | 5.791124 |
| 9. | 1 | 9 | 4.504411 |
| 10. | 1 | 10 | 4.896333 |
| 11. | 2 | 1 | 5.693386 |
| 12. | 2 | 2 | 4.508785 |
| 13. | 2 | 3 | 5.10376 |
| 14. | 2 | 4 | 5.753985 |
| 15. | 2 | 5 | 5.092277 |
| 16. | 2 | 6 | 4.496496 |
| 17. | 2 | 7 | 6.339948 |
| 18. | 2 | 8 | 4.820389 |
| 19. | 2 | 9 | 5.686253 |
| 20. | 2 | 10 | 6.951727 |

These data may be analyzed using anova.
. anova y treatment subject

|  | Number of obs Root MSE | $\begin{array}{r} 50 \\ .914159 \end{array}$ | R-squared $=$ <br> Adj R -squared $=$ |  | $\begin{aligned} & 0.3544 \\ & 0.1213 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | 16.518219 | 13 | 1.2706322 | 1.52 | 0.1574 |
| treatment | 13.022671 | 9 | 1.4469634 | 1.73 | 0.1174 |
| subject | 3.4955481 | 4 | . 87388703 | 1.05 | 0.3973 |
| Residual | 30.08475 | 36 | . 83568751 |  |  |
| Total | 46.602969 | 49 | . 951081 |  |  |

anova gives a $p$-value of 0.1174 for the treatment effect. This $p$-value is calculated with the assumption of normality for the distribution of the outcome conditional on the means of each treatment and subject effect.

Suppose we do not want to assume normality. The treatments were assigned in a random order to each of the subjects. A null hypothesis of no treatment effect means that the observed values of $y$ and their order were determined by factors other than the treatments. The treatments were essentially labels that had nothing to do with the outcomes, and any other ordering of the labels would be a possible occurrence. That is, we imagine running the experiment multiple times, each with a different ordering of the treatments, but each time, we get the same observed values of $y$. This is the permutation-based formulation of the null hypothesis.

What about the subjects? Each subject gets each of the 10 treatments, so clearly we must permute the treatments within each subject independently of the permutations for the other subjects. We can do this using the strata() option with permute.

If we type ereturn list after anova, we see that the $F$ statistic for treatment is stored in $e\left(F \_1\right)$. This is our test statistic for our permutation test.

We save the dataset containing all the permutations of the test statistic using the saving () option. Specifying the test statistic as F_treatment=e(F_1) labels the test statistic as F_treatment in the output and is also the name of the variable containing the test statistic in permanova.dta, the dataset created by saving(). We set the seed for the random-number generator and also specify the nodots option to suppress the dots in the output.

```
. permute treatment F_treatment=e(F_1), reps(10000) strata(subject)
> saving(permanova) rseed(1234) nodots: anova y treatment subject
```

Monte Carlo permutation results Number of observations $=50$
Permutation variable: treatment Number of strata $=5$
Number of permutations $=10,000$
Command: anova y treatment subject
F_treatment: e(F_1)

| T | T(obs) | Test | c | n | p | Monte Carlo error |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  |  |  | SE(p) | [95\% | $\mathrm{CI}(\mathrm{p})]$ |
| F_treatment | 1.731465 | lower | 8788 | 10000 | . 8788 | . 0033 | . 8722 | . 8851 |
|  |  | upper | 1212 | 10000 | . 1212 | . 0033 | . 1149 | . 1278 |
|  |  | two-sided |  |  | . 2424 | . 0043 | . 2340 | . 2508 |

Notes: For lower one-sided test, $c=\#\{T<=T(o b s)\}$ and $p=p \_l o w e r=c / n$. For upper one-sided test, $c=\#\{T>=T$ (obs) $\}$ and $p=p \_u p p e r=c / n$. For two-sided test, $\mathrm{p}=2 * \min \left(\mathrm{p}_{-}\right.$lower, p _upper) ; SE and CI approximate.
Our test statistic is an $F$ statistic, so we are interested in the number of permutations that have a larger (more extreme) statistic than the 1.73 we obtained with our original data. Therefore, we want the upper one-sided $p$-value, which is 0.1212 . This value is close to the $p$-value given by anova of 0.1174 for the treatment effect.

For an additional example of a permutation test, with an application in epidemiology, see Hayes and Moulton (2017, 237-241).

## Enumeration

When the number of observations, $N$, in a dataset is small, it may be possible to enumerate all possible permutations and obtain $p$-values without the error involved in computing Monte Carlo $p$-values.

When permute does an enumeration, not only does $N$ matter, but the number of distinct values of the permutation variable matters as well. permute does the enumeration by computing only permutations that give different arrangements of the permutation variable; that is, it does not compute any duplicate permutations. So the number of distinct values (and their multiplicity) of the permutation variable determines the number of permutations enumerated and so determines whether enumeration is feasible. See Efficiency considerations for enumeration below for details.
> Example 3: Wilcoxon rank-sum test using enumeration
Here we repeat example 1, but this time we do it by enumerating all possible permutations. We load the data:

The data consist of an outcome y grouped by the variable group. If we tabulate group
. tabulate group

| Group | Freq. | Percent | Cum. |
| ---: | ---: | ---: | ---: |
| 0 | 11 | 64.71 | 64.71 |
| 1 | 6 | 35.29 | 100.00 |
| Total | 17 | 100.00 |  |

we see that group consists of 11 zeros and 6 ones. Hence, there are only $\binom{17}{6}=12,376$ possible distinct permutations of group.

In example 1, we used ranksum to compute the test statistic, but each time ranksum is called, it computes the ranks of $y$. It is unnecessary to recompute the ranks for each permutation. It is better to compute the ranks just once at the outset. We can do this using the rank() function of egen:

```
. egen r = rank(y)
```

The test statistic is the sum of the ranks for either one of the groups. The sum can be computed efficiently using summarize with an if restriction and the meanonly option.

```
. permute group r(sum), enumerate nodrop nowarning dots(100): summarize r
> if group == 1, meanonly
(running summarize on estimation sample)
(enumerating all 12,376 possible permutations)
Permutations (12,376): .........1,000.........2,000..........3,000.........4,000
> ........5,000........6,000.........7,000.........8,000.........9,000.......
> ..10,000.........11,000.........12,000.... done
```

Enumeration permutation results
Number of observations $=17$
Number of permutations $=12,376$
Permutation variable: group
Command: summarize $r$ if group $==1$, meanonly
_pm_1: r(sum)

| T | T (obs) | Test | c | n | p |
| ---: | ---: | ---: | ---: | ---: | ---: |
| pm_1 | 74 | lower | 12142 | 12376 | .9811 |
|  |  | upper | 270 | 12376 | .0218 |
|  |  | two-sided |  |  | .0436 |

Notes: For lower one-sided test, $c=\#\{T<=T(o b s)\}$ and $p=$ p_lower $=c / n$.
For upper one-sided test, $c=\#\{T>=T(o b s)\}$ and $p=p \_u p p e r=c / n$.
For two-sided test, $\mathrm{p}=2 * \min (\mathrm{p}$ _lower, p _upper) .

Note that it is necessary to specify the nodrop option. Otherwise, permute would drop all observations not satisfying if group $==1$ before doing the permutations, and that would not give us what we want.
permute with the enumerate option gave a two-sided $p$-value of 0.0436 , which is the same as the exact $p$-value reported by ranksum, as it should be.

With group as the variable being permuted, the number of distinct permutations is quite small. If, however, we attempt to do the enumeration for all the distinct permutations of $r$ :

```
. permute r r(sum), enumerate nodrop nowarning dots(100):
> summarize r if group == 1, meanonly
(running summarize on estimation sample)
(enumerating all 3.71e+12 possible permutations)
Permutations (3,705,077,376,000): ..........1,000..........2,000.......... 3,000
> .........4,000........-Break-
r(1);
```

Each permutation takes about 0.1 millisecond on our computer. Thus, the enumeration will take $0.1 \times 3.71 \times 10^{12} /(365 \times 24 \times 60 \times 60 \times 1000) \approx 12$ years, so we pressed Break.

## Efficiency considerations for Monte Carlo permutations

Suppose you want to perform a randomization two-sample $t$ test, which is like the two-sample $t$ test that assumes normality, only the randomization test is based on permuting the variable that defines the samples. It is the same as the Wilcoxon rank-sum test, except the observed values of the outcome, rather than their ranks, are used for the test statistic.

So say we have a variable x with continuous outcomes for two groups defined by the variable group, with values 0 or 1 . The randomization two-sample $t$ test could be done using Monte Carlo permutations by typing

```
. permute x r(mu_1), reps(10000): ttest x, by(group)
```

Or by typing

```
. permute group r(mu_1), reps(10000): ttest x, by(group)
```

In the first case, $x$ was permuted, and in the second, group. Both are valid ways to do Monte Carlo permutations. permute is smart, however, and treats a $0 / 1$ variable differently from how it treats a variable with lots of distinct values.

Suppose there are fewer 1 s than 0 s in group. Rather than randomly permuting all the 0 s and 1 s , permute randomly shuffles the 1 s into the 0 s . If there are only a few 1 s relative to the number of 0 s , this method is much faster than permuting all the values. Hence, if you are doing a permutation test that involves two variables, pick the one with the fewest distinct values to be the permvar.

When the strata() option is specified, permute uses special code for the case in which the permvar is dichotomous (with, say, values $y_{0}$ and $y_{1}$ ) and each stratum contains a single observation equal to $y_{1}$ and all other observations in the stratum equal to $y_{0}$ (and $y_{0}$ and $y_{1}$ do not flip or change in value across strata). If you are doing a stratified permutation test and you have such a variable whose permutations will give the test you want, be sure to make it the permvar.

For all types of data, Monte Carlo permutations of the permvar are computed quickly. If the command calculating the test statistic for each permutation is not fast, it is unlikely you will notice the greater speed of permuting a dichotomous variable. If, however, you are using a fast command such as regress, you likely will notice the greater speed.

## Efficiency considerations for enumeration

Doing an enumeration of all possible distinct permutations using the enumerate option, however, is a different story. Here the selection of the permvar is crucial and typically determines whether it is feasible to do the enumeration.

Consider performing the randomization $t$ test we described earlier using enumeration:

```
. permute group r(mu_1), enumerate: ttest x, by(group)
```

Suppose there are 20 observations, 10 with group $=0$ and 10 with group $=1$. Typing permute group ... will enumerate all $\binom{20}{10}=184,756$ possible distinct permutations of group. (When we say "distinct permutations", we mean that duplicate permutations are not computed.)

If, however, we type

```
. permute x r(mu_1), enumerate: ttest x, by(group)
```

permute $\mathrm{x} .$. will attempt to enumerate all possible permutations of x . If all the values of x are unique, there are $20!\approx 2.4 \times 10^{18}$ possible permutations of x , which is much larger than 184,756 .

Hence, a dichotomous variable or a variable with few distinct values should always be chosen as the permvar rather than another variable with many distinct values whenever possible. (To be precise, both the number of distinct values and their multiplicities determine the number of permutations. See Methods and formulas.)

See example 3 for an example using enumeration.

## Stored results

permute stores the following in $r()$ :

Scalars
$r(N)$
r(n_reps)
$r$ (k_exp)
r(k_eexp)
$r$ (n_strata)
r(level)
Macros
$r$ (cmd)
$r$ (command)
$r$ (permvar)
$r$ (enumerate)
r(title)
$r$ (rngstate)
r(exp\#)
r(strata)
r(missing)
Matrices
$r$ (b)
r(b_std)
$r(n)$
r(c_lower)
r(c_upper)
r(p_lower)
r(p_upper)
r(p_twosided)
r(se_p_lower)
r(se_p_upper)
number of observations for command number of permutations performed number of standard expressions number of _b and _se expressions number of strata, if strata() specified confidence level

## permute

command following colon
permutation variable
"enumerate", if enumerate specified
title in output
random-number state used for Monte Carlo permutations
\#th expression
strata variable, if strata() specified
"missing" when one or more expressions equal missing value
observed statistics
standardized observed statistics, if standardize specified
number of nonmissing results
counts for lower one-sided $p$-values
counts for upper one-sided $p$-values
lower one-sided $p$-values
upper one-sided $p$-values
two-sided $p$-values
Monte Carlo standard errors of lower one-sided $p$-values
Monte Carlo standard errors of upper one-sided $p$-values

```
r(se_p_twosided)
r(ci_p_lower)
r(ci_p_upper)
r(ci_p_twosided)
```

Monte Carlo standard errors of two-sided $p$-values
Monte Carlo confidence intervals of lower one-sided $p$-values
Monte Carlo confidence intervals of upper one-sided $p$-values
Monte Carlo confidence intervals of two-sided $p$-values

## Methods and formulas

One-sided $p$-values are based on counts of the test statistic $T$ calculated for each permutation that are more extreme than the observed value $T_{\text {obs }}$. The lower one-sided $p$-value uses the count $c=\#\left\{T \leq T_{\text {obs }}\right\}$, and the upper one-sided $p$-value uses $c=\#\left\{T \geq T_{\text {obs }}\right\}$.

The counts from Monte Carlo permutations are assumed to have a binomial distribution. Standard errors and confidence intervals are computed using cii proportions $n c$, where $n$ is the number of permutations that yielded nonmissing results and $c$ is the count. The confidence intervals are exact binomial confidence intervals. See Methods and formulas in [R] ci.
permute calculates the two-sided $p$-value as $p=\min \left[1,2 \min \left(p_{\text {lower }}, p_{\text {upper }}\right)\right]$, where $p_{\text {lower }}$ is the lower one-sided $p$-value and $p_{\text {upper }}$ the upper one-sided $p$-value. Because the definition of the two-sided $p$-value does not yield a simple formula for the standard error or confidence interval for Monte Carlo permutations, the following ad hoc procedure is used. If $p_{\text {lower }}$ is the minimum one-sided $p$-value, its count $c_{\text {lower }}$ is doubled. If $p_{\text {upper }}$ is the minimum one-sided $p$-value, its count $c_{\text {upper }}$ is doubled. More precisely, the value $c_{2}=\min \left[n, 2 \min \left(c_{\text {lower }}, c_{\text {upper }}\right)\right]$ is used, and its distribution is assumed to be approximately binomial. Standard errors and confidence intervals are computed using cii proportions $n c_{2}$, wald. The confidence intervals produced are asymptotic binomial confidence intervals.

When enumerate is specified, the $p$-values have no error.
Suppose there are $N$ observations and the variable being permuted contains $K$ distinct values, each with multiplicity $n_{k}, k=1, \ldots, K$. The total number of distinct permutations is

$$
\frac{N!}{n_{1}!n_{2}!\cdots n_{K}!}
$$

This is the number of permutations computed when enumerate is specified.
When standardize is specified, instead of displaying the observed test statistic $T_{\text {obs }}$, the standardized statistic

$$
\frac{T_{\mathrm{obs}}-\operatorname{mean}(T)}{\sqrt{\operatorname{Var}(T)}}
$$

is displayed where mean $(T)$ and $\operatorname{Var}(T)$ are the mean and variance of the permutation distribution of $T$ :

$$
\begin{aligned}
\operatorname{mean}(T) & =\frac{1}{n} \sum_{i=1}^{n} T_{i} \\
\operatorname{Var}(T) & =\frac{1}{n} \sum_{i=1}^{n}\left\{T_{i}-\operatorname{mean}(T)\right\}^{2}
\end{aligned}
$$

$n$ is the number of permutations, and $T_{i}$ is the test statistic calculated for the $i$ th permutation.

## References

Ängquist, L. 2010. Stata tip 92: Manual implementation of permutations and bootstraps. Stata Journal 10: 686-688.
Gallis, J. A., F. Li, H. Yu, and E. L. Turner. 2018. cverand and cptest: Commands for efficient design and analysis of cluster randomized trials using constrained randomization and permutation tests. Stata Journal 18: 357-378.
Good, P. I. 2006. Resampling Methods: A Practical Guide to Data Analysis. 3rd ed. Boston: Birkhäuser.
Hayes, R. J., and L. H. Moulton. 2017. Cluster Randomised Trials. 2nd ed. Boca Raton, FL: CRC Press.
Kaiser, J. 2007. An exact and a Monte Carlo proposal to the Fisher-Pitman permutation tests for paired replicates and for independent samples. Stata Journal 7: 402-412.
Kaiser, J., and M. G. Lacy. 2009. A general-purpose method for two-group randomization tests. Stata Journal 9: 70-85.

## Also see

[R] bootstrap - Bootstrap sampling and estimation
[R] jackknife - Jackknife estimation
[R] simulate - Monte Carlo simulations
Description Remarks and examples References Also see

## Description

The term pk refers to pharmacokinetic data and the Stata commands, all of which begin with the letters pk , designed to do some of the analyses commonly performed in the pharmaceutical industry. The system is intended for the analysis of pharmacokinetic data, although some of the commands are for general use.

The pk commands are

| pkexamine | $[R]$ pkexamine | Calculate pharmacokinetic measures |
| :--- | :--- | :--- |
| pksumm | $[R]$ pksumm | Summarize pharmacokinetic data |
| pkshape | $[R]$ pkshape | Reshape (pharmacokinetic) Latin-square data |
| pkcross | $[R]$ pkcross | Analyze crossover experiments |
| pkequiv | $[R]$ pkequiv | Perform bioequivalence tests |
| pkcollapse | $[R]$ pkcollapse | Generate pharmacokinetic measurement dataset |

Also see [ME] menl for fitting pharmacokinetic models using nonlinear mixed-effects models; for instance, see example 15 in [ME] menl.

## Remarks and examples

Several types of clinical trials are commonly performed in the pharmaceutical industry. Examples include combination trials, multicenter trials, equivalence trials, and active control trials. For each type of trial, there is an optimal study design for estimating the effects of interest. The pk system can be used to analyze equivalence trials, which are usually conducted using a crossover design; however, it is possible to use a parallel design and still draw conclusions about equivalence.

Equivalence trials assess bioequivalence between two drugs. Although proving that two drugs behave the same is impossible, regulatory agencies believe that if the absorption properties of two drugs are similar, the two drugs will produce similar effects and have similar safety profiles. Generally, the goal of an equivalence trial is to assess the equivalence of a generic drug to an existing drug. This goal is commonly accomplished by comparing a confidence interval about the difference between a pharmacokinetic measurement of two drugs with an equivalence limit constructed from regulations. If the confidence interval is entirely within the equivalence limit, the drugs are declared bioequivalent. Another approach to assessing bioequivalence is to use the method of interval hypotheses testing. pkequiv is used to conduct these tests of bioequivalence.

Several pharmacokinetic measures can be used to ascertain how available a drug is for cellular absorption. The most common measure is the area under the concentration-time curve (AUC). Another common measure of drug availability is the maximum concentration ( $C_{\max }$ ) achieved by the drug during the follow-up period. Stata reports these and other less common measures of drug availability, including the time at which the maximum drug concentration was observed and the duration of the period during which the subject was being measured. Stata also reports the elimination rate, that is, the rate at which the drug is metabolized, and the drug's half-life, that is, the time it takes for the drug concentration to fall to one-half of its maximum concentration.
pkexamine computes and reports all the pharmacokinetic measures that Stata produces, including four calculations of the AUC. The standard AUC from 0 to the maximum observed time ( $\mathrm{AUC}_{0, t_{\max }}$ ) is computed using cubic splines or the trapezoidal rule. Additionally, pkexamine also computes the AUC from 0 to infinity by extending the standard concentration-time curve from the maximum observed time using three different methods. The first method simply extends the standard curve by using a least-squares linear fit through the last few data points. The second method extends the standard curve by fitting a decreasing exponential curve through the last few data points. The third method extends the curve by fitting a least-squares linear regression line on the log concentration. The mathematical details of these extensions are described in Methods and formulas of [R] pkexamine.

Data from an equivalence trial may also be analyzed using methods appropriate to the particular study design. When you have a crossover design, pkcross can be used to fit an appropriate ANOVA model. A crossover design is simply a restricted Latin square; therefore, pkcross can also be used to analyze any Latin-square design.

Some practical concerns arise when dealing with data from equivalence trials. Primarily, the data must be organized in a manner that Stata can use. The pk commands include pkcollapse and pkshape, which are designed to help transform data from a common format to one that is suitable for analysis with Stata.

In the following examples, we illustrate several different data formats that are often encountered in pharmaceutical research and describe how these formats can be transformed to formats that can be analyzed with Stata.
$>$ Example 1
Assume that we have one subject and are interested in determining the drug profile for that subject. A reasonable experiment would be to give the subject the drug and then measure the concentration of the drug in the subject's blood over a given period. For example, here is part of a dataset from Chow and Liu (2009, 13):

```
. use https://www.stata-press.com/data/r18/auc
(Primidone concentrations)
. list, abbrev(14)
```

| id | time | concentration |
| :---: | ---: | ---: |
| 1 | 0 | 0 |
| 1 | .5 | 0 |
| 1 | 1 | 2.8 |
| 1 | 1.5 | 4.4 |
| 1 | 2 | 4.4 |
| 1 | 3 | 4.7 |
| 1 | 4 | 4.1 |
| 1 | 6 | 4 |
| 1 | 8 | 3.6 |
| 1 | 12 | 3 |
| 1 | 16 | 2.5 |
| 1 | 24 | 2 |
| 1 | 32 | 1.6 |

Examining these data, we notice that the concentration quickly increases, plateaus for a short period, and then slowly decreases over time. pkexamine is used to calculate the pharmacokinetic measures of interest. pkexamine is explained in detail in [R] pkexamine. The output is

```
. pkexamine time concentration
```

$$
\begin{array}{rlr}
\text { Maximum concentration } & = & 4.7 \\
\text { Time of maximum concentration } & = & 3 \\
\text { Time of last observation (Tmax) } & = & 32 \\
\text { Elimination rate } & = & 0.0279 \\
\text { Half life } & =34.8503
\end{array}
$$

Area under the curve

| AUC [0, Tmax] | AUC [0, inf.) <br> Linear of log conc. | AUC [0, inf.) <br> Linear fit | AUC [0, inf.) <br> Exponential fit |
| :---: | :---: | :---: | :---: |
| 85.24 | 142.603 | 107.759 | 142.603 |

Fit based on last 3 points.

## Example 2

Clinical trials require that data be collected on more than one subject. There are several ways to enter raw measured data collected on several subjects. It would be reasonable to enter for each subject the drug concentration value at specific points in time. Such data could be

| id | conc1 | conc2 | conc3 | conc4 | conc5 | conc6 | conc7 |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| 1 | 0 | 1 | 4 | 7 | 5 | 3 | 1 |
| 2 | 0 | 2 | 6 | 5 | 4 | 3 | 2 |
| 3 | 0 | 1 | 2 | 3 | 5 | 4 | 1 |

where conc1 is the concentration at the first measured time, conc2 is the concentration at the second measured time, etc. This format requires that each drug concentration measurement be made at the same time on each subject. Another more flexible way to enter the data is to have an observation with three variables for each time measurement on a subject. Each observation would have a subject ID, the time at which the measurement was made, and the corresponding drug concentration at that time. The data would be as follows:

```
. use https://www.stata-press.com/data/r18/pkdata
(Fictional drug concentration data)
. list id conc1 time, sepby(id)
```

| id | conc1 | time |
| :---: | :---: | ---: |
| 1 | 0 | 0 |
| 1 | 3.073403 | .5 |
| 1 | 5.188444 | 1 |
| 1 | 5.898577 | 1.5 |
| 1 | 5.096378 | 2 |
| 1 | 6.094085 | 3 |
| 1 | 5.158772 | 4 |
| 1 | 5.7065 | 6 |
| 1 | 5.272467 | 8 |
| 1 | 4.4576 | 12 |
| 1 | 5.146423 | 16 |
| 1 | 4.947427 | 24 |
| 1 | 1.920421 | 32 |
| 2 |  | 0 |
| 2 | 2.48462 | .5 |
| 2 | 4.883569 | 1 |
| 2 | 7.253442 | 1.5 |
| 2 | 5.849345 | 2 |
| 2 | 6.761085 | 3 |
| 2 | 4.33839 | 4 |
| 2 | 5.04199 | 6 |
| 2 | 4.25128 | 8 |
| 2 | 6.205004 | 12 |
| 2 | 5.566165 | 16 |
| 2 | 3.689007 | 24 |
| 2 | 3.644063 | 32 |

Stata expects the data to be organized in the second form, as shown in pkdata.dta. If your data are organized as described in the first format, you will need to reshape the data to the second form; see [D] reshape. Because the data in the second (or long) format contain information for one drug on several subjects, pksumm can be used to produce summary statistics of the pharmacokinetic measurements. The output is

| Summary statistics for the pharmacokinetic measures |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | Number of | servations | $=16$ |
| Measure | Mean | Median | Variance | Skewness | Kurtosis | p-value |
| auc | 151.63 | 152.18 | 127.58 | -0.34 | 2.07 | 0.55 |
| aucline | 397.09 | 219.83 | 178276.59 | 2.69 | 9.61 | 0.00 |
| aucexp | 668.60 | 302.96 | 720356.98 | 2.67 | 9.54 | 0.00 |
| auclog | 665.95 | 298.03 | 752573.34 | 2.71 | 9.70 | 0.00 |
| half | 90.68 | 29.12 | 17750.70 | 2.36 | 7.92 | 0.00 |
| ke | 0.02 | 0.02 | 0.00 | 0.88 | 3.87 | 0.08 |
| cmax | 7.37 | 7.42 | 0.40 | -0.64 | 2.75 | 0.36 |
| tomc | 3.38 | 3.00 | 7.25 | 2.27 | 7.70 | 0.00 |
| tmax | 32.00 | 32.00 | 0.00 | . | . |  |

Until now, we have been concerned with the profile of only one drug. We have characterized the profile of that drug by individual subjects by using pkexamine and by a group of subjects by using pksumm. The goal of an equivalence trial, however, is to compare two drugs, which we will do in the rest of this example.

For equivalence trials, the study design most often used is the crossover design. For a complete discussion of crossover designs, see Ratkowsky, Evans, and Alldredge (1993).

In brief, crossover designs require that each subject be given both treatments at two different times. The order in which the treatments are applied changes between groups. For example, if we had 16 subjects numbered $1-16$, the first 8 would receive reference treatment " $R$ " during the first period of the study, and then they would be given test treatment " T ". The second 8 subjects would be given treatment " $T$ " during the first period of the study, and then they would be given treatment " $R$ ". Each subject in the study will have four variables that describe the observation: a subject identifier, a sequence identifier that indicates the order of treatment, and two outcome variables, one for each treatment. The outcome variables for each subject are the pharmacokinetic measures. The data must be transformed from a series of measurements on individual subjects to data containing the pharmacokinetic measures for each subject. In Stata parlance, this is referred to as a collapse, which can be done with pkcollapse; see [R] pkcollapse.

Here is part of our data:

|  | id | seq | time | conc1 | conc2 |
| :---: | :---: | :---: | :---: | :---: | :---: |
| 1. | 1 | 1 | 0 | 0 | 0 |
| 2. | 1 | 1 | . 5 | 3.073403 | 3.712592 |
| 3. | 1 | 1 | 1 | 5.188444 | 6.230602 |
| 4. | 1 | 1 | 1.5 | 5.898577 | 7.885944 |
| 5. | 1 | 1 | 2 | 5.096378 | 9.241735 |
| 6. | 1 | 1 | 3 | 6.094085 | 13.10507 |
| 7. | 1 | 1 | 4 | 5.158772 | . 169429 |
| 8. | 1 | 1 | 6 | 5.7065 | 8.759894 |
| 9. | 1 | 1 | 8 | 5.272467 | 7.985409 |
| 10. | 1 | 1 | 12 | 4.4576 | 7.740126 |
| 11. | 1 | 1 | 16 | 5.146423 | 7.607208 |
| 12. | 1 | 1 | 24 | 4.947427 | 7.588428 |
| 13. | 1 | 1 | 32 | 1.920421 | 2.791115 |
| 14. | 2 | 1 | 0 | 0 | 0 |
| 15. | 2 | 1 | . 5 | 2.48462 | . 9209593 |
| 16. | 2 | 1 | 1 | 4.883569 | 5.925818 |
| 17. | 2 | 1 | 1.5 | 7.253442 | 8.710549 |
| 18. | 2 | 1 | 2 | 5.849345 | 10.90552 |
| 19. | 2 | 1 | 3 | 6.761085 | 8.429898 |
| 20. | 2 | 1 | 4 | 4.33839 | 5.573152 |
| 21. | 2 | 1 | 6 | 5.04199 | 6.32341 |
| 22. | 2 | 1 | 8 | 4.25128 | . 5251224 |
| 23. | 2 | 1 | 12 | 6.205004 | 7.415988 |
| 24. | 2 | 1 | 16 | 5.566165 | 6.323938 |
| 25. | 2 | 1 | 24 | 3.689007 | 1.133553 |
| 26. | 2 | 1 | 32 | 3.644063 | 5.759489 |
| 27. | (output omitted) |  |  |  | 0 |
| 207. | 16 | 2 | 24 | 4.673281 | 6.059818 |
| 208. | 16 | 2 | 32 | 3.487347 | 5.213639 |

This format is similar to the second format described above, except that now we have measurements for two drugs at each time for each subject. We transform these data with pkcollapse:

```
. pkcollapse time conc1 conc2, id(id) keep(seq) stat(auc)
. list, sep(8) abbrev(10)
```

| id | seq | auc_conc1 | auc_conc2 |
| ---: | ---: | ---: | ---: |
| 1 | 1 | 150.9643 | 218.5551 |
| 2 | 1 | 146.7606 | 133.3201 |
| 3 | 1 | 160.6548 | 126.0635 |
| 4 | 1 | 157.8622 | 96.17461 |
| 5 | 1 | 133.6957 | 188.9038 |
| 6 | 1 | 160.639 | 223.6922 |
| 7 | 1 | 131.2604 | 104.0139 |
| 8 | 1 | 168.5186 | 237.8962 |
| 9 | 2 | 137.0627 | 139.7382 |
| 10 | 2 | 153.4038 | 202.3942 |
| 11 | 2 | 163.4593 | 136.7848 |
| 12 | 2 | 146.0462 | 104.5191 |
| 13 | 2 | 158.1457 | 165.8654 |
| 14 | 2 | 147.1977 | 139.235 |
| 15 | 2 | 164.9988 | 166.2391 |
| 16 | 2 | 145.3823 | 158.5146 |

For this example, we chose to use the $\mathrm{AUC}_{0, t_{\max }}$ for two drugs as our pharmacokinetic measure. We could have used any of the measures computed by pkexamine. In addition to the AUCs, the dataset also contains a sequence variable for each subject indicating when each treatment was administered.

The data produced by pkcollapse are in what Stata calls wide format; that is, there is one observation per subject containing two or more outcomes. To use pkcross and pkequiv, we need to transform these data to long format, which we can do using pkshape; see [R] pkshape.

Consider the first subject in the dataset. This subject is in sequence 1 , which means that treatment " $R$ " was applied during the first period of the study and treatment " T " was applied in the second period of the study. We need to split the first observation into two observations so that the outcome measure is only in one variable. We also need two new variables, one indicating the treatment the subject received and another recording the period of the study when the subject received that treatment. We might expect the expansion of the first subject to be

| id | sequence | auc | treat | period |
| ---: | ---: | ---: | ---: | ---: |
| 1 | 1 | 150.9643 | $R$ | 1 |
| 1 | 1 | 218.5551 | $T$ | 2 |

We see that subject number 1 was in sequence 1 , had an $\mathrm{AUC}_{0, t_{\max }}$ of 150.9643 when treatment " R ", was applied in the first period of the study, and had an $\mathrm{AUC}_{0, t_{\max }}$ of 218.5551 when treatment " T " was applied in the second period.

Similarly, the expansion of subject 9 (the first subject in sequence 2 ) would be

| id | sequence | auc | treat | period |
| ---: | ---: | ---: | ---: | ---: |
| 9 | 2 | 137.0627 | $T$ | 1 |
| 9 | 2 | 139.7382 | R | 2 |

Here treatment "T" was applied to the subject during the first period of the study, and treatment "R" was applied to the subject during the second period of the study.

An additional complication is common in crossover study designs. The treatment applied in the first period of the study might still have some effect on the outcome in the second period. In this example,
each subject was given one treatment followed by another treatment. To get accurate estimates of treatment effects, it is necessary to account for the carryover effect, the effect that the first treatment has in the second period of the study. We must, therefore, have a variable that indicates which treatment was applied in the first treatment period. pkshape creates a variable that indicates the carryover effect. For treatments applied during the first treatment period, there will never be a carryover effect. The sequence, treatment, and carryover variables all receive value labels. Thus, the expanded data created by pkshape for subject 1 will be

| id | sequence | outcome | treat | period | carry |
| ---: | ---: | ---: | ---: | ---: | ---: |
| 1 | RT | 150.9643 | R | 1 | 0 |
| 1 | RT | 218.5551 | T | 2 | R |

and the expanded data for subject 9 will be

| id | sequence | outcome | treat | period | carry |
| ---: | ---: | ---: | ---: | ---: | ---: |
| 9 | TR | 137.0627 | $T$ | 1 | 0 |
| 9 | TR | 139.7382 | $R$ | 2 | T |

We pkshape the data:

```
. pkshape id seq auc*, order(RT TR)
. sort id sequence period
. list, sep(16)
```

|  | id | sequence | outcome | treat | carry | period |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 1. | 1 | RT | 150.9643 | R | 0 | 1 |
| 2. | 1 | RT | 218.5551 | T | R | 2 |
| 3. | 2 | RT | 146.7606 | R | 0 | 1 |
| 4. | 2 | RT | 133.3201 | T | R | 2 |
| 5. | 3 | RT | 160.6548 | R | 0 | 1 |
| 6. | 3 | RT | 126.0635 | T | R | 2 |
| 7. | 4 | RT | 157.8622 | R | 0 | 1 |
| 8. | 4 | RT | 96.17461 | T | R | 2 |
| 9. | 5 | RT | 133.6957 | R | 0 | 1 |
| 10. | 5 | RT | 188.9038 | T | R | 2 |
| 11. | 6 | RT | 160.639 | R | 0 | 1 |
| 12. | 6 | RT | 223.6922 | T | R | 2 |
| 13. | 7 | RT | 131.2604 | R | 0 | 1 |
| 14. | 7 | RT | 104.0139 | T | R | 2 |
| 15. | 8 | RT | 168.5186 | R | 0 | 1 |
| 16. | 8 | RT | 237.8962 | T | R | 2 |
| 17. | 9 | TR | 137.0627 | T | 0 | 1 |
| 18. | 9 | TR | 139.7382 | R | T | 2 |
| 19. | 10 | TR | 153.4038 | T | 0 | 1 |
| 20. | 10 | TR | 202.3942 | R | T | 2 |
| 21. | 11 | TR | 163.4593 | T | 0 | 1 |
| 22. | 11 | TR | 136.7848 | R | T | 2 |
| 23. | 12 | TR | 146.0462 | T | 0 | 1 |
| 24. | 12 | TR | 104.5191 | R | T | 2 |
| 25. | 13 | TR | 158.1457 | T | 0 | 1 |
| 26. | 13 | TR | 165.8654 | R | T | 2 |
| 27. | 14 | TR | 147.1977 | T | 0 | 1 |
| 28. | 14 | TR | 139.235 | R | T | 2 |
| 29. | 15 | TR | 164.9988 | T | 0 | 1 |
| 30. | 15 | TR | 166.2391 | R | T | 2 |
| 31. | 16 | TR | 145.3823 | T | 0 | 1 |
| 32. | 16 | TR | 158.5146 | R | T | 2 |

Crossover designs do not require that each subject receive each treatment, but if they do, the crossover design is referred to as a complete crossover design.

The dataset (pkdata.dta) in this example is organized in a manner that can be analyzed with Stata. To fit an ANOVA model to these data, we can use anova or pkcross. To conduct equivalence tests, we can use pkequiv.

## References

Chow, S.-C., and J.-P. Liu. 2009. Design and Analysis of Bioavailability and Bioequivalence Studies. 3rd ed. Boca Raton, FL: Chapman and Hall/CRC.

Ratkowsky, D. A., M. A. Evans, and J. R. Alldredge. 1993. Cross-over Experiments: Design, Analysis, and Application. New York: Dekker.

## Also see

[ME] menl - Nonlinear mixed-effects regression

## Title

pkcollapse - Generate pharmacokinetic measurement dataset

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Methods and formulas | Also see |

## Description

pkcollapse generates new variables with the pharmacokinetic summary measures of interest. pkcollapse is one of the pk commands. Please read [R] pk before reading this entry.

## Quick start

Single concentration, v1, measured over time, tvar, for patients identified by idvar pkcollapse tvar v1, id(idvar)

Same as above, but add additional drug concentration data stored in v2
pkcollapse tvar v1 v2, id(idvar)
Same as above, but use trapezoidal rule for calculating area under the concentration-time curve $\left(\mathrm{AUC}_{0, t_{\max }}\right)$
pkcollapse tvar v1 v2, id(idvar) trapezoid
Same as above, and increase the number of data points used to estimate $\mathrm{AUC}_{0, \infty}$ to 10
pkcollapse tvar v1 v2, id(idvar) trapezoid fit(10)
Retain variables v3 and v4 when collapsing dataset
pkcollapse tvar v1 v2, id(idvar) keep(v3 v4)

## Menu

Statistics $>$ Epidemiology and related $>$ Other $>$ Generate pharmacokinetic measurement dataset

## Syntax

pkcollapse time concentration $[$ concentration $[\ldots]][i f]$, id(id_var) [options]
options Description
Main
*id(id_var) subject ID variable
stat(measures) create specified measures; default is all
trapezoid
use trapezoidal rule; default is cubic splines
fit(\#) use \# points to estimate $\mathrm{AUC}_{0, \infty}$; default is fit(3)
keep (varlist) keep variables in varlist
force force collapse
nodots suppress dots during calculation
*id(id_var) is required.
measures Description
auc $\quad \mathrm{AUC}_{0, t_{\max }}$
aucline
aucexp
auclog
half half-life of the drug
ke
cmax
tmax
tomc
$\operatorname{AUC}_{0, \infty}$ using a linear extension
$\mathrm{AUC}_{0, \infty}$ using an exponential extension
area under the concentration-time curve from 0 to $\infty$ extended with a linear fit to log concentration
elimination rate
maximum concentration
time at last concentration
time of maximum concentration

## Options

$\qquad$
id (id_var) is required and specifies the variable that contains the subject ID over which pkcollapse is to operate.
stat(measures) specifies the measures to be generated. The default is to generate all the measures. trapezoid tells Stata to use the trapezoidal rule when calculating the $\mathrm{AUC}_{0, t_{\text {max }}}$. The default is to use cubic splines, which give better results for most functions. When the curve is irregular, trapezoid may give better results.
fit(\#) specifies the number of points to use in estimating the $\operatorname{AUC}_{0, \infty}$. The default is $f i t(3)$, the last three points. This number should be viewed as a minimum; the appropriate number of points will depend on your data.
keep (varlist) specifies the variables to be kept during the collapse. Variables not specified with the keep () option will be dropped. When keep () is specified, the kept variables are checked to ensure that all values of the variables are the same within id_var.
force forces the collapse, even when values of the keep() variables differ within id_var.
nodots suppresses the display of dots during calculation.

## Remarks and examples

pkcollapse generates all the summary pharmacokinetic measures.

## > Example 1

We demonstrate the use of pkcollapse with pkdata.dta described in example 2 of $[\mathrm{R}] \mathbf{p k}$. We have drug concentration data on 16 subjects. Each subject is measured at 13 time points over a 32-hour period. Some of the records are as follows:

```
. use https://www.stata-press.com/data/r18/pkdata
(Fictional drug concentration data)
. list, sep(0)
```

| id | seq | time | conc1 | conc2 |
| ---: | ---: | ---: | ---: | ---: |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | .5 | 3.073403 | 3.712592 |
| 1 | 1 | 1 | 5.188444 | 6.230602 |
| 1 | 1 | 1.5 | 5.898577 | 7.885944 |
| 1 | 1 | 2 | 5.096378 | 9.241735 |
| 1 | 1 | 3 | 6.094085 | 13.10507 |


| 14. | 2 | 1 | 0 | 0 | 0 |
| :--- | ---: | ---: | ---: | ---: | ---: |
| 15. | 2 | 1 | .5 | 2.48462 | .9209593 |
| 16. | 2 | 1 | 1 | 4.883569 | 5.925818 |
| 17. | 2 | 1 | 1.5 | 7.253442 | 8.710549 |
| 18. | 2 | 1 | 2 | 5.849345 | 10.90552 |
| 19. | 2 | 1 | 3 | 6.761085 | 8.429898 |

Although pksumm allows us to view all the pharmacokinetic measures, we can create a dataset with the measures by using pkcollapse.

```
. pkcollapse time conc1 conc2, id(id) stat(auc) keep(seq)
. list, sep(8) abbrev(10)
```

|  | id |  | auc_conc1 | auc_conc2 |
| :---: | :---: | :---: | :---: | :---: |
| 1. | 1 | 1 | 150.9643 | 218.5551 |
| 2. | 2 | 1 | 146.7606 | 133.3201 |
| 3. | 3 | 1 | 160.6548 | 126.0635 |
| 4. | 4 | 1 | 157.8622 | 96.17461 |
| 5. | 5 | 1 | 133.6957 | 188.9038 |
| 6. | 6 | 1 | 160.639 | 223.6922 |
| 7. | 7 | 1 | 131.2604 | 104.0139 |
| 8. | 8 | 1 | 168.5186 | 237.8962 |
| 9. | 9 | 2 | 137.0627 | 139.7382 |
| 10. | 10 | 2 | 153.4038 | 202.3942 |
| 11. | 11 | 2 | 163.4593 | 136.7848 |
| 12. | 12 | 2 | 146.0462 | 104.5191 |
| 13. | 13 | 2 | 158.1457 | 165.8654 |
| 14. | 14 | 2 | 147.1977 | 139.235 |
| 15. | 15 | 2 | 164.9988 | 166.2391 |
| 16. | 16 | 2 | 145.3823 | 158.5146 |

The resulting dataset contains one observation per subject and is in wide format. If we want to use pkcross or pkequiv, we must transform these data to long format with the pkshape command, which we do in example 2 of [R] pk.

## Methods and formulas

The statistics generated by pkcollapse are described in [R] pkexamine.

## Also see

[R] pk - Pharmacokinetic (biopharmaceutical) data

## Title

pkcross - Analyze crossover experiments

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Methods and formulas | References |
| Also see |  |  |  |

## Description

pkcross analyzes data from a crossover design experiment. When analyzing pharmaceutical trial data, if the treatment, carryover, and sequence variables are known, the omnibus test for separability of the treatment and carryover effects is calculated.
pkcross is one of the pk commands. Please read [R] pk before reading this entry.

## Quick start

For pharmacokinetic outcome y measured for subjects identified by idvar given treatments treat in sequences identified by seq in periods period with potential carryover effects from previous treatment carry:

Sequence, treatment, and period effects for a $2 \times 2$ design
pkcross y, param(3) id(idvar) sequence(seq) treatment(treat) /// period(period)

Same as above, but estimate the carryover effect instead of the sequence effect
pkcross y, param(1) id(idvar) treatment(treat) period(period) ///
carryover (carry)
Only estimate sequence, treatment, and period effects in higher-order designs
pkcross y, id(idvar) sequence(seq) treatment (treat) carryover(none)
Also estimate carryover effect and omnibus measure of separability of treatment and carryover effects

```
pkcross y, model(seq / idvar|seq treat carry period) ///
    treatment(treat) carryover(carry) sequence(seq)
```


## Menu

Statistics $>$ Epidemiology and related $>$ Other $>$ Analyze crossover experiments

## Syntax

pkcross outcome [if] [in] [, options]
options
Model
sequence (varname)
treatment (varname)
period(varname)
id(varname)
carryover (varname)
carryover (none)
model (string)
sequential
Parameterization
param (3)
param(1)
param(2)
param(4)
sequence variable; default is sequence (sequence)
treatment variable; default is treatment (treat)
period variable; default is period(period)
ID variable; default is id(id)
carryover variable; default is carryover (carry)
omit carryover effects from model; default is carryover (carry)
specify the model to fit
estimate sequential instead of partial sums of squares
estimate mean and the period, treatment, and sequence effects; assume no carryover effects exist; the default
estimate mean and the period, treatment, and carryover effects; assume no sequence effects exist
estimate mean, period and treatment effects, and period-by-treatment interaction; assume no sequence or carryover effects exist
estimate mean, sequence and treatment effects, and sequence-by-treatment interaction; assume no period or carryover effects exist

## Options

Model
sequence (varname) specifies the variable that contains the sequence in which the treatment was administered. If this option is not specified, sequence (sequence) is assumed.
treatment (varname) specifies the variable that contains the treatment information. If this option is not specified, treatment (treat) is assumed.
period (varname) specifies the variable that contains the period information. If this option is not specified, period(period) is assumed.
id (varname) specifies the variable that contains the subject identifiers. If this option is not specified, id(id) is assumed.
carryover (varname ${ }^{\text {none) }}$ specifies the variable that contains the carryover information. If carry (none) is specified, the carryover effects are omitted from the model. If this option is not specified, carryover (carry) is assumed.
model (string) specifies the model to be fit. For higher-order crossover designs, this option can be useful if you want to fit a model other than the default. However, anova (see [R] anova) can also be used to fit a crossover model. The default model for higher-order crossover designs is outcome predicted by sequence, period, treatment, and carryover effects. By default, the model statement is model (sequence period treat carry).
sequential specifies that sequential sums of squares be estimated.

Parameterization
param(\#) specifies which of the four parameterizations to use for the analysis of a $2 \times 2$ crossover experiment. This option is ignored with higher-order crossover designs. The default is param (3). See the technical note in this entry for $2 \times 2$ crossover designs for more details.
param(3) estimates the overall mean, the period effects, the treatment effects, and the sequence effects, assuming that no carryover effects exist. This is the default parameterization.
param(1) estimates the overall mean, the period effects, the treatment effects, and the carryover effects, assuming that no sequence effects exist.
param(2) estimates the overall mean, the period effects, the treatment effects, and the period-bytreatment interaction, assuming that no sequence or carryover effects exist.
param (4) estimates the overall mean, the sequence effects, the treatment effects, and the sequence-by-treatment interaction, assuming that no period or carryover effects exist. When the sequence-by-treatment interaction is equivalent to the period effect, this reduces to the third parameterization.

## Remarks and examples

pkcross is designed to analyze crossover experiments. Use pkshape first to reshape your data; see [R] pkshape. pkcross assumes that the data were reshaped by pkshape or are organized in the same manner as produced with pkshape. Washout periods are indicated by the number 0 . See the technical note below for more information on analyzing $2 \times 2$ crossover experiments.

## - Technical note

The $2 \times 2$ crossover design cannot be used to estimate more than four parameters because there are only four pieces of information (the four cell means) collected. pkcross uses ANOVA models to analyze the data, so one of the four parameters must be the overall mean of the model, leaving just 3 degrees of freedom to estimate the remaining effects (period, sequence, treatment, and carryover). Thus, the model is overparameterized. Estimation of treatment and carryover effects requires the assumption of either no period effects or no sequence effects. Some researchers maintain that estimating carryover effects at the expense of other effects is a bad idea. This is a limitation of this design. pkcross implements four parameterizations for this model. They are numbered sequentially from one to four and are described in Options.
$>$ Example 1
Consider the example data published in Chow and Liu (2009, 71). There are 24 patients, 12 in each sequence. Sequence 1 is the reference formulation followed by the test formulation; sequence 2 is the test formulation followed by the reference formulation. After reshaping the data with pkshape, we have variables that identify the subjects, periods, treatments, sequence, and carryover treatment. The outcome variable contains the reported $\mathrm{AUC}_{0, t_{\text {max }}}$. To compute the ANOVA table, use pkcross:
. use https://www.stata-press.com/data/r18/chowliu
. pkshape id seq period1 period2, order (RT TR)

- pkcross outcome


There is evidence of intersubject variability, but there are no other significant effects. The omnibus test for separability is a measure reflecting the degree to which the study design allows the treatment effects to be estimated independently of the carryover effects. The measure of separability of the treatment and carryover effects indicates approximately $29 \%$ separability, which can be interpreted as the degree to which the treatment and carryover effects are orthogonal. This is a characteristic of the design of the study. For a complete discussion, see Ratkowsky, Evans, and Alldredge (1993). Compared with the output in Chow and Liu (2009), the sequence effect is mislabeled as a carryover effect. See Ratkowsky, Evans, and Alldredge (1993, sec. 3.2) for a complete discussion of the mislabeling.

By specifying param(1), we obtain parameterization 1 for this model.

| . pkcross outcome, param(1) |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  | Sequenc Perio <br> Treatmen Carryove | $\begin{aligned} & \text { riabl } \\ & \text { riabl } \\ & \text { riabl } \\ & \text { riabl } \\ & \text { riabl } \end{aligned}$ | sequence <br> period <br> treat <br> carry <br> id |
| Analysis of variance (ANOVA) for a 2 x 2 crossover study |  |  |  |  |  |
| Source of variation | Partial SS | df | MS | F | Prob > F |
| Treatment effect | 301.04 | 1 | 301.04 | 0.67 | 0.4189 |
| Period effect | 255.62 | 1 | 255.62 | 0.57 | 0.4561 |
| Carryover effect | 276.00 | 1 | 276.00 | 0.61 | 0.4388 |
| Residuals | 19890.92 | 44 | 452.07 |  |  |
| Total | 20265.68 | 47 |  |  |  |
| Omnibus measure of separability of treatment and carryover $=29.2893 \%$ |  |  |  |  |  |

## $>$ Example 2

Consider the case of a two-treatment, four-sequence, two-period crossover design. This design is commonly referred to as Balaam's design (Balaam 1968). Ratkowsky, Evans, and Alldredge (1993, 140) published the following data from an amantadine trial, originally published by Taka and Armitage (1983):
. use https://www.stata-press.com/data/r18/balaam, clear
. list, sep(0)

| id | seq | period1 | period2 | period3 |
| ---: | ---: | ---: | ---: | ---: |
| 1 | $-a b$ | 9 | 8.75 | 8.75 |
| 2 | $-a b$ | 12 | 10.5 | 9.75 |
| 3 | $-a b$ | 17 | 15 | 18.5 |
| 4 | $-a b$ | 21 | 21 | 21.5 |
| 1 | -ba | 23 | 22 | 18 |
| 2 | -ba | 15 | 15 | 13 |
| 3 | -ba | 13 | 14 | 13.75 |
| 4 | -ba | 24 | 22.75 | 21.5 |
| 5 | -ba | 18 | 17.75 | 16.75 |
| 1 | -aa | 14 | 12.5 | 14 |
| 2 | -aa | 27 | 24.25 | 22.5 |
| 3 | -aa | 19 | 17.25 | 16.25 |
| 4 | -aa | 30 | 28.25 | 29.75 |
| 1 | -bb | 21 | 20 | 19.51 |
| 2 | -bb | 11 | 10.5 | 10 |
| 3 | -bb | 20 | 19.5 | 20.75 |
| 4 | -bb | 25 | 22.5 | 23.5 |

The sequence identifier must be a string with 0s to indicate washout or baseline periods, or a number. If the sequence identifier is numeric, the order () option must be specified with pkshape. If the sequence identifier is a string, pkshape will use the string values to create sequence, period, and treatment variables. In this example, the dash is used to indicate a baseline period, which is an invalid code for this purpose. Therefore, we use the subinstr () function to replace each dash with a 0 . After doing so, we can use pkshape to format the data in a way that can be used with pkcross. We leave most pkcross options at their defaults, but we specify the sequential option to calculate sequential sums of squares instead of the default partial sums of squares.

```
. replace seq = subinstr(seq, "-", "0", .)
(17 real changes made)
. pkshape id seq period1 period2 period3
. pkcross outcome, sequential
\[
\begin{aligned}
\text { Sequence variable } & =\text { sequence } \\
\text { Period variable } & =\text { period } \\
\text { Treatment variable } & =\text { treat } \\
\text { Carryover variable } & =\text { carry } \\
\text { ID variable } & =\text { id }
\end{aligned}
\]
\begin{tabular}{r|rrrrrr} 
Analysis of variance (ANOVA) for a crossover study \\
Source of variation & SS & df & MS & F & Prob > F \\
\hline Intersubjects & & & & & \\
Sequence effect & 285.82 & 3 & 95.27 & 1.01 & 0.4180 \\
Residuals & 1221.49 & 13 & 93.96 & 59.96 & 0.0000 \\
\hline Intrasubjects & & & & & \\
Period effect & 15.13 & 2 & 7.56 & 6.34 & 0.0048 \\
Treatment effect & 8.48 & 1 & 8.48 & 8.86 & 0.0056 \\
Carryover effect & 0.11 & 1 & 0.11 & 0.12 & 0.7366 \\
Residuals & 29.56 & 30 & 0.99 & & \\
\hline Total & 1560.59 & 50 & & &
\end{tabular}
Omnibus measure of separability of treatment and carryover \(=64.6447 \%\)
```


## - Example 3

For this example, we return to pkdata.dta from example 2 of $[R] \mathbf{p k}$ and use pkcollapse and pkshape on the data as discussed in that example.

```
. use https://www.stata-press.com/data/r18/pkdata, clear
(Fictional drug concentration data)
. pkcollapse time conc1 conc2, id(id) keep(seq) stat(auc)
. pkshape id seq auc*, order(RT TR)
```

After sorting the data with sort, our data appear as follows:

|  | id | sequence | outcome | treat | carry | period |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 1. | 1 | RT | 150.9643 | R | 0 | 1 |
| 2. | 2 | RT | 146.7606 | R | 0 | 1 |
| 3. | 3 | RT | 160.6548 | R | 0 | 1 |
| 4. | 4 | RT | 157.8622 | R | 0 | 1 |
| 5. | 5 | RT | 133.6957 | R | 0 | 1 |
| 6. | 6 | RT | 160.639 | R | 0 | 1 |
| 7. | 7 | RT | 131.2604 | R | 0 | 1 |
| 8. | 8 | RT | 168.5186 | R | 0 | 1 |
| 9. | 9 | TR | 137.0627 | T | 0 | 1 |
| 10. | 10 | TR | 153.4038 | T | 0 | 1 |
| 11. | 11 | TR | 163.4593 | T | 0 | 1 |
| 12. | 12 | TR | 146.0462 | T | 0 | 1 |
| 13. | 13 | TR | 158.1457 | T | 0 | 1 |
| 14. | 14 | TR | 147.1977 | T | 0 | 1 |
| 15. | 15 | TR | 164.9988 | T | 0 | 1 |
| 16. | 16 | TR | 145.3823 | T | 0 | 1 |
| 17. | 1 | RT | 218.5551 | T | R | 2 |
| 18. | 2 | RT | 133.3201 | T | R | 2 |
| 19. | 3 | RT | 126.0635 | T | R | 2 |
| 20. | 4 | RT | 96.17461 | T | R | 2 |
| 21. | 5 | RT | 188.9038 | T | R | 2 |
| 22. | 6 | RT | 223.6922 | T | R | 2 |
| 23. | 7 | RT | 104.0139 | T | R | 2 |
| 24. | 8 | RT | 237.8962 | T | R | 2 |
| 25. | 9 | TR | 139.7382 | R | T | 2 |
| 26. | 10 | TR | 202.3942 | R | T | 2 |
| 27. | 11 | TR | 136.7848 | R | T | 2 |
| 28. | 12 | TR | 104.5191 | R | T | 2 |
| 29. | 13 | TR | 165.8654 | R | T | 2 |
| 30. | 14 | TR | 139.235 | R | T | 2 |
| 31. | 15 | TR | 166.2391 | R | T | 2 |
| 32. | 16 | TR | 158.5146 | R | T | 2 |

We now fit an ANOVA model using pkcross:


## Example 4

Consider the case of a six-treatment crossover trial in which the squares are not variance balanced. The following dataset is from a partially balanced crossover trial published by Patterson and Lucas (1962) and reproduced in Ratkowsky, Evans, and Alldredge (1993, 231):
. use https://www.stata-press.com/data/r18/nobalance, clear
. list, $\operatorname{sep}(4)$

|  | cow | seq | period1 | period2 | period3 | period4 | block |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 1. | 1 | adbe | 38.7 | 37.4 | 34.3 | 31.3 | 1 |
| 2. | 2 | baed | 48.9 | 46.9 | 42 | 39.6 | 1 |
| 3. | 3 | ebda | 34.6 | 32.3 | 28.5 | 27.1 | 1 |
| 4. | 4 | deab | 35.2 | 33.5 | 28.4 | 25.1 | 1 |
| 5. | 1 | dafc | 32.9 | 33.1 | 27.5 | 25.1 | 2 |
| 6. | 2 | fdca | 30.4 | 29.5 | 26.7 | 23.1 | 2 |
| 7. | 3 | cfad | 30.8 | 29.3 | 26.4 | 23.2 | 2 |
| 8. | 4 | acdf | 25.7 | 26.1 | 23.4 | 18.7 | 2 |
| 9. | 1 | efbc | 25.4 | 26 | 23.9 | 19.9 | 3 |
| 10. | 2 | becf | 21.8 | 23.9 | 21.7 | 17.6 | 3 |
| 11. | 3 | fceb | 21.4 | 22 | 19.4 | 16.6 | 3 |
| 12. | 4 | cbfe | 22.8 | 21 | 18.6 | 16.1 | 3 |

When there is no variance balance in the design, a square or blocking variable is needed to indicate in which treatment cell a sequence was observed, but the mechanical steps are the same.

```
. pkshape cow seq period1 period2 period3 period4
. pkcross outcome, model(block cow|block period|block treat carry) sequential
\begin{tabular}{|c|c|c|c|c|c|}
\hline \multirow[b]{2}{*}{Source} & \multicolumn{3}{|l|}{\[
\begin{array}{llr}
\text { Number of obs } & = & 48 \\
\text { Root MSE } & = & .740408
\end{array}
\]} & R -squared & \[
\begin{aligned}
& =0.9965 \\
& =0.9903
\end{aligned}
\] \\
\hline & Seq. SS & df & MS & F & Prob > F \\
\hline Model & 2650.1331 & 30 & 88.3377701 & 161.14 & 0.0000 \\
\hline block & 1607.01128 & 2 & 803.505642 & 1465.71 & 0.0000 \\
\hline cow|block & 628.706274 & 9 & 69.8562527 & 127.43 & 0.0000 \\
\hline period|block & 408.031253 & 9 & 45.3368059 & 82.70 & 0.0000 \\
\hline treat & 2.50000057 & 5 & . 500000114 & 0.91 & 0.4964 \\
\hline carry & 3.88428906 & 5 & . 776857812 & 1.42 & 0.2680 \\
\hline Residual & 9.31945887 & 17 & . 548203463 & & \\
\hline Total & 2659.45256 & 47 & 56.584097 & & \\
\hline
\end{tabular}
```

When the model statement is used and the omnibus measure of separability is desired, specify the variables in the treatment(), carryover(), and sequence() options to pkcross.

## Methods and formulas

pkcross uses ANOVA to fit models for crossover experiments; see [R] anova.
The omnibus measure of separability is

$$
S=100(1-V) \%
$$

where $V$ is Cramér's $V$ and is defined as

$$
V=\left\{\frac{\frac{\chi^{2}}{N}}{\min (r-1, c-1)}\right\}^{\frac{1}{2}}
$$

$N$ is the sample size and $\chi^{2}$ is calculated as

$$
\chi^{2}=\sum_{i} \sum_{j}\left\{\frac{\left(O_{i j}-E_{i j}\right)^{2}}{E_{i j}}\right\}
$$

where $O$ and $E$ are the observed and expected counts in an $r \times c$ table of the number of times each treatment is followed by the other treatments.

## References

Balaam, L. N. 1968. A two-period design with $t^{2}$ experimental units. Biometrics 24: 61-73. https://doi.org/10.2307/2528460.
Chow, S.-C., and J.-P. Liu. 2009. Design and Analysis of Bioavailability and Bioequivalence Studies. 3rd ed. Boca Raton, FL: Chapman and Hall/CRC.

Kutner, M. H., C. J. Nachtsheim, J. Neter, and W. Li. 2005. Applied Linear Statistical Models. 5th ed. New York: McGraw-Hill/Irwin.

Patterson, H. D., and H. L. Lucas. 1962. Change-over designs. Technical Bulletin 147, North Carolina Agricultural Experiment Station and the USDA.
Ratkowsky, D. A., M. A. Evans, and J. R. Alldredge. 1993. Cross-over Experiments: Design, Analysis, and Application. New York: Dekker.

Taka, M. T., and P. Armitage. 1983. Autoregressive models in clinical trials. Communications in Statistics-Theory and Methods 12: 865-876. https://doi.org/10.1080/03610928308828502.

## Also see

[R] pk - Pharmacokinetic (biopharmaceutical) data

## Title

pkequiv - Perform bioequivalence tests

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

pkequiv performs bioequivalence testing for two treatments. By default, pkequiv calculates a standard confidence interval (CI) symmetric about the difference between the two treatment means. pkequiv also calculates CIs symmetric about 0 and intervals based on Fieller's theorem. Also, pkequiv can perform interval hypothesis tests for bioequivalence.
pkequiv is one of the pk commands. Please read [R] pk before reading this entry.

## Quick start

Classic CI for difference in pharmacokinetic outcome y1 between treatments v1 given over period v2 in sequence v3 with subjects identified by idvar
pkequiv y1 v1 v2 v3 idvar
Same as above, but calculate an exact CI by Fieller's theorem
pkequiv y1 v1 v2 v3 idvar, fieller
Schuirmann's two one-sided tests for bioequivalence
pkequiv y1 v1 v2 v3 idvar, tost
Specify the two treatments, 2 and 3, that are to be tested for equivalence pkequiv y1 v1 v2 v3 idvar, compare (2 3)

## Menu

Statistics $>$ Epidemiology and related $>$ Other $>$ Bioequivalence tests

## Syntax

pkequiv outcome treatment period sequence id [if] [in] [, options] options Description

Options
compare (string)
limit(\#)
level(\#)
fieller
symmetric
anderson
tost
noboot
compare the two specified values of the treatment variable
equivalence limit (between 0.01 and 0.99 ); default is limit(0.2)
set confidence level; default is level (90)
calculate CI by Fieller's theorem
calculate symmetric CI
Anderson and Hauck hypothesis test for bioequivalence
two one-sided hypothesis tests for bioequivalence
do not estimate probability that CI lies within confidence limits
collect is allowed; see [U] 11.1.10 Prefix commands.

## Options

$\rightleftharpoons$ Options
compare (string) specifies the two treatments to be tested for equivalence. Sometimes there may be more than two treatments, but the equivalence can be determined only between any two treatments.
limit(\#) specifies the equivalence limit. The default is limit ( 0.2 ). The equivalence limit can be changed only symmetrically; for example, it is not possible to have a 0.15 lower limit and a 0.2 upper limit in the same test.
level(\#) specifies the confidence level, as a percentage, for CIs. The default is level(90). This setting is not controlled by the set level command.
fieller specifies that a CI based on Fieller's theorem be calculated.
symmetric specifies that a symmetric CI be calculated.
anderson specifies that the Anderson and Hauck (1983) hypothesis test for bioequivalence be computed. This option is ignored when calculating a CI based on Fieller's theorem or when calculating a CI that is symmetric about 0 .
tost specifies that the two one-sided hypothesis tests for bioequivalence be computed. This option is ignored when calculating a CI based on Fieller's theorem or when calculating a CI that is symmetric about 0 .
noboot prevents the estimation of the probability that the CI lies within the equivalence limits. If this option is not specified, this probability is estimated by resampling the data.

## Remarks and examples

pkequiv is designed to conduct tests for bioequivalence based on data from a crossover experiment. pkequiv requires that the user specify the outcome, treatment, period, sequence, and id variables. The data must be in the same format as that produced by pkshape; see $[R]$ pkshape.

## Example 1

We use pkdata.dta from example 2 of [R] pk. We use pkcollapse and pkshape on the data as discussed in that example. After sorting the data with sort, our data appear as follows:

```
. use https://www.stata-press.com/data/r18/pkdata, clear
(Fictional drug concentration data)
. pkcollapse time conc1 conc2, id(id) keep(seq) stat(auc)
. pkshape id seq auc*, order(RT TR)
. sort period id
. list, sep(4)
```

|  | id | sequence | outcome | treat | carry | period |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 1. | 1 | RT | 150.9643 | R | 0 | 1 |
| 2. | 2 | RT | 146.7606 | R | 0 | 1 |
| 3. | 3 | RT | 160.6548 | R | 0 | 1 |
| 4. | 4 | RT | 157.8622 | R | 0 | 1 |
| 5. | 5 | RT | 133.6957 | R | 0 | 1 |
| 6. | 6 | RT | 160.639 | R | 0 | 1 |
| 7. | 7 | RT | 131.2604 | R | 0 | 1 |
| 8. | 8 | RT | 168.5186 | R | 0 | 1 |
| 9. | 9 | TR | 137.0627 | T | 0 | 1 |
| 10. | 10 | TR | 153.4038 | T | 0 | 1 |
| 11. | 11 | TR | 163.4593 | T | 0 | 1 |
| 12. | 12 | TR | 146.0462 | T | 0 | 1 |
| 13. | 13 | TR | 158.1457 | T | 0 | 1 |
| 14. | 14 | TR | 147.1977 | T | 0 | 1 |
| 15. | 15 | TR | 164.9988 | T | 0 | 1 |
| 16. | 16 | TR | 145.3823 | T | 0 | 1 |
| 17. | 1 | RT | 218.5551 | T | R | 2 |
| 18. | 2 | RT | 133.3201 | T | R | 2 |
| 19. | 3 | RT | 126.0635 | T | R | 2 |
| 20. | 4 | RT | 96.17461 | T | R | 2 |
| 21. | 5 | RT | 188.9038 | T | R | 2 |
| 22. | 6 | RT | 223.6922 | T | R | 2 |
| 23. | 7 | RT | 104.0139 | T | R | 2 |
| 24. | 8 | RT | 237.8962 | T | R | 2 |
| 25. | 9 | TR | 139.7382 | R | T | 2 |
| 26. | 10 | TR | 202.3942 | R | T | 2 |
| 27. | 11 | TR | 136.7848 | R | T | 2 |
| 28. | 12 | TR | 104.5191 | R | T | 2 |
| 29. | 13 | TR | 165.8654 | R | T | 2 |
| 30. | 14 | TR | 139.235 | R | T | 2 |
| 31. | 15 | TR | 166.2391 | R | T | 2 |
| 32. | 16 | TR | 158.5146 | R | T | 2 |

We use pkequiv to conduct a bioequivalence test between reference treatment " $R$ " and test treatment " T ".


The default output for pkequiv shows a CI for the difference between the means (test limits), the ratio of the means, and the user-specified equivalence limits. The classic CI can be constructed around the difference between the average measure of effect for the two drugs or around the ratio of the average measure of effect for the two drugs. pkequiv reports both the difference measure and the ratio measure. Following Chow and Liu (2009), we can apply the $\pm 20$ rule to the difference and ratio measures to determine equivalence limits. For these data, the CI for the difference must be entirely contained within the range $[-30.296,30.296]$ and for the ratio between $80 \%$ and $120 \%$. Here the test limits are within the equivalence limits. Although the test limits are inside the equivalence limits, there is only a $66 \%$ assurance that the observed CI will be within the equivalence limits in the long run. This is an interesting case because, although this sample shows bioequivalence, the evaluation of the long-run performance indicates possible problems. These fictitious data were generated with high intrasubject variability, which causes poor long-run performance.

We now use the data published in Chow and Liu (2009, 71), which we describe in example 1 of $[R]$ pkshape. As shown in example 1, we also use pkshape to reshape the data. We use the same pkequiv command used above to conduct a bioequivalence test on the data.

```
. use https://www.stata-press.com/data/r18/chowliu, clear
. pkshape id seq period1 period2, order(RT TR)
. set seed 123
. pkequiv outcome treat period seq id
    Classic confidence interval for bioequivalence
\begin{tabular}{rrrlr}
\hline & [equivalence limits] & [ & test & limits \(]\) \\
\hline difference: & -16.512 & 16.512 & -8.698 & 4.123 \\
ratio: & \(80 \%\) & \(120 \%\) & \(89.464 \%\) & \(104.994 \%\) \\
\hline Probability test limits are within equivalence & limits \(=\) & 0.9940 \\
Note: Reference treatment \(=1\).
\end{tabular}
```

For these data, the test limits are well within the equivalence limits, and the probability that the test limits are within the equivalence limits is $99.4 \%$.

## > Example 2

pkequiv displays interval hypothesis tests of bioequivalence if you specify the tost or the anderson option, or both. For example,

```
. set seed 123
. pkequiv outcome treat period seq id, tost anderson
Classic confidence interval for bioequivalence
\begin{tabular}{rrrll}
\hline & [equivalence limits] & [ & \multicolumn{1}{c}{ test limits } & ] \\
\hline difference: & -16.512 & 16.512 & -8.698 & 4.123 \\
ratio: & \(80 \%\) & \(120 \%\) & \(89.464 \%\) & \(104.994 \%\) \\
\hline
\end{tabular}
Probability test limits are within equivalence limits = 0.9940
Schuirmann's two one-sided tests
\begin{tabular}{lrl} 
upper test statistic \(=\) & -5.036 & p-value \(=\) \\
lower test statistic \(=\) & 3.810 & p-value \(=\)
\end{tabular}
Anderson and Hauck's test
```



```
Note: Reference treatment = 1.
```

Both of Schuirmann's one-sided tests are highly significant, suggesting that the two drugs are bioequivalent. A similar conclusion is drawn from the Anderson and Hauck test of bioequivalence.

## Stored results

pkequiv stores the following in $r()$ :
Scalars

| $r$ (stddev) | pooled-sample standard deviation of period differences from both sequences |
| :--- | :--- |
| r(uci) | upper limit of a classic CI |
| $r$ (lci) | lower limit of a classic CI |
| $r$ (delta) | delta value used in calculating a symmetric CI |
| r(u3) | upper limit of Fieller's CI |
| $r(13)$ | lower limit of Fieller's CI |

## Methods and formulas

The lower limit for the difference in the two treatments for the classic shortest CI is

$$
L_{1}=\left(\bar{Y}_{T}-\bar{Y}_{R}\right)-t_{\left(\alpha, n_{1}+n_{2}-2\right)} \widehat{\sigma}_{d} \sqrt{\frac{1}{n_{1}}+\frac{1}{n_{2}}}
$$

The upper limit is

$$
U_{1}=\left(\bar{Y}_{T}-\bar{Y}_{R}\right)+t_{\left(\alpha, n_{1}+n_{2}-2\right)} \widehat{\sigma}_{d} \sqrt{\frac{1}{n_{1}}+\frac{1}{n_{2}}}
$$

The limits for the ratio measure are

$$
L_{2}=\left(\frac{L_{1}}{\bar{Y}_{R}}+1\right) 100 \%
$$

and

$$
U_{2}=\left(\frac{U_{1}}{\bar{Y}_{R}}+1\right) 100 \%
$$

where $\bar{Y}_{T}$ is the marginal mean of the test formulation of the drug, $\bar{Y}_{R}$ is the marginal mean of the reference formulation of the drug, and $t_{\left(\alpha, n_{1}+n_{2}-2\right)}$ is the $t$ distribution with $n_{1}+n_{2}-2$ degrees of freedom. $\widehat{\sigma}_{d}$ is the pooled sample variance of the period differences from both sequences, defined as

$$
\widehat{\sigma}_{d}=\frac{1}{n_{1}+n_{2}-2} \sum_{k=1}^{2} \sum_{i=1}^{n_{k}}\left(d_{i k}-\bar{d}_{. k}\right)^{2}
$$

The finite sample performance of the classical CI is assessed via bootstrap simulation of the CI. One thousand bootstrap samples are drawn using the patient IDs as clusters. For each sample, the classical CI is constructed and compared with the equivalence limits.

The upper and lower limits for the symmetric CI are $\bar{Y}_{R}+\Delta$ and $\bar{Y}_{R}-\Delta$, where

$$
\Delta=k_{1} \widehat{\sigma}_{d} \sqrt{\frac{1}{n_{1}}+\frac{1}{n_{2}}}-\left(\bar{Y}_{R}-\bar{Y}_{T}\right)
$$

and (simultaneously)

$$
\Delta=-k_{2} \widehat{\sigma}_{d} \sqrt{\frac{1}{n_{1}}+\frac{1}{n_{2}}}+\left(\bar{Y}_{R}-\bar{Y}_{T}\right)
$$

and $k_{1}$ and $k_{2}$ are computed iteratively to satisfy the above equalities and the condition

$$
\int_{k_{1}}^{k_{2}} f(t) d t=1-2 \alpha
$$

where $f(t)$ is the probability density function of the $t$ distribution with $n_{1}+n_{2}-2$ degrees of freedom.

See Chow and Liu (2009, 88-92) for details about calculating the CI based on Fieller's theorem.
The two test statistics for the two one-sided tests of equivalence are

$$
T_{L}=\frac{\left(\bar{Y}_{T}-\bar{Y}_{R}\right)-\theta_{L}}{\widehat{\sigma}_{d} \sqrt{\frac{1}{n_{1}}+\frac{1}{n_{2}}}}
$$

and

$$
T_{U}=\frac{\left(\bar{Y}_{T}-\bar{Y}_{R}\right)-\theta_{U}}{\widehat{\sigma}_{d} \sqrt{\frac{1}{n_{1}}+\frac{1}{n_{2}}}}
$$

where $-\theta_{L}=\theta_{U}$, both of these being the regulated confidence limits.

The logic of the Anderson and Hauck test is tricky; see Chow and Liu (2009) for a complete explanation. However, the test statistic is

$$
T_{A H}=\frac{\left(\bar{Y}_{T}-\bar{Y}_{R}\right)-\left(\frac{\theta_{L}+\theta_{U}}{2}\right)}{\widehat{\sigma}_{d} \sqrt{\frac{1}{n_{1}}+\frac{1}{n_{2}}}}
$$

and the noncentrality parameter is estimated by

$$
\widehat{\delta}=\frac{\theta_{U}-\theta_{L}}{2 \widehat{\sigma}_{d} \sqrt{\frac{1}{n_{1}}+\frac{1}{n_{2}}}}
$$

The empirical $p$-value is calculated as

$$
p=F_{t}\left(\left|T_{A H}\right|-\widehat{\delta}\right)-F_{t}\left(-\left|T_{A H}\right|-\widehat{\delta}\right)
$$

where $F_{t}$ is the cumulative distribution function of the $t$ distribution with $n_{1}+n_{2}-2$ degrees of freedom.

## References

Anderson, S., and W. W. Hauck. 1983. A new procedure for testing equivalence in comparative bioavailability and other clinical trials. Communications in Statistics-Theory and Methods 12: 2663-2692. https://doi.org/10.1080/03610928308828634.
Chow, S.-C., and J.-P. Liu. 2009. Design and Analysis of Bioavailability and Bioequivalence Studies. 3rd ed. Boca Raton, FL: Chapman and Hall/CRC.
Fieller, E. C. 1954. Some problems in interval estimation. Journal of the Royal Statistical Society, Series B 16: 175-185. https://doi.org/10.1111/j.2517-6161.1954.tb00159.x.
Kutner, M. H., C. J. Nachtsheim, J. Neter, and W. Li. 2005. Applied Linear Statistical Models. 5th ed. New York: McGraw-Hill/Irwin.
Locke, C. S. 1984. An exact confidence interval from untransformed data for the ratio of two formulation means. Journal of Pharmacokinetics and Biopharmaceutics 12: 649-655. https://doi.org/10.1007/BF01059558.
Schuirmann, D. J. 1989. Confidence intervals for the ratio of two means from a cross-over study. In Proceedings of the Biopharmaceutical Section, 121-126. Washington, DC: American Statistical Association.
Westlake, W. J. 1976. Symmetrical confidence intervals for bioequivalence trials. Biometrics 32: 741-744. https://doi.org/10.2307/2529259.

## Also see

[R] pk - Pharmacokinetic (biopharmaceutical) data

## Title

pkexamine - Calculate pharmacokinetic measures

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| Reference | Also see |  |  |

## Description

pkexamine calculates pharmacokinetic measures from concentration-and-time subject-level data. pkexamine computes and displays the maximum measured concentration, the time at the maximum measured concentration, the time of the last measurement, the elimination time, the half-life, and the area under the concentration-time curve $\left(\mathrm{AUC}_{0, t_{\text {max }}}\right)$. Three estimates of the AUC from 0 to infinity $\left(\mathrm{AUC}_{0, \infty}\right)$ are also calculated.
pkexamine is one of the pk commands. Please read [R] pk before reading this entry.

## Quick start

Pharmacokinetic measures for concentrations y at times tvar where idvar $=4$ pkexamine tvar y if idvar==4

Same as above, but use trapezoidal rule to calculate $\mathrm{AUC}_{0}, t_{\text {max }}$ pkexamine tvar y if idvar==4, trapezoid

Plot concentration-time curve where idvar $=2$
pkexamine tvar y if idvar==2, graph
Same as above, and save graph as mygraph
pkexamine tvar y if idvar==2, graph saving(mygraph)

## Menu

Statistics $>$ Epidemiology and related $>$ Other $>$ Pharmacokinetic measures

## Syntax

pkexamine time concentration [if] [in] [, options]
options
Description
Main
fit(\#) use \# points to estimate $\mathrm{AUC}_{0, \infty}$; default is fit(3)
trapezoid use trapezoidal rule; default is cubic splines
graph graph the AUC
line graph the linear extension
log graph the log extension
$\exp (\#) \quad$ plot the exponential fit for the $\mathrm{AUC}_{0, \infty}$
AUC plot
cline_options
marker_options
marker_label_options
affect rendition of plotted points connected by lines
change look of markers (color, size, etc.)
add marker labels; change look or position

## Add plots

addplot (plot) add other plots to the generated graph
Y axis, X axis, Titles, Legend, Overall
twoway_options any options other than by () documented in [G-3] twoway_options
by and collect are allowed; see [U] 11.1.10 Prefix commands.

## Options

Main
fit(\#) specifies the number of points, counting back from the last measurement, to use in fitting the extension to estimate the $\mathrm{AUC}_{0, \infty}$. The default is fit(3), or the last three points. This value should be viewed as a minimum; the appropriate number of points will depend on your data.
trapezoid specifies that the trapezoidal rule be used to calculate the $\mathrm{AUC}_{0}, t_{\max }$. The default is cubic splines, which give better results for most functions. When the curve is irregular, trapezoid may give better results.
graph tells pkexamine to graph the concentration-time curve.
line and log specify the estimates of the $\mathrm{AUC}_{0, \infty}$ to display when graphing the $\mathrm{AUC}_{0, \infty}$. If the graph option is not also specified, then these options are ignored.
$\exp (\#)$ specifies that the exponential fit for the $\mathrm{AUC}_{0, \infty}$ be plotted. You must specify the maximum time value to which you want to plot the curve, and this time value must be greater than the maximum time measurement in the data. If you specify 0 , the curve will be plotted to the point at which the linear extension would cross the $x$ axis. If the graph option is not also specified, then this option is ignored. This option is not valid with the line or log option.

AUC plot
cline_options affect the rendition of the plotted points connected by lines; see [G-3] cline_options. marker_options specify the look of markers. This look includes the marker symbol, size, color, and outline; see [G-3] marker_options.
marker_label_options specify if and how the markers are to be labeled; see [G-3] marker_label_options.
$\qquad$ Add plots
addplot (plot) provides a way to add other plots to the generated graph; see [G-3] addplot_option.
$\qquad$ Y axis, X axis, Titles, Legend, Overall
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

## Remarks and examples

pkexamine computes summary statistics for a given patient in a pharmacokinetic trial. If by idvar: is specified, statistics will be displayed for each subject in the data.

## $>$ Example 1

Chow and Liu $(2009,13)$ present data on a study examining primidone concentrations versus time over a 32 -hour period after dosing a subject.

```
. use https://www.stata-press.com/data/r18/auc
(Primidone concentrations)
. list, abbrev(14)
```

| id | time | concentration |
| ---: | ---: | ---: |
| 1 | 0 | 0 |
| 1 | .5 | 0 |
| 1 | 1 | 2.8 |
| 1 | 1.5 | 4.4 |
| 1 | 2 | 4.4 |
| 1 | 3 | 4.7 |
| 1 | 4 | 4.1 |
| 1 | 6 | 4 |
| 1 | 8 | 3.6 |
| 1 | 12 | 3 |
| 1 | 16 | 2.5 |
| 1 | 24 | 2 |
| 1 | 32 | 1.6 |

We use pkexamine to produce the summary statistics:
. pkexamine time concentration, graph

| Maximum concentration | $=$ | 4.7 |
| ---: | :--- | ---: |
| Time of maximum concentration | $=$ | 3 |
| Time of last observation (Tmax) | $=$ | 32 |
| Elimination rate | $=$ | 0.0279 |
| Half life | $=$ | 24.8503 |

Area under the curve

| AUC [0, Tmax] | AUC [0, inf.) <br> Linear of log conc. | AUC [0, inf.) <br> Linear fit | AUC [0, inf.) <br> Exponential fit |
| :---: | :---: | :---: | :---: |
| 85.24 | 142.603 | 107.759 | 142.603 |

Fit based on last 3 points.


The maximum concentration of 4.7 occurs at time 3, and the time of the last observation (Tmax) is 32. In addition to $\mathrm{AUC}_{0}, t_{\text {max }}$, which is calculated from 0 to the maximum value of time, pkexamine also reports $\mathrm{AUC}_{0}, \infty$, the AUC computed by extending the curve with each of three methods: a linear fit to the $\log$ of the concentration, a linear regression line, and a decreasing exponential regression line. See Methods and formulas for details on these three methods.

By default, all extensions to the AUC are based on the last three points. In looking at the graph for these data, it seems more appropriate to use the last seven points to estimate the $\mathrm{AUC}_{0, \infty}$ :

- pkexamine time concentration, fit(7)

| Maximum concentration | $=$ | 4.7 |
| ---: | :--- | ---: |
| Time of maximum concentration | $=$ | 3 |
| Time of last observation (Tmax) | $=$ | 32 |
| Elimination rate | $=$ | 0.0349 |
| Half life | $=$ | 19.8354 |

Area under the curve

| AUC [0, Tmax] | AUC [0, inf.) <br> Linear of log conc. | AUC [0, inf.) <br> Linear fit | AUC [0, inf.) <br> Exponential fit |
| :---: | :---: | :---: | :---: |
| 85.24 | 131.027 | 96.805 | 129.181 |

Fit based on last 7 points.

This approach decreased the estimate of the $\mathrm{AUC}_{0, \infty}$ for all extensions. To see a graph of the $\mathrm{AUC}_{0, \infty}$ using a linear extension, specify the graph and line options.
pkexamine time concentration, fit(7) graph line
Maximum concentration $=4.7$
Time of maximum concentration $=$
Time of last observation (Tmax) = 32
Elimination rate $=0.0349$
Half life $=19.8354$
Area under the curve

| AUC [0, Tmax] | AUC [0, inf.) <br> Linear of log conc. | AUC [0, inf.) <br> Linear fit | AUC [0, inf.) <br> Exponential fit |
| :---: | :---: | :---: | :---: |
| 85.24 | 131.027 | 96.805 | 129.181 |

Fit based on last 7 points.


## Stored results

pkexamine stores the following in $r()$ :
Scalars

$$
\begin{array}{ll}
r(\text { auc }) & \text { AUC } \\
r(\text { half }) & \text { half-life of the drug } \\
r(\mathrm{ke}) & \text { elimination rate } \\
r(\text { tmax }) & \text { time at last concentration measurement } \\
r(c \operatorname{cmax}) & \text { maximum concentration } \\
r(t a m c) & \text { time of maximum concentration } \\
r(\text { auc_line }) & \text { AUC } 0, \infty \text { estimated with a linear fit } \\
r\left(a u c \_ \text {exp }\right) & \text { AUC }_{0}, \infty \text { estimated with an exponential fit } \\
r(\text { auc_ln }) & \text { AUC } 0, \infty \text { estimated with a linear fit of the natural log }
\end{array}
$$

## Methods and formulas

Let $i$ index the observations sorted by time, let $k$ be the number of observations, and let $f$ be the number of points specified in the fit(\#) option.

The $\mathrm{AUC}_{0, t_{\text {max }}}$ is defined as

$$
\mathrm{AUC}_{0, t_{\max }}=\int_{0}^{t_{\max }} C_{t} d t
$$

where $C_{t}$ is the concentration at time $t$. By default, the integral is calculated numerically using cubic splines. However, if the trapezoidal rule is used, the $\mathrm{AUC}_{0, t_{\text {max }}}$ is given as

$$
\mathrm{AUC}_{0, t_{\max }}=\sum_{i=2}^{k} \frac{C_{i-1}+C_{i}}{2}\left(t_{i}-t_{i-1}\right)
$$

The $\mathrm{AUC}_{0, \infty}$ is the $\mathrm{AUC}_{0, t_{\text {max }}}+\mathrm{AUC}_{t_{\text {max }}, \infty}$, or

$$
\mathrm{AUC}_{0, \infty}=\int_{0}^{t_{\max }} C_{t} d t+\int_{t_{\max }}^{\infty} C_{t} d t
$$

When using the linear extension to the $\mathrm{AUC}_{0, t_{\text {max }}}$, the integration is cut off when the line crosses the $x$ axis. The $\log$ extension is a linear extension on the $\log$ concentration scale. The area for the exponential extension is

$$
\operatorname{AUC}_{t_{\max }, \infty}=\int_{t_{\max }}^{\infty} e^{\beta_{0}+t \beta_{1}} d t=\frac{e^{\beta_{0}+t_{\max } \beta_{1}}}{-\beta_{1}}
$$

where $\beta_{0}>0$ and $\beta_{1}<0$ are the intercept and slope, respectively, of an exponential accelerated failure-time regression of concentration on time.

The elimination rate $K_{\text {eq }}$ is the negative of the slope from a linear regression of log concentration on time fit to the number of points specified in the fit(\#) option:

$$
K_{\mathrm{eq}}=-\frac{\sum_{i=k-f+1}^{k}\left(t_{i}-\bar{t}\right)\left(\ln C_{i}-\overline{\ln C}\right)}{\sum_{i=k-f+1}^{k}\left(t_{i}-\bar{t}\right)^{2}}
$$

The half-life is

$$
t_{\text {half }}=\frac{\ln 2}{K_{\mathrm{eq}}}
$$

## Reference

Chow, S.-C., and J.-P. Liu. 2009. Design and Analysis of Bioavailability and Bioequivalence Studies. 3rd ed. Boca Raton, FL: Chapman and Hall/CRC.

## Also see

[R] pk - Pharmacokinetic (biopharmaceutical) data

## Title

pkshape - Reshape (pharmacokinetic) Latin-square data

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | References | Also see |

## Description

pkshape reshapes data for use with anova, pkcross, and pkequiv; see $[R]$ anova, $[R]$ pkcross, and $[R]$ pkequiv. Latin-square and crossover data are often organized in a manner that cannot be analyzed easily with Stata. pkshape reorganizes the data in memory for use in Stata.
pkshape is one of the pk commands. Please read [R] pk before reading this entry.

## Quick start

Reshape data when string sequence variable seq $=T R$ or $R T$ for patients identified by idvar observed at tvar1 and tvar2 pkshape idvar seq tvar1 tvar2

Same as above, but with numeric seq $=1$ indicating TR and seq $=2$ indicating RT pkshape idvar seq tvar1 tvar2, order (TR RT)

Indicate that period 2 is a washout and the second treatment is administered in period 3
pkshape idvar seq tvar1 tvar2 tvar3, order (TOR ROT)

## Menu

Statistics $>$ Epidemiology and related $>$ Other $>$ Reshape pharmacokinetic Latin-square data

## Syntax

pkshape id sequence periodl period 2 [periodlist] [, options]
Variable id specifies unique subject identifiers. Variable sequence specifies the sequence (numeric or string) in which treatments were received. Variables periodl, period2, and so on specify the pharmacokinetic measurements such as AUC in the corresponding periods.
options
Description
order (string)
outcome(newvar)
treatment (newvar)
carryover(newvar)
sequence(newvar)
period(newvar)
apply treatments in specified order; required with numeric sequence name for outcome variable; default is outcome (outcome) name for treatment variable; default is treatment (treat) name for carryover variable; default is carryover (carry) name for sequence variable; default is sequence (sequence) name for period variable; default is period(period)

## Options

order (string) specifies the order in which treatments were applied when generating the sequence, treatment, and carryover variables in the reorganized data. This option is required if the input sequence variable, sequence, is numeric. It is not allowed if sequence is a string variable. For crossover designs, any washout periods can be indicated with the number 0.
outcome (newvar) specifies the name for the outcome variable in the reorganized data. By default, outcome (outcome) is used.
treatment (newvar) specifies the name for the treatment variable in the reorganized data. By default, treatment (treat) is used.
carryover (newvar) specifies the name for the carryover variable in the reorganized data. By default, carryover (carry) is used.
sequence (newvar) specifies the name for the sequence variable in the reorganized data. By default, sequence (sequence) is used.
period(newvar) specifies the name for the period variable in the reorganized data. By default, period(period) is used.

## Remarks and examples

Often, data from a Latin-square experiment are naturally organized in a manner that Stata cannot manage easily. pkshape reorganizes Latin-square data so that they can be used with anova (see $[\mathrm{R}]$ anova) or any pk command. This includes the classic $2 \times 2$ crossover design commonly used in pharmaceutical research, as well as many other Latin-square designs. When using pkshape, newly created variables will automatically be labeled and system value labels will be created. The value label _treatlbl will be attached to the treatment and carrover variables, to indicate which treatment is applied in a given period and which treatment is being carried over from the previous period. The value label _seqlbl will be attached to the sequence variable, indicating the sequence of treatments.
pkshape expects the data to be organized in the same format as that produced by [R] pkcollapsewith variables representing time periods of the study.

## > Example 1

Consider the example data published in Chow and Liu (2009, 71). There are 24 patients, 12 in each sequence. Sequence 1 is the reference formulation followed by the test formulation; sequence 2 is the test formulation followed by the reference formulation. The measurements reported are the $\mathrm{AUC}_{0, t_{\text {max }}}$ for each patient and for each period.
. use https://www.stata-press.com/data/r18/chowliu
. list, sep(4)

|  | id | seq | period1 | period2 |
| :---: | :---: | :---: | :---: | :---: |
| 1. | 1 | 1 | 74.675 | 73.675 |
| 2. | 4 | 1 | 96.4 | 93.25 |
| 3. | 5 | 1 | 101.95 | 102.125 |
| 4. | 6 | 1 | 79.05 | 69.45 |
| 5. | 11 | 1 | 79.05 | 69.025 |
| 6. | 12 | 1 | 85.95 | 68.7 |
| 7. | 15 | 1 | 69.725 | 59.425 |
| 8. | 16 | 1 | 86.275 | 76.125 |
| 9. | 19 | 1 | 112.675 | 114.875 |
| 10. | 20 | 1 | 99.525 | 116.25 |
| 11. | 23 | 1 | 89.425 | 64.175 |
| 12. | 24 | 1 | 55.175 | 74.575 |
| 13. | 2 | 2 | 74.825 | 37.35 |
| 14. | 3 | 2 | 86.875 | 51.925 |
| 15. | 7 | 2 | 81.675 | 72.175 |
| 16. | 8 | 2 | 92.7 | 77.5 |
| 17. | 9 | 2 | 50.45 | 71.875 |
| 18. | 10 | 2 | 66.125 | 94.025 |
| 19. | 13 | 2 | 122.45 | 124.975 |
| 20. | 14 | 2 | 99.075 | 85.225 |
| 21. | 17 | 2 | 86.35 | 95.925 |
| 22. | 18 | 2 | 49.925 | 67.1 |
| 23. | 21 | 2 | 42.7 | 59.425 |
| 24. | 22 | 2 | 91.725 | 114.05 |

Because the outcome for one person is in two different variables, the treatment that was applied to an individual is a function of the period and the sequence. To analyze this treatment using anova, all the outcomes must be in one variable, and each covariate must be in its own variable. To reorganize these data, use pkshape:
. pkshape id seq period1 period2, order (RT TR)
. sort seq id period
. list, sep(8)

|  | id | sequence | outcome | treat | carry | period |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 1. | 1 | RT | 74.675 | R | 0 | 1 |
| 2. | 1 | RT | 73.675 | T | R | 2 |
| 3. | 4 | RT | 96.4 | R | 0 | 1 |
| 4. | 4 | RT | 93.25 | T | R | 2 |
| 5. | 5 | RT | 101.95 | R | 0 | 1 |
| 6. | 5 | RT | 102.125 | T | R | 2 |
| 7. | 6 | RT | 79.05 | R | 0 | 1 |
| 8. | 6 | RT | 69.45 | T | R | 2 |
| 9. | 11 | RT | 79.05 | R | 0 | 1 |
| 10. | 11 | RT | 69.025 | T | R | 2 |
| 11. | 12 | RT | 85.95 | R | 0 | 1 |
| 12. | 12 | RT | 68.7 | T | R | 2 |
| 13. | 15 | RT | 69.725 | R | 0 | 1 |
| 14. | 15 | RT | 59.425 | T | R | 2 |
| 15. | 16 | RT | 86.275 | R | 0 | 1 |
| 16. | 16 | RT | 76.125 | T | R | 2 |
| 17. | 19 | RT | 112.675 | R | 0 | 1 |
| 18. | 19 | RT | 114.875 | T | R | 2 |
| 19. | 20 | RT | 99.525 | R | 0 | 1 |
| 20. | 20 | RT | 116.25 | T | R | 2 |
| 21. | 23 | RT | 89.425 | R | 0 | 1 |
| 22. | 23 | RT | 64.175 | T | R | 2 |
| 23. | 24 | RT | 55.175 | R | 0 | 1 |
| 24. | 24 | RT | 74.575 | T | R | 2 |
| 25. | 2 | TR | 74.825 | T | 0 | 1 |
| 26. | 2 | TR | 37.35 | R | T | 2 |
| 27. | 3 | TR | 86.875 | T | 0 | 1 |
| 28. | 3 | TR | 51.925 | R | T | 2 |
| 29. | 7 | TR | 81.675 | T | 0 | 1 |
| 30. | 7 | TR | 72.175 | R | T | 2 |
| 31. | 8 | TR | 92.7 | T | 0 | 1 |
| 32. | 8 | TR | 77.5 | R | T | 2 |
| 33. | 9 | TR | 50.45 | T | 0 | 1 |
| 34. | 9 | TR | 71.875 | R | T | 2 |
| 35. | 10 | TR | 66.125 | T | 0 | 1 |
| 36. | 10 | TR | 94.025 | R | T | 2 |
| 37. | 13 | TR | 122.45 | T | 0 | 1 |
| 38. | 13 | TR | 124.975 | R | T | 2 |
| 39. | 14 | TR | 99.075 | T | 0 | 1 |
| 40. | 14 | TR | 85.225 | R | T | 2 |
| 41. | 17 | TR | 86.35 | T | 0 | 1 |
| 42. | 17 | TR | 95.925 | R | T | 2 |
| 43. | 18 | TR | 49.925 | T | 0 | 1 |
| 44. | 18 | TR | 67.1 | R | T | 2 |
| 45. | 21 | TR | 42.7 | T | 0 | 1 |
| 46. | 21 | TR | 59.425 | R | T | 2 |
| 47. | 22 | TR | 91.725 | T | 0 | 1 |
| 48. | 22 | TR | 114.05 | R | T | 2 |

Now, the data are organized into separate variables that indicate each factor level for each of the covariates, so the data may be used with anova or pkcross; see [ $R$ ] anova and [ $R$ ] pkeross.

Initially, the output from list displayed sequence values 1 and 2, but now we see sequences RT and $T R$ listed for the individuals. pkshape used the information we provided in the order () option to assign value labels to the numeric variables sequence, treat, and carry. Because we did not specify any new variable names, the default names were used.

## $>$ Example 2

Consider the study of background music on bank teller productivity published in Kutner et al. (2005). The data are

| Week | Monday | Tuesday | Wednesday | Thursday | Friday |
| ---: | :---: | :---: | :---: | :---: | :---: |
| 1 | $18(\mathrm{D})$ | $17(\mathrm{C})$ | $14(\mathrm{~A})$ | $21(\mathrm{~B})$ | $17(\mathrm{E})$ |
| 2 | $13(\mathrm{C})$ | $34(\mathrm{~B})$ | $21(\mathrm{E})$ | $16(\mathrm{~A})$ | $15(\mathrm{D})$ |
| 3 | $7(\mathrm{~A})$ | $29(\mathrm{D})$ | $32(\mathrm{~B})$ | $27(\mathrm{E})$ | $13(\mathrm{C})$ |
| 4 | $17(\mathrm{E})$ | $13(\mathrm{~A})$ | $24(\mathrm{C})$ | $31(\mathrm{D})$ | $25(\mathrm{~B})$ |
| 5 | $21(\mathrm{~B})$ | $26(\mathrm{E})$ | $26(\mathrm{D})$ | $31(\mathrm{C})$ | $7(\mathrm{~A})$ |

The numbers are the productivity scores, and the letters represent the treatment. We entered the data into Stata:

|  | id | seq | day1 | day2 | day3 | day 4 | day5 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 1. | 1 | dcabe | 18 | 17 | 14 | 21 | 17 |
| 2. | 2 | cbead | 13 | 34 | 21 | 16 | 15 |
| 3. | 3 | adbec | 7 | 29 | 32 | 27 | 13 |
| 4. | 4 | eacdb | 17 | 13 | 24 | 31 | 25 |
| 5. | 5 | bedca | 21 | 26 | 26 | 31 | 7 |

We reshape these data with pkshape:

```
. pkshape id seq day1 day2 day3 day4 day5
. sort id period
. list, sep(0)
```

|  | id | sequence | outcome | treat | carry | period |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 1. | 1 | dcabe | 18 | d | 0 | 1 |
| 2. | 1 | dcabe | 17 | c | d | 2 |
| 3. | 1 | dcabe | 14 | a | c | 3 |
| 4. | 1 | dcabe | 21 | b | a | 4 |
| 5. | 1 | dcabe | 17 | e | b | 5 |
| 6. | 2 | cbead | 13 | c | 0 | 1 |
| 7. | 2 | cbead | 34 | b | c | 2 |
| 8. | 2 | cbead | 21 | e | b | 3 |
| 9. | 2 | cbead | 16 | a | e | 4 |
| 10. | 2 | cbead | 15 | d | a | 5 |
| 11. | 3 | adbec | 7 | a | 0 | 1 |
| 12. | 3 | adbec | 29 | d | a | 2 |
| 13. | 3 | adbec | 32 | b | d | 3 |
| 14. | 3 | adbec | 27 | e | b | 4 |
| 15. | 3 | adbec | 13 | c | e | 5 |
| 16. | 4 | eacdb | 17 | e | 0 | 1 |
| 17. | 4 | eacdb | 13 | a | e | 2 |
| 18. | 4 | eacdb | 24 | c | a | 3 |
| 19. | 4 | eacdb | 31 | d | c | 4 |
| 20. | 4 | eacdb | 25 | b | d | 5 |
| 21. | 5 | bedca | 21 | b | 0 | 1 |
| 22. | 5 | bedca | 26 | e | b | 2 |
| 23. | 5 | bedca | 26 | d | e | 3 |
| 24. | 5 | bedca | 31 | c | d | 4 |
| 25. | 5 | bedca | 7 | a | c | 5 |

Here the sequence variable is a string variable that specifies how the treatments were applied. The characters in this string variable are used to assign value labels to the newly created sequence, treat, and carry variables. We could now produce an ANOVA table:

- anova outcome sequence period treat



## > Example 3

Consider the Latin-square crossover example published in Kutner et al. (2005). The example is about apple sales given different methods for displaying apples.

| Pattern | Store | Week 1 | Week 2 | Week 3 |
| ---: | :---: | :---: | :---: | :---: |
| 1 | 1 | $9(\mathrm{~B})$ | $12(\mathrm{C})$ | $15(\mathrm{~A})$ |
|  | 2 | 4(B) | $12(\mathrm{C})$ | $9(\mathrm{~A})$ |
| 2 | 1 | $12(\mathrm{~A})$ | $14(\mathrm{~B})$ | $3(\mathrm{C})$ |
|  | 2 | $13(\mathrm{~A})$ | $14(\mathrm{~B})$ | $3(\mathrm{C})$ |
| 3 | 1 | $7(\mathrm{C})$ | $18(\mathrm{~A})$ | $6(\mathrm{~B})$ |
|  | 2 | $5(\mathrm{C})$ | $20(\mathrm{~A})$ | $4(\mathrm{~B})$ |

We entered the data into Stata:

```
. use https://www.stata-press.com/data/r18/applesales, clear
(Display impact on apple sales)
. list, sep(2)
```

|  | id | seq | p1 | p2 | p3 | square |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: |
| 1. | 1 | 1 | 9 | 12 | 15 | 1 |
| 2. | 2 | 1 | 4 | 12 | 9 | 2 |
| 3. | 3 | 2 | 12 | 14 | 3 | 1 |
| 4. | 4 | 2 | 13 | 14 | 3 | 2 |
|  | 5 | 5 | 3 | 7 | 18 | 6 |

Now, the data can be reorganized using descriptive names for the outcome variables.
. pkshape id seq p1 p2 p3, order(bca abc cab) seq(pattern) treat(displays)
. anova outcome pattern period displays id|pattern

|  | $\begin{array}{ll} \text { Number of obs } & = \\ \text { Root MSE } & = \end{array}$ | $\begin{array}{r} 18 \\ 1.59426 \end{array}$ | R-squared |  | $\begin{aligned} & 0.9562 \\ & 0.9069 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | 443.66667 | 9 | 49.296296 | 19.40 | 0.0002 |
| pattern | . 33333333 | 2 | . 16666667 | 0.07 | 0.9370 |
| period | 233.33333 | 2 | 116.66667 | 45.90 | 0.0000 |
| displays | 189 | 2 | 94.5 | 37.18 | 0.0001 |
| id\|pattern | 21 | 3 | 7 | 2.75 | 0.1120 |
| Residual | 20.333333 | 8 | 2.5416667 |  |  |
| Total | 464 | 17 | 27.294118 |  |  |

These are the same results reported by Kutner et al. (2005).

## References

Chow, S.-C., and J.-P. Liu. 2009. Design and Analysis of Bioavailability and Bioequivalence Studies. 3rd ed. Boca Raton, FL: Chapman and Hall/CRC.
Kutner, M. H., C. J. Nachtsheim, J. Neter, and W. Li. 2005. Applied Linear Statistical Models. 5th ed. New York: McGraw-Hill/Irwin.

## Also see

[R] pk - Pharmacokinetic (biopharmaceutical) data

## Title

pksumm - Summarize pharmacokinetic data

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Methods and formulas | Also see |

## Description

pksumm obtains summary measures based on the first four moments from the empirical distribution of each pharmacokinetic measurement and tests the null hypothesis that the distribution of that measurement is normally distributed.
pksumm is one of the pk commands. Please read [R] pk before reading this entry.

## Quick start

Table of pharmacokinetic measures for concentrations y at times tvar for patients identified by idvar pksumm idvar tvar y

Add a histogram of $\mathrm{AUC}_{0, t_{\text {max }}}$ values pksumm idvar tvar y, graph

Same as above
pksumm idvar tvar y, graph stat(auc)
Same as above, but plot AUC calculated from 0 to $\infty$ using a linear extension pksumm idvar tvar y, graph stat(aucline)

Same as above, but use 10 bins for the histogram pksumm idvar tvar y, graph stat(aucline) bins(10)

## Menu

Statistics $>$ Epidemiology and related $>$ Other $>$ Summarize pharmacokinetic data

## Syntax

pksumm id time concentration [if] [in] [, options]
options Description

Main
trapezoid use trapezoidal rule to calculate $\mathrm{AUC}_{0, t_{\text {max }}}$; default is cubic splines
fit(\#) use \# points to estimate $\mathrm{AUC}_{0, \infty}$; default is fit(3)
notimechk do not check whether follow-up time for all subjects is the same
nodots suppress the dots during calculation
graph graph the distribution of statistic
stat (statistic) graph the specified statistic; default is stat(auc)
Histogram, Density plots, Y axis, X axis, Titles, Legend, Overall
histogram_options any option other than by () documented in [R] histogram
statistic Description
auc area under the concentration-time curve $\left(\mathrm{AUC}_{0, t_{\text {max }}}\right)$; the default
aucline $\quad$ AUC from 0 to $\infty$ using a linear extension
aucexp $\quad$ AUC from 0 to $\infty$ using an exponential extension
auclog area under the concentration-time curve from 0 to $\infty$ extended with a
linear fit to $\log$ concentration
half half-life of the drug
ke elimination rate
cmax maximum concentration
tmax time at last concentration
tomc time of maximum concentration

## Options

$\qquad$ Main
trapezoid specifies that the trapezoidal rule be used to calculate the $\mathrm{AUC}_{0, t_{\max }}$. The default is cubic splines, which give better results for most situations. When the curve is irregular, the trapezoidal rule may give better results.
fit(\#) specifies the number of points, counting back from the last time measurement, to use in fitting the extension to estimate the $\mathrm{AUC}_{0, \infty}$. The default is fit(3), the last three points. This default should be viewed as a minimum; the appropriate number of points will depend on the data. notimechk suppresses the check that the follow-up time for all subjects is the same. By default, pksumm expects the maximum follow-up time to be equal for all subjects.
nodots suppresses the progress dots during calculation. By default, a dot (a period) is displayed for every call to calculate the pharmacokinetic measures.
graph requests a graph of the distribution of the statistic specified with stat().
stat (statistic) specifies the statistic that pksumm should graph. The default is stat(auc). If the graph option is not also specified, then this option is ignored.
histogram_options are any of the options documented in [R] histogram, excluding by (). For pksumm, fraction is the default, not density.

## Remarks and examples

pksumm produces summary statistics for the distribution of nine common pharmacokinetic measurements. If there are more than eight subjects, pksumm also computes a test for normality on each measurement. The nine measurements summarized by pksumm are listed above and are described in Methods and formulas of $[\mathrm{R}]$ pkexamine.

## > Example 1

We demonstrate the use of pksumm on a variation of the data described in [R] pk. We have drug concentration data on 15 subjects, each measured at 13 time points over a 32-hour period. A few of the records are as follows:

```
. use https://www.stata-press.com/data/r18/pksumm
. list, sep(0)
```

| id | time | conc |
| :---: | ---: | ---: |
| 1 | 0 | 0 |
| 1 | .5 | 3.073403 |
| 1 | 1 | 5.188444 |
| 1 | 1.5 | 5.898577 |
| 1 | 2 | 5.096378 |
| 1 | 3 | 6.094085 |
| (output omitted) |  |  |
| 15 | 0 | 0 |
| 15 | .5 | 3.86493 |
| 15 | 1 | 6.432444 |
| 15 | 1.5 | 6.969195 |
| 15 | 2 | 6.307024 |
| 15 | 3 | 6.509584 |
| 15 | 4 | 6.555091 |
| 15 | 6 | 7.318319 |
| 15 | 8 | 5.329813 |
| 15 | 12 | 5.411624 |
| 15 | 16 | 3.891397 |
| 15 | 24 | 5.167516 |
| 15 | 32 | 2.649686 |

We can use pksumm to view the summary statistics for all the pharmacokinetic parameters.

| Summary statistics for the pharmacokinetic measures |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | Number of | servation | 5 |
| Measure | Mean | Median | Variance | Skewness | Kurtosis | p-value |
| auc | 150.74 | 150.96 | 123.07 | -0.26 | 2.10 | 0.69 |
| aucline | 408.30 | 214.17 | 188856.87 | 2.57 | 8.93 | 0.00 |
| aucexp | 691.68 | 297.08 | 762679.94 | 2.56 | 8.87 | 0.00 |
| auclog | 688.98 | 297.67 | 797237.24 | 2.59 | 9.02 | 0.00 |
| half | 94.84 | 29.39 | 18722.13 | 2.26 | 7.37 | 0.00 |
| ke | 0.02 | 0.02 | 0.00 | 0.89 | 3.70 | 0.09 |
| cmax | 7.36 | 7.42 | 0.42 | -0.60 | 2.56 | 0.44 |
| tomc | 3.47 | 3.00 | 7.62 | 2.17 | 7.18 | 0.00 |
| tmax | 32.00 | 32.00 | 0.00 | . | . |  |

For the 15 subjects, the mean $\mathrm{AUC}_{0, t_{\max }}$ is 150.74 , and $\sigma^{2}=123.07$. The skewness of -0.26 indicates that the distribution is slightly skewed left. The $p$-value of 0.69 for the $\chi^{2}$ test of normality indicates that we cannot reject the null hypothesis that the distribution is normal.
If we were to consider any of the three variants of the $\mathrm{AUC}_{0, \infty}$, we would see that there is huge variability and that the distribution is heavily skewed. A skewness different from 0 and a kurtosis different from 3 are expected because the distribution of the $\mathrm{AUC}_{0, \infty}$ is not normal.
We now graph the distribution of $\mathrm{AUC}_{0, t_{\text {max }}}$ by specifying the graph option.
. pksumm id time conc, graph bins(20)
.................
Summary statistics for the pharmacokinetic measures
Number of observations $=15$

| Measure | Mean | Median | Variance | Skewness | Kurtosis | p-value |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| auc | 150.74 | 150.96 | 123.07 | -0.26 | 2.10 | 0.69 |
| aucline | 408.30 | 214.17 | 188856.87 | 2.57 | 8.93 | 0.00 |
| aucexp | 691.68 | 297.08 | 762679.94 | 2.56 | 8.87 | 0.00 |
| auclog | 688.98 | 297.67 | 797237.24 | 2.59 | 9.02 | 0.00 |
| half | 94.84 | 29.39 | 18722.13 | 2.26 | 7.37 | 0.00 |
| ke | 0.02 | 0.02 | 0.00 | 0.89 | 3.70 | 0.09 |
| cmax | 7.36 | 7.42 | 0.42 | -0.60 | 2.56 | 0.44 |
| tomc | 3.47 | 3.00 | 7.62 | 2.17 | 7.18 | 0.00 |
| tmax | 32.00 | 32.00 | 0.00 | . | . | . |


graph, by default, plots $\mathrm{AUC}_{0, t_{\max }}$. To plot a graph of one of the other pharmacokinetic measurements, we need to specify the stat () option. For example, we can ask Stata to produce a plot of the $\mathrm{AUC}_{0, \infty}$ using the log extension:
. pksumm id time conc, stat(auclog) graph bins(20)
.................
Summary statistics for the pharmacokinetic measures
Number of observations $=15$

| Measure | Mean | Median | Variance | Skewness | Kurtosis | p-value |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| auc | 150.74 | 150.96 | 123.07 | -0.26 | 2.10 | 0.69 |
| aucline | 408.30 | 214.17 | 188856.87 | 2.57 | 8.93 | 0.00 |
| aucexp | 691.68 | 297.08 | 762679.94 | 2.56 | 8.87 | 0.00 |
| auclog | 688.98 | 297.67 | 797237.24 | 2.59 | 9.02 | 0.00 |
| half | 94.84 | 29.39 | 18722.13 | 2.26 | 7.37 | 0.00 |
| ke | 0.02 | 0.02 | 0.00 | 0.89 | 3.70 | 0.09 |
| cmax | 7.36 | 7.42 | 0.42 | -0.60 | 2.56 | 0.44 |
| tomc | 3.47 | 3.00 | 7.62 | 2.17 | 7.18 | 0.00 |
| tmax | 32.00 | 32.00 | 0.00 | . | . | . |



## Methods and formulas

The $\chi^{2}$ test for normality is conducted with sktest; see $[\mathrm{R}]$ sktest for more information on the test of normality.

The statistics reported by pksumm are identical to those reported by summarize and sktest; see $[R]$ summarize and $[R]$ sktest.

## Also see

[R] pk - Pharmacokinetic (biopharmaceutical) data

## Title

poisson - Poisson regression

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

poisson fits a Poisson regression of depvar on indepvars, where depvar is a nonnegative count variable.

If you have panel data, see [XT] xtpoisson.

## Quick start

Poisson regression of $y$ on $x$
poisson y x
Add categorical variable a
poisson y x i.a
Add exposure variable $v$
poisson y x i.a, exposure(v)
With robust standard errors

```
poisson y x i.a, vce(robust)
```

Report results as incidence-rate ratios

```
poisson y x i.a, irr
```

Replace data in memory with the results of running a Poisson regression model on each level of catvar
statsby, by (catvar) clear: poisson y x

## Menu

Statistics $>$ Count outcomes $>$ Poisson regression

## Syntax

poisson depvar [indepvars] [if] [in] [weight] [, options]
options Description

## Model

noconstant
exposure (varname ${ }_{e}$ )
offset ( varname $_{o}$ )
constraints (constraints)
SE/Robust
vce (vcetype)

Reporting
level(\#)
irr
nocnsreport
display_options
suppress constant term
include $\ln \left(\right.$ varname $\left._{e}\right)$ in model with coefficient constrained to 1
include varname ${ }_{o}$ in model with coefficient constrained to 1
apply specified linear constraints
vcetype may be oim, robust, cluster clustvar, opg, bootstrap, or jackknife
set confidence level; default is level (95)
report incidence-rate ratios
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
control the maximization process; seldom used
keep collinear variables
display legend instead of statistics
indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
depvar, indepvars, varname $e_{e}$, and varname $_{o}$ may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bayes, bootstrap, by, fmm, fp, jackknife, mfp, mi estimate, nestreg, rolling, statsby, stepwise,
and svy are allowed; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: poisson and [FMM] fmm: poisson.
vce(bootstrap) and vce(jackknife) are not allowed with the mi estimate prefix; see [MI] mi estimate.
Weights are not allowed with the bootstrap prefix; see $[\mathrm{R}]$ bootstrap.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

[^46]vce(vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.

Reporting
level(\#); see [R] Estimation options.
irr reports estimated coefficients transformed to incidence-rate ratios, that is, $e^{\beta_{i}}$ rather than $\beta_{i}$. Standard errors and confidence intervals are similarly transformed. This option affects how results are displayed, not how they are estimated or stored. irr may be specified at estimation or when replaying previously estimated results.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\% fmt), sformat (\% fmt), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate(\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance (\#), nonrtolerance, and from(init_specs); see [R] Maximize. These options are seldom used.

Setting the optimization type to technique (bhhh) resets the default vcetype to vce (opg).
The following options are available with poisson but are not shown in the dialog box: collinear, coeflegend; see [R] Estimation options.

## Remarks and examples

The basic idea of Poisson regression was outlined by Coleman (1964, 378-379). See Cameron and Trivedi (2013; 2022, chap. 20) and Johnson, Kemp, and Kotz (2005, chap. 4) for information about the Poisson distribution. See Cameron and Trivedi (2013), Long (1997, chap. 8), Long and Freese (2014, chap. 9), McNeil (1996, chap. 6), and Selvin (2011, chap. 6) for an introduction to Poisson regression. Also see Selvin (2004, chap. 5) for a discussion of the analysis of spatial distributions, which includes a discussion of the Poisson distribution. An early example of Poisson regression was Cochran (1940).

Poisson regression fits models of the number of occurrences (counts) of an event. The Poisson distribution has been applied to diverse events, such as the number of soldiers kicked to death by horses in the Prussian army (von Bortkiewicz 1898); the pattern of hits by buzz bombs launched against London during World War II (Clarke 1946); telephone connections to a wrong number (Thorndike 1926); and disease incidence, typically with respect to time, but occasionally with respect to space. The basic assumptions are as follows:

1. There is a quantity called the incidence rate that is the rate at which events occur. Examples are 5 per second, 20 per 1,000 person-years, 17 per square meter, and 38 per cubic centimeter.
2. The incidence rate can be multiplied by exposure to obtain the expected number of observed events. For example, a rate of 5 per second multiplied by 30 seconds means that 150 events are expected; a rate of 20 per 1,000 person-years multiplied by 2,000 person-years means that 40 events are expected; and so on.
3. Over very small exposures $\epsilon$, the probability of finding more than one event is small compared with $\epsilon$.
4. Nonoverlapping exposures are mutually independent.

With these assumptions, to find the probability of $k$ events in an exposure of size $E$, you divide $E$ into $n$ subintervals $E_{1}, E_{2}, \ldots, E_{n}$, and approximate the answer as the binomial probability of observing $k$ successes in $n$ trials. If you let $n \rightarrow \infty$, you obtain the Poisson distribution.

In the Poisson regression model, the incidence rate for the $j$ th observation is assumed to be given by

$$
r_{j}=e^{\beta_{0}+\beta_{1} x_{1, j}+\cdots+\beta_{k} x_{k, j}}
$$

If $E_{j}$ is the exposure, the expected number of events, $C_{j}$, will be

$$
\begin{aligned}
C_{j} & =E_{j} e^{\beta_{0}+\beta_{1} x_{1, j}+\cdots+\beta_{k} x_{k, j}} \\
& =e^{\ln \left(E_{j}\right)+\beta_{0}+\beta_{1} x_{1, j}+\cdots+\beta_{k} x_{k, j}}
\end{aligned}
$$

This model is fit by poisson. Without the exposure() or offset () options, $E_{j}$ is assumed to be 1 (equivalent to assuming that exposure is unknown), and controlling for exposure, if necessary, is your responsibility.

Comparing rates is most easily done by calculating incidence-rate ratios (IRRs). For instance, what is the relative incidence rate of chromosome interchanges in cells as the intensity of radiation increases; the relative incidence rate of telephone connections to a wrong number as load increases; or the relative incidence rate of deaths due to cancer for females relative to males? That is, you want to hold all the $x$ 's in the model constant except one, say, the $i$ th. The IRR for a one-unit change in $x_{i}$ is

$$
\frac{e^{\ln (E)+\beta_{1} x_{1}+\cdots+\beta_{i}\left(x_{i}+1\right)+\cdots+\beta_{k} x_{k}}}{e^{\ln (E)+\beta_{1} x_{1}+\cdots+\beta_{i} x_{i}+\cdots+\beta_{k} x_{k}}}=e^{\beta_{i}}
$$

More generally, the IRR for a $\Delta x_{i}$ change in $x_{i}$ is $e^{\beta_{i} \Delta x_{i}}$. The lincom command can be used after poisson to display incidence-rate ratios for any group relative to another; see [R] lincom.

## $>$ Example 1

Chatterjee and Hadi $(2012,174)$ give the number of injury incidents and the proportion of flights for each airline out of the total number of flights from New York for nine major US airlines in one year:
. use https://www.stata-press.com/data/r18/airline
. list

| airline | injuries | $n$ | XYZowned |
| ---: | ---: | ---: | ---: |
| 1 | 11 | 0.0950 | 1 |
| 2 | 7 | 0.1920 | 0 |
| 3 | 7 | 0.0750 | 0 |
| 4 | 19 | 0.2078 | 0 |
| 5 | 9 | 0.1382 | 0 |
| 6 | 4 | 0.0540 | 1 |
| 7 | 3 | 0.1292 | 0 |
| 8 | 1 | 0.0503 | 0 |
| 9 | 3 | 0.0629 | 1 |

To their data, we have added a fictional variable, XYZowned. We will imagine that an accusation is made that the airlines owned by XYZ Company have a higher injury rate.

```
. poisson injuries XYZowned, exposure(n) irr
Iteration 0: Log likelihood = -23.027197
Iteration 1: Log likelihood = -23.027177
Iteration 2: Log likelihood = -23.027177
```

Poisson regression $\quad$ Number of obs $=\quad 9$
LR chi2 (1) $=1.77$
Prob > chi2 $=0.1836$
Pseudo R2 $=0.0370$

| injuries | IRR | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| XYZowned | 1.463467 | .406872 | 1.37 | 0.171 | .8486578 | 2.523675 |
| _cons | 58.04416 | 8.558145 | 27.54 | 0.000 | 43.47662 | 77.49281 |
| $\ln (\mathrm{n})$ | 1 | (exposure) |  |  |  |  |

Note: _cons estimates baseline incidence rate.
We specified irr to see the IRRs rather than the underlying coefficients. We estimate that XYZ Airlines' injury rate is 1.46 times larger than that for other airlines, but the $95 \%$ confidence interval is 0.85 to 2.52; we cannot even reject the hypothesis that XYZ Airlines has a lower injury rate.

## - Technical note

In example 1, we assumed that each airline's exposure was proportional to its fraction of flights out of New York. What if "large" airlines, however, also used larger planes, and so had even more passengers than would be expected, given this measure of exposure? A better measure would be each airline's fraction of passengers on flights out of New York, a number that we do not have. Even so, we suppose that n represents this number to some extent, so a better estimate of the effect might be


Here rather than specifying the exposure() option, we explicitly included the variable that would normalize for exposure in the model. We did not specify the irr option, so we see coefficients rather than IRRs. We started with the model

$$
\text { rate }=e^{\beta_{0}+\beta_{1} \text { XYZowned }}
$$

The observed counts are therefore

$$
\text { count }=n e^{\beta_{0}+\beta_{1} \text { XYZowned }}=e^{\ln (n)+\beta_{0}+\beta_{1} \mathrm{XYZowned}}
$$

which amounts to constraining the coefficient on $\ln (\mathrm{n})$ to 1 . This is what was estimated when we specified the exposure (n) option. In the above model, we included the normalizing exposure ourselves and, rather than constraining the coefficient to be 1 , estimated the coefficient.

The estimated coefficient is 1.42 , a respectable distance away from 1 , and is consistent with our speculation that larger airlines also use larger airplanes. With this small amount of data, however, we also have a wide confidence interval that includes 1 .

Our estimated coefficient on XYZowned is now 0.684 , and the implied IRR is $e^{0.684} \approx 1.98$ (which we could also see by typing poisson, irr). The $95 \%$ confidence interval for the coefficient still includes 0 (the interval for the IRR includes 1 ), so although the point estimate is now larger, we still cannot be certain of our results.

Our expert opinion would be that, although there is not enough evidence to support the charge, there is enough evidence to justify collecting more data.

## > Example 2

In a famous age-specific study of coronary disease deaths among male British doctors, Doll and Hill (1966) reported the following data (reprinted in Lash et al. [2021, 417]):

|  | Smokers |  | Nonsmokers |  |
| :--- | ---: | ---: | ---: | ---: |
| Age | Deaths | Person-years | Deaths | Person-years |
| $35-44$ | 32 | 52,407 | 2 | 18,790 |
| $45-54$ | 104 | 43,248 | 12 | 10,673 |
| $55-64$ | 206 | 28,612 | 28 | 5,710 |
| $65-74$ | 186 | 12,663 | 28 | 2,585 |
| $75-84$ | 102 | 5,317 | 31 | 1,462 |

The first step is to enter these data into Stata, which we have done:

```
. use https://www.stata-press.com/data/r18/dollhill3, clear
(Doll and Hill (1966))
. list
```

|  | agecat | smokes | deaths | pyears |
| ---: | ---: | ---: | ---: | ---: |
| 1. | $35-44$ | 1 | 32 | 52,407 |
| 2. | $45-54$ | 1 | 104 | 43,248 |
| 3. | $55-64$ | 1 | 206 | 28,612 |
| 4. | $65-74$ | 1 | 186 | 12,663 |
| 5. | $75-84$ | 1 | 102 | 5,317 |
| 6. | $35-44$ | 0 | 2 | 18,790 |
| 7. | $45-54$ | 0 | 12 | 10,673 |
| 8. | $55-64$ | 0 | 28 | 5,710 |
| 9. | $65-74$ | 0 | 28 | 2,585 |
| 10. | $75-84$ | 0 | 31 | 1,462 |

The most "natural" analysis of these data would begin by introducing indicator variables for each age category and one indicator for smoking:

```
. poisson deaths smokes i.agecat, exposure(pyears) irr
Iteration 0: Log likelihood = -33.823284
Iteration 1: Log likelihood = -33.600471
Iteration 2: Log likelihood = -33.600153
Iteration 3: Log likelihood = -33.600153
```

| Poisson regression | Number of obs $=10$ |
| :--- | :--- |
|  | LR chi2 $(5)$ |
|  | $=922.93$ |
| Log likelihood $=-33.600153$ | Prob $>$ chi2 $=0.0000$ |
|  | Pseudo R2 |


| deaths | IRR | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| :---: | ---: | :---: | ---: | :---: | ---: | ---: |
| smokes | 1.425519 | .1530638 | 3.30 | 0.001 | 1.154984 | 1.759421 |
| agecat |  |  |  |  |  |  |
| $45-54$ | 4.410584 | .8605197 | 7.61 | 0.000 | 3.009011 | 6.464997 |
| $55-64$ | 13.8392 | 2.542638 | 14.30 | 0.000 | 9.654328 | 19.83809 |
| $65-74$ | 28.51678 | 5.269878 | 18.13 | 0.000 | 19.85177 | 40.96395 |
| $75-84$ | 40.45121 | 7.775511 | 19.25 | 0.000 | 27.75326 | 58.95885 |
| _cons | .0003636 | .0000697 | -41.30 | 0.000 | .0002497 | .0005296 |
| ln(pyears) | 1 | (exposure) |  |  |  |  |

Note: _cons estimates baseline incidence rate.
In the above, we specified irr to obtain IRRs. We estimate that smokers have 1.43 times the mortality rate of nonsmokers. See, however, example 1 in $[R]$ poisson postestimation.

## Stored results

poisson stores the following in e():
Scalars

```
e(N)
    e(k)
    e(k_eq)
    e(k_eq_model)
    e(k_dv)
    e(df_m)
    e(r2_p)
    e(ll)
    e(ll_0)
    e(N_clust)
    e(chi2)
    e(p)
    e(rank)
    e(ic)
    e(rc)
    e(converged)
```

Macros
Matrices

Functions
e(sample)
number of observations
number of parameters
number of equations in $e(b)$
number of equations in overall model test
number of dependent variables
model degrees of freedom
pseudo- $R^{2}$
log likelihood
log likelihood, constant-only model
number of clusters
$\chi^{2}$
$p$-value for model test
rank of $e(V)$
number of iterations
return code
1 if converged, 0 otherwise
Macros

```
        e(cmd)
        e(cmdline)
    e(depvar)
    e(wtype)
    e(wexp)
e(title)
e(clustvar)
e(offset)
e(chi2type)
e(vce)
e(vcetype)
    e(opt)
    e(which)
    e(ml_method)
    e(user)
    e(technique)
    e(properties)
    e(estat_cmd)
    e(predict)
    e(asbalanced)
    e(asobserved)
```

poisson
command as typed
name of dependent variable
weight type
weight expression
title in estimation output
name of cluster variable
linear offset variable
Wald or LR; type of model $\chi^{2}$ test
vcetype specified in vce()
title used to label Std. err.
type of optimization
max or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement estat
program used to implement predict
factor variables fvset as asbalanced
factor variables fvset as asobserved

Matrices

```
e(b)
e(Cns)
e(ilog)
e(gradient)
e(V)
    e(V_modelbased)
```

coefficient vector
constraints matrix
iteration $\log$ (up to 20 iterations)
gradient vector
variance-covariance matrix of the estimators
model-based variance
Functions
e(sample)
marks estimation sample

In addition to the above, the following is stored in r() :

```
Matrices
    r(table)
```

matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

The log likelihood (with weights $w_{j}$ and offsets) is given by

$$
\begin{aligned}
\operatorname{Pr}(Y=y) & =\frac{e^{-\lambda} \lambda^{y}}{y!} \\
\xi_{j} & =\mathbf{x}_{j} \boldsymbol{\beta}+\operatorname{offset}_{j} \\
f\left(y_{j}\right) & =\frac{e^{-\exp \left(\xi_{j}\right)} e^{\xi_{j} y_{j}}}{y_{j}!} \\
\ln L & =\sum_{j=1}^{n} w_{j}\left\{-e^{\xi_{j}}+\xi_{j} y_{j}-\ln \left(y_{j}!\right)\right\}
\end{aligned}
$$

This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce(robust) and vce(cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.
poisson also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

Siméon-Denis Poisson (1781-1840) was a French mathematician and physicist who contributed to several fields: his name is perpetuated in Poisson brackets, Poisson's constant, Poisson's differential equation, Poisson's integral, and Poisson's ratio. Among many other results, he produced a version of the law of large numbers. His rather misleadingly titled Recherches sur la probabilité des jugements embraces a complete treatise on probability, as the subtitle indicates, including what is now known as the Poisson distribution. That, however, was discovered earlier by the Huguenot-British mathematician Abraham de Moivre (1667-1754).

## References

Bru, B. 2001. Siméon-Denis Poisson. In Statisticians of the Centuries, ed. C. C. Heyde and E. Seneta, 123-126. New York: Springer.

Cameron, A. C., and P. K. Trivedi. 2013. Regression Analysis of Count Data. 2nd ed. New York: Cambridge University Press.
——. 2022. Microeconometrics Using Stata. 2nd ed. College Station, TX: Stata Press.
Chatterjee, S., and A. S. Hadi. 2012. Regression Analysis by Example. 5th ed. New York: Wiley.
Clarke, R. D. 1946. An application of the Poisson distribution. Journal of the Institute of Actuaries 72: 481. https://doi.org/10.1017/S0020268100035435.
Cochran, W. G. 1940. The analysis of variance when experimental errors follow the Poisson or binomial laws. Annals of Mathematical Statistics 11: 335-347. https://doi.org/10.1214/aoms/1177731871.

- 1982. Contributions to Statistics. New York: Wiley.

Coleman, J. S. 1964. Introduction to Mathematical Sociology. New York: Free Press.
Cummings, T. H., J. W. Hardin, A. C. McLain, J. R. Hussey, K. J. Bennett, and G. M. Wingood. 2015. Modeling heaped count data. Stata Journal 15: 457-479.

Deb, P., E. C. Norton, and W. G. Manning. 2017. Health Econometrics Using Stata. College Station, TX: Stata Press.
Doll, R., and A. B. Hill. 1966. Mortality of British doctors in relation to smoking: Observations on coronary thrombosis. Journal of the National Cancer Institute, Monographs 19: 205-268.

Gould, W. W. 2011. Use poisson rather than regress; tell a friend. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2011/08/22/use-poisson-rather-than-regress-tell-a-friend/.

Harris, T., Z. Yang, and J. W. Hardin. 2012. Modeling underdispersed count data with generalized Poisson regression. Stata Journal 12: 736-747.
Hilbe, J. M. 2014. Modeling Count Data. New York: Cambridge University Press.
Jochmans, K., and V. Verardi. 2020. Fitting exponential regression models with two-way fixed effects. Stata Journal 20: 468-480.
Johnson, N. L., A. W. Kemp, and S. Kotz. 2005. Univariate Discrete Distributions. 3rd ed. New York: Wiley.
Lash, T. L., T. J. VanderWeele, S. Haneuse, and K. J. Rothman. 2021. Modern Epidemiology. 4th ed. Philadelphia: Wolters Kluwer.

Long, J. S. 1997. Regression Models for Categorical and Limited Dependent Variables. Thousand Oaks, CA: Sage.
Long, J. S., and J. Freese. 2001. Predicted probabilities for count models. Stata Journal 1: 51-57.
__. 2014. Regression Models for Categorical Dependent Variables Using Stata. 3rd ed. College Station, TX: Stata Press.

McNeil, D. 1996. Epidemiological Research Methods. Chichester, UK: Wiley.
Miranda, A., and S. Rabe-Hesketh. 2006. Maximum likelihood estimation of endogenous switching and sample selection models for binary, ordinal, and count variables. Stata Journal 6: 285-308.

Newman, S. C. 2001. Biostatistical Methods in Epidemiology. New York: Wiley.
Poisson, S. D. 1837. Recherches sur la probabilité des jugements en matière criminelle et en matière civile: précédées des règles générales du calcul des probabilités. Paris: Bachelier.

Raciborski, R. 2011. Right-censored Poisson regression model. Stata Journal 11: 95-105.
Rutherford, E., J. Chadwick, and C. D. Ellis. 1930. Radiations from Radioactive Substances. Cambridge: Cambridge University Press.

Rutherford, M. J., P. C. Lambert, and J. Thompson. 2010. Age-period-cohort modeling. Stata Journal 10: 606-627.
Sasieni, P. D. 2012. Age-period-cohort models in Stata. Stata Journal 12: 45-60.
Schonlau, M. 2005. Boosted regression (boosting): An introductory tutorial and a Stata plugin. Stata Journal 5: 330-354.

Selvin, S. 2004. Statistical Analysis of Epidemiologic Data. 3rd ed. New York: Oxford University Press.
——. 2011. Statistical Tools for Epidemiologic Research. New York: Oxford University Press.
Thorndike, F. 1926. Applications of Poisson's probability summation. Bell System Technical Journal 5: 604-624. https://doi.org/10.1002/j.1538-7305.1926.tb00126.x.
von Bortkiewicz, L. 1898. Das Gesetz der Kleinen Zahlen. Leipzig: Teubner.
Xu, X., and J. W. Hardin. 2016. Regression models for bivariate count outcomes. Stata Journal 16: 301-315.

## Also see

[R] poisson postestimation - Postestimation tools for poisson
[ R ] glm - Generalized linear models
[R] heckpoisson - Poisson regression with sample selection
[R] nbreg - Negative binomial regression
[R] npregress kernel - Nonparametric kernel regression
[R] npregress series - Nonparametric series regression
[R] tpoisson - Truncated Poisson regression
[R] zip - Zero-inflated Poisson regression
[BAYES] bayes: poisson - Bayesian Poisson regression
[FMM] fmm: poisson - Finite mixtures of Poisson regression models
[LASSO] Lasso intro - Introduction to lasso
[ME] mepoisson - Multilevel mixed-effects Poisson regression
[MI] Estimation - Estimation commands for use with mi estimate
[SVY] svy estimation - Estimation commands for survey data
[XT] xtpoisson - Fixed-effects, random-effects, and population-averaged Poisson models
[U] 20 Estimation and postestimation commands

## Title

poisson postestimation - Postestimation tools for poisson

| Postestimation commands | predict | margins | estat |
| :--- | :--- | :--- | :--- |
| Remarks and examples | Stored results | Methods and formulas | Reference |

Also see

## Postestimation commands

The following postestimation commands are of special interest after poisson:

| Command | Description |
| :--- | :--- |
| estat gof | goodness-of-fit test |
| lassogof | calculate goodness-of-fit predictions |

estat gof is not appropriate with svy estimation results.
The following standard postestimation commands are also available:

| Command | Description |
| :--- | :--- |
| contrast | contrasts and ANOVA-style joint tests of estimates |
| estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian in- <br> formation criteria (AIC, CAIC, AICc, and BIC) |

estat summarize summary statistics for the estimation sample
estat vce variance-covariance matrix of the estimators (VCE)
estat (svy) postestimation statistics for survey data
estimates cataloging estimation results
etable table of estimation results
*forecast dynamic forecasts and simulations
*hausman Hausman's specification test
lincom point estimates, standard errors, testing, and inference for linear combinations of coefficients
linktest link test for model specification

* lrtest likelihood-ratio test
margins marginal means, predictive margins, marginal effects, and average marginal effects marginsplot graph the results from margins (profile plots, interaction plots, etc.)
nlcom point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients
predict number of events, incidence rates, probabilities, etc.
predictnl point estimates, standard errors, testing, and inference for generalized predictions
pwcompare pairwise comparisons of estimates
suest seemingly unrelated estimation
test Wald tests of simple and composite linear hypotheses
testnl Wald tests of nonlinear hypotheses

[^47]
## predict

## Description for predict

predict creates a new variable containing predictions such as numbers of events, incidence rates, probabilities, linear predictions, standard errors, and the equation-level score.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

predict [type] newvar [if] [in] [, statistic nooffset]
statistic Description

```
Main
```

n
ir
pr( $n$ )
$\operatorname{pr}(a, b)$
xb
stdp
score
number of events; the default
incidence rate
probability $\operatorname{Pr}\left(y_{j}=n\right)$
probability $\operatorname{Pr}\left(a \leq y_{j} \leq b\right)$
linear prediction
standard error of the linear prediction
first derivative of the log likelihood with respect to $\mathbf{x}_{j} \boldsymbol{\beta}$

These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

## Options for predict

$\qquad$ Main
n , the default, calculates the predicted number of events, which is $\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}\right)$ if neither offset () nor exposure() was specified when the model was fit; $\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}+\right.$ offset $\left._{j}\right)$ if offset() was specified; or $\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}\right) \times$ exposure $_{j}$ if exposure() was specified.
ir calculates the incidence rate $\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}\right)$, which is the predicted number of events when exposure is 1 . Specifying ir is equivalent to specifying $n$ when neither offset() nor exposure() was specified when the model was fit.
$\operatorname{pr}(n)$ calculates the probability $\operatorname{Pr}\left(y_{j}=n\right)$, where $n$ is a nonnegative integer that may be specified as a number or a variable.
$\operatorname{pr}(a, b)$ calculates the probability $\operatorname{Pr}\left(a \leq y_{j} \leq b\right)$, where $a$ and $b$ are nonnegative integers that may be specified as numbers or variables;
$b$ missing ( $b \geq$.) means $+\infty$;
$\operatorname{pr}(20,$.$) calculates \operatorname{Pr}\left(y_{j} \geq 20\right)$;
$\operatorname{pr}(20, b)$ calculates $\operatorname{Pr}\left(y_{j} \geq 20\right)$ in observations for which $b \geq$. and calculates $\operatorname{Pr}\left(20 \leq y_{j} \leq b\right)$ elsewhere.
$\operatorname{pr}(., b)$ produces a syntax error. A missing value in an observation of the variable $a$ causes a missing value in that observation for $\operatorname{pr}(a, b)$.
xb calculates the linear prediction, which is $\mathbf{x}_{j} \boldsymbol{\beta}$ if neither offset () nor exposure() was specified; $\mathbf{x}_{j} \boldsymbol{\beta}+$ offset $_{j}$ if offset () was specified; or $\mathbf{x}_{j} \boldsymbol{\beta}+\ln$ (exposure ${ }_{j}$ ) if exposure() was specified; see nooffset below.
stdp calculates the standard error of the linear prediction.
score calculates the equation-level score, $\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)$.
nooffset is relevant only if you specified offset() or exposure() when you fit the model. It modifies the calculations made by predict so that they ignore the offset or exposure variable; the linear prediction is treated as $\mathbf{x}_{j} \boldsymbol{\beta}$ rather than as $\mathbf{x}_{j} \boldsymbol{\beta}+$ offset $_{j}$ or $\mathbf{x}_{j} \boldsymbol{\beta}+\ln \left(\right.$ exposure $\left._{j}\right)$. Specifying predict ..., nooffset is equivalent to specifying predict ..., ir.

## margins

## Description for margins

margins estimates margins of response for numbers of events, incidence rates, probabilities, and linear predictions.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...) ...] [options]
```

statistic Description

| n | number of events; the default |
| :--- | :--- |
| ir | incidence rate |
| $\operatorname{pr}(n)$ | probability $\operatorname{Pr}\left(y_{j}=n\right)$ |
| $\operatorname{pr}(a, b)$ | probability $\operatorname{Pr}\left(a \leq y_{j} \leq b\right)$ |
| xb | linear prediction |
| stdp | not allowed with margins |
| score | not allowed with margins |

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see $[R]$ margins.

## estat

## Description for estat

estat gof performs a goodness-of-fit test of the model. Both the deviance statistic and the Pearson statistic are reported. If the tests are significant, the Poisson regression model is inappropriate.

## Menu for estat

Statistics $>$ Postestimation

## Syntax for estat

estat gof
collect is allowed; see [U] 11.1.10 Prefix commands.

## Remarks and examples

## Example 1

Continuing with example 2 of [R] poisson, we use estat gof to determine whether the model fits the data well.

```
. use https://www.stata-press.com/data/r18/dollhill3
(Doll and Hill (1966))
. poisson deaths smokes i.agecat, exp(pyears) irr
    (output omitted)
. estat gof
    Deviance goodness-of-fit = 12.13237
    Prob > chi2(4) = 0.0164
    Pearson goodness-of-fit = 11.15533
    Prob > chi2(4) = 0.0249
```

The deviance goodness-of-fit test tells us that, given the model, we can reject the hypothesis that these data are Poisson distributed at the $1.64 \%$ significance level. The Pearson goodness-of-fit test tells us that we can reject the hypothesis at the $2.49 \%$ significance level.

So let us now back up and be more careful. We can most easily obtain the incidence-rate ratios within age categories by using ir; see [R] Epitab:
. ir deaths smokes pyears, by (agecat) nohet
Stratified incidence-rate analysis

| Age category | IRR | [95\% conf. interval] | M-H weight |  |
| ---: | ---: | ---: | ---: | ---: |
| $35-44$ | 5.736638 | 1.463557 | 49.40468 | 1.472169 (exact) |
| $45-54$ | 2.138812 | 1.173714 | 4.272545 | 9.624747 (exact) |
| $55-64$ | 1.46824 | .9863624 | 2.264107 | 23.34176 (exact) |
| $65-74$ | 1.35606 | .9081925 | 2.096412 | 23.25315 (exact) |
| $75-84$ | .9047304 | .6000757 | 1.399687 | 24.31435 (exact) |
| Crude | 1.719823 | 1.391992 | 2.14353 |  |
| M-H combined | 1.424682 | 1.154703 | 1.757784 |  |

We find that the mortality incidence ratios are greatly different within age category, being highest for the youngest categories and actually dropping below 1 for the oldest. (In the last case, we might argue that those who smoke and who have not died by age 75 are self-selected to be particularly robust.)

Seeing this, we will now parameterize the smoking effects separately for each category, although we will begin by constraining the smoking effects on third and fourth age categories to be equivalent:


Note: _cons estimates baseline incidence rate.

- estat gof

| Deviance goodness-of-fit | $=$ | .0773491 |
| :--- | :--- | ---: |
| Prob $>$ chi2 (1) | $=$ | 0.7809 |
| Pearson goodness-of-fit | $=$ | .0773885 |
| Prob > chi2(1) | $=$ | 0.7809 |

The goodness-of-fit is now small; we are no longer running roughshod over the data. Let us now consider simplifying the model. The point estimate of the incidence-rate ratio for smoking in age category 1 is much larger than that for smoking in age category 2 , but the confidence interval for smokes\#1.agecat is similarly wide. Is the difference real?

```
. test smokes#1.agecat = smokes#2.agecat
( 1) [deaths]1b.agecat#c.smokes - [deaths]2.agecat#c.smokes = 0
    chi2( 1) = 
```

The point estimates of the incidence-rate ratio for smoking in the 35-44 age category is much larger than that for smoking in the $45-54$ age category, but there is insufficient data, and we may be
observing random differences. With that success, might we also combine the smokers in the third and fourth categories with those in the first and second categories?

```
. test smokes#2.agecat = smokes#3.agecat, accum
( 1) [deaths]1b.agecat#c.smokes - [deaths]2.agecat#c.smokes = 0
( 2) [deaths]2.agecat#c.smokes - [deaths]3.agecat#c.smokes = 0
    chi2( 2) = 4.73
    Prob > chi2 = 0.0938
```

Combining the first four categories may be overdoing it - the $9.38 \%$ significance level is enough to stop us, although others may disagree.

Thus, we now fit our final model:


Note: _cons estimates baseline incidence rate.

The above strikes us as a fair representation of the data. The probabilities of observing the deaths seen in these data are estimated using the following predict command:

```
. predict p, pr(0, deaths)
```

. list deaths p

|  | deaths | p |
| ---: | ---: | ---: |
| 1. | 32 | .6891766 |
| 2. | 104 | .4456625 |
| 3. | 206 | .5455328 |
| 4. | 186 | .4910622 |
| 5. | 102 | .5263011 |
| 6. | 2 | .227953 |
| 7. | 12 | .7981917 |
| 8. | 28 | .4772961 |
| 9. | 28 | .6227565 |
| 10. | 31 | .5475718 |

The probability $\operatorname{Pr}(y \leq$ deaths $)$ ranges from 0.23 to 0.80 .

## Stored results

estat gof after poisson stores the following in $r()$ :
Scalars
$r(d f) \quad$ degrees of freedom (Pearson and deviance)
r(chi2_p)
$\chi^{2}$ (Pearson)
$r\left(\right.$ chi2_d) $\quad \chi^{2}$ (deviance)
$r\left(p \_p\right) \quad p$-value for $\chi^{2}$ test (Pearson)
$r\left(\right.$ p_d) $\quad p$-value for $\chi^{2}$ test (deviance)

## Methods and formulas

In the following, we use the same notation as in $[R]$ poisson.
The equation-level score is given by

$$
\operatorname{score}(\mathbf{x} \boldsymbol{\beta})_{j}=y_{j}-e^{\xi_{j}}
$$

The deviance ( D ) and Pearson ( P ) goodness-of-fit statistics are given by

$$
\begin{aligned}
\ln L_{\max } & =\sum_{j=1}^{n} w_{j}\left[y_{j}\left\{\ln \left(y_{j}\right)-1\right\}-\ln \left(y_{j}!\right)\right] \\
\chi_{D}^{2} & =-2\left\{\ln L-\ln L_{\max }\right\} \\
\chi_{P}^{2} & =\sum_{j=1}^{n} \frac{w_{j}\left(y_{j}-e^{\xi_{j}}\right)^{2}}{e^{\xi_{j}}}
\end{aligned}
$$

## Reference

Manjón, M., and O. Martínez. 2014. The chi-squared goodness-of-fit test for count-data models. Stata Journal 14: 798-816.

## Also see

[R] poisson - Poisson regression
[LASSO] lassogof - Goodness of fit after lasso for prediction
[U] 20 Estimation and postestimation commands

## Title

postest — Postestimation Selector

Description Menu Syntax Remarks and examples Also see

## Description

Launch the Postestimation Selector window. The window contains a list of all postestimation features that are available for the currently active estimation results. To launch the dialog box for an item in the list, select an item and click on Launch. The list is automatically updated when estimation commands are run or estimates are restored from memory or disk.

## Menu

Statistics $>$ Postestimation

## Syntax

## postest

## Remarks and examples

Remarks are presented under the following headings:

Overview<br>Video example

## Overview

Stata uses an estimation followed by postestimation analysis paradigm. You type regress ... to fit a regression model, then you type test ... to test linear relationships among the estimated parameters, or you type contrast ... to compare marginal means, or you type rvfplot to see a residual-versus-fitted plot, or you type one of a myriad of other postestimation commands. This is an extension of Stata's "type a little, get a little" concept. The Postestimation Selector exposes this type of postestimation analysis to those who prefer to use the dialog boxes to fit and analyze models, or at least they sometimes prefer the dialogs when exploring their data. We might call this "click a little, get a little".

The Postestimation Selector knows what is available after any estimation command. It shows the full list of postestimation features that are available after any estimation, and it shows only those that are available. For example, if you fit a linear regression, you can choose from 59 postestimation analyses, including "Likelihood-ratio test comparing models". If, however, you fit that linear regression using survey estimation, the likelihood-ratio test is not available because that test has no meaning for survey estimation. You will, however, see 8 new postestimation features for survey-data analysis, including "Design and misspecification effects".

If you are using the menus and dialogs, we recommend you launch the Postestimation Selector and just leave it up-select Statistics > Postestimation, or type postest at the command line. All available postestimation features for whatever model you are analyzing will then be just a click away.

Here is what the Selector looks like after a linear regression,


And, here is what it looks like with all the groupings expanded.


## Video example

Postestimation Selector

## Also see

[U] 20 Estimation and postestimation commands

## Title

predict - Obtain predictions, residuals, etc., after estimation

| Description | Quick start | Menu for predict | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Methods and formulas | References |
| Also see |  |  |  |

## Description

predict calculates predictions, residuals, influence statistics, and the like after estimation. Exactly what predict can do is determined by the previous estimation command; command-specific options are documented with each estimation command. Regardless of command-specific options, the actions of predict share certain similarities across estimation commands:

1. predict newvar creates newvar containing "predicted values"-numbers related to the $E\left(y_{j} \mid \mathbf{x}_{j}\right)$. For instance, after linear regression, predict newvar creates $\mathbf{x}_{j} \mathbf{b}$ and, after probit, creates the probability $\Phi\left(\mathbf{x}_{j} \mathbf{b}\right)$.
2. predict newvar, xb creates newvar containing $\mathbf{x}_{j} \mathbf{b}$. This may be the same result as option 1 (for example, linear regression) or different (for example, probit), but regardless, option xb is allowed.
3. predict newvar, stdp creates newvar containing the standard error of the linear prediction $\mathbf{x}_{j} \mathbf{b}$.
4. predict newvar, other_options may create newvar containing other useful quantities; see help or the reference manual entry for the particular estimation command to find out about other available options.
5. nooffset added to any of the above commands requests that the calculation ignore any offset or exposure variable specified by including the offset (varname ${ }_{o}$ ) or exposure (varname ${ }_{e}$ ) option when you fit the model.
predict can be used to make in-sample or out-of-sample predictions:
6. predict calculates the requested statistic for all possible observations, whether they were used in fitting the model or not. predict does this for standard options 1 through 3 and generally does this for estimator-specific options 4.
7. predict newvar if e(sample), ... restricts the prediction to the estimation subsample.
8. Some statistics make sense only with respect to the estimation subsample. In such cases, the calculation is automatically restricted to the estimation subsample, and the documentation for the specific option states this. Even so, you can still specify if e (sample) if you are uncertain.
9. predict can make out-of-sample predictions even using other datasets. In particular, you can
. use ds1

- (fit a model)
. use two
/* another dataset */
. predict yhat, ... /* fill in the predictions */


## Quick start

Create newvar1 containing the default prediction for the previous estimation command predict newvar1

Create newvar2 containing the linear prediction
predict newvar2, xb
Same as above, but only for observations used in the previous estimation
predict newvar2 if e(sample), xb
Create newvar3, the default prediction for the first equation in a multiple-equation model predict newvar3, equation(\#1)

Same as above when y1 is the name of the first equation
predict newvar3, equation(y1)
Note: For a complete list of options available with predict after an estimation command, see the corresponding postestimation entry.

## Menu for predict

Statistics $>$ Postestimation

## Syntax

After single-equation (SE) models
predict [type] newvar [if] [in] [, single_options]

After multiple-equation (ME) models

```
predict [type] newvar [if] [in] [, multiple_options]
predict [type] stub* [if] [in], scores
```

single_options
Description
Main
xb calculate linear prediction
stdp calculate standard error of the prediction
score
calculate first derivative of the $\log$ likelihood with respect to $\mathbf{x}_{j} \boldsymbol{\beta}$
Options
nooffset
other_options
multiple_options
Description
Main
equation (eqno [, eqno]) specify equations
xb
stdp
stddp
Options
nooffset
other_options
calculate linear prediction command-specific options
ignore any offset() or exposure() variable command-specific options
calculate standard error of the prediction
calculate the difference in linear predictions
ignore any offset () or exposure() variable

## Options

$\qquad$ Main
xb calculates the linear prediction from the fitted model. That is, all models can be thought of as estimating a set of parameters $b_{1}, b_{2}, \ldots, b_{k}$, and the linear prediction is $\widehat{y}_{j}=b_{1} x_{1 j}+b_{2} x_{2 j}+$ $\cdots+b_{k} x_{k j}$, often written in matrix notation as $\widehat{\mathbf{y}}_{j}=\mathbf{x}_{j} \mathbf{b}$. For linear regression, the values $\widehat{y}_{j}$ are called the predicted values or, for out-of-sample predictions, the forecast. For logit and probit, for example, $\widehat{y}_{j}$ is called the logit or probit index.
$x_{1 j}, x_{2 j}, \ldots, x_{k j}$ are obtained from the data currently in memory and do not necessarily correspond to the data on the independent variables used to fit the model (obtaining $b_{1}, b_{2}, \ldots, b_{k}$ ).
stdp calculates the standard error of the linear prediction. Here the prediction means the same thing as the "index", namely, $\mathbf{x}_{j} \mathbf{b}$. The statistic produced by stdp can be thought of as the standard error of the predicted expected value, or mean index, for the observation's covariate pattern. The standard error of the prediction is also commonly referred to as the standard error of the fitted value. The calculation can be made in or out of sample.
stddp is allowed only after you have previously fit a multiple-equation model. The standard error of the difference in linear predictions $\left(\mathbf{x}_{1 j} \mathbf{b}-\mathbf{x}_{2 j} \mathbf{b}\right)$ between equations 1 and 2 is calculated. This option requires that equation (eqno, eqno 2 ) be specified.
score calculates the equation-level score, $\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)$. Here $\ln L$ refers to the log-likelihood function.
scores is the ME model equivalent of the score option, resulting in multiple equation-level score variables. An equation-level score variable is created for each equation in the model; ancillary parameters-such as $\ln \sigma$ and $\operatorname{atanh} \rho$ —make up separate equations.
equation (eqno [, eqno]) -synonym outcome() -is relevant only when you have previously fit a multiple-equation model. It specifies the equation to which you are referring.
equation() is typically filled in with one eqno-it would be filled in that way with options xb and stdp, for instance. equation(\#1) would mean the calculation is to be made for the first equation, equation(\#2) would mean the second, and so on. You could also refer to the equations by their names. equation(income) would refer to the equation named income and equation(hours) to the equation named hours.
If you do not specify equation(), results are the same as if you specified equation(\#1).
Other statistics, such as stddp, refer to between-equation concepts. In those cases, you might specify equation (\#1, \#2) or equation(income, hours). When two equations must be specified, equation() is required.

## Options

nooffset may be combined with most statistics and specifies that the calculation should be made, ignoring any offset or exposure variable specified when the model was fit.

This option is available, even if it is not documented for predict after a specific command. If neither the offset (varname $e_{o}$ ) option nor the exposure (varname ${ }_{e}$ ) option was specified when the model was fit, specifying nooffset does nothing.
other_options refers to command-specific options that are documented with each command.

## Remarks and examples

Remarks are presented under the following headings:
Estimation-sample predictions
Out-of-sample predictions
Residuals
Single-equation (SE) models
SE model scores
Multiple-equation (ME) models
ME model scores
Most of the examples are presented using linear regression, but the general syntax is applicable to all estimators.

You can think of any estimation command as estimating a set of coefficients $b_{1}, b_{2}, \ldots, b_{k}$ corresponding to the variables $x_{1}, x_{2}, \ldots, x_{k}$, along with a (possibly empty) set of ancillary statistics $\gamma_{1}, \gamma_{2}, \ldots, \gamma_{m}$. All estimation commands store the $b_{i} \mathrm{~s}$ and $\gamma_{i}$ s. predict accesses that stored information and combines it with the data currently in memory to make various calculations. For instance, predict can calculate the linear prediction, $\widehat{y}_{j}=b_{1} x_{1 j}+b_{2} x_{2 j}+\cdots+b_{k} x_{k j}$. The data on which predict makes the calculation can be the same data used to fit the model or a different
dataset-it does not matter. predict uses the stored parameter estimates from the model, obtains the corresponding values of $x$ for each observation in the data, and then combines them to produce the desired result.

## Estimation-sample predictions

## > Example 1

We have a 74-observation dataset on automobiles, including the mileage rating (mpg), the car's weight (weight), and whether the car is foreign (foreign). We fit the model

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
```

. regress mpg weight if foreign

| Source | SS | df | MS | Number of obs | = | 22 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | F (1, 20) | = | 17.47 |
| Model | 427.990298 | 1 | 427.990298 | Prob > F | = | 0.0005 |
| Residual | 489.873338 | 20 | 24.4936669 | R -squared | = | 0.4663 |
| Total | 917.863636 | 21 | 43.7077922 | Adj R-squared Root MSE | = | 0.4396 4.9491 |


| mpg | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| weight | -.010426 | .0024942 | -4.18 | 0.000 | -.0156287 | -.0052232 |
| _cons | 48.9183 | 5.871851 | 8.33 | 0.000 | 36.66983 | 61.16676 |

If we were to type predict pmpg now, we would obtain the linear predictions for all 74 observations. To obtain the predictions just for the sample on which we fit the model, we could type

```
. predict pmpg if e(sample)
(option xb assumed; fitted values)
(52 missing values generated)
```

Here e(sample) is true only for foreign cars because we typed if foreign when we fit the model and because there are no missing values among the relevant variables. If there had been missing values, e(sample) would also account for those.

By the way, the if e(sample) restriction can be used with any Stata command, so we could obtain summary statistics on the estimation sample by typing

```
. summarize if e(sample)
(output omitted)
```


## Out-of-sample predictions

By out-of-sample predictions, we mean predictions extending beyond the estimation sample. In the example above, typing predict pmpg would generate linear predictions using all 74 observations.
predict will work on other datasets, too. You can use a new dataset and type predict to obtain results for that sample.

## > Example 2

Using the same auto dataset, assume that we wish to fit the model

$$
\mathrm{mpg}=\beta_{1} \text { weight }+\beta_{2} \ln (\text { weight })+\beta_{3} \text { foreign }+\beta_{4}
$$

We first create the $\ln ($ weight ) variable, and then type the regress command:

| . use https://www.stata-press.com/data (1978 automobile data) <br> . generate lnweight = ln(weight) <br> . regress mpg weight lnweight foreign |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Source | SS | df | MS | Number of obs |  | 74 |
|  |  |  |  | F (3, 70) |  | 52.36 |
| Model | 1690.27997 | 3 | 563.426657 | Prob > F |  | 0.0000 |
| Residual | 753.179489 | 70 | 10.759707 | R-squared |  | 0.6918 |
|  |  |  |  | Adj R-squared |  | 0.6785 |
| Total | 2443.45946 | 73 | 33.4720474 | Root MSE |  | 3.2802 |


| mpg | Coefficient | Std. err. | t | $\mathrm{P}>\mid \mathrm{t\mid}$ | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | ---: | ---: |
| weight | .003304 | .0038995 | 0.85 | 0.400 | -.0044734 | .0110813 |
| lnweight | -29.59133 | 11.52018 | -2.57 | 0.012 | -52.5676 | -6.615061 |
| foreign | -2.125299 | 1.052324 | -2.02 | 0.047 | -4.224093 | -.0265044 |
| _cons | 248.0548 | 80.37079 | 3.09 | 0.003 | 87.76035 | 408.3493 |

If we typed predict pmpg now, we would obtain predictions for all 74 cars in the current data. Instead, we are going to use a new dataset.
newautos.dta contains the make, weight, and place of manufacture of two cars, the Pontiac Sunbird and the Volvo 260. Let's use the dataset and create the predictions:

```
. use https://www.stata-press.com/data/r18/newautos, clear
(New automobile models)
. list
```

|  | make | weight | foreign |
| :--- | ---: | ---: | ---: |
| 1. | Pont. Sunbird | 2690 | Domestic |
| 2. | Volvo 260 | 3170 | Foreign |
|  |  |  |  |

```
. predict mpg
(option xb assumed; fitted values)
variable lnweight not found
r(111);
```

Things did not work. We typed predict mpg, and Stata responded with the message "variable lnweight not found". predict can calculate predicted values on a different dataset only if that dataset contains the variables that went into the model. Here our dataset does not contain a variable called lnweight. lnweight is just the $\log$ of weight, so we can create it and try again:

```
. generate lnweight = ln(weight)
- predict mpg
(option xb assumed; fitted values)
. list
```

| make | weight | foreign | lnweight | mpg |
| ---: | ---: | ---: | ---: | ---: |
| Pont. Sunbird | 2690 | Domestic | 7.897296 | 23.25097 |
| Volvo 260 | 3170 | Foreign | 8.061487 | 17.85295 |

We obtained our predicted values. The Pontiac Sunbird has a predicted mileage rating of 23.3 mpg , whereas the Volvo 260 has a predicted rating of 17.9 mpg .

## Residuals

## > Example 3

With many estimators, predict can calculate more than predicted values. With most regressiontype estimators, we can, for instance, obtain residuals. Using our regression example, we return to our original data and obtain residuals by typing

```
. use https://www.stata-press.com/data/r18/auto, clear
(1978 automobile data)
. generate lnweight = ln(weight)
. regress mpg weight lnweight foreign
    (output omitted)
. predict double resid, residuals
. summarize resid
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline resid & 74 & \(-1.31 \mathrm{e}-14\) & 3.212091 & -5.453078 & 13.83719
\end{tabular}
```

We could do this without refitting the model. Stata always remembers the last set of estimates, even as we use new datasets.

It was not necessary to type the double in predict double resid, residuals, but we wanted to remind you that you can specify the type of a variable in front of the variable's name; see [U] 11.4.2 Lists of new variables. We made the new variable resid a double rather than the default float.

If we had not specified $\%$ double, the mean of resid would have been roughly $9 \times 10^{-9}$ rather than $1 \times 10^{-14}$. Although $1 \times 10^{-14}$ sounds more precise than $9 \times 10^{-9}$, the difference really does not matter.

For linear regression, predict can also calculate standardized residuals and Studentized residuals with the options rstandard and rstudent; for examples, see [R] regress postestimation.

## Single-equation (SE) models

If you have not read the discussion above on using predict after linear regression, please do so. And predict's default calculation almost always produces a statistic in the same metric as the dependent variable of the fitted model-for example, predicted counts for Poisson regression. In any case, xb can always be specified to obtain the linear prediction.
predict can calculate the standard error of the prediction, which is obtained by using the covariance matrix of the estimators.
$\rangle$ Example 4
After most binary outcome models (for example, logistic, logit, probit, cloglog, scobit), predict calculates the probability of a positive outcome if we do not tell it otherwise. We can specify the xb option if we want the linear prediction (also known as the logit or probit index). The odd abbreviation xb is meant to suggest $\mathbf{x} \boldsymbol{\beta}$. In logit and probit models, for example, the predicted probability is $p=F(\mathbf{x} \boldsymbol{\beta})$, where $F()$ is the logistic or normal cumulative distribution function, respectively.

```
. logistic foreign mpg weight
    (output omitted)
. predict phat
(option pr assumed; Pr(foreign))
. predict idxhat, xb
. summarize foreign phat idxhat
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline foreign & 74 & .2972973 & .4601885 & 0 & 1 \\
phat & 74 & .2972973 & .3052979 & .000729 & .8980594 \\
idxhat & 74 & -1.678202 & 2.321509 & -7.223107 & 2.175845
\end{tabular}
```

Because this is a logit model, we could obtain the predicted probabilities ourselves from the predicted index

```
. generate phat2 = exp(idxhat)/(1+exp(idxhat))
```

but using predict without options is easier.

## > Example 5

For all models, predict attempts to produce a predicted value in the same metric as the dependent variable of the model. We have seen that for dichotomous outcome models, the default statistic produced by predict is the probability of a success. Similarly, for Poisson regression, the default statistic produced by predict is the predicted count for the dependent variable. You can always specify the xb option to obtain the linear combination of the coefficients with an observation's $x$ values (the inner product of the coefficients and $x$ values). For poisson (without an explicit exposure), this is the natural log of the count.

```
. use https://www.stata-press.com/data/r18/airline, clear
. poisson injuries XYZowned
(output omitted)
```

```
. predict injhat
(option n assumed; predicted number of events)
. predict idx, xb
. generate exp_idx = exp(idx)
. summarize injuries injhat exp_idx idx
```

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| injuries | 9 | 7.111111 | 5.487359 | 1 | 19 |
| injhat | 9 | 7.111111 | .8333333 | 6 | 7.666667 |
| exp_idx | 9 | 7.111111 | .8333333 | 6 | 7.666667 |
| idx | 9 | 1.955174 | .1225612 | 1.791759 | 2.036882 |

We note that our "hand-computed" prediction of the count (exp_idx) matches what was produced by the default operation of predict.

If our model has an exposure-time variable, we can use predict to obtain the linear prediction with or without the exposure. Let's verify what we are getting by obtaining the linear prediction with and without exposure, transforming these predictions to count predictions and comparing them with the default count prediction from predict. We must remember to multiply by the exposure time when using predict..., nooffset.

```
. use https://www.stata-press.com/data/r18/airline, clear
. poisson injuries XYZowned, exposure(n)
(output omitted)
. predict double injhat
(option n assumed; predicted number of events)
. predict double idx, xb
. generate double exp_idx = exp(idx)
. predict double idxn, xb nooffset
. generate double exp_idxn = exp(idxn)*n
. summarize injuries injhat exp_idx exp_idxn idx idxn
```

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| injuries | 9 | 7.111111 | 5.487359 | 1 | 19 |
| injhat | 9 | 7.111111 | 3.10936 | 2.919621 | 12.06158 |
| exp_idx | 9 | 7.111111 | 3.10936 | 2.919621 | 12.06158 |
| exp_idxn | 9 | 7.111111 | 3.10936 | 2.919621 | 12.06158 |
| idx | 9 | 1.869722 | .4671044 | 1.071454 | 2.490025 |
| idxn | 9 | 4.18814 | .1904042 | 4.061204 | 4.442013 |

Looking at the identical means and standard deviations for injhat, exp_idx, and exp_idxn, we see that we can reproduce the default computations of predict for poisson estimations. We have also demonstrated the relationship between the count predictions and the linear predictions with and without exposure.

## SE model scores

## $>$ Example 6

With most maximum likelihood estimators, predict can calculate equation-level scores. The first derivative of the $\log$ likelihood with respect to $\mathbf{x}_{j} \boldsymbol{\beta}$ is the equation-level score.

```
. use https://www.stata-press.com/data/r18/auto, clear
(1978 automobile data)
. logistic foreign mpg weight
    (output omitted)
. predict double sc, score
. summarize sc
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline sc & 74 & \(-1.37 e-12\) & .3533133 & -.8760856 & .8821309
\end{tabular}
```

See $[\mathrm{P}]$ _robust and [SVY] Variance estimation for details regarding the role equation-level scores play in linearization-based variance estimators.

## - Technical note

predict after some estimation commands, such as regress and cnsreg, allows the score option as a synonym for the residuals option.

## Multiple-equation (ME) models

If you have not read the above discussion on using predict after SE models, please do so. With the exception of the ability to select specific equations to predict from, the use of predict after ME models follows almost the same form that it does for SE models.

## > Example 7

The details of prediction statistics that are specific to particular ME models are documented with the estimation command. If you are using ME commands that do not have separate discussions on obtaining predictions, read Obtaining predicted values in [R] mlogit postestimation, even if your interest is not in multinomial logistic regression. As a general introduction to the ME models, we will demonstrate predict after sureg:

```
. use https://www.stata-press.com/data/r18/auto, clear
(1978 automobile data)
. sureg (price foreign displ) (weight foreign length)
Seemingly unrelated regression
```


sureg estimated two equations, one called price and the other weight; see $[R]$ sureg.

```
. predict pred_p, equation(price)
(option xb assumed; fitted values)
. predict pred_w, equation(weight)
(option xb assumed; fitted values)
```

. summarize price pred_p weight pred_w

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| price | 74 | 6165.257 | 2949.496 | 3291 | 15906 |
| pred_p | 74 | 6165.257 | 1678.805 | 2664.81 | 10485.33 |
| weight | 74 | 3019.459 | 777.1936 | 1760 | 4840 |
| pred_w | 74 | 3019.459 | 726.0468 | 1501.602 | 4447.996 |

You may specify the equation by name, as we did above, or by number: equation(\#1) means the same thing as equation(price) in this case.

## ME model scores

## > Example 8

For ME models, predict allows you to specify a stub when generating equation-level score variables. predict generates new variables using this stub by appending an equation index. Depending upon the command, the index will start with 0 or 1 . Here is an example where predict starts indexing the score variables with 0 .

```
. ologit rep78 mpg weight
(output omitted)
. predict double sc*, scores
. summarize sc*
```

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| sc0 | 69 | $-1.33 e-11$ | .5337363 | -.9854088 | .921433 |
| sc1 | 69 | $-7.69 \mathrm{e}-13$ | .186919 | -.2738537 | .9854088 |
| sc2 | 69 | $-2.87 \mathrm{e}-11$ | .4061637 | -.5188487 | 1.130178 |
| sc3 | 69 | $-1.04 \mathrm{e}-10$ | .5315368 | -1.067351 | .8194842 |
| sc4 | 69 | $1.47 \mathrm{e}-10$ | .360525 | -.921433 | .6140182 |

Although it involves much more typing, we could also specify the new variable names individually.

```
. predict double (sc_xb sc_1 sc_2 sc_3 sc_4), scores
```

. summarize sc_*

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| sc_xb | 69 | $-1.33 \mathrm{e}-11$ | .5337363 | -.9854088 | .921433 |
| sc_1 | 69 | $-7.69 \mathrm{e}-13$ | .186919 | -.2738537 | .9854088 |
| sc_2 | 69 | $-2.87 \mathrm{e}-11$ | .4061637 | -.5188487 | 1.130178 |
| sc_3 | 69 | $-1.04 \mathrm{e}-10$ | .5315368 | -1.067351 | .8194842 |
| sc_4 | 69 | $1.47 \mathrm{e}-10$ | .360525 | -.921433 | .6140182 |

## Methods and formulas

Denote the previously estimated coefficient vector as $\mathbf{b}$ and its estimated variance matrix as $\mathbf{V}$. predict works by recalling various aspects of the model, such as $\mathbf{b}$, and combining that information with the data currently in memory. Let's write $\mathbf{x}_{j}$ for the $j$ th observation currently in memory.

The predicted value ( xb option) is defined as $\widehat{y}_{j}=\mathbf{x}_{j} \mathbf{b}+$ offset $_{j}$
The standard error of the prediction (the stdp option) is defined as $s_{p_{j}}=\sqrt{\mathbf{x}_{j} \mathbf{V} \mathbf{x}_{j}^{\prime}}$
The standard error of the difference in linear predictions between equations 1 and 2 is defined as

$$
s_{d p_{j}}=\left\{\left(\mathbf{x}_{1 j},-\mathbf{x}_{2 j}, \mathbf{0}, \ldots, \mathbf{0}\right) \mathbf{V}\left(\mathbf{x}_{1 j},-\mathbf{x}_{2 j}, \mathbf{0}, \ldots, \mathbf{0}\right)^{\prime}\right\}^{\frac{1}{2}}
$$

See the individual estimation commands for information about calculating command-specific predict statistics.

## References

Huber, C. 2013a. Multilevel linear models in Stata, part 1: Components of variance. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2013/02/04/multilevel-linear-models-in-stata-part-1-components-of-variance/.
——. 2013b. Multilevel linear models in Stata, part 2: Longitudinal data. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2013/02/18/multilevel-linear-models-in-stata-part-2-longitudinal-data/.
Skeels, C. L., and L. W. Taylor. 2015. Prediction in linear index models with endogenous regressors. Stata Journal 15: 627-644.


#### Abstract

Also see $[\mathrm{R}]$ predictnl - Obtain nonlinear predictions, standard errors, etc., after estimation [P] _predict - Obtain predictions, residuals, etc., after estimation programming command [U] 20 Estimation and postestimation commands


## Title

predictnl - Obtain nonlinear predictions, standard errors, etc., after estimation

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Methods and formulas | References |
| Also see |  |  |  |

## Description

predictnl calculates (possibly) nonlinear predictions after any Stata estimation command and optionally calculates the variances, standard errors, Wald test statistics, $p$-values, and confidence limits for these predictions. Unlike its companion nonlinear postestimation commands testnl and nlcom, predictnl generates functions of the data (that is, predictions), not scalars. The quantities generated by predictnl are thus vectorized over the observations in the data.

Consider some general prediction, $g\left(\boldsymbol{\theta}, \mathbf{x}_{i}\right)$, for $i=1, \ldots, n$, where $\boldsymbol{\theta}$ are the model parameters and $\mathbf{x}_{i}$ are some data for the $i$ th observation; $\mathbf{x}_{i}$ is assumed fixed. Typically, $g\left(\boldsymbol{\theta}, \mathbf{x}_{i}\right)$ is estimated by $g\left(\widehat{\boldsymbol{\theta}}, \mathbf{x}_{i}\right)$, where $\widehat{\boldsymbol{\theta}}$ are the estimated model parameters, which are stored in e(b) following any Stata estimation command.

In its most common use, predictnl generates two variables: one containing the estimated prediction, $g\left(\widehat{\boldsymbol{\theta}}, \mathbf{x}_{i}\right)$, the other containing the estimated standard error of $g\left(\widehat{\boldsymbol{\theta}}, \mathbf{x}_{i}\right)$. The calculation of standard errors (and other obtainable quantities that are based on the standard errors, such as test statistics) is based on the delta method, an approximation appropriate in large samples; see Methods and formulas.
predictnl can be used with svy estimation results (assuming that predict is also allowed), see [SVY] svy postestimation.

The specification of $g\left(\widehat{\boldsymbol{\theta}}, \mathbf{x}_{i}\right)$ is handled by specifying pnl_exp, and the values of $g\left(\widehat{\boldsymbol{\theta}}, \mathbf{x}_{i}\right)$ are stored in the new variable newvar of storage type type. pnl_exp is any valid Stata expression and may also contain calls to two special functions unique to predictnl:

1. predict ([predict_options]): When you are evaluating pnl_exp, predict() is a convenience function that replicates the calculation performed by the command
```
predict ..., predict_options
```

As such, the predict() function may be used either as a shorthand for the formula used to make this prediction or when the formula is not readily available. When used without arguments, predict() replicates the default prediction for that particular estimation command.
2. $\mathrm{xb}([$ eqno $])$ : The xb() function replicates the calculation of the linear predictor $\mathbf{x}_{i} \mathbf{b}$ for equation eqno. If xb() is specified without eqno, the linear predictor for the first equation (or the only equation in single-equation estimation) is obtained.

For example, $\mathrm{xb}(\# 1)$ (or equivalently, xb() with no arguments) translates to the linear predictor for the first equation, $\mathrm{xb}(\# 2)$ for the second, and so on. You could also refer to the equations by their names, such as xb (income).
When specifying pnl_exp, both of these functions may be used repeatedly, in combination, and in combination with other Stata functions and expressions. See Remarks and examples for examples that use both of these functions.

## Quick start

After regress, create yhat containing the default linear prediction
predictnl yhat $=$ predict ()
After probit y x1 x2, create pr1 containing predicted probability that $\mathrm{y}=1$
predictnl pr1 = predict(pr)
Same as above, and create lb1 and ub1 containing the upper and lower bounds of the $95 \%$ confidence interval for the prediction
predictnl pr1 = predict(pr), ci(lb1 ub1)
Same as above, specified as a function of the linear prediction
predictnl pr1 = normal(xb()), ci(lb1 ub1)
Same as above, specified as a function of the coefficients
predictnl pr1 = normal(_b[_cons]+_b[x1]*x1+_b[x2]*x2), ci(lb1 ub1)
Same as above, but create the prediction using x 1 equal to 0
predictnl pr1 = normal (_b[_cons]+_b[x1]*0+_b[x2]*x2), ci(lb1 ub1)
Same as above, but create variable se1 containing the standard error of the prediction
predictnl pr1 = normal (_b[_cons]+_b[x1]*0+_b[x2]*x2), se(se1)
After a multiple-equation model, create ratio as the ratio of the linear predictions from the second and third equations

```
predictnl ratio = xb(#2)/xb(#3)
```


## Menu

Statistics $>$ Postestimation

## Syntax

$$
\text { predictnl [type] newvar }=\text { pnl_exp }[\text { if }][\text { in }][, \text { options }]
$$

## options

Main
se(newvar)
variance (newvar)
wald(newvar)
p(newvar)
ci(newvars)
level(\#)
g (stub)

```
Advanced
    iterate(#)
    force
force
```

    \(\mathrm{df}(\#) \quad\) use \(F\) distribution with \# denominator degrees of freedom for the
    reference distribution of the test statistic
    $\mathrm{df}(\#) \quad$ use $F$ distribution with \# denominator degrees of freedom for the reference distribution of the test statistic
$\mathrm{df}(\#)$ does not appear in the dialog box.

## Options

$\qquad$ Main
se(newvar) adds newvar of storage type type, where for each i in the prediction sample, newvar [i] contains the estimated standard error of $g\left(\widehat{\boldsymbol{\theta}}, \mathbf{x}_{i}\right)$.
variance (newvar) adds newvar of storage type type, where for each i in the prediction sample, newvar $[i]$ contains the estimated variance of $g\left(\widehat{\boldsymbol{\theta}}, \mathbf{x}_{i}\right)$.
wald (newvar) adds newvar of storage type type, where for each i in the prediction sample, newvar [i] contains the Wald test statistic for the test of the hypothesis $H_{0}: g\left(\boldsymbol{\theta}, \mathbf{x}_{i}\right)=0$.
p (newvar) adds newvar of storage type type, where newvar[i] contains the $p$-value for the Wald test of $H_{0}: g\left(\boldsymbol{\theta}, \mathbf{x}_{i}\right)=0$ versus the two-sided alternative.
ci (newvars) requires the specification of two newvars, such that the $i$ th observation of each will contain the left and right endpoints (respectively) of a confidence interval for $g\left(\boldsymbol{\theta}, \mathbf{x}_{i}\right)$. The level of the confidence intervals is determined by level (\#).
level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals.
$\mathrm{g}(s t u b)$ specifies that new variables, stub1, stub2, $\ldots$, stubk be created, where k is the dimension of $\boldsymbol{\theta}$. stub1 will contain the observation-specific derivatives of $g\left(\boldsymbol{\theta}, \mathbf{x}_{i}\right)$ with respect to the first element, $\theta_{1}$, of $\boldsymbol{\theta}$; stub2 will contain the derivatives of $g\left(\boldsymbol{\theta}, \mathbf{x}_{i}\right)$ with respect to $\theta_{2}$, etc.; If the derivative of $g\left(\boldsymbol{\theta}, \mathbf{x}_{i}\right)$ with respect to a particular coefficient in $\boldsymbol{\theta}$ equals zero for all observations in the prediction sample, the stub variable for that coefficient is not created. The ordering of the parameters in $\boldsymbol{\theta}$ is precisely that of the stored vector of parameter estimates e(b).

Advanced
iterate(\#) specifies the maximum number of iterations used to find the optimal step size in the calculation of numerical derivatives of $g\left(\boldsymbol{\theta}, \mathbf{x}_{i}\right)$ with respect to $\boldsymbol{\theta}$. By default, the maximum number of iterations is 100 , but convergence is usually achieved after only a few iterations. You should rarely have to use this option.
force forces the calculation of standard errors and other inference-related quantities in situations where predictnl would otherwise refuse to do so. The calculation of standard errors takes place by evaluating (at $\widehat{\boldsymbol{\theta}}$ ) the numerical derivative of $g\left(\boldsymbol{\theta}, \mathbf{x}_{i}\right)$ with respect to $\boldsymbol{\theta}$. If predictnl detects that $g(\cdot)$ is possibly a function of random quantities other than $\widehat{\boldsymbol{\theta}}$, it will refuse to calculate standard errors or any other quantity derived from them. The force option forces the calculation to take place anyway. If you use the force option, there is no guarantee that any inference quantities (for example, standard errors) will be correct or that the values obtained can be interpreted.

The following option is available with predictnl but is not shown in the dialog box:
$\mathrm{df}(\#)$ specifies that the $F$ distribution with \# denominator degrees of freedom be used for the reference distribution of the test statistic.

## Remarks and examples

Remarks are presented under the following headings:
Introduction
Nonlinear transformations and standard errors
Using $x b()$ and predict()
Multiple-equation (ME) estimators
Test statistics and $p$-values
Manipulability
Confidence intervals

## Introduction

predictnl and nlcom both use the delta method. They take a nonlinear transformation of the estimated parameter vector from some fitted model and apply the delta method to calculate the variance, standard error, Wald test statistic, etc., of this transformation. nlcom is designed for scalar functions of the parameters, and predictnl is designed for functions of the parameters and of the data, that is, for predictions.

## Nonlinear transformations and standard errors

We begin by fitting a probit model to the low-birthweight data of Hosmer, Lemeshow, and Sturdivant $(2013,24)$. The data are described in detail in example 1 of $[R]$ logistic.

```
. use https://www.stata-press.com/data/r18/lbw
(Hosmer & Lemeshow data)
```

. probit low lwt smoke ptl ht
Iteration 0: Log likelihood $=-117.336$
Iteration 1: Log likelihood $=-106.75886$
Iteration 2: Log likelihood $=-106.67852$
Iteration 3: Log likelihood $=-106.67851$
Probit regression
Log likelihood $=-106.67851$

| Number of obs | $=189$ |
| :--- | ---: | ---: |
| LR chi2(4) | $=21.31$ |
| Prob > chi2 | $=0.0003$ |
| Pseudo R2 | $=0.0908$ |


| low | Coefficient | Std. err. | $z$ | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | :---: | ---: | ---: | ---: |
| lwt | -.0095164 | .0036875 | -2.58 | 0.010 | -.0167438 | -.0022891 |
| smoke | .3487004 | .2041772 | 1.71 | 0.088 | -.0514794 | .7488803 |
| ptl | .365667 | .1921201 | 1.90 | 0.057 | -.0108815 | .7422154 |
| ht | 1.082355 | .410673 | 2.64 | 0.008 | .2774503 | 1.887259 |
| _cons | .4238985 | .4823224 | 0.88 | 0.379 | -.5214361 | 1.369233 |

After we fit such a model, we first would want to generate the predicted probabilities of a low birthweight, given the covariate values in the estimation sample. This is easily done using predict after probit, but it doesn't answer the question, "What are the standard errors of those predictions?"

For the time being, we will consider ourselves ignorant of any automated way to obtain the predicted probabilities after probit. The formula for the prediction is

$$
\operatorname{Pr}\left(y \neq 0 \mid \mathbf{x}_{i}\right)=\Phi\left(\mathbf{x}_{i} \boldsymbol{\beta}\right)
$$

where $\Phi$ is the standard cumulative normal. Thus for this example, $g\left(\boldsymbol{\theta}, \mathbf{x}_{i}\right)=\Phi\left(\mathbf{x}_{i} \boldsymbol{\beta}\right)$. Armed with the formula, we can use predictnl to generate the predictions and their standard errors:

```
. predictnl phat = normal(_b[_cons] + _b[ht]*ht + _b[ptl]*ptl +
> _b[smoke]*smoke + _b[lwt]*lwt), se(phat_se)
. list phat phat_se lwt smoke ptl ht in -10/l
```

|  | phat | phat_se | lwt | smoke | ptl | ht |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: |
|  | 180. | .2363556 | .042707 | 120 | Nonsmoker | 0 |
| 181. | .6577712 | .1580714 | 154 | Nonsmoker | 1 | 1 |
| 182. | .2793261 | .0519958 | 106 | Nonsmoker | 0 | 0 |
| 183. | .1502118 | .0676339 | 190 | Smoker | 0 | 0 |
| 184. | .5702871 | .0819911 | 101 | Smoker | 1 | 0 |
| 185. | .4477045 | .079889 | 95 | Smoker | 0 | 0 |
| 186. | .2988379 | .0576306 | 100 | Nonsmoker | 0 | 0 |
| 187. | .4514706 | .080815 | 94 | Smoker | 0 | 0 |
| 188. | .5615571 | .1551051 | 142 | Nonsmoker | 0 | 1 |
| 189. | .7316517 | .1361469 | 130 | Smoker | 0 | 1 |

Thus, subject 180 in our data has an estimated probability of low birthweight of $23.6 \%$ with standard error $4.3 \%$.

Used without options, predictnl is not much different from generate. By specifying the se (phat_se) option, we were able to obtain a variable containing the standard errors of the predictions; therein lies the utility of predictnl.

## Using $\mathbf{x b}()$ and predict()

As was the case above, a prediction is often not a function of a few isolated parameters and their corresponding variables but instead is some (possibly elaborate) function of the entire linear predictor. For models with many predictors, the brute-force expression for the linear predictor can be cumbersome to type. An alternative is to use the inline function xb()$. \mathrm{xb}()$ is a shortcut for having to type _b[_cons] + _b[ht]*ht + _b[ptl]*ptl + ...,

```
. drop phat phat_se
. predictnl phat = norm(xb()), se(phat_se)
. list phat phat_se lwt smoke ptl ht in -10/l
```

|  | phat | phat_se | lwt | smoke | ptl | ht |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: |
|  | 180. | .2363556 | .042707 | 120 | Nonsmoker | 0 |
| 181. | .6577712 | .1580714 | 154 | Nonsmoker | 1 | 1 |
| 182. | .2793261 | .0519958 | 106 | Nonsmoker | 0 | 0 |
| 183. | .1502118 | .0676339 | 190 | Smoker | 0 | 0 |
| 184. | .5702871 | .0819911 | 101 | Smoker | 1 | 0 |
| 185. | .4477045 | .079889 | 95 | Smoker | 0 | 0 |
| 186. | .2988379 | .0576306 | 100 | Nonsmoker | 0 | 0 |
| 187. | .4514706 | .080815 | 94 | Smoker | 0 | 0 |
| 188. | .5615571 | .1551051 | 142 | Nonsmoker | 0 | 1 |
| 189. | .7316517 | .1361469 | 130 | Smoker | 0 | 1 |

which yields the same results. This approach is easier, produces more readable code, and is less prone to error, such as forgetting to include a term in the sum.

Here we used xb() without arguments because we have only one equation in our model. In multiple-equation (ME) settings, xb() (or equivalently $\mathrm{xb}(\# 1)$ ) yields the linear predictor from the first equation, $\mathrm{xb}(\# 2)$ from the second, etc. You can also refer to equations by their names, for example, xb(income).

## - Technical note

Most estimation commands in Stata allow the postestimation calculation of linear predictors and their standard errors via predict. For example, to obtain these for the first (or only) equation in the model, you could type

```
predict xbvar, xb
predict stdpvar, stdp
```

Equivalently, you could type

```
predictnl xbvar = xb(), se(stdpvar)
```

but we recommend the first method, because it is faster. As we demonstrated above, however, predictnl is more general.

Returning to our probit example, we can further simplify the calculation by using the inline function predict(). predict(pred_options) works by substituting, within our predictnl expression, the calculation performed by

```
predict ..., pred_options
```

In our example, we are interested in the predicted probabilities after a probit regression, normally obtained via

```
predict ..., p
```

We can obtain these predictions (and standard errors) by using

```
. drop phat phat_se
. predictnl phat = predict(p), se(phat_se)
. list phat phat_se lwt smoke ptl ht in -10/l
```

|  | phat | phat_se | lwt | smoke | ptl | ht |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 180. | . 2363556 | . 042707 | 120 | Nonsmoker | 0 | 0 |
| 181. | . 6577712 | . 1580714 | 154 | Nonsmoker | 1 | 1 |
| 182. | . 2793261 | . 0519958 | 106 | Nonsmoker | 0 | 0 |
| 183. | . 1502118 | . 0676339 | 190 | Smoker | 0 | 0 |
| 184. | . 5702871 | . 0819911 | 101 | Smoker | 1 | 0 |
| 185. | . 4477045 | . 079889 | 95 | Smoker | 0 | 0 |
| 186. | . 2988379 | . 0576306 | 100 | Nonsmoker | 0 | 0 |
| 187. | . 4514706 | . 080815 | 94 | Smoker | 0 | 0 |
| 188. | . 5615571 | . 1551051 | 142 | Nonsmoker | 0 | 1 |
| 189. | . 7316517 | . 1361469 | 130 | Smoker | 0 | 1 |

which again replicates what we have already done by other means. However, this version did not require knowledge of the formula for the predicted probabilities after a probit regression—predict (p) took care of that for us.

Because the predicted probability is the default prediction after probit, we could have just used predict() without arguments, namely,

```
. predictnl phat = predict(), se(phat_se)
```

Also, the expression pnl_exp can be inordinately complicated, with multiple calls to predict () and xb() . For example,

```
. predictnl phat = normal(invnormal(predict()) + predict(xb)/xb() - 1),
> se(phat_se)
```

is perfectly valid and will give the same result as before, albeit a bit inefficiently.

## Technical note

When using predict() and xb() , the formula for the calculation is substituted within pnl_exp, not the values that result from the application of that formula. To see this, note the subtle difference between

```
. predict xbeta, xb
. predictnl phat = normal(xbeta), se(phat_se)
```

and

```
. predictnl phat = normal(xb()), se(phat_se)
```

Both sequences will yield the same phat, yet for the first sequence, phat_se will equal zero for all observations. The reason is that, once evaluated, xbeta will contain the values of the linear predictor, yet these values are treated as fixed and nonstochastic as far as predictnl is concerned. By contrast, because xb() is shorthand for the formula used to calculate the linear predictor, it contains not values, but references to the estimated regression coefficients and corresponding variables. Thus, the second method produces the desired result.

## Multiple-equation (ME) estimators

In [R] mlogit, data on insurance choice (Tarlov et al. 1989; Wells et al. 1989) were examined, and a multinomial logit was used to assess the effects of age, gender, race, and site of study (one of three sites) on the type of insurance:
. use https://www.stata-press.com/data/r18/sysdsn1, clear
(Health insurance data)
. mlogit insure age male nonwhite i.site, nolog

Multinomial logistic regression $\quad$\begin{tabular}{l}

Number of obs $=$| 615 |
| ---: |
|  |
| Log likelihood $=-534.36165$ | <br>

\end{tabular}

| insure | Coefficient | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Indemnity | (base outcome) |  |  |  |  |  |
| Prepaid |  |  |  |  |  |  |
| age | -.011745 | .0061946 | -1.90 | 0.058 | -.0238862 | .0003962 |
| male | .5616934 | .2027465 | 2.77 | 0.006 | .1643175 | .9590693 |
| nonwhite | .9747768 | .2363213 | 4.12 | 0.000 | .5115955 | 1.437958 |
| site |  |  |  |  |  |  |
| 2 | .1130359 | .2101903 | 0.54 | 0.591 | -.2989296 | .5250013 |
| 3 | -.5879879 | .2279351 | -2.58 | 0.010 | -1.034733 | -.1412433 |
| _cons | .2697127 | .3284422 | 0.82 | 0.412 | -.3740222 | .9134476 |
| age | -.0077961 | .0114418 | -0.68 | 0.496 | -.0302217 | .0146294 |
| male | .4518496 | .3674867 | 1.23 | 0.219 | -.268411 | 1.17211 |
| nonwhite | .2170589 | .4256361 | 0.51 | 0.610 | -.6171725 | 1.05129 |
| site |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |
| 3 | -1.211563 | .4705127 | -2.57 | 0.010 | -2.133751 | -.2893747 |
| Uninsure | -.2078123 | .3662926 | -0.57 | 0.570 | -.9257327 | .510108 |
| _cons | -1.286943 | .5923219 | -2.17 | 0.030 | -2.447872 | -.1260134 |

Of particular interest is the estimation of the relative risk, which, for a given selection, is the ratio of the probability of making that selection to the probability of selecting the base category (Indemnity here), given a set of covariate values. In a multinomial logit model, the relative risk (when comparing to the base category) simplifies to the exponentiated linear predictor for that selection.

Using this example, we can estimate the observation-specific relative risks of selecting a prepaid plan over the base category (with standard errors) by either referring to the Prepaid equation by name or number,

```
. predictnl RRppaid = exp(xb(Prepaid)), se(SERRppaid)
```

or

```
. predictnl RRppaid = exp(xb(#1)), se(SERRppaid)
```

because Prepaid is the first equation in the model.
Those of us for whom the simplified formula for the relative risk does not immediately come to mind may prefer to calculate the relative risk directly from its definition, that is, as a ratio of two predicted probabilities. After mlogit, the predicted probability for a category may be obtained using predict, but we must specify the category as the outcome:

```
. predictnl RRppaid = predict(outcome(Prepaid))/predict(outcome(Indemnity)),
> se(SERRppaid)
(1 missing value generated)
. list RRppaid SERRppaid age male nonwhite site in 1/10
```

| RRppaid | SERRpp~d | age | male | nonwhite | site |
| ---: | ---: | ---: | ---: | ---: | ---: |
| .6168578 | .1503759 | 73.722107 | 0 | 0 | 2 |
| 1.056658 | .1790703 | 27.89595 | 0 | 0 | 2 |
| .8426442 | .1511281 | 37.541397 | 0 | 0 | 1 |
| 1.460581 | .3671465 | 23.641327 | 0 | 1 | 3 |
| .9115747 | .1324168 | 40.470901 | 0 | 0 | 2 |
| 1.034701 | .1696923 | 29.683777 | 0 | 0 | 2 |
| .9223664 | .1344981 | 39.468857 | 0 | 0 | 2 |
| 1.678312 | .4216626 | 26.702255 | 1 | 0 | 1 |
| .9188519 | .2256017 | 63.101974 | 0 | 1 | 3 |
| .5766296 | .1334877 | 69.839828 | 0 | 0 | 1 |

The "(1 missing value generated)" message is not an error; further examination of the data would reveal that age is missing in one observation and that the offending observation (among others) is not in the estimation sample. Just as with predict, predictnl can generate predictions in or out of the estimation sample.

Thus, we estimate (among other things) that a white, female, 73-year-old from site 2 is less likely to choose a prepaid plan over an indemnity plan-her relative risk is about $62 \%$ with standard error $15 \%$.

## Test statistics and p -values

Often, a standard error calculation is just a means to an end, and what is really desired is a test of the hypothesis,

$$
H_{0}: g\left(\boldsymbol{\theta}, \mathbf{x}_{i}\right)=0
$$

versus the two-sided alternative.
We can use predictnl to obtain the Wald test statistics or $p$-values (or both) for the above tests, whether or not we want standard errors. To obtain the Wald test statistics, we use the wald() option; for $p$-values, we use p() .

Returning to our mlogit example, suppose that we wanted for each observation a test of whether the relative risk of choosing a prepaid plan over an indemnity plan is different from one. One way to do this would be to define $g(\cdot)$ to be the relative risk minus one and then test whether $g(\cdot)$ is different from zero.

```
. predictnl RRm1 = exp(xb(Prepaid)) - 1, wald(W_RRm1) p(sig_RRm1)
(1 missing value generated)
note: p-values are with respect to the chi-squared(1) distribution.
. list RRm1 W_RRm1 sig_RRm1 age male nonwhite in 1/10
```

| RRm1 | W_RRm1 | sig_RRm1 | age | male | nonwhite |
| ---: | ---: | ---: | ---: | ---: | ---: |
| -.3831422 | 6.491778 | .0108375 | 73.722107 | 0 | 0 |
| .0566578 | .100109 | .7516989 | 27.89595 | 0 | 0 |
| -.1573559 | 1.084116 | .2977787 | 37.541397 | 0 | 0 |
| .4605812 | 1.573743 | .2096643 | 23.641327 | 0 | 1 |
| -.0884253 | .4459299 | .5042742 | 40.470901 | 0 | 0 |
| .0347015 | .0418188 | .8379655 | 29.683777 | 0 | 0 |
| -.0776336 | .3331707 | .563798 | 39.468857 | 0 | 0 |
| .6783119 | 2.587788 | .1076906 | 26.702255 | 1 | 0 |
| -.0811482 | .1293816 | .719074 | 63.101974 | 0 | 1 |
| -.4233705 | 10.05909 | .001516 | 69.839828 | 0 | 0 |

The newly created variable W_RRm1 contains the Wald test statistic for each observation, and sig_RRm1 contains the $p$-value. Thus, our 73-year-old white female represented by the first observation would have a relative risk of choosing prepaid over indemnity that is significantly different from 1 , at least at the $5 \%$ level. For this test, it was not necessary to generate a variable containing the standard error of the relative risk minus 1 , but we could have done so had we wanted. We could have also omitted specifying wald ( $\mathrm{W} \_$RRm1) if all we cared about were, say, the $p$-values for the tests.

In this regard, predictnl acts as an observation-specific version of testnl, with the test results vectorized over the observations in the data. The $p$-values are pointwise-they are not adjusted to reflect any simultaneous testing over the observations in the data.

## Manipulability

There are many ways to specify $g\left(\boldsymbol{\theta}, \mathbf{x}_{i}\right)$ to yield tests such that, for multiple specifications of $g(\cdot)$, the theoretical conditions for which

$$
H_{0}: g\left(\boldsymbol{\theta}, \mathbf{x}_{i}\right)=0
$$

is true will be equivalent. However, this does not mean that the tests themselves will be equivalent. This is known as the manipulability of the Wald test for nonlinear hypotheses; also see [R] boxcox.

As an example, consider the previous section where we defined $g(\cdot)$ to be the relative risk between choosing a prepaid plan over an indemnity plan, minus 1 . We could also have defined $g(\cdot)$ to be the risk difference - the probability of choosing a prepaid plan minus the probability of choosing an indemnity plan. Either specification of $g(\cdot)$ yields a mathematically equivalent specification of $H_{0}: g(\cdot)=0$; that is, the risk difference will equal zero when the relative risk equals one. However, the tests themselves do not give the same results:

```
. predictnl RD = predict(outcome(Prepaid)) - predict(outcome(Indemnity)),
> wald(W_RD) p(sig_RD)
(1 missing value generated)
note: p-values are with respect to the chi-squared(1) distribution.
list RD W_RD sig_RD RRm1 W_RRm1 sig_RRm1 in 1/10
```

|  | RD | W_RD | sig_RD | RRm1 | W_RRm1 | sig_RRm1 |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| 1. | -.2303744 | 4.230243 | .0397097 | -.3831422 | 6.491778 | .0108375 |
| 2. | .0266902 | .1058542 | .7449144 | .0566578 | .100109 | .7516989 |
| 3. | -.0768078 | .9187646 | .3377995 | -.1573559 | 1.084116 | .2977787 |
| 4. | .1710702 | 2.366535 | .1239619 | .4605812 | 1.573743 | .2096643 |
| 5. | -.0448509 | .4072922 | .5233471 | -.0884253 | .4459299 | .5042742 |
| 6. | .0165251 | .0432816 | .835196 | .0347015 | .0418188 | .8379655 |
| 7. | -.0391535 | .3077611 | .5790573 | -.0776336 | .3331707 | .563798 |
| 8. | .22382 | 4.539085 | .0331293 | .6783119 | 2.587788 | .1076906 |
| 9. | -.0388409 | .1190183 | .7301016 | -.0811482 | .1293816 | .719074 |
| 10. | -.2437626 | 6.151558 | .0131296 | -.4233705 | 10.05909 | .001516 |

In certain cases (such as subject 8 ), the difference can be severe enough to potentially change the conclusion. The reason for this inconsistency is that the nonlinear Wald test is actually a standard Wald test of a first-order Taylor approximation of $g(\cdot)$, and this approximation can differ according to how $g(\cdot)$ is specified.

As such, keep in mind the manipulability of nonlinear Wald tests when drawing scientific conclusions.

## Confidence intervals

We can also use predictnl to obtain confidence intervals for the observation-specific $g\left(\boldsymbol{\theta}, \mathbf{x}_{i}\right)$ by using the ci() option to specify two new variables to contain the left and right endpoints of the confidence interval, respectively. For example, we could generate confidence intervals for the risk differences calculated previously:

```
. drop RD
. predictnl RD = predict(outcome(Prepaid)) - predict(outcome(Indemnity)),
> ci(RD_lcl RD_rcl)
(1 missing value generated)
note: confidence intervals calculated using Z critical values.
. list RD RD_lcl RD_rcl age male nonwhite in 1/10
```

| RD | RD_lcl | RD_rcl | age | male | nonwhite |
| ---: | ---: | ---: | ---: | ---: | ---: |
| -.2303744 | -.4499073 | -.0108415 | 73.722107 | 0 | 0 |
| .0266902 | -.1340948 | .1874752 | 27.89595 | 0 | 0 |
| -.0768078 | -.2338625 | .080247 | 37.541397 | 0 | 0 |
| .1710702 | -.0468844 | .3890248 | 23.641327 | 0 | 1 |
| -.0448509 | -.1825929 | .092891 | 40.470901 | 0 | 0 |
| .0165251 | -.1391577 | .1722078 | 29.683777 | 0 | 0 |
| -.0391535 | -.177482 | .099175 | 39.468857 | 0 | 0 |
| .22382 | .0179169 | .4297231 | 26.702255 | 1 | 0 |
| -.0388409 | -.2595044 | .1818226 | 63.101974 | 0 | 1 |
| -.2437626 | -.4363919 | -.0511332 | 69.839828 | 0 | 0 |

The confidence level, here, $95 \%$, is either set using the level() option or obtained from the current default level, c(level); see [U] 20.8 Specifying the width of confidence intervals.

From the above output, we can see that, for subjects 1,8 , and 10 , a $95 \%$ confidence interval for the risk difference does not contain zero, meaning that, for these subjects, there is some evidence of a significant difference in risks.

The confidence intervals calculated by predictnl are pointwise; there is no adjustment (such as a Bonferroni correction) made so that these confidence intervals may be considered jointly at the specified level.

## Methods and formulas

For the $i$ th observation, consider the transformation $g\left(\boldsymbol{\theta}, \mathbf{x}_{i}\right)$, estimated by $g\left(\widehat{\boldsymbol{\theta}}, \mathbf{x}_{i}\right)$, for the $1 \times k$ parameter vector $\boldsymbol{\theta}$ and data $\mathbf{x}_{i}$ ( $\mathbf{x}_{i}$ is assumed fixed). The variance of $g\left(\widehat{\boldsymbol{\theta}}, \mathbf{x}_{i}\right)$ is estimated by

$$
\widehat{\operatorname{Var}}\left\{g\left(\widehat{\boldsymbol{\theta}}, \mathbf{x}_{i}\right)\right\}=\mathbf{G V G}^{\prime}
$$

where $\mathbf{G}$ is the vector of derivatives

$$
\mathbf{G}=\left\{\left.\frac{\partial g\left(\boldsymbol{\theta}, \mathbf{x}_{i}\right)}{\partial \boldsymbol{\theta}}\right|_{\boldsymbol{\theta}=\widehat{\boldsymbol{\theta}}}\right\}_{(1 \times k)}
$$

and $\mathbf{V}$ is the estimated variance-covariance matrix of $\widehat{\boldsymbol{\theta}}$. Standard errors, $\widehat{\operatorname{se}}\left\{g\left(\widehat{\boldsymbol{\theta}}, \mathbf{x}_{i}\right)\right\}$, are obtained as the square roots of the variances.

The Wald test statistic for testing

$$
H_{0}: g\left(\boldsymbol{\theta}, \mathbf{x}_{i}\right)=0
$$

versus the two-sided alternative is given by

$$
W_{i}=\frac{\left\{g\left(\widehat{\boldsymbol{\theta}}, \mathbf{x}_{i}\right)\right\}^{2}}{\widehat{\operatorname{Var}}\left\{g\left(\widehat{\boldsymbol{\theta}}, \mathbf{x}_{i}\right)\right\}}
$$

When the variance-covariance matrix of $\widehat{\boldsymbol{\theta}}$ is an asymptotic covariance matrix, $W_{i}$ is approximately distributed as $\chi^{2}$ with 1 degree of freedom. For linear regression, $W_{i}$ is taken to be approximately distributed as $F_{1, r}$, where $r$ is the residual degrees of freedom from the original model fit. The $p$-values for the observation-by-observation tests of $H_{0}$ versus the two-sided alternative are given by

$$
p_{i}=\operatorname{Pr}\left(T>W_{i}\right)
$$

where $T$ is either a $\chi^{2}$ - or $F$-distributed random variable, as described above.
A $(1-\alpha) \times 100 \%$ confidence interval for $g\left(\boldsymbol{\theta}, \mathbf{x}_{i}\right)$ is given by

$$
g\left(\widehat{\boldsymbol{\theta}}, \mathbf{x}_{i}\right) \pm z_{\alpha / 2}\left[\widehat{\operatorname{se}}\left\{g\left(\widehat{\boldsymbol{\theta}}, \mathbf{x}_{i}\right)\right\}\right]
$$

when $W_{i}$ is $\chi^{2}$-distributed, and

$$
g\left(\widehat{\boldsymbol{\theta}}, \mathbf{x}_{i}\right) \pm t_{\alpha / 2, r}\left[\widehat{\operatorname{se}}\left\{g\left(\widehat{\boldsymbol{\theta}}, \mathbf{x}_{i}\right)\right\}\right]
$$

when $W_{i}$ is $F$-distributed. $z_{p}$ is the $1-p$ quantile of the standard normal distribution, and $t_{p, r}$ is the $1-p$ quantile of the $t$ distribution with $r$ degrees of freedom.

## References

Drukker, D. M. 2016a. Probability differences and odds ratios measure conditional-on-covariate effects and population-parameter effects. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2016/07/26/probability-differences-and-odds-ratios-measure-conditional-on-covariate-effects-and-population-parameter-effects/.
-_. 2016b. Quantile regression allows covariate effects to differ by quantile. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2016/09/27/quantile-regression-allows-covariate-effects-to-differ-by-quantile/.
Hosmer, D. W., Jr., S. A. Lemeshow, and R. X. Sturdivant. 2013. Applied Logistic Regression. 3rd ed. Hoboken, NJ: Wiley.
Phillips, P. C. B., and J. Y. Park. 1988. On the formulation of Wald tests of nonlinear restrictions. Econometrica 56: 1065-1083. https://doi.org/10.2307/1911359.
Tarlov, A. R., J. E. Ware, Jr., S. Greenfield, E. C. Nelson, E. Perrin, and M. Zubkoff. 1989. The medical outcomes study. An application of methods for monitoring the results of medical care. Journal of the American Medical Association 262: 925-930. https://doi.org/10.1001/jama.1989.03430070073033.
Wells, K. B., R. D. Hays, M. A. Burnam, W. H. Rogers, S. Greenfield, and J. E. Ware, Jr. 1989. Detection of depressive disorder for patients receiving prepaid or fee-for-service care. Results from the Medical Outcomes Survey. Journal of the American Medical Association 262: 3298-3302. https://doi.org/10.1001/jama.1989.03430230083030.

## Also see

[ $R$ ] lincom - Linear combinations of parameters
[R] nlcom - Nonlinear combinations of parameters
[ R$]$ predict - Obtain predictions, residuals, etc., after estimation
[R] test - Test linear hypotheses after estimation
[R] testnl - Test nonlinear hypotheses after estimation
[U] 20 Estimation and postestimation commands

## Title

probit — Probit regression

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

probit fits a probit model for a binary dependent variable, assuming that the probability of a positive outcome is determined by the standard normal cumulative distribution function. probit can compute robust and cluster-robust standard errors and adjust results for complex survey designs.

## Quick start

Probit model of y on continuous variable x 1
probit y x1
Add square of x 1
probit y c.x1\#\#c.x1
Same as above, but report bootstrap standard errors
probit y c.x1\#\#c.x1, vce(bootstrap)
Bootstrap estimates of coefficients
bootstrap _b: probit y c.x1\#\#c.x1
Adjust for complex survey design using svyset data and add x2
svy: probit y c.x1\#\#c.x1 x2

## Menu

Statistics $>$ Binary outcomes $>$ Probit regression

## Syntax

probit depvar [indepvars] [if] [in] [weight] [, options]
options
Description
Model
noconstant
offset (varname)
asis
constraints (constraints)
SE/Robust
vce(vcetype)

Reporting
level (\#)
nocnsreport
display_options
suppress constant term
include varname in model with coefficient constrained to 1
retain perfect predictor variables
apply specified linear constraints
vcetype may be oim, opg, robust, cluster clustvar, bootstrap, or jackknife
set confidence level; default is level (95)
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

Maximization
maximize_options
nocoef
collinear
coeflegend
control the maximization process; seldom used
do not display the coefficient table; seldom used
keep collinear variables
display legend instead of statistics
indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
depvar and indepvars may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bayes, bootstrap, by, collect, fmm, fp, jackknife, mfp, mi estimate, nestreg, rolling, statsby, stepwise, and svy are allowed; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: probit and [FMM] fmm: probit.
vce (bootstrap) and vce (jackknife) are not allowed with the mi estimate prefix; see [MI] mi estimate.
Weights are not allowed with the bootstrap prefix; see [R] bootstrap.
vce(), nocoef, and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
nocoef, collinear, and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

$\qquad$ Model
noconstant, offset (varname), constraints (constraints); see [R] Estimation options.
asis specifies that all specified variables and observations be retained in the maximization process. This option is typically not specified and may introduce numerical instability. Normally probit omits variables that perfectly predict success or failure in the dependent variable along with their associated observations. In those cases, the effective coefficient on the omitted variables is infinity (negative infinity) for variables that completely determine a success (failure). Dropping the variable
and perfectly predicted observations has no effect on the likelihood or estimates of the remaining coefficients and increases the numerical stability of the optimization process. Specifying this option forces retention of perfect predictor variables and their associated observations.

## SE/Robust

vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.

## Reporting

level (\#); see [R] Estimation options.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fmt), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate(\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance (\#), nonrtolerance, and from(init_specs); see [R] Maximize. These options are seldom used.

The following options are available with probit but are not shown in the dialog box:
nocoef specifies that the coefficient table not be displayed. This option is sometimes used by programmers but is of no use interactively.
collinear, coeflegend; see [R] Estimation options.

## Remarks and examples

Remarks are presented under the following headings:
Robust standard errors
Model identification
Video examples
probit fits maximum likelihood models with dichotomous dependent (left-hand-side) variables coded as $0 / 1$ (more precisely, coded as 0 and not 0 ).

For grouped data or data in binomial form, a probit model can be fit using glm with the family(binomial) and link(probit) options.

## > Example 1

We have data on the make, weight, and mileage rating of 22 foreign and 52 domestic automobiles. We wish to fit a probit model explaining whether a car is foreign based on its weight and mileage. Here is an overview of our data:

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. keep make mpg weight foreign
. describe
Contains data from https://www.stata-press.com/data/r18/auto.dta
    Observations: 74 1978 automobile data
        Variables: 4 13 Apr 2022 17:45
                                (_dta has notes)
\begin{tabular}{lccll}
\hline \begin{tabular}{r} 
Variable \\
name
\end{tabular} & \begin{tabular}{r} 
Storage \\
type
\end{tabular} & \begin{tabular}{c} 
Display \\
format
\end{tabular} & \begin{tabular}{l} 
Value \\
label
\end{tabular} & Variable label \\
\hline make & str18 & \(\%-18 \mathrm{~s}\) & & Make and model \\
mpg & int & \(\% 8.0 \mathrm{~g}\) & & Mileage (mpg) \\
weight & int & \(\% 8.0 \mathrm{gc}\) & & Weight (lbs.) \\
foreign & byte & \(\% 8.0 \mathrm{~g}\) & origin & Car origin \\
\hline
\end{tabular}
Sorted by: foreign
        Note: Dataset has changed since last saved.
    . inspect foreign
foreign: Car origin
Number of observations
|\mp@code{# rrarl}
\begin{tabular}{lrrr} 
& Total & Integers & Nonintegers \\
Negative & - & - & - \\
Zero & 52 & 52 & - \\
Positive & 22 & 22 & - \\
\cline { 2 - 2 } \cline { 4 - 5 } & 74 & 74 & - \\
Total & - & &
\end{tabular}
(2 unique values)
foreign is labeled and all values are documented in the label.
```

The foreign variable takes on two unique values, 0 and 1 . The value 0 denotes a domestic car, and 1 denotes a foreign car.

The model that we wish to fit is

$$
\operatorname{Pr}(\text { foreign }=1)=\Phi\left(\beta_{0}+\beta_{1} \text { weight }+\beta_{2} \mathrm{mpg}\right)
$$

where $\Phi$ is the cumulative normal distribution.
To fit this model, we type

```
. probit foreign weight mpg
Iteration 0: Log likelihood = -45.03321
Iteration 1: Log likelihood = -27.914626
    (output omitted)
Iteration 5: Log likelihood = -26.844189
Probit regression Number of obs = 74
                                    LR chi2(2) = 36.38
                                    Prob > chi2 = 0.0000
                                    Pseudo R2 = 0.4039
Log likelihood = -26.844189
Pseudo R2 \(=0.4039\)
```

| foreign | Coefficient | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | ---: | ---: |
| weight | -.0023355 | .0005661 | -4.13 | 0.000 | -.003445 | -.0012261 |
| mpg | -.1039503 | .0515689 | -2.02 | 0.044 | -.2050235 | -.0028772 |
| _cons | 8.275464 | 2.554142 | 3.24 | 0.001 | 3.269437 | 13.28149 |

We find that heavier cars are less likely to be foreign and that cars yielding better gas mileage are also less likely to be foreign, at least holding the weight of the car constant.

See [R] Maximize for an explanation of the output.

## - Technical note

Stata interprets a value of 0 as a negative outcome (failure) and treats all other values (except missing) as positive outcomes (successes). Thus if your dependent variable takes on the values 0 and 1 , then 0 is interpreted as failure and 1 as success. If your dependent variable takes on the values 0 , 1 , and 2 , then 0 is still interpreted as failure, but both 1 and 2 are treated as successes.

If you prefer a more formal mathematical statement, when you type probit $y x$, Stata fits the model

$$
\operatorname{Pr}\left(y_{j} \neq 0 \mid \mathbf{x}_{j}\right)=\Phi\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)
$$

where $\Phi$ is the standard cumulative normal.

## Robust standard errors

If you specify the vce (robust) option, probit reports robust standard errors; see [U] $20.22 \mathrm{Ob}-$ taining robust variance estimates.
> Example 2
For the model from example 1, the robust calculation increases the standard error of the coefficient on mpg by almost $15 \%$ :

- probit foreign weight mpg, vce (robust) nolog
Probit regression
Log pseudolikelihood $=-26.844189$

Without vce (robust), the standard error for the coefficient on mpg was reported to be 0.052 with a resulting confidence interval of $[-0.21,-0.00]$.
> Example 3
The vce(cluster clustvar) option can relax the independence assumption required by the probit estimator to independence between clusters. To demonstrate, we will switch to a different dataset.

We are studying unionization of women in the United States and have a dataset with 26,200 observations on 4,434 women between 1970 and 1988. We will use the variables age (the women were $14-26$ in 1968, and our data span the age range of 16-46), grade (years of schooling completed, ranging from 0 to 18), not_smsa ( $28 \%$ of the person-time was spent living outside an SMSA—standard metropolitan statistical area), south ( $41 \%$ of the person-time was in the South), and year. Each of
these variables is included in the regression as a covariate along with the interaction between south and year. This interaction, along with the south and year variables, is specified in the probit command using factor-variables notation, south\#\#c.year. We also have variable union, indicating union membership. Overall, $22 \%$ of the person-time is marked as time under union membership, and $44 \%$ of these women have belonged to a union.

We fit the following model, ignoring that the women are observed an average of 5.9 times each in these data:
. use https://www.stata-press.com/data/r18/union, clear
(NLS Women 14-24 in 1968)
. probit union age grade not_smsa south\#\#c.year
Iteration 0: Log likelihood $=-13864.23$
Iteration 1: Log likelihood $=-13545.541$
Iteration 2: Log likelihood $=-13544.385$
Iteration 3: Log likelihood $=-13544.385$
Probit regression

| Number of obs | $=26,200$ |
| :--- | :--- |
| LR chi2 $(6)$ | $=639.69$ |
| Prob $>$ chi2 | $=0.0000$ |
| Pseudo R2 | $=0.0231$ |


| union | Coefficient | Std. err. | $z$ | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | ---: | ---: |
| age | .0118481 | .0029072 | 4.08 | 0.000 | .0061502 | .017546 |
| grade | .0267365 | .0036689 | 7.29 | 0.000 | .0195457 | .0339273 |
| not_smsa | -.1293525 | .0202595 | -6.38 | 0.000 | -.1690604 | -.0896445 |
| 1.south | -.8281077 | .2472219 | -3.35 | 0.001 | -1.312654 | -.3435618 |
| year | -.0080931 | .0033469 | -2.42 | 0.016 | -.0146529 | -.0015333 |
| south\#c.year |  |  |  |  |  |  |
| 1 | .0057369 | .0030917 | 1.86 | 0.064 | -.0003226 | .0117965 |
| _cons | -.6542487 | .2007777 | -3.26 | 0.001 | -1.047766 | -.2607316 |

The reported standard errors in this model are probably meaningless. Women are observed repeatedly, and so the observations are not independent. Looking at the coefficients, we find a large southern effect against unionization and a time trend for the south that is almost significantly different from the overall downward trend. The vce (cluster clustvar) option provides a way to fit this model and obtains correct standard errors:

| Iteration 0: Log pseudolikelihood $=-13864.23$ |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Iteration 1: Log pseudolikelihood = -13545.541 |  |  |  |  |  |  |
| Iteration 2: Log pseudolikelihood = -13544.385 |  |  |  |  |  |  |
| Iteration 3: Log pseudolikelihood = -13544.385 |  |  |  |  |  |  |
| Probit regression |  |  |  |  | $\begin{aligned} \text { Number of obs } & =26,200 \\ \text { Wald chi2 }(6) & =166.53 \end{aligned}$ |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  | Prob > chi2 | $=0.0000$ |
| Log pseudolikelihood $=-13544.385$ |  |  |  |  | Pseudo R2 $=0.0231$ |  |
| (Std. err. adjusted for 4,434 clusters in idcode) |  |  |  |  |  |  |
|  |  | Robust |  |  |  |  |
| union | Coefficient | std. err. | z | $P>\|z\|$ | [95\% conf | interval] |
| age | . 0118481 | . 0056625 | 2.09 | 0.036 | . 0007499 | . 0229463 |
| grade | . 0267365 | . 0078124 | 3.42 | 0.001 | . 0114244 | . 0420486 |
| not_smsa | -. 1293525 | . 0403885 | -3.20 | 0.001 | -. 2085125 | -. 0501925 |
| 1. south | -. 8281077 | . 3201584 | -2.59 | 0.010 | -1.455607 | -. 2006089 |
| year | -. 0080931 | . 0060829 | -1.33 | 0.183 | -. 0200153 | . 0038292 |
| south\#c.year |  |  |  |  |  |  |
| 1 | . 0057369 | . 0040133 | 1.43 | 0.153 | -. 002129 | . 0136029 |
| _cons | -. 6542487 | . 3485976 | -1.88 | 0.061 | -1.337487 | . 02899 |

These standard errors are larger than those reported by the inappropriate conventional calculation. By comparison, another model we could fit is an equal-correlation population-averaged probit model:

| Iteration 1: Tolerance $=.12544249$ |  |  |
| :---: | :---: | :---: |
| Iteration 2: Tolerance $=.0034686$ |  |  |
| Iteration 3: Tolerance $=.00017448$ |  |  |
| Iteration 4: Tolerance $=8.382 \mathrm{e}-06$ |  |  |
| Iteration 5: Tolerance $=3.997 \mathrm{e}-07$ |  |  |
| GEE population-averaged model | Number of obs | = 26,200 |
| Group variable: idcode | Number of groups | $=4,434$ |
| Family: Binomial | Obs per group: |  |
| Link: Probit | min | $=1$ |
| Correlation: exchangeable | avg | $=5.9$ |
|  | max | 12 |
|  | Wald chi2(6) | $=242.57$ |
| Scale parameter = 1 | Prob > chi2 | $=0.0000$ |


| union | Coefficient | Std. err. | $z$ | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | ---: | ---: |
| age | .0089699 | .0053208 | 1.69 | 0.092 | -.0014586 | .0193985 |
| grade | .0333174 | .0062352 | 5.34 | 0.000 | .0210966 | .0455382 |
| not_smsa | -.0715717 | .027543 | -2.60 | 0.009 | -.1255551 | -.0175884 |
| 1.south | -1.017368 | .207931 | -4.89 | 0.000 | -1.424905 | -.6098308 |
| year | -.0062708 | .0055314 | -1.13 | 0.257 | -.0171122 | .0045706 |
| south\#c.year |  |  |  |  |  |  |
| 1 | .0086294 | .00258 | 3.34 | 0.001 | .0035727 | .013686 |
| _cons | -.8670997 | .294771 | -2.94 | 0.003 | -1.44484 | -.2893592 |

The coefficient estimates are similar, but these standard errors are smaller than those produced by probit, vce(cluster clustvar), as we would expect. If the equal-correlation assumption is valid, the population-averaged probit estimator above should be more efficient.

Is the assumption valid? That is a difficult question to answer. The default population-averaged estimates correspond to an assumption of exchangeable correlation within person. It would not be unreasonable to assume an $\operatorname{AR}(1)$ correlation within person or to assume that the observations are correlated but that we do not wish to impose any structure. See [XT] xtprobit and [XT] xtgee for full details.
probit, vce(cluster clustvar) is robust to assumptions about within-cluster correlation. That is, it inefficiently sums within cluster for the standard error calculation rather than attempting to exploit what might be assumed about the within-cluster correlation.

## Model identification

The probit command has one more feature that is probably the most useful. It will automatically check the model for identification and, if the model is underidentified, omit whatever variables and observations are necessary for estimation to proceed.
$>$ Example 4
Have you ever fit a probit model where one or more of your independent variables perfectly predicted one or the other outcome?

For instance, consider the following data:

| Outcome $y$ | Independent variable $x$ |
| :---: | :---: |
| 0 | 1 |
| 0 | 1 |
| 0 | 0 |
| 1 | 0 |

Say that we wish to predict the outcome on the basis of the independent variable. The outcome is always zero when the independent variable is one. In our data, $\operatorname{Pr}(y=0 \mid x=1)=1$, which means that the probit coefficient on $x$ must be minus infinity with a corresponding infinite standard error. At this point, you may suspect that we have a problem.

Unfortunately, not all such problems are so easily detected, especially if you have many independent variables in your model. If you have ever had such difficulties, then you have experienced one of the more unpleasant aspects of computer optimization. The computer has no idea that it is trying to solve for an infinite coefficient as it begins its iterative process. All it knows is that, at each step, making the coefficient a little bigger, or a little smaller, works wonders. It continues on its merry way until either 1) the whole thing comes crashing to the ground when a numerical overflow error occurs or 2) it reaches some predetermined cutoff that stops the process. Meanwhile, you have been waiting. And the estimates that you finally receive, if any, may be nothing more than numerical roundoff.

Stata watches for these sorts of problems, alerts you, fixes them, and then properly fits the model.
Let's return to our automobile data. Among the variables we have in the data is one called repair that takes on three values. A value of 1 indicates that the car has a poor repair record, 2 indicates an average record, and 3 indicates a better-than-average record. Here is a tabulation of our data:

```
. use https://www.stata-press.com/data/r18/repair
(1978 automobile data)
. tabulate foreign repair
\begin{tabular}{|c|c|c|c|c|}
\hline \multirow[b]{2}{*}{Car origin} & \multicolumn{3}{|c|}{Repair} & \\
\hline & 1 & 2 & 3 & Total \\
\hline Domestic & 10 & 27 & 9 & 46 \\
\hline Foreign & 0 & 3 & 9 & 12 \\
\hline Total & 10 & 30 & 18 & 58 \\
\hline
\end{tabular}
```

All the cars with poor repair records (repair $=1$ ) are domestic. If we were to attempt to predict foreign on the basis of the repair records, the predicted probability for the repair $=1$ category would have to be zero. This in turn means that the probit coefficient must be minus infinity, and that would set most computer programs buzzing.

Let's try using Stata on this problem.

```
. probit foreign b3.repair
note: 1.repair != 0 predicts failure perfectly;
    1.repair omitted and 10 obs not used.
Iteration 0: Log likelihood = -26.992087
Iteration 1: Log likelihood = -22.276479
Iteration 2: Log likelihood = -22.229184
Iteration 3: Log likelihood = -22.229138
Iteration 4: Log likelihood = -22.229138
Probit regression
Number of obs = 48
                                LR chi2(1) = 9.53
                                Prob > chi2 = 0.0020
Log likelihood = -22.229138
Pseudo R2 = 0.1765
```

| foreign | Coefficient | Std. err. | $z$ | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | :--- | :--- | :--- | :--- | :--- |
| repair | 0 | (empty) |  |  |  |  |
| 1 |  |  |  |  |  |  |
| 2 | -1.281552 | .4297326 | -2.98 | 0.003 | -2.123812 | -.4392911 |
| _cons | $1.16 \mathrm{e}-16$ | .295409 | 0.00 | 1.000 | -.578991 | .578991 |

Remember that all the cars with poor repair records (repair $=1$ ) are domestic, so the model cannot be fit, or at least it cannot be fit if we restrict ourselves to finite coefficients. Stata noted that fact "note: 1.repair != 0 predicts failure perfectly". This is Stata's mathematically precise way of saying what we said in English. When repair is 1, the car is domestic.

Stata then went on to say, "1.repair omitted and 10 obs not used". This is Stata eliminating the problem. First, 1.repair had to be removed from the model because it would have an infinite coefficient. Then, the 10 observations that led to the problem had to be eliminated, as well, so as not to bias the remaining coefficients in the model. The 10 observations that are not used are the 10 domestic cars that have poor repair records.

Stata then fit what was left of the model, using the remaining observations. Because no observations remained for cars with poor repair records, Stata reports "(empty)" in the row for repair $=1$.

## - Technical note

Stata is pretty smart about catching these problems. It will catch "one-way causation by a dummy variable", as we demonstrated above.

Stata also watches for "two-way causation", that is, a variable that perfectly determines the outcome, both successes and failures. Here Stata says that the variable "predicts outcome perfectly" and stops. Statistics dictate that no model can be fit.

Stata also checks your data for collinear variables; it will say "so-and-so omitted because of collinearity". No observations need to be eliminated here and model fitting will proceed without the offending variable.

It will also catch a subtle problem that can arise with continuous data. For instance, if we were estimating the chances of surviving the first year after an operation, and if we included in our model age, and if all the persons over 65 died within the year, Stata will say, "age $>65$ predicts failure perfectly". It will then inform us about how it resolves the issue and fit what can be fit of our model.
probit (and logit, logistic, and ivprobit) will also occasionally fail to converge and then display messages such as

Note: 4 failures and 0 successes completely determined.
The cause of this message and what to do if you see it are described in [R] logit.

## Video examples

Probit regression with categorical covariates
Probit regression with continuous covariates
Probit regression with categorical and continuous covariates

## Stored results

probit stores the following in e():
Scalars

```
        e(N)
        e(N_cds)
        e(N_cdf)
        e(k)
        e(k_eq)
        e(k_eq_model)
        e(k_dv)
        e(df_m)
        e(r2_p)
        e(11)
        e(ll_0)
        e(N_clust)
        e(chi2)
        e(p)
        e(rank)
        e(ic)
        e(rc)
        e(converged)
        number of observations
        number of completely determined successes
        number of completely determined failures
        number of parameters
        number of equations in e(b)
        number of equations in overall model test
        number of dependent variables
        model degrees of freedom
        pseudo-}\mp@subsup{R}{}{2
        log likelihood
        log likelihood, constant-only model
        number of clusters
        \chi2
        p-value for model test
        rank of e(V)
        number of iterations
        return code
        1 if converged, 0 otherwise
```

Macros
e(cmd)
e(cmdline)
e(depvar)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(offset)
e(chi2type)
e(vce)
e(vcetype)
e(opt)
e(which)
e(ml_method)
e(user)
e(technique)
e(properties)
e(estat_cmd)
e(predict)
e(marginsok)
e(marginsnotok)
e(asbalanced)
e(asobserved)
Matrices
e(b)
e(Cns)
e(ilog)
e(gradient)
e(mns)
e(rules)
e(V)
e(V_modelbased)
Functions
e(sample) marks estimation sample

In addition to the above, the following is stored in r() :

```
Matrices
```

    \(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values,
    and confidence intervals
    Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

Probit analysis originated in connection with bioassay, and the word probit, a contraction of "probability unit", was suggested by Bliss (1934a, 1934b). For an introduction to probit and logit, see, for example, Aldrich and Nelson (1984), Cameron and Trivedi (2022), Long (1997), Pampel (2021), or Powers and Xie (2008). Long and Freese (2014, chap. 5 and 6) and Jones (2007, chap. 3) provide introductions to probit and logit, along with Stata examples.

The log-likelihood function for probit is

$$
\ln L=\sum_{j \in S} w_{j} \ln \Phi\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)+\sum_{j \notin S} w_{j} \ln \left\{1-\Phi\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)\right\}
$$

where $\Phi$ is the cumulative normal and $w_{j}$ denotes the optional weights. $\ln L$ is maximized, as described in [R] Maximize.

This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce (robust) and vce(cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas. The scores are calculated as $\mathbf{u}_{j}=$ $\left\{\phi\left(\mathbf{x}_{j} \mathbf{b}\right) / \Phi\left(\mathbf{x}_{j} \mathbf{b}\right)\right\} \mathbf{x}_{j}$ for the positive outcomes and $-\left[\phi\left(\mathbf{x}_{j} \mathbf{b}\right) /\left\{1-\Phi\left(\mathbf{x}_{j} \mathbf{b}\right)\right\}\right] \mathbf{x}_{j}$ for the negative outcomes, where $\phi$ is the normal density.
probit also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

Chester Ittner Bliss (1899-1979) was born in Ohio. He was educated as an entomologist, earning degrees from Ohio State and Columbia, and was employed by the United States Department of Agriculture until 1933. When he lost his job because of the Depression, Bliss then worked with R. A. Fisher in London and at the Institute of Plant Protection in Leningrad before returning to a post at the Connecticut Agricultural Experiment Station in 1938. He was also a lecturer at Yale for 25 years. Among many contributions to biostatistics, his development and application of probit methods to biological problems are outstanding.

## References

Aldrich, J. H., and F. D. Nelson. 1984. Linear Probability, Logit, and Probit Models. Newbury Park, CA: Sage.
Berkson, J. 1944. Application of the logistic function to bio-assay. Journal of the American Statistical Association 39: 357-365. https://doi.org/10.2307/2280041.
Bliss, C. I. 1934a. The method of probits. Science 79: 38-39. https://doi.org/10.1126/science.79.2037.38.
——. 1934b. The method of probits—a correction. Science 79: 409-410. https://doi.org/10.1126/science.79.2053.409.
Cameron, A. C., and P. K. Trivedi. 2022. Microeconometrics Using Stata. 2nd ed. College Station, TX: Stata Press.
Cochran, W. G., and D. J. Finney. 1979. Chester Ittner Bliss 1899-1979. Biometrics 35: 715-717.
De Luca, G. 2008. SNP and SML estimation of univariate and bivariate binary-choice models. Stata Journal 8: 190-220.
Jones, A. M. 2007. Applied Econometrics for Health Economists: A Practical Guide. 2nd ed. Abingdon, UK: Radcliffe.
Judge, G. G., W. E. Griffiths, R. C. Hill, H. Lütkepohl, and T.-C. Lee. 1985. The Theory and Practice of Econometrics. 2nd ed. New York: Wiley.
Long, J. S. 1997. Regression Models for Categorical and Limited Dependent Variables. Thousand Oaks, CA: Sage.
Long, J. S., and J. Freese. 2014. Regression Models for Categorical Dependent Variables Using Stata. 3rd ed. College Station, TX: Stata Press.

Miranda, A., and S. Rabe-Hesketh. 2006. Maximum likelihood estimation of endogenous switching and sample selection models for binary, ordinal, and count variables. Stata Journal 6: 285-308.
Newson, R. B., and M. Falcaro. 2023. Robit regression in Stata. Stata Journal 23: 658-682.
Pampel, F. C. 2021. Logistic Regression: A Primer. 2nd ed. Thousand Oaks, CA: Sage.
Pedace, R. 2013. Econometrics for Dummies. Hoboken, NJ: Wiley.
Pinzon, E. 2016. Effects of nonlinear models with interactions of discrete and continuous variables: Estimating, graphing, and interpreting. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2016/07/12/effects-for-nonlinear-models-with-interactions-of-discrete-and-continuous-variables-estimating-graphing-and-interpreting/.

Powers, D. A., and Y. Xie. 2008. Statistical Methods for Categorical Data Analysis. 2nd ed. Bingley, UK: Emerald.
Xu, J., and J. S. Long. 2005. Confidence intervals for predicted outcomes in regression models for categorical outcomes. Stata Journal 5: 537-559.

## Also see

[R] probit postestimation - Postestimation tools for probit
[R] biprobit - Bivariate probit regression
[R] brier - Brier score decomposition
[R] glm - Generalized linear models
$[R]$ heckoprobit - Ordered probit model with sample selection
[R] hetprobit - Heteroskedastic probit model
[R] ivprobit - Probit model with continuous endogenous covariates
$[\mathrm{R}]$ logistic - Logistic regression, reporting odds ratios
[R] logit - Logistic regression, reporting coefficients
[R] mprobit - Multinomial probit regression
[R] npregress kernel - Nonparametric kernel regression
[R] npregress series - Nonparametric series regression
$[R]$ roc - Receiver operating characteristic (ROC) analysis
[R] scobit - Skewed logistic regression
[BAYES] bayes: probit - Bayesian probit regression
[CM] cmmprobit - Multinomial probit choice model
[ERM] eprobit - Extended probit regression
[FMM] fmm: probit — Finite mixtures of probit regression models
[LASSO] Lasso intro - Introduction to lasso
[ME] meprobit - Multilevel mixed-effects probit regression
[MI] Estimation - Estimation commands for use with mi estimate
[SVY] svy estimation - Estimation commands for survey data
[XT] xtprobit - Random-effects and population-averaged probit models
[U] 20 Estimation and postestimation commands

## Title

probit postestimation - Postestimation tools for probit

| Postestimation commands | predict | margins |
| :--- | :--- | :--- |
| Remarks and examples | Methods and formulas | Also see |

## Postestimation commands

The following postestimation commands are of special interest after probit:
Command Description

```
estat classification report various summary statistics, including the classification table
estat gof Pearson or Hosmer-Lemeshow goodness-of-fit test
lroc compute area under ROC curve and graph the curve
lsens graph sensitivity and specificity versus probability cutoff
lassogof calculate goodness-of-fit predictions
```

These commands are not appropriate with svy estimation results.

The following standard postestimation commands are also available:

| Command | Description |
| :---: | :---: |
| contrast | contrasts and ANOVA-style joint tests of estimates |
| estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estat (svy) | postestimation statistics for survey data |
| estimates | cataloging estimation results |
| etable | table of estimation results |
| * forecast | dynamic forecasts and simulations |
| * hausman | Hausman's specification test |
| lincom | point estimates, standard errors, testing, and inference for linear combinations of coefficients |
| linktest | link test for model specification |
| * lrtest | likelihood-ratio test |
| margins | marginal means, predictive margins, marginal effects, and average marginal effects |
| marginsplot | graph the results from margins (profile plots, interaction plots, etc.) |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients |
| predict | probabilities, linear predictions and their SEs, etc. |
| predictnl | point estimates, standard errors, testing, and inference for generalized predictions |

```
pwcompare pairwise comparisons of estimates
suest seemingly unrelated estimation
test Wald tests of simple and composite linear hypotheses
testnl Wald tests of nonlinear hypotheses
```

*forecast, hausman, and lrtest are not appropriate with svy estimation results. forecast is also not appropriate with mi estimation results.

## predict

## Description for predict

predict creates a new variable containing predictions such as probabilities, linear predictions, standard errors, deviance residuals, and the equation-level score.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

```
predict [type] newvar [if] [in] [, statistic nooffset rules asif]
```

statistic Description

Main
pr probability of a positive outcome; the default
$\mathrm{xb} \quad$ linear prediction
stdp standard error of the linear prediction
*deviance
deviance residual
score
first derivative of the $\log$ likelihood with respect to $\mathbf{x}_{j} \boldsymbol{\beta}$
Unstarred statistics are available both in and out of sample; type predict . . . if e (sample) ... if wanted only for the estimation sample. Starred statistics are calculated only for the estimation sample, even when if e(sample) is not specified.

## Options for predict

$\qquad$ Main
pr, the default, calculates the probability of a positive outcome.
xb calculates the linear prediction.
stdp calculates the standard error of the linear prediction.
deviance calculates the deviance residual.
score calculates the equation-level score, $\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)$.
nooffset is relevant only if you specified offset (varname) for probit. It modifies the calculations made by predict so that they ignore the offset variable; the linear prediction is treated as $\mathbf{x}_{j} \mathbf{b}$ rather than as $\mathbf{x}_{j} \mathbf{b}+$ offset $_{j}$.
rules requests that Stata use any rules that were used to identify the model when making the prediction. By default, Stata calculates missing for excluded observations.
asif requests that Stata ignore the rules and exclusion criteria and calculate predictions for all observations possible using the estimated parameter from the model.

## margins

## Description for margins

margins estimates margins of response for probabilities and linear predictions.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
```


statistic Description

| pr | probability of a positive outcome; the default |
| :--- | :--- |
| xb | linear prediction |
| stdp | not allowed with margins |
| deviance | not allowed with margins |
| score | not allowed with margins |

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

## Remarks and examples

Remarks are presented under the following headings:
Obtaining predicted values
Performing hypothesis tests

## Obtaining predicted values

Once you have fit a probit model, you can obtain the predicted probabilities by using the predict command for both the estimation sample and other samples; see [U] 20 Estimation and postestimation commands and $[R]$ predict. Here we will make only a few additional comments.
predict without arguments calculates the predicted probability of a positive outcome. With the xb option, predict calculates the linear combination $\mathbf{x}_{j} \mathbf{b}$, where $\mathbf{x}_{j}$ are the independent variables in the $j$ th observation and $\mathbf{b}$ is the estimated parameter vector. This is known as the index function because the cumulative density indexed at this value is the probability of a positive outcome.

In both cases, Stata remembers any rules used to identify the model and calculates missing for excluded observations unless rules or asif is specified. This is covered in the following example.

With the stdp option, predict calculates the standard error of the prediction, which is not adjusted for replicated covariate patterns in the data.

You can calculate the unadjusted-for-replicated-covariate-patterns diagonal elements of the hat matrix, or leverage, by typing
. predict pred
. predict stdp, stdp
. generate hat $=$ stdp^ $2 *$ pred $*(1-$ pred $)$

## > Example 1

In example 4 of $[\mathrm{R}]$ probit, we fit the probit model probit foreign b3.repair. To obtain predicted probabilities, we type

```
. predict p
(option pr assumed; Pr(foreign))
(10 missing values generated)
. summarize foreign p
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline foreign & 58 & .2068966 & .4086186 & 0 & 1 \\
\(p\) & 48 & .25 & .1956984 & .1 & .5
\end{tabular}
```

Stata remembers any rules used to identify the model and sets predictions to missing for any excluded observations. In example 4 of [R] probit, probit omitted the variable 1.repair from our model and excluded 10 observations. When we typed predict $p$, those same 10 observations were again excluded and their predictions set to missing.
predict's rules option uses the rules in the prediction. During estimation, we were told, "1.repair
 probability of success or a positive outcome:

```
. predict p2, rules
(option pr assumed; Pr(foreign))
. summarize foreign p p2
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline foreign & 58 & .2068966 & .4086186 & 0 & 1 \\
p & 48 & .25 & .1956984 & .1 & .5 \\
p2 & 58 & .2068966 & .2016268 & 0 & .5
\end{tabular}
```

predict's asif option ignores the rules and the exclusion criteria and calculates predictions for all observations possible using the estimated parameters from the model:

| Variable | Obs | Mean | Std. dev. | Min | Max |
| :---: | :---: | :---: | :---: | :---: | :---: |
| foreign | 58 | . 2068966 | . 4086186 | 0 | 1 |
| p | 48 | . 25 | . 1956984 | . 1 | . 5 |
| p2 | 58 | . 2068966 | . 2016268 | 0 | . 5 |
| p3 | 58 | . 2931034 | . 2016268 | . 1 | . 5 |

Which is right? By default, predict uses the most conservative approach. If many observations had been excluded due to a simple rule, we could be reasonably certain that the rules prediction is correct. The asif prediction is correct only if the exclusion is a fluke and we would be willing to exclude the variable from the analysis, anyway. Then, however, we should refit the model to include the excluded observations.

## Performing hypothesis tests

After estimation with probit, you can perform hypothesis tests by using the test or testnl command; see [U] 20 Estimation and postestimation commands.

## Methods and formulas

Let index $j$ be used to index observations. Define $M_{j}$ for each observation as the total number of observations sharing $j$ 's covariate pattern. Define $Y_{j}$ as the total number of positive responses among observations sharing $j$ 's covariate pattern. Define $p_{j}$ as the predicted probability of a positive outcome for observation $j$.

For $M_{j}>1$, the deviance residual $d_{j}$ is defined as

$$
d_{j}= \pm\left(2\left[Y_{j} \ln \left(\frac{Y_{j}}{M_{j} p_{j}}\right)+\left(M_{j}-Y_{j}\right) \ln \left\{\frac{M_{j}-Y_{j}}{M_{j}\left(1-p_{j}\right)}\right\}\right]\right)^{1 / 2}
$$

where the sign is the same as the sign of $\left(Y_{j}-M_{j} p_{j}\right)$. In the limiting cases, the deviance residual is given by

$$
d_{j}= \begin{cases}-\sqrt{2 M_{j}\left|\ln \left(1-p_{j}\right)\right|} & \text { if } Y_{j}=0 \\ \sqrt{2 M_{j}\left|\ln p_{j}\right|} & \text { if } Y_{j}=M_{j}\end{cases}
$$

## Also see

[R] probit - Probit regression
[R] estat classification - Classification statistics and table
[R] estat gof - Pearson or Hosmer-Lemeshow goodness-of-fit test
[R] Iroc - Compute area under ROC curve and graph the curve
[R] Isens - Graph sensitivity and specificity versus probability cutoff
[LASSO] lassogof - Goodness of fit after lasso for prediction
[U] 20 Estimation and postestimation commands

## Title

proportion - Estimate proportions

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

proportion produces estimates of proportions, along with standard errors, for the categories identified by the values in each variable of varlist.

## Quick start

Proportions, standard errors, and $95 \%$ CIs for each level of v1
proportion v1
Also compute statistics for v2
proportion v1 v2
Same as above, for each subpopulation defined by the levels of catvar
proportion v 1 v 2 , over(catvar)
Standardizing across strata defined by svar with stratum weight wvar1
proportion v1, stdize(svar) stdweight(wvar1)
Weighting by sampling weight wvar2
proportion v1 [pweight=wvar2]

## Menu

Statistics $>$ Summaries, tables, and tests $>$ Summary and descriptive statistics $>$ Proportions

## Syntax

proportion varlist [if] [in] [weight] [, options]
options
Description
Model
stdize (varname)
stdweight (varname)
nostdrescale
iffin/over
over (varlisto)
SE/Cluster
vce (vcetype)

Reporting
level (\#)
citype (citype)
percent
noheader
display_options
coeflegend
variable identifying strata for standardization
weight variable for standardization
do not rescale the standard weight variable
group over subpopulations defined by varlisto
vcetype may be analytic, cluster clustvar, bootstrap, or jackknife
set confidence level; default is level (95)
method to compute limits of confidence intervals; default is citype(logit)
report estimated proportions as percentages
suppress table header
control column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
display legend instead of statistics
varlist may contain factor variables; see [U] 11.4.3 Factor variables.
Only numeric, nonnegative, integer-valued variables are allowed in varlist.
bootstrap, collect, jackknife, mi estimate, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands.
vce (bootstrap) and vce(jackknife) are not allowed with the mi estimate prefix; see [MI] mi estimate.
Weights are not allowed with the bootstrap prefix; see [R] bootstrap.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
coeflegend does not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.
citype Description
logit calculate logit-transformed confidence intervals; the default
agresti
exact
jeffreys
normal
wald
wilson

## Options

Model
stdize (varname) specifies that the point estimates be adjusted by direct standardization across the strata identified by varname. This option requires the stdweight() option.
stdweight (varname) specifies the weight variable associated with the standard strata identified in the stdize () option. The standardization weights must be constant within the standard strata. nostdrescale prevents the standardization weights from being rescaled within the over() groups. This option requires stdize() but is ignored if the over() option is not specified.

## if/in/over

over (varlist $t_{o}$ ) specifies that estimates be computed for multiple subpopulations, which are identified by the different values of the variables in varlist ${ }_{o}$. Only numeric, nonnegative, integer-valued variables are allowed in over (varlisto).

## SE/Cluster

vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (analytic), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
vce(analytic), the default, uses the analytically derived variance estimator associated with the sample proportion.

## Reporting

level (\#); see [R] Estimation options.
citype (citype) specifies how to compute the limits of confidence intervals. citype may be one of logit (default), agresti, exact, jeffreys, normal, wald, or wilson.
percent specifies that the proportions be reported as percentages.
noheader prevents the table header from being displayed.
display_options: noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap (\#), fvwrapon ( $\overline{\text { style }), ~ c f o r m a t ~(\% f i t) ~, ~ a n d ~ n o l s t r e t c h ; ~ s e e ~[R] ~ E s t i m a-~}$ tion options.

The following option is available with proportion but is not shown in the dialog box: coeflegend; see [R] Estimation options.

## Remarks and examples

## Example 1

We can estimate the proportion of each repair rating in auto2.dta:

marginsplot will produce a graph of the results from proportion:

```
. marginsplot
```

Variables that uniquely identify proportions: rep78

Estimated proportions of rep78 with 95\% Cls


## > Example 2

We can also estimate proportions over groups:

```
. proportion rep78, over(foreign)
Proportion estimation Number of obs = 69
```

|  |  |  | Logit <br> [95\% conf. interval] |  |
| ---: | ---: | :---: | ---: | :--- |
|  | Proportion | Std. err. |  |  |
| rep78@foreign |  |  |  |  |
| Poor Domestic | .0416667 | .0288424 | .0101825 | .1552326 |
| Poor Foreign | 0 | (no observations) |  |  |
| Fair Domestic | .1666667 | .0537914 | .084534 | .3022522 |
| Fair Foreign | 0 | (no observations) |  |  |
| Average Domestic | .5625 | .0716027 | .4184154 | .6967587 |
| Average Foreign | .1428571 | .0763604 | .0458191 | .3664757 |
| Good Domestic | .1875 | .0563367 | .0993684 | .3255432 |
| Good Foreign | .4285714 | .1079898 | .2372889 | .6438783 |
| Excellent Domestic | .0416667 | .0288424 | .0101825 | .1552326 |
| Excellent Foreign | .4285714 | .1079898 | .2372889 | .6438783 |

To see the results as percentages instead of proportions, we add the percent option:

| . proportion rep78, over(foreign) percent |  |
| :---: | ---: | :---: | :---: | :---: |
| Percent estimation |  |

We can now use marginsplot to graph the percentages for each group. We add the bydimension(foreign) option to plot the groups in separate graph panels. The xlabel(, angle(30)) option prevents the $x$-axis labels from running into each other.
. marginsplot, bydimension(foreign) xlabel(, angle(30))
Variables that uniquely identify proportions: rep78 foreign
Estimated percentages of rep78 with $95 \%$ Cls


We estimate that only $19 \%$ of domestic cars have good repair records and only $4 \%$ have excellent repair records. For foreign cars, however, we find that $43 \%$ have good repair records and $43 \%$ have excellent repair records.

## $>$ Example 3

Instead of estimating percentages within the foreign and domestic groupings, we might want to know overall percentages. For instance, what percentage of all cars are foreign and have excellent repair records? What percentage are domestic and have average records? We can obtain all such percentages by specifying an interaction between rep78 and foreign.

| Percent estimation | Number of obs $=69$ |  |  |  |
| :---: | :---: | :---: | :---: | :---: |
|  | Percent | Std. err. | Logit |  |
| rep78\#foreign |  |  |  |  |
| Poor\#Domestic | 2.90 | 2.02 | 0.71 | 11.11 |
| Poor\#Foreign | 0.00 | (no observ |  |  |
| Fair\#Domestic | 11.59 | 3.85 | 5.83 | 21.74 |
| Fair\#Foreign | 0.00 | (no observ |  |  |
| Average\#Domestic | 39.13 | 5.88 | 28.21 | 51.26 |
| Average\#Foreign | 4.35 | 2.46 | 1.38 | 12.86 |
| Good\#Domestic | 13.04 | 4.05 | 6.85 | 23.44 |
| Good\#Foreign | 13.04 | 4.05 | 6.85 | 23.44 |
| Excellent\#Domestic | 2.90 | 2.02 | 0.71 | 11.11 |
| Excellent\#Foreign | 13.04 | 4.05 | 6.85 | 23.44 |

Looking at the last line of this output, we estimate that $13 \%$ of all cars are foreign with excellent repair records.

## Stored results

proportion stores the following in e():
Scalars

| $e(N)$ | number of observations |
| :--- | :--- |
| $e\left(N_{\text {_over }}\right)$ | number of subpopulations |
| $e\left(N_{\text {stdize }}\right)$ | number of standard strata |
| $e\left(N_{c l u}\right)$ | number of clusters |
| $e\left(k \_e q\right)$ | number of equations in $e(b)$ |
| $e(d f-r)$ | sample degrees of freedom |
| $e(r a n k)$ | rank of $e(V)$ |

Macros
e(cmd) proportion
e(cmdline) command as typed
e(varlist) varlist
e(stdize)
e(stdweight)
varname from stdize()
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(over)
e(vce)
e(vcetype)
varname from stdweight()
weight type
weight expression
title in estimation output
name of cluster variable
varlist from over()
vcetype specified in vce()
e(properties)
title used to label Std. err.
e(estat_cmd)
b V
e(marginsnotok)
program used to implement estat
predictions disallowed by margins
Matrices
e(b)
e(V)
vector of proportion estimates
(co)variance estimates
e(_N)
e(_N_stdsum)
vector of numbers of nonmissing observations
number of nonmissing observations within the standard strata
e(_p_stdize)
standardizing proportions
e(freq)
vector of frequency estimates
e(error)
error code corresponding to e(b)
Functions
e(sample) marks estimation sample
In addition to the above, the following is stored in r() :
Matrices
$r$ (table) matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

The point estimate of a proportion, $\widehat{p}$, is computed as the mean of an indicator variable; see [R] mean.

Methods and formulas are presented under the following headings:

## Confidence intervals

Survey data and sampling weights

## Confidence intervals

For an overview of confidence interval methods for binomial proportions, see Dean and Pagano (2015).

Given $k$ successes of $n$ trials, the estimated proportion (probability of a success) is $\widehat{p}=k / n$ with estimated standard error $\widehat{s}=\sqrt{\widehat{p}(1-\widehat{p}) / n}$.

For weighted data, $k=\sum_{i=1}^{N} w_{i} \mathbf{1}$ (obs $i$ is a success) and $n=\sum_{i=1}^{N} w_{i}$, where $N$ is the total number of observations, $w_{i}$ is the weight for the $i$ th observation, and $\mathbf{1}(\cdot)$ denotes the indicator function that evaluates to 1 if observation $i$ is a success and 0 otherwise.

The logit-transformed confidence interval is given by

$$
\log \left(\frac{\widehat{p}}{1-\widehat{p}}\right) \pm t_{1-\alpha / 2, \nu} \frac{\widehat{s}}{\widehat{p}(1-\widehat{p})}
$$

where $t_{p, \nu}$ is the $p$ th quantile of Student's $t$ distribution with $\nu$ degrees of freedom.
The endpoints of this confidence interval are transformed back to the proportion metric by using the inverse of the logit transform

$$
f^{-1}(y)=\frac{e^{y}}{1+e^{y}}
$$

Hence, the displayed confidence intervals for proportions are

$$
f^{-1}\left\{\ln \left(\frac{\widehat{p}}{1-\widehat{p}}\right) \pm t_{1-\alpha / 2, \nu} \frac{\widehat{s}}{\widehat{p}(1-\widehat{p})}\right\}
$$

The Wald-type $100(1-\alpha) \%$ confidence interval is given by

$$
\widehat{p} \pm t_{1-\alpha / 2, \nu} \widehat{s}
$$

The formulas below (for the Wilson, exact, Jeffreys, and Agresti-Coull intervals) are slightly different when using the svy prefix, pweights, or the vce(cluster clustvar) option; see Survey data and sampling weights below.

The Wilson interval is given by

$$
\frac{\widehat{p}+z_{1-\alpha / 2}^{2} / 2 n \pm z_{1-\alpha / 2} \sqrt{\widehat{p}(1-\widehat{p}) / n+z_{1-\alpha / 2}^{2} / 4 n^{2}}}{1+z_{1-\alpha / 2}^{2} / n}
$$

where $z_{p}$ is the $p$ th quantile of the standard normal distribution.
The exact (Clopper-Pearson) interval is given by

$$
\left\{\frac{\nu_{1} F_{\alpha / 2, \nu_{1}, \nu_{2}}}{\nu_{2}+\nu_{1} F_{\alpha / 2, \nu_{1}, \nu_{2}}} ; \quad \frac{\nu_{3} F_{1-\alpha / 2, \nu_{3}, \nu_{4}}}{\nu_{4}+\nu_{3} F_{\alpha / 2, \nu_{3}, \nu_{4}}}\right\}
$$

where $\nu_{1}=2 k, \nu_{2}=2(n-k+1), \nu_{3}=2(k+1), \nu_{4}=2(n-k)$, and $F_{p, \nu_{1}, \nu_{2}}$ is the $p$ th quantile of an $F$ distribution with $\nu_{1}$ and $\nu_{2}$ degrees of freedom.

The Jeffreys interval is given by

$$
\left\{\operatorname{Beta}_{\alpha / 2, \alpha_{1}, \beta_{1}} ; \operatorname{Beta}_{1-\alpha / 2, \alpha_{1}, \beta_{1}}\right\}
$$

where $\alpha_{1}=k+0.5, \beta_{1}=n-k+0.5$, and $\operatorname{Beta}_{p, \alpha_{1}, \beta_{1}}$ is the $p$ th quantile of a Beta distribution with $\alpha_{1}$ and $\beta_{1}$ degrees of freedom.

The Agresti-Coull interval is given by

$$
\widetilde{p} \pm z_{1-\alpha / 2} \sqrt{\widetilde{p}(1-\widetilde{p}) / \widetilde{n}}
$$

where $\widetilde{k}=k+z_{1-\alpha / 2}^{2} / 2, \widetilde{n}=n+z_{1-\alpha / 2}^{2}$, and $\widetilde{p}=\widetilde{k} / \widetilde{n}$.

## Survey data and sampling weights

With the svy prefix, pweights, or the vce(cluster clustvar) option, the Wilson, exact, Jeffreys, and Agresti-Coull intervals use $k^{*}$ in place of $k$ and $n^{*}$ in place of $n$, where $k^{*}=n^{*} \widehat{p}$,

$$
n^{*}=\frac{\widehat{p}(1-\widehat{p})}{\widehat{s}^{2}}\left\{\frac{z_{1-\alpha / 2}}{t_{1-\alpha / 2, \nu}}\right\}^{2}
$$

and $\widehat{s}$ is the linearized standard error estimate. For more details about the linearized standard error estimate, see Ratios and other functions of survey data in [SVY] Variance estimation.

## References

Cochran, W. G. 1977. Sampling Techniques. 3rd ed. New York: Wiley.
Dean, N., and M. Pagano. 2015. Evaluating confidence interval methods for binomial proportions in clustered surveys. Journal of Survey Statistics and Methodology 3: 484-503. https://doi.org/10.1093/jssam/smv024.

Stuart, A., and J. K. Ord. 1994. Kendall's Advanced Theory of Statistics: Distribution Theory, Vol. 1. 6th ed. London: Arnold.

## Also see

[R] proportion postestimation - Postestimation tools for proportion
[R] mean - Estimate means
[R] ratio - Estimate ratios
[R] total - Estimate totals
[MI] Estimation - Estimation commands for use with mi estimate
[SVY] Direct standardization - Direct standardization of means, proportions, and ratios
[SVY] Poststratification - Poststratification for survey data
[SVY] Subpopulation estimation - Subpopulation estimation for survey data
[SVY] svy estimation - Estimation commands for survey data
[SVY] Variance estimation - Variance estimation for survey data
[U] 20 Estimation and postestimation commands

## Title

proportion postestimation - Postestimation tools for proportion

Postestimation commands Remarks and examples Also see

## Postestimation commands

The following postestimation commands are available after proportion:

| Command | Description |
| :---: | :---: |
| contrast | contrasts and ANOVA-style joint tests of estimates |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estat (svy) | postestimation statistics for survey data |
| estimates | cataloging estimation results |
| etable | table of estimation results |
| lincom | point estimates, standard errors, testing, and inference for linear combinations of coefficients |
| marginsplot | graph the results from proportion |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients |
| pwcompare | pairwise comparisons of estimates |
| test | Wald tests of simple and composite linear hypotheses |
| testnl | Wald tests of nonlinear hypotheses |

## Remarks and examples

## > Example 1

In example 2 of [R] proportion, we computed the proportions of cars with different repair records for each group, foreign or domestic. We use test to test whether the proportion of cars with repair record equal to 4 is the same for domestic and foreign cars.

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. proportion rep78, over(foreign)
    (output omitted)
. test 4.rep78@0.foreign=4.rep78@1.foreign
    ( 1) 4.rep78@Obn.foreign - 4.rep78@1.foreign = 0
            F( 1, 68) = 3.92
            Prob > F = 0.0518
```

There is not a significant difference between those proportions at the $5 \%$ level.

## > Example 2

Continuing with auto.dta from example 1, we generate a new variable, highprice, that indicates if the price is larger than $\$ 5,000$ and then use proportion to see the proportion of cars with high price among domestic and foreign cars separately.

| Proportion estimation |  |  | Number of obs $=74$ |  |
| :---: | :---: | :---: | :---: | :---: |
|  | Proportion | Std. err. | [95\% conf | interval] |
| highprice@foreign |  |  |  |  |
| 0 Domestic | . 5576923 | . 0688744 | . 4195373 | . 6874611 |
| 0 Foreign | . 3636364 | . 1025593 | . 191094 | . 5802222 |
| 1 Domestic | . 4423077 | . 0688744 | . 3125389 | . 5804627 |
| 1 Foreign | . 6363636 | . 1025593 | . 4197778 | . 808906 |

We will compute the odds ratio of having a high price in group Foreign to having a high price in group Domestic. Usually, odds ratios are computed by using the logistic command, but here we will perform the computation by using nlcom after proportion.

```
. nlcom OR: (_b[1.highprice@1.foreign]/_b[0.highprice@1.foreign])/(_b[1.highpri
> ce@0.foreign]/_b[0.highprice@0.foreign])
    OR: (_b[1.highprice@1.foreign]/_b[0.highprice@1.foreign])/(_b[1.high
> price@0.foreign]/_b[0.highprice@0.foreign])
\begin{tabular}{r|rlcrrr}
\hline Proportion & Coefficient & Std. err. & z & P>|z| & [95\% conf. interval] \\
\hline OR & 2.206522 & 1.155825 & 1.91 & 0.056 & -.0588533 & 4.471897 \\
\hline
\end{tabular}
```

This is the same odds ratio that we would obtain from

```
. logistic highprice foreign
```

The odds ratio is slightly larger than 2 , which means that the odds of having a high price among foreign cars are more than twice that of having a high price among domestic cars.

## Also see

[R] proportion - Estimate proportions
[U] 20 Estimation and postestimation commands

## Title

prtest — Tests of proportions

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options for prtest | Options for prtesti | Remarks and examples | Stored results |
| Methods and formulas | References | Also see |  |

## Description

prtest performs tests on the equality of proportions using large-sample statistics. The test can be performed for one sample against a hypothesized population value or for no difference in population proportions estimated from two samples. Clustered data are supported.
prtesti is the immediate form of prtest; see [U] 19 Immediate commands.

## Quick start

One-sample test that the proportion of 1 s in v is equal to 0.1

```
prtest v == 0.1
```

Same as above, but using the $90 \%$ confidence level and adjusting for clustering with clusters defined by cvar and an intraclass correlation of 0.5

```
prtest v == 0.1, level(90) cluster(cvar) rho(0.5)
```

Test that the proportion of 1 s in v is equal between two groups defined by catvar

```
prtest v, by(catvar)
```

Same as above, and adjust for clustering with clusters defined by cvar and an intraclass correlation of 0.5 in the two groups
prtest v, by (catvar) cluster(cvar) rho(0.5)
Test equality of proportions between v 1 and v 2
prtest v1 == v2
Test $p_{1}=p_{2}$ if $\widehat{p}_{1}=0.10, \widehat{p}_{2}=0.17, n_{1}=29$, and $n_{2}=36$
prtesti 290.10360 .17

## Menu

prtest
Statistics $>$ Summaries, tables, and tests $>$ Classical tests of hypotheses $>$ Proportion test

## prtesti

Statistics $>$ Summaries, tables, and tests $>$ Classical tests of hypotheses $>$ Proportion test calculator

## Syntax

One-sample test of proportion

$$
\text { prtest varname }==\#_{p}[\text { if }][\text { in }][\text {, onesampleopts }]
$$

Two-sample test of proportions using groups

$$
\text { prtest varname }[\text { if }][\text { in }] \text {, by (groupvar) [twosamplegropts }]
$$

Two-sample test of proportions using variables

$$
\text { prtest } \text { varname }_{1}==\text { varname }_{2}[\text { if }][\text { in }][\text { level (\#) }]
$$

Immediate form of one-sample test of proportion

```
prtesti # obs1 # p1 #p2 [, level(#) count ]
```

Immediate form of two-sample test of proportions

```
prtesti # obs1 # p1 # obs2 # #2 [, level(#) count ]
```

onesampleopts Description
Main

| level(\#) | confidence level; default is level (95) |
| :--- | :--- |
| cluster (varname) | variable defining the clusters |
| rho(\#) | intraclass correlation |

twosamplegropts Description

Main

| *by (groupvar) | variable defining the groups |
| :--- | :--- |
| level(\#) | confidence level; default is level (95) |
| cluster (varname) | variable defining the clusters |
| rho(\#) | common intraclass correlation |
| rho1(\#) | intraclass correlation for group 1 |
| rho2(\#) | intraclass correlation for group 2 |

*by (groupvar) is required.
by is allowed with prtest, and collect is allowed with prtest and prtesti; see [U] 11.1.10 Prefix commands.

## Options for prtest

$\qquad$ may be specified.
level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals.
cluster (varname) specifies the variable that identifies clusters. The cluster() option is required to adjust the computation for clustering.
rho(\#) specifies the intraclass correlation for a one-sample test or the common intraclass correlation for a two-sample test. The rho() option is required to adjust the computation for clustering for a one-sample test.
rho1(\#) specifies the intraclass correlation of the first group for a two-sample test using groups. The rho() option or both rho1() and rho2() options are required to adjust the computation for clustering.
rho2(\#) specifies the intraclass correlation of the second group for a two-sample test using groups. The rho() option or both rho1() and rho2() options are required to adjust the computation for clustering.

## Options for prtesti

level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals.
count specifies that integer counts instead of proportions be used in the immediate forms of prtest. In the first syntax, prtesti expects that $\#_{\mathrm{obs} 1}$ and $\#_{p 1}$ are counts- $\#_{p 1} \leq \#_{\mathrm{obs} 1}$-and $\#_{p 2}$ is a proportion. In the second syntax, prtesti expects that all four numbers are integer counts, that $\#_{\mathrm{obs} 1} \geq \#_{p 1}$, and that $\#_{\mathrm{obs} 2} \geq \#_{p 2}$.

## Remarks and examples

Remarks are presented under the following headings:
Tests of proportions
Adjust for clustering
Immediate form

## Tests of proportions

The prtest output follows the output of ttest in providing a lot of information. Each proportion is presented along with a confidence interval. The appropriate one- or two-sample test is performed, and the two-sided and both one-sided results are included at the bottom of the output. For a twosample test, the calculated difference is also presented with its confidence interval. This command may be used for both large-sample testing and large-sample interval estimation. For one-sample tests of proportions with small-sample sizes and to obtain exact $p$-values, researchers should use bitest; see [R] bitest.
$\lambda$ Example 1: One-sample test of proportion
In the first form, prtest tests whether the mean of the sample is equal to a known constant. Assume that we have a sample of 74 automobiles. We wish to test whether the proportion of automobiles that are foreign is different from $40 \%$.

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
```



The test indicates that we cannot reject the hypothesis that the proportion of foreign automobiles is 0.40 at the $5 \%$ significance level.

## > Example 2: Two-sample test of proportions

We have two headache remedies that we give to patients. Each remedy's effect is recorded as 0 for failing to relieve the headache and 1 for relieving the headache. We wish to test the equality of the proportion of people relieved by the two treatments.


We find that the proportions are statistically different from each other at any level greater than $3.9 \%$.

## Adjust for clustering

When observations are not independent and can be grouped into clusters, we need to adjust for clustering in a proportion test. For example, in a cluster randomized design, groups of individuals are randomized instead of individuals. To adjust for clustering, we need to specify the cluster identifier variable in the cluster () option. In the case of a one-sample proportion test, we need to also specify the intraclass correlation in the rho() option. In the case of a two sample proportions test, we need to also specify the common population intraclass correlation in the rho() option or group-specific population intraclass correlations in the rho1() and rho2() options.

## > Example 3: One-sample test of proportion, adjusting for clusters

Consider data from Hujoel, Moulton, and Loesche (1990) on the accuracy of an enzymatic diagnostic test (EDT) of bacterial infections for 29 patients with multiple sites. The EDT was conducted on each site, a specific area in a patient's mouth, to determine infection by two strings of bacteria. A separate reference test was also conducted on each site with an antibody assay against the two strings of bacteria. The data record whether there was a positive EDT result at each infected site, a true positive result.

We want to test whether the proportion of infected sites that were correctly diagnosed by the EDT is different from 0.6. Because we have multiple infections per patient, we cluster by the patient-identifier subject and use a value of 0.2 from Ahn, Heo, and Zhang $(2015,33)$ for the intrapatient correlation.

To perform the test, we specify the cluster (subject) and rho(0.2) options:


We do not find statistical evidence to reject the null hypothesis of $H_{0}: P_{\text {infection }}=0.6$ versus the two-sided alternative $H_{a}: P_{\text {infection }} \neq 0.6$ at the $5 \%$ significance level; the $p$-value $=0.2660>0.05$.
> Example 4: Two-sample test of proportions using groups, adjusting for clusters
Consider a dataset provided by Hayes and Moulton (2009), which contains a random subsample of the original participants in a cluster randomized trial of a pneumococcal conjugate vaccine in American Indian populations in the southwestern United States. There are two groups of infants with 18 clusters in each group. The control group received a meningococcal Conjugate vaccine ( MnCC ), and the experimental group received the seven-valent pneumococcal conjugate vaccine (PnCRM7). The two groups are identified by the vaccine variable, and the pneumonia variable records 1 if an infant had at least one bacterial pneumonia episode and 0 otherwise. These data are originally from O'Brien et al. (2003).

We want to test the equality of the proportion of cases of pneumonia in the two vaccine groups. We assume a common known intraclass correlation of 0.02 . To perform the test, we type

```
. use https://www.stata-press.com/data/r18/pneumoniacrt
(Bacterial pneumonia episodes data from CRT (Hayes and Moulton 2009))
. prtest pneumonia, by(vaccine) cluster(cluster) rho(0.02)
```

Two-sample test of proportions
Cluster variable: cluster
Group: MnCC Group: PnCRM7

| Number of obs | $=$ | 238 |  | Number of obs |
| :--- | ---: | :--- | :--- | :--- |
| Number of clusters | $=$ | 18 |  | 211 |
| Avg. cluster size | $=$ | 13.22 | Number of clusters | $=$ |
| CV cluster size | $=$ | 0.9605 | Avg. cluster size | $=$ |
| Intraclass corr. | $=$ | 0.0200 | CV cluster size | $=$ |


| Group | Mean | Std. err. | z | $P>\|z\|$ | [95\% con | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| MnCC | . 2226891 | . 0329017 |  |  | . 1582029 | . 2871753 |
| PnCRM7 | . 1658768 | . 0299027 |  |  | . 1072686 | . 224485 |
| diff | . 0568123 | . 04446 |  |  | -. 0303278 | . 1439524 |
|  | under H0: | . 0447641 | 1.27 | 0.204 |  |  |
| diff $=\operatorname{prop}(\mathrm{MnCC})-\operatorname{prop}($ PnCRM $) ~$ |  |  |  |  | $z=1.2691$ |  |

    HO: diff \(=0\)
    Ha: diff < 0
$\operatorname{Pr}(Z<z)=0.8978$

Ha: diff $!=0$
$\operatorname{Pr}(|Z|>|z|)=0.2044$

Ha: diff > 0
$\operatorname{Pr}(Z>z)=0.1022$

We do not find statistical evidence to reject the null hypothesis of $H_{0}: P_{\text {diff }}=0$ versus the two-sided alternative $H_{a}: P_{\text {diff }} \neq 0$ at the $5 \%$ significance level; the $p$-value $=0.2044>0.05$.

## Immediate form

## > Example 5: Immediate form of one-sample test of proportion

prtesti is like prtest, except that you specify summary statistics rather than variables as arguments. For instance, we are reading an article that reports the proportion of registered voters among 50 randomly selected eligible voters as 0.52 . We wish to test whether the proportion is 0.7 :

| . prtesti 50 0.52 0.70 |
| :--- |
| One-sample test of proportion |

## > Example 6: Immediate form of two-sample test of proportions

To judge teacher effectiveness, we wish to test whether the same proportion of people from two classes will answer an advanced question correctly. In the first classroom of 30 students, $40 \%$ answered the question correctly, whereas in the second classroom of 45 students, $67 \%$ answered the question correctly.

| Two-sample test of proportions |  |  |  | x : Number of obs $=30$ |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | Mean | Std. err. | z | $P>\|z\|$ | [95\% conf | interval] |
| x | . 4 | . 0894427 |  |  | . 2246955 | . 5753045 |
| y | . 67 | . 0700952 |  |  | . 532616 | . 807384 |
| diff | -. 27 | . 1136368 |  |  | -. 4927241 | -. 0472759 |
|  | under H0: | . 1169416 | -2.31 | 0.021 |  |  |
| diff $=\operatorname{prop}(\mathrm{x})-\operatorname{prop}(\mathrm{y}) \quad \mathrm{z}=-2.3088$ |  |  |  |  |  |  |
| HO: diff = 0 |  |  |  |  |  |  |
| Ha: diff < 0 |  | Ha: diff != 0 |  |  | Ha: diff > 0 |  |
| $\operatorname{Pr}(\mathrm{Z}<\mathrm{z})=0.0105$ |  | $\operatorname{Pr}(\|Z\|>\|z\|)=0.0210$ |  |  | $\operatorname{Pr}(\mathrm{Z}>\mathrm{z})=0.9895$ |  |

## Stored results

One-sample prtest and prtesti store the following in r() :
Scalars

```
r(N) sample size
r(P) sample proportion
r(se) standard error of sample proportion
r(lb) lower confidence bound of sample proportion
r(ub) upper confidence bound of sample proportion
r(z) z statistic
r(p_l) lower one-sided p-value
r(p) two-sided p-value
r(p_u) upper one-sided p-value
r(level) confidence level
```

Cluster-adjusted one-sample prtest also stores the following in r() :
Scalars

```
r(K) number of clusters K
    r(M) cluster size M
    r(rho) intraclass correlation
    r(CV_cluster) coefficient of variation for cluster sizes
```

Two-sample prtest and two-sample prtesti store the following in $r()$ :
Scalars

| $r(N 1)$ | sample size of population one |
| :--- | :--- |
| $r(N 2)$ | sample size of population two |
| $r(P 1)$ | sample proportion for population one |
| $r(P 2)$ | sample proportion for population two |
| $r\left(P \_d i f f\right)$ | difference of proportions |
| $r(\operatorname{se} 1)$ | standard error of population-one sample proportion |
| $r(\operatorname{se2})$ | standard error of population-two sample proportion |

```
r(se_diff) standard error of the difference of proportions
r(se_diff0) standard error of the difference of proportions under H0
r(lb1)
r(ub1)
r(lb2)
r(ub2)
r(lb_diff)
r(ub_diff)
r(z)
r(p_l)
r(p)
r(p_u)
r(level)
standard error of the difference of proportions standard error of the difference of proportions under \(H_{0}\) lower confidence bound of population-one sample proportion upper confidence bound of population-one sample proportion lower confidence bound of population-two sample proportion upper confidence bound of population-two sample proportion lower confidence bound of the difference of proportions upper confidence bound of the difference of proportions \(z\) statistic
lower one-sided \(p\)-value
two-sided \(p\)-value
upper one-sided \(p\)-value
r(level) confidence level
```

Cluster-adjusted two-sample prtest using the by () option also stores the following in $r()$ :
Scalars
r(K1)
$r(K 2)$
$r$ (M1) population-one cluster size $M_{1}$
$r$ (M2) population-two cluster size $M_{2}$
r(rho)
r(rho1)
r(rho2)
r(CV_cluster1)
r(CV_cluster2)
population-one number of clusters $K_{1}$ population-two number of clusters $K_{2}$ common intraclass correlation population-one intraclass correlation population-two intraclass correlation
population-one coefficient of variation for cluster sizes population-two coefficient of variation for cluster sizes

## Methods and formulas

Remarks are presented under the following headings:
One-sample test Two-sample test
For all the tests below, the test statistic $z$ has an asymptotic standard normal distribution, and the $p$-value is computed as

$$
p= \begin{cases}1-\Phi(z) & \text { for an upper one-sided test } \\ \Phi(z) & \text { for a lower one-sided test } \\ 2\{1-\Phi(|z|)\} & \text { for a two-sided test }\end{cases}
$$

where $\Phi(\cdot)$ is the cdf of a standard normal distribution and $|z|$ is an absolute value of $z$.
See Acock (2023, 158-164) for additional examples of tests of proportions using Stata.

## One-sample test

Let $n$ be the number of observations, $\widehat{p}$ be the observed proportion, and $\widehat{q}=1-\widehat{p}$.
The one-tailed and two-tailed tests of a population proportion use an asymptotically normally distributed test statistic calculated as

$$
z=\frac{\widehat{p}-p_{0}}{s_{0}}
$$

where $p_{0}$ is the hypothesized proportion, $q_{0}=1-p_{0}$, and $s_{0}=\sqrt{p_{0} q_{0} / n}$ is the standard error of $\widehat{p}$ under the null hypothesis of $p=p_{0}$.

A large-sample $100(1-\alpha) \%$ confidence interval for a proportion $p$ is

$$
\widehat{p} \pm z_{1-\alpha / 2} s
$$

where $s=\sqrt{\widehat{p} \widehat{q} / n}$ and $z_{1-\alpha / 2}$ is the $(1-\alpha / 2)$ th quantile of the standard normal distribution.
With clustered data, suppose that there are $K$ clusters, each of size $M_{i}$ such that $n=\sum_{i=1}^{K} M_{i}$. Let $\rho$ be the intraclass correlation. Following Ahn, Heo, and Zhang (2015), we assume that the cluster sizes $M_{i}$ are independent and identically distributed. Let $C_{\text {adj }}$ be the adjustment to the standard error for clustered data,

$$
C_{\mathrm{adj}}=\sqrt{\sum_{i=1}^{K} M_{i}\left\{1+\rho\left(M_{i}-1\right)\right\} / n}
$$

such that $s_{0, \mathrm{cl}}=C_{\mathrm{adj}} s_{0}$ and $s_{\mathrm{cl}}=C_{\mathrm{adj}} s$.
$C_{\text {adj }}$ can be equivalently written as

$$
C_{\mathrm{adj}}=\sqrt{1+\rho(\bar{M}-1)+\rho \bar{M} \mathrm{CV}_{\mathrm{cl}}^{2}}
$$

where $\bar{M}=\sum_{i=1}^{K} M_{i} / K$ is the average cluster size and $\mathrm{CV}_{\mathrm{cl}}$ is the coefficient of variation for cluster sizes:

$$
\mathrm{CV}_{\mathrm{cl}}=\frac{\sqrt{\sum_{i=1}^{K}\left(M_{i}-\bar{M}\right)^{2} / K}}{\bar{M}}
$$

To adjust the test statistic $z$ and the confidence interval for clustering, replace $s_{0}$ with $s_{0, \mathrm{cl}}$ and $s$ with $s_{\mathrm{cl}}$ in the corresponding formulas. In the presence of clustering, the test statistic $z$ is asymptotically normally distributed conditional on the empirical distribution of $M_{i}$ 's.

## Two-sample test

Let $n_{1}$ be the number of observations in population one and $n_{2}$ be the number of observations in population two, $\widehat{p}_{1}$ be the observed proportion in population one and $\widehat{p}_{2}$ be the observed proportion in population two, and $\widehat{q}_{1}=1-\widehat{p}_{1}$ and $\widehat{q}_{2}=1-\widehat{p}_{2}$. Let $x_{1}$ and $x_{2}$ be the total number of successes in the two populations.

A test of the difference of two proportions uses an asymptotically normally distributed test statistic calculated as

$$
z=\frac{\widehat{p}_{1}-\widehat{p}_{2}}{s_{d 0}}
$$

where $s_{d 0}=\sqrt{\widehat{p}_{p} \widehat{q}_{p}\left(1 / n_{1}+1 / n_{2}\right)}$ is the standard error of $\widehat{p}_{1}-\widehat{p}_{2}$ under the null hypothesis of $p_{1}=p_{2}$, with $\widehat{p}_{p}=\left(x_{1}+x_{2}\right) /\left(n_{1}+n_{2}\right)$ and $\widehat{q}_{p}=1-\widehat{p}_{p}$.

The $100(1-\alpha) \%$ confidence interval for the difference of two proportions is given by

$$
\left(\widehat{p}_{1}-\widehat{p}_{2}\right) \pm z_{1-\alpha / 2} \sqrt{s_{1}^{2}+s_{2}^{2}}
$$

where $s_{1}=\sqrt{\widehat{p}_{1} \widehat{q}_{1} / n_{1}}$ and $s_{2}=\sqrt{\widehat{p}_{2} \widehat{q}_{2} / n_{2}}$ are the standard errors of the two sample proportions and $z_{1-\alpha / 2}$ is the $(1-\alpha / 2)$ th quantile of the standard normal distribution.

With clustered data, suppose that there are $K_{1}$ and $K_{2}$ clusters in population one and population two with the corresponding average cluster sizes of $\bar{M}_{1}$ and $\bar{M}_{2}$. Let $\rho_{1}$ and $\rho_{2}$ be the intraclass correlations and $\mathrm{CV}_{\mathrm{cl}, 1}$ and $\mathrm{CV}_{\mathrm{cl}, 2}$ be the coefficients of variation for cluster sizes for population one and population two. Let $C_{\mathrm{adj}, 1}$ and $C_{\mathrm{adj}, 2}$ be the adjustments to standard errors of the two sample proportions for clustered data, defined analogously to $C_{\text {adj }}$ in One-sample test for each population.

Let $s_{d 0, \mathrm{cl}}=\sqrt{\widehat{p}_{p} \widehat{q}_{p}\left(C_{\mathrm{adj}, 1}^{2} / n_{1}+C_{\mathrm{adj}, 2}^{2} / n_{2}\right)}$ be the standard error of $\widehat{p}_{1}-\widehat{p}_{2}$ under the null hypothesis of $p_{1}=p_{2}$ adjusted for clustered data. Also, let $s_{1, \mathrm{cl}}=C_{\text {adj, } 1} s_{1}$ and $s_{2, \mathrm{cl}}=C_{\text {adj, } 2} s_{2}$ be the standard errors of $\widehat{p}_{1}$ and $\widehat{p}_{2}$ adjusted for clustered data. To adjust the two-sample test statistic and the confidence interval for clustering, replace $s_{d 0}$ with $s_{d 0, \mathrm{cl}}, s_{1}$ with $s_{1, \mathrm{cl}}$, and $s_{2}$ with $s_{2, \mathrm{cl}}$ in the corresponding formulas.

## References

Acock, A. C. 2023. A Gentle Introduction to Stata. Rev. 6th ed. College Station, TX: Stata Press.
Ahn, C., M. Heo, and S. Zhang. 2015. Sample Size Calculations for Clustered and Longitudinal Outcomes in Clinical Research. Boca Raton, FL: CRC Press.
Hayes, R. J., and L. H. Moulton. 2009. Cluster Randomised Trials. Boca Raton, FL: CRC Press.
——. 2017. Cluster Randomised Trials. 2nd ed. Boca Raton, FL: CRC Press.
Hujoel, P. P., L. H. Moulton, and W. J. Loesche. 1990. Estimation of sensitivity and specificity of site-specific diagnostic tests. Journal of Periodontal Research 25: 193-196. https://doi.org/10.1111/j.1600-0765.1990.tb00903.x.
O’Brien, K. L., L. H. Moulton, R. Reid, R. Weatherholt, J. Oski, L. B. Brown, G. Kumar, A. Parkinson, D. Hu, J. Hackell, I. Chang, R. Kohberger, G. Siber, and M. Santosham. 2003. Efficacy and safety of seven-valent conjugate pneumococcal vaccine in American Indian children: Group randomised trial. Lancet 362: 355-361. https://doi.org/10.1016/S0140-6736(03)14022-6.

## Also see

[R] bitest - Binomial probability test
[R] proportion - Estimate proportions
$[\mathrm{R}]$ ttest $-t$ tests (mean-comparison tests)
[MV] hotelling - Hotelling's $T^{2}$ generalized means test
[PSS-2] power oneproportion - Power analysis for a one-sample proportion test
[PSS-2] power oneproportion, cluster - Power analysis for a one-sample proportion test, CRD
[PSS-2] power twoproportions - Power analysis for a two-sample proportions test
[PSS-2] power twoproportions, cluster - Power analysis for a two-sample proportions test, CRD

## Title

pwcompare - Pairwise comparisons

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

pwcompare performs pairwise comparisons across the levels of factor variables from the most recently fit model. pwcompare can compare estimated cell means, marginal means, intercepts, marginal intercepts, slopes, or marginal slopes-collectively called margins. pwcompare reports the comparisons as contrasts (differences) of margins along with significance tests or confidence intervals for the contrasts. The tests and confidence intervals can be adjusted for multiple comparisons.
pwcompare can be used with svy estimation results; see [SVY] svy postestimation.
See [R] margins, pwcompare for performing pairwise comparisons of margins of linear and nonlinear predictions.

## Quick start

All pairwise comparisons of the means of y across levels of a after regress y i.a pwcompare a

Same as above, and report test statistics and $p$-values for tests of differences in means pwcompare a, effects

Adjust $p$-values and confidence intervals for multiple comparisons using Tukey's method pwcompare a, effects mcompare(tukey)

Same as above, but adjust for multiple comparisons using Bonferroni's method pwcompare a, effects mcompare(bonferroni)

Report means for the levels of a, and group those that are not significantly different pwcompare a, groups

Pairwise comparisons of cell means after regress y1 a\#\#b pwcompare a\#b

Pairwise comparisons of the marginal means of a
pwcompare a
Pairwise comparisons of slopes for continuous $x$ after regress $y 1$ a\#\#c. $x$ pwcompare a\#c.x

Pairwise comparisons of $\log$ odds after logit y2 i.a pwcompare a

Pairwise comparisons of the means of y2 across levels of a after mvreg y1 y2 y3 = i.a pwcompare a, equation(y2)

Same as above, but report pairwise comparisons of a for each equation pwcompare a, atequations

Pairwise comparisons of overall margins of $\mathrm{y} 1, \mathrm{y} 2$, and y 3 pwcompare _eqns

## Menu

Statistics $>$ Postestimation

## Syntax

pwcompare marginlist [, options]
where marginlist is a list of factor variables or interactions that appear in the current estimation results or _eqns to reference equations. The variables may be typed with or without the i. prefix, and you may use any factor-variable syntax:
. pwcompare i.sex i.group i.sex\#i.group
. pwcompare sex group sex\#group
. pwcompare sex\#\#group
options
Description
Main
mcompare (method)
asobserved
Equations
equation (eqspec)
atequations
Advanced
emptycells (empspec)
noestimcheck

Reporting
level(\#)
cieffects
pveffects
effects
cimargins
groups
sort
post
display_options
eform_option
df (\#)
adjust for multiple comparisons; default is mcompare(noadjust)
treat all factor variables as observed
perform comparisons within equation eqspec
perform comparisons within each equation

| Advanced |  |
| :---: | :---: |
| emptycells (empspec) | treatment of empty cells for balanced factors |
| noestimcheck | suppress estimability checks |
| Reporting |  |
| level (\#) | confidence level; default is level (95) |
| cieffects | show effects table with confidence intervals; the default |
| pveffects | show effects table with $p$-values |
| effects | show effects table with confidence intervals and $p$-values |
| cimargins | show table of margins and confidence intervals |
| groups | show table of margins and group codes |
| sort | sort the margins or contrasts within each term |
| post | post margins and their VCEs as estimation results |
| display_options | control column formats, row spacing, line width, and factor-variable labeling |
| eform_option | report exponentiated contrasts |
| df (\#) | use $t$ distribution with \# degrees of freedom for computing $p$-values and confidence intervals |

$\mathrm{df}(\#)$ does not appear in the dialog box.

## method

## Description

noadjust
bonferroni [adjustall] Bonferroni's method; adjust across all terms
sidak [adjustall] Šidák's method; adjust across all terms
scheffe
*tukey
*snk

* duncan
* dunnett
do not adjust for multiple comparisons; the default

Scheffé's method
Tukey's method
Student-Newman-Keuls's method
Duncan's method
Dunnett's method
*tukey, snk, duncan, and dunnett are only allowed with results from anova, manova, regress, and mvreg. tukey, snk, duncan, and dunnett are not allowed with results from svy.

Time-series operators are allowed if they were used in the estimation.
collect is allowed; see [U] 11.1.10 Prefix commands.

## Options

$\qquad$
mcompare (method) specifies the method for computing $p$-values and confidence intervals that account for multiple comparisons within a factor-variable term.
Most methods adjust the comparisonwise error rate, $\alpha_{c}$, to achieve a prespecified experimentwise error rate, $\alpha_{e}$.
mcompare(noadjust) is the default; it specifies no adjustment.

$$
\alpha_{c}=\alpha_{e}
$$

mcompare(bonferroni) adjusts the comparisonwise error rate based on the upper limit of the Bonferroni inequality:

$$
\alpha_{e} \leq m \alpha_{c}
$$

where $m$ is the number of comparisons within the term.
The adjusted comparisonwise error rate is

$$
\alpha_{c}=\alpha_{e} / m
$$

mcompare (sidak) adjusts the comparisonwise error rate based on the upper limit of the probability inequality

$$
\alpha_{e} \leq 1-\left(1-\alpha_{c}\right)^{m}
$$

where $m$ is the number of comparisons within the term.
The adjusted comparisonwise error rate is

$$
\alpha_{c}=1-\left(1-\alpha_{e}\right)^{1 / m}
$$

This adjustment is exact when the $m$ comparisons are independent.
mcompare (scheffe) controls the experimentwise error rate using the $F$ (or $\chi^{2}$ ) distribution with degrees of freedom equal to the rank of the term.

For results from anova, regress, manova, and mvreg (see [R] anova, [R] regress, [MV] manova, and [MV] mvreg), pwcompare allows the following additional methods. These methods are not allowed with results that used vce(robust) or vce(cluster clustvar).
mcompare (tukey) uses what is commonly referred to as Tukey's honestly significant difference. This method uses the Studentized range distribution instead of the $t$ distribution.
mcompare (snk) is a variation on mcompare (tukey) that counts only the number of margins in the range for a given comparison instead of the full number of margins.
mcompare (duncan) is a variation on mcompare (snk) with additional adjustment to the significance probabilities.
mcompare (dunnett) uses Dunnett's method for making comparisons with a reference category. mcompare(method adjustall) specifies that the multiple-comparison adjustments count all comparisons across all terms rather than performing multiple comparisons term by term. This leads to more conservative adjustments when multiple variables or terms are specified in marginlist. This option is compatible only with the bonferroni and sidak methods.
asobserved specifies that factor covariates be evaluated using the cell frequencies observed when the model was fit. The default is to treat all factor covariates as though there were an equal number of observations at each level.

Equations
equation (eqspec) specifies the equation from which margins are to be computed. The default is to compute margins from the first equation.
atequations specifies that the margins be computed within each equation.

Advanced
emptycells (empspec) specifies how empty cells are handled in interactions involving factor variables that are being treated as balanced.
emptycells(strict) is the default; it specifies that margins involving empty cells be treated as not estimable.
emptycells (reweight) specifies that the effects of the observed cells be increased to accommodate any missing cells. This makes the margins estimable but changes their interpretation.
noestimcheck specifies that pwcompare not check for estimability. By default, the requested margins are checked and those found not estimable are reported as such. Nonestimability is usually caused by empty cells. If noestimcheck is specified, estimates are computed in the usual way and reported even though the resulting estimates are manipulable, which is to say they can differ across equivalent models having different parameterizations.

## Reporting

level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals. The significance level used by the groups option is $100-\#$, expressed as a percentage.
cieffects specifies that a table of the pairwise comparisons with their standard errors and confidence intervals be reported. This is the default.
pveffects specifies that a table of the pairwise comparisons with their standard errors, test statistics, and $p$-values be reported.
effects specifies that a table of the pairwise comparisons with their standard errors, test statistics, $p$-values, and confidence intervals be reported.
cimargins specifies that a table of the margins with their standard errors and confidence intervals be reported.
groups specifies that a table of the margins with their standard errors and group codes be reported. Margins with the same letter in the group code are not significantly different at the specified significance level.
sort specifies that the reported tables be sorted on the margins or differences in each term.
post causes pwcompare to behave like a Stata estimation (e-class) command. pwcompare posts the vector of estimated margins along with the estimated variance-covariance matrix to e(), so you can treat the estimated margins just as you would results from any other estimation command. For example, you could use test to perform simultaneous tests of hypotheses on the margins, or you could use lincom to create linear combinations.
display_options: vsquish, nofvlabel, fvwrap(\#), fvwrapon(style), cformat(\%fmt), pformat (\%fmt), sformat (\%fmt), and nolstretch.
vsquish specifies that the blank space separating factor-variable terms or time-series-operated variables from other variables in the model be suppressed.
nofvlabel displays factor-variable level values rather than attached value labels. This option overrides the fvlabel setting; see [R] set showbaselevels.
fvwrap(\#) specifies how many lines to allow when long value labels must be wrapped. Labels requiring more than \# lines are truncated. This option overrides the fvwrap setting; see [R] set showbaselevels.
fvwrapon(style) specifies whether value labels that wrap will break at word boundaries or break based on available space.
fvwrapon(word), the default, specifies that value labels break at word boundaries.
fvwrapon(width) specifies that value labels break based on available space.
This option overrides the fvwrapon setting; see [R] set showbaselevels.
cformat (\% fint) specifies how to format contrasts or margins, standard errors, and confidence limits in the table of pairwise comparisons.
pformat (\%fmt) specifies how to format $p$-values in the table of pairwise comparisons.
sformat (\%fmt) specifies how to format test statistics in the table of pairwise comparisons.
nolstretch specifies that the width of the table of pairwise comparisons not be automatically widened to accommodate longer variable names. The default, lstretch, is to automatically widen the table of pairwise comparisons up to the width of the Results window. Specifying lstretch or nolstretch overrides the setting given by set lstretch. If set lstretch has not been set, the default is lstretch. nolstretch is not shown in the dialog box.
eform_option specifies that the contrasts table be displayed in exponentiated form. $e^{\text {contrast }}$ is displayed rather than contrast. Standard errors and confidence intervals are also transformed. See [R] eform_option for the list of available options.

The following option is available with pwcompare but is not shown in the dialog box:
df (\#) specifies that the $t$ distribution with \# degrees of freedom be used for computing $p$-values and confidence intervals. The default is to use e(df_r) degrees of freedom or the standard normal distribution if $e\left(d f \_r\right)$ is missing.

## Remarks and examples

pwcompare performs pairwise comparisons of margins across the levels of factor variables from the most recently fit model. The margins can be estimated cell means, marginal means, intercepts,
marginal intercepts, slopes, or marginal slopes. With the exception of slopes, we can also consider these margins to be marginal linear predictions.

The margins are calculated as linear combinations of the coefficients. Let $k$ be the number of levels for a factor term in our model; then there are $k$ margins for that term, and

$$
m=\binom{k}{2}=\frac{k(k-1)}{2}
$$

unique pairwise comparisons of those margins.
The confidence intervals and $p$-values for these pairwise comparisons can be adjusted to account for multiple comparisons. Bonferroni's, Šidák's, and Scheffé's adjustments can be made for multiple comparisons after fitting any type of model. In addition, Tukey's, Student-Newman-Keuls's, Duncan's, and Dunnett's adjustments are available when fitting ANOVA, linear regression, MANOVA, or multivariate regression models.

Remarks are presented under the following headings:

```
Pairwise comparisons of means
    Marginal means
    All pairwise comparisons
Overview of multiple-comparison methods
    Fisher's protected least-significant difference (LSD)
    Bonferroni's adjustment
    Sidák's adjustment
    Scheffé's adjustment
    Tukey's HSD adjustment
    Student-Newman-Keuls's adjustment
    Duncan's adjustment
    Dunnett's adjustment
Example adjustments using one-way models
    Fisher's protected LSD
    Tukey's HSD
    Dunnett's method for comparisons to a control
Two-way models
Pairwise comparisons of slopes
Nonlinear models
Multiple-equation models
Unbalanced data
Empty cells
```


## Pairwise comparisons of means

Suppose we are interested in the effects of five different fertilizers on wheat yield. We could estimate the following linear regression model to determine the effect of each type of fertilizer on the yield.

```
. use https://www.stata-press.com/data/r18/yield
(Artificial wheat yield dataset)
. regress yield i.fertilizer
```

| Source | SS | df | MS |
| ---: | :---: | ---: | ---: |
| Model | 1078.84207 | 4 | 269.710517 |
| Residual | 9859.55334 | 195 | 50.561812 |
| Total | 10938.3954 | 199 | 54.9668111 |


| Number of obs | $=$ | 200 |
| :--- | :--- | ---: |
| $\mathrm{~F}(4,195)$ | $=$ | 5.33 |
| Prob > F | $=$ | 0.0004 |
| R-squared | $=$ | 0.0986 |
| Adj R-squared | $=$ | 0.0801 |
| Root MSE | $=$ | 7.1107 |


| yield | Coefficient | Std. err. | t | $\mathrm{P}>\mid \mathrm{tI}$ | [95\% conf. interval] |  |
| :---: | ---: | :--- | ---: | ---: | ---: | ---: |
| fertilizer |  |  |  |  |  |  |
| $10-08-22$ | 3.62272 | 1.589997 | 2.28 | 0.024 | .4869212 | 6.758518 |
| $16-04-08$ | .4906299 | 1.589997 | 0.31 | 0.758 | -2.645169 | 3.626428 |
| $18-24-06$ | 4.922803 | 1.589997 | 3.10 | 0.002 | 1.787005 | 8.058602 |
| $29-03-04$ | -1.238328 | 1.589997 | -0.78 | 0.437 | -4.374127 | 1.89747 |
| _cons | 41.36243 | 1.124298 | 36.79 | 0.000 | 39.14509 | 43.57977 |

In this simple case, the coefficients for fertilizers 10-08-22, 16-04-08, 18-24-06, and 29-03-04 indicate the difference in the mean yield for that fertilizer versus the mean yield for fertilizer 10-10-10. That the standard errors of all four coefficients are identical results from having perfectly balanced data.

## Marginal means

We can use pwcompare with the cimargins option to compute the mean yield for each of the fertilizers.

```
. pwcompare fertilizer, cimargins
Pairwise comparisons of marginal linear predictions
Margins: asbalanced
```

|  |  |  | Unadjusted <br> [95\% conf. interval] |  |
| :---: | ---: | :---: | ---: | ---: |
|  | Margin | Std. err. |  |  |
| fertilizer |  |  | 39.14509 | 43.57977 |
| $10-10-10$ | 41.36243 | 1.124298 | 42.7678 | 47.20249 |
| $10-08-22$ | 44.98515 | 1.124298 | 39.63571 | 44.0704 |
| $16-04-08$ | 41.85306 | 1.124298 | 44.06789 | 48.50258 |
| $18-24-06$ | 46.28523 | 1.124298 | 37.90676 | 42.34145 |
| $29-03-04$ | 40.1241 | 1.124298 |  |  |

Looking at the confidence intervals for fertilizers 10-10-10 and 10-08-22 in the table above, we might be tempted to conclude that these means are not significantly different because the intervals overlap. However, as discussed in Interaction plots of [R] marginsplot, we cannot draw conclusions about the differences in means by looking at confidence intervals for the means themselves. Instead, we would need to look at confidence intervals for the difference in means.

## All pairwise comparisons

By default, pwcompare calculates all pairwise differences of the margins, in this case pairwise differences of the mean yields.

```
. pwcompare fertilizer
Pairwise comparisons of marginal linear predictions
Margins: asbalanced
```

|  | Contrast | Std. err. | $\begin{array}{cc}  & \text { Una } \\ {[95 \%} & \text { con } \end{array}$ | $\begin{aligned} & \text { isted } \\ & \text { interval] } \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: |
| fertilizer |  |  |  |  |
| $10-08-22$ vs $10-10-10$ | 3.62272 | 1.589997 | . 4869212 | 6.758518 |
| $16-04-08$ vs $10-10-10$ | . 4906299 | 1.589997 | -2.645169 | 3.626428 |
| $18-24-06$ vs 10-10-10 | 4.922803 | 1.589997 | 1.787005 | 8.058602 |
| $29-03-04$ vs 10-10-10 | -1.238328 | 1.589997 | -4.374127 | 1.89747 |
| $16-04-08$ vs 10-08-22 | -3.13209 | 1.589997 | -6.267889 | . 0037086 |
| $18-24-06$ vs 10-08-22 | 1.300083 | 1.589997 | -1.835715 | 4.435882 |
| $29-03-04$ vs 10-08-22 | -4.861048 | 1.589997 | -7.996847 | -1.725249 |
| $18-24-06$ vs 16-04-08 | 4.432173 | 1.589997 | 1.296375 | 7.567972 |
| 29-03-04 vs 16-04-08 | -1.728958 | 1.589997 | -4.864757 | 1.406841 |
| 29-03-04 vs 18-24-06 | -6.161132 | 1.589997 | -9.29693 | -3.025333 |

If a confidence interval does not include zero, the means for the compared fertilizers are significantly different. Therefore, at the $5 \%$ significance level, we would reject the hypothesis that the means for fertilizers $10-10-10$ and 10-08-22 are equivalent-as we would do for 18-24-06 vs 10-10-10, $29-03-04$ vs $10-08-22,18-24-06$ vs $16-04-08$, and $29-03-04$ vs $18-24-06$.

We may prefer to see the $p$-values instead of looking at confidence intervals to determine whether the pairwise differences are significantly different from zero. We could use the pveffects option to see the differences with standard errors and $p$-values, or we could use the effects option to see both $p$-values and confidence intervals in the same table. Here we specify effects as well as the sort option so that the differences are sorted from smallest to largest.

```
. pwcompare fertilizer, effects sort
Pairwise comparisons of marginal linear predictions
Margins: asbalanced
```

|  | Contrast | Std. err. | Unadjusted |  | Unadjusted |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | $p>\|t\|$ | [95\% conf | interval] |
| $\begin{gathered} \text { fertilizer } \\ 29-03-04 \end{gathered}$ |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 18-24-06 | -6.161132 | 1.589997 | -3.87 | 0.000 | -9.29693 | -3.025333 |
| 29-03-04 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-08-22 | -4.861048 | 1.589997 | -3.06 | 0.003 | -7.996847 | -1.725249 |
| 16-04-08 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-08-22 | -3.13209 | 1.589997 | -1.97 | 0.050 | -6.267889 | . 0037086 |
| 29-03-04 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 16-04-08 | -1.728958 | 1.589997 | -1.09 | 0.278 | $-4.864757$ | 1.406841 |
| 29-03-04 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | -1.238328 | 1.589997 | -0.78 | 0.437 | $-4.374127$ | 1.89747 |
| 16-04-08 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | . 4906299 | 1.589997 | 0.31 | 0.758 | -2.645169 | 3.626428 |
| 18-24-06 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-08-22 | 1.300083 | 1.589997 | 0.82 | 0.415 | $-1.835715$ | 4.435882 |
| 10-08-22 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | 3.62272 | 1.589997 | 2.28 | 0.024 | . 4869212 | 6.758518 |
| 18-24-06 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 16-04-08 | 4.432173 | 1.589997 | 2.79 | 0.006 | 1.296375 | 7.567972 |
| 18-24-06 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | 4.922803 | 1.589997 | 3.10 | 0.002 | 1.787005 | 8.058602 |

We find that 5 of the 10 pairs of means are significantly different at the $5 \%$ significance level.
We can use the groups option to obtain a table that identifies groups whose means are not significantly different by assigning them the same letter.

```
. pwcompare fertilizer, groups sort
Pairwise comparisons of marginal linear predictions
Margins: asbalanced
```

|  | Margin | Std. err. | Unadjusted <br> groups |
| :---: | ---: | :---: | ---: |
| fertilizer |  |  |  |
| $29-03-04$ | 40.1241 | 1.124298 | A |
| $10-10-10$ | 41.36243 | 1.124298 | A |
| $16-04-08$ | 41.85306 | 1.124298 | AB |
| $10-08-22$ | 44.98515 | 1.124298 | BC |
| $18-24-06$ | 46.28523 | 1.124298 | C |

Note: Margins sharing a letter in the group label are not significantly different at the 5\% level.

The letter A that is assigned to fertilizers 29-03-04, 10-10-10, and 16-04-08 designates that the mean yields for these fertilizers are not different at the $5 \%$ level.

## Overview of multiple-comparison methods

For a single test, if we choose a 5\% significance level, we would have a 5\% chance of concluding that two margins are different when the population values are actually equal. This is known as making a type I error. When we perform $m=k(k-1) / 2$ pairwise comparisons of the $k$ margins, we have $m$ opportunities to make a type I error.
pwcompare with the mcompare () option allows us to adjust the confidence intervals and $p$-values for each comparison to account for the increased probability of making a type I error when making multiple comparisons. Bonferroni's adjustment, Šidák's adjustment, and Scheffé's adjustment can be used when making pairwise comparisons of the margins after any estimation command. Tukey's honestly significant difference, Student-Newman-Keuls's method, Duncan's method, and Dunnett's method are only available when fitting linear models after anova, manova, regress, or mvreg.

## Fisher's protected least-significant difference (LSD)

pwcompare does not offer an mcompare() option specifically for Fisher's protected least-significant difference (LSD). In this methodology, no adjustment is made to the confidence intervals or $p$-values. However, it is protected in the sense that no pairwise comparisons are tested unless the joint test for the corresponding term in the model is significant. Therefore, the default mcompare (noadjust) corresponds to Fisher's protected LSD assuming that the corresponding joint test was performed before using pwcompare.

Milliken and Johnson (2009) recommend using this methodology for planned comparisons, assuming the corresponding joint test is significant.

## Bonferroni's adjustment

mcompare(bonferroni) adjusts significance levels based on the Bonferroni inequality, which, in the case of multiple testing, tells us that the maximum error rate for all comparisons is the sum of the error rates for the individual comparisons. Assuming that we are using the same significance level for all tests, the experimentwise error rate is the error rate for a single test multiplied by the
number of comparisons. Therefore, a $p$-value for each comparison can be computed by multiplying the unadjusted $p$-value by the total number of comparisons. If the adjusted $p$-value is greater than 1 , then pwcompare will report a $p$-value of 1 .

Bonferroni's adjustment is popular because it is easy to compute manually and because it can be applied to any set of tests, not only the pairwise comparisons available in pwcompare. In addition, this method does not require equal sample sizes.

Because Bonferroni's adjustment is so general, it is more conservative than many of the other adjustments. It is especially conservative when a large number of tests is being performed.

## Šidák's adjustment

mcompare (sidak) performs an adjustment using Šidák's method. This adjustment, like Bonferroni's adjustment, is derived from an inequality. However, in this case, the inequality is based on the probability of not making a type I error. For a single test, the probability that we do not make a type I error is $1-\alpha$. For two independent tests, both using $\alpha$ as a significance level, the probability is $(1-\alpha)(1-\alpha)$. Likewise, for $m$ independent tests, the probability of not making a type I error is $(1-\alpha)^{m}$. Therefore, the probability of making one or more type I errors is $1-(1-\alpha)^{m}$. When tests are not independent, the probability of making at least one error is less than $1-(1-\alpha)^{m}$. Therefore, we can compute an adjusted $p$-value as $1-\left(1-{ }_{u} p\right)^{m}$, where ${ }_{u} p$ is the unadjusted $p$-value for a single comparison.

Šidák's method is also conservative although slightly less so than Bonferroni's method. Like Bonferroni's method, this method does not require equal sample sizes.

## Scheffé's adjustment

Scheffe's adjustment is used when mcompare (scheffe) is specified. This adjustment is derived from the joint $F$ test and its correspondence to the maximum normalized comparison. To adjust for multiple comparisons, the absolute value of the $t$ statistic for a particular comparison can be compared with a critical value of $\sqrt{(k-1) F_{k-1, \nu}}$, where $\nu$ is the residual degrees of freedom. $F_{k-1, \nu}$ is the distribution of the joint $F$ test for the corresponding term in a one-way ANOVA model. Winer, Brown, and Michels (1991, 191-195) discuss this in detail. For estimation commands that report $z$ statistics instead of $t$ statistics for the tests on coefficients, a $\chi^{2}$ distribution is used instead of an $F$ distribution.

Scheffés method allows for making all possible comparisons of the $k$ margins, not just the pairwise comparisons. Unlike the methods described above, it does not take into account the number of comparisons that are currently being made. Therefore, this method is even more conservative than the others. Because this method adjusts for all possible comparisons of the levels of the term, Milliken and Johnson (2009) recommend using this procedure when making unplanned contrasts that are suggested by the data. As Winer, Brown, and Michels $(1991,191)$ put it, this method is often used to adjust for "unfettered data snooping". When using this adjustment, a contrast will never be significant if the joint $F$ or $\chi^{2}$ test for the term is not also significant.

This is another method that does not require equal sample sizes.

## Tukey's HSD adjustment

Tukey's adjustment is also referred to as Tukey's honestly significant difference (HSD) and is used when mcompare (tukey) is specified. It is often applied to all pairwise comparisons of means. Tukey's HSD is commonly used as a post hoc test although this is not a requirement.

To adjust for multiple comparisons, Tukey's method compares the absolute value of the $t$ statistic from the individual comparison with a critical value based on a Studentized range distribution with parameter equal to the number of levels in the term. When applied to pairwise comparisons of means,

$$
q=\frac{\text { mean }_{\max }-\text { mean }_{\min }}{s}
$$

follows a Studentized range distribution with parameter $k$ and $\nu$ degrees of freedom. Here mean $\max$ and mean $\min$ are the largest and smallest marginal means, and $s$ is an estimate of the standard error of the means.

Now for the comparison of the smallest and largest means, we can say that the probability of not making a type I error is

$$
\operatorname{Pr}\left(\frac{\operatorname{mean}_{\max }-\text { mean }_{\min }}{s} \leq q_{k, \nu}\right)=1-\alpha
$$

Then, the following inequality holds for all pairs of means simultaneously:

$$
\operatorname{Pr}\left(\frac{\left|\operatorname{mean}_{i}-\operatorname{mean}_{j}\right|}{s} \leq q_{k, \nu}\right) \geq 1-\alpha
$$

Based on this procedure, Tukey's HSD computes the $p$-value for each of the individual comparisons using the Studentized range distribution. However, because the equality holds only for the difference in the largest and smallest means, this procedure produces conservative tests for the remaining comparisons. Winer, Brown, and Michels (1991, 172-182) discuss this in further detail.

With unequal sample sizes, mcompare(tukey) produces the Tukey-Kramer adjustment (Tukey 1953; Kramer 1956).

## Student-Newman-Keuls's adjustment

The Student-Newman-Keuls (SNK) method is used when mcompare (snk) is specified. It is a modification to Tukey's method and is less conservative. In this procedure, we first order the means. We then test the difference in the smallest and largest means using a critical value from the Studentized range distribution with parameter $k$, where $k$ is the number of levels in the term. This step uses the same methodology as in Tukey's procedure. However, in the next step, we will then test for differences in the two sets of means that are the endpoints of the two ranges including $k-1$ means. Specifically, we test the difference in the smallest mean and the second-largest mean using a critical value from the Studentized range distribution with parameter $k-1$. We would also test the difference in the second-smallest mean and the largest mean using this critical value. Likewise, the means that are the endpoints of ranges including $k-2$ means when ordered are tested using the Studentized range distribution with parameter $k-2$, and so on.

Equal sample sizes are required for this method.

## Duncan's adjustment

When mcompare (duncan) is specified, tests are adjusted for multiple comparisons using Duncan's method, which is sometimes referred to as Duncan's new multiple range method. This adjustment produces tests that are less conservative than both Tukey's HSD and SNK. This procedure is performed in the same manner as SNK except that the $p$-values for the individual comparisons are adjusted as $1-\left(1-{ }_{\mathrm{snk}} p_{i}\right)^{1 /(r+1)}$, where $\mathrm{snk} p$ is the $p$-value computed using the SNK method and $r$ represents the number of means that, when ordered, fall between the two that are being compared.

Again, equal sample sizes are required for this adjustment.

## Dunnett's adjustment

Dunnett's adjustment is obtained by specifying mcompare (dunnett). It is used when one of the levels of a factor can be considered a control or reference level with which each of the other levels is being compared. When Dunnett's adjustment is requested, $k-1$ instead of $k(k-1) / 2$ pairwise comparisons are made. Dunnett $(1955,1964)$ developed tables of critical values for what Miller (1981, 76) refers to as the "many-one $t$ statistic". The $t$ statistics for individual comparisons are compared with these critical values when making many comparisons to a single reference level.

This method also requires equal sample sizes.

## Example adjustments using one-way models

## Fisher's protected LSD

Fisher's protected LSD requires that we first verify that the joint test for a term in our model is significant before proceeding with pairwise comparisons. Using our previous example, we could have first used the contrast command to obtain a joint test for the effects of fertilizer.

| . contrast fertilizer |
| :--- |
| Contrasts of marginal linear predictions |
| Margins: asbalanced |

This test for the effects of fertilizer is highly significant. Now we can say we are using Fisher's protected LSD when looking at the unadjusted $p$-values that were obtained from our previous command,

## Tukey's HSD

Because we fit a linear regression model and are interested in all pairwise comparisons of the marginal means, we may instead choose to use Tukey's HSD.

```
. pwcompare fertilizer, effects sort mcompare(tukey)
Pairwise comparisons of marginal linear predictions
Margins: asbalanced
```

|  | Number of <br> comparisons |
| ---: | ---: |
| fertilizer | 10 |


|  | Contrast | Std. err. | Tukey |  | Tukey |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  | t | $P>\|t\|$ | [95\% con | interval] |
| $\begin{gathered} \text { fertilizer } \\ 29-03-04 \end{gathered}$ |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 18-24-06 | -6.161132 | 1.589997 | -3.87 | 0.001 | -10.53914 | -1.78312 |
| 29-03-04 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-08-22 | -4.861048 | 1.589997 | -3.06 | 0.021 | -9.239059 | -. 4830368 |
| 16-04-08 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-08-22 | -3.13209 | 1.589997 | -1.97 | 0.285 | -7.510101 | 1.245921 |
| 29-03-04 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 16-04-08 | -1.728958 | 1.589997 | -1.09 | 0.813 | -6.106969 | 2.649053 |
| 29-03-04 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | -1.238328 | 1.589997 | -0.78 | 0.936 | -5.616339 | 3.139683 |
| 16-04-08 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | . 4906299 | 1.589997 | 0.31 | 0.998 | -3.887381 | 4.868641 |
| 18-24-06 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-08-22 | 1.300083 | 1.589997 | 0.82 | 0.925 | -3.077928 | 5.678095 |
| 10-08-22 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | 3.62272 | 1.589997 | 2.28 | 0.156 | -. 7552913 | 8.000731 |
| 18-24-06 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 16-04-08 | 4.432173 | 1.589997 | 2.79 | 0.046 | . 0541623 | 8.810185 |
| 18-24-06 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | 4.922803 | 1.589997 | 3.10 | 0.019 | . 5447922 | 9.300815 |

This time, our $p$-values have been modified, and we find that only four of the pairwise differences are considered significantly different from zero at the 5\% level.

If we are interested only in performing pairwise comparisons of a subset of our means, we can use factor-variable operators to select the levels of the factor that we want to compare. Here we exclude all comparisons involving fertilizer 10-10-10.

```
. pwcompare i(2/5).fertilizer, effects sort mcompare(tukey)
Pairwise comparisons of marginal linear predictions
Margins: asbalanced
```

|  | Number of <br> comparisons |
| :--- | ---: |
| fertilizer | 6 |


|  | Contrast | Std. err. | Tukey |  | Tukey |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  | t | $P>\|t\|$ | [95\% con | interval] |
| fertilizer |  |  |  |  |  |  |
| 29-03-04 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 18-24-06 | -6.161132 | 1.589997 | -3.87 | 0.001 | -10.28133 | -2.040937 |
| 29-03-04 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-08-22 | -4.861048 | 1.589997 | -3.06 | 0.013 | -8.981242 | -. 7408538 |
| 16-04-08 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-08-22 | -3.13209 | 1.589997 | -1.97 | 0.203 | -7.252284 | . 9881042 |
| 29-03-04 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 16-04-08 | -1.728958 | 1.589997 | -1.09 | 0.698 | -5.849152 | 2.391236 |
| 18-24-06 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-08-22 | 1.300083 | 1.589997 | 0.82 | 0.846 | -2.820111 | 5.420278 |
| 18-24-06 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 16-04-08 | 4.432173 | 1.589997 | 2.79 | 0.030 | . 3119792 | 8.552368 |

The adjusted $p$-values and confidence intervals differ from those in the previous output because Tukey's adjustment takes into account the total number of comparisons being made when determining the appropriate degrees of freedom to use for the Studentized range distribution.

## Dunnett's method for comparisons to a control

If one of our five fertilizer groups represents fields where no fertilizer was applied, we may want to use Dunnett's method to compare each of the four fertilizers with the control group. In this case, we make only $k-1$ comparisons for $k$ groups.

```
. pwcompare fertilizer, effects mcompare(dunnett)
Pairwise comparisons of marginal linear predictions
Margins: asbalanced
```

|  | Number of <br> comparisons |
| :--- | ---: |
| fertilizer | 4 |


|  | Contrast | Std. err. | Dunnett |  | Dunnett |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf | interval] |
| $\begin{gathered} \text { fertilizer } \\ 10-08-22 \end{gathered}$ |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | 3.62272 | 1.589997 | 2.28 | 0.079 | -. 2918331 | 7.537273 |
| 16-04-08 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | . 4906299 | 1.589997 | 0.31 | 0.994 | -3.423923 | 4.405183 |
| 18-24-06 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | 4.922803 | 1.589997 | 3.10 | 0.008 | 1.00825 | 8.837356 |
| 29-03-04 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | -1.238328 | 1.589997 | -0.78 | 0.852 | -5.152881 | 2.676225 |

In our previous regress command, fertilizer $10-10-10$ was treated as the base. Therefore, by default, it was treated as the control when using Dunnett's adjustment, and the pairwise comparisons are equivalent to the coefficients reported by regress. Based on our regress output, we would conclude that fertilizers 10-08-22 and 18-24-06 are different from fertilizer 10-10-10 at the 5\% level. However, using Dunnett's adjustment, we find only fertilizer 18-24-06 to be different from fertilizer 10-10-10 at this same significance level.

If the model is fit without a base level for a factor variable, then pwcompare will choose the first level as the reference level. If we want to make comparisons with a different level than the one mcompare (dunnett) chooses by default, we can use the b. operator to override the default. Here we use fertilizer 5 (29-03-04) as the reference level.
. pwcompare b5.fertilizer, effects sort mcompare(dunnett)
Pairwise comparisons of marginal linear predictions
Margins: asbalanced

|  | Number of <br> comparisons |
| :--- | ---: |
| fertilizer | 4 |


|  | Contrast | Std. err. | Dunnett <br> t |  | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |
| :--- | :--- | :--- | :--- | :--- | :--- | :--- |

## Two-way models

In the previous examples, we have performed pairwise comparisons after fitting a model with a single factor. Now, we include two factors and their interaction in our model.

| Source | SS | df | MS | Number of obs $F(9,190)$ <br> Prob > F <br> R-squared <br> Adj R-squared <br> Root MSE |  | 200 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Model | 6200.81605 | 9 | 688.979561 |  |  | 0.0000 |
| Residual | 4737.57936 | 190 | 24.9346282 |  |  | 0.5669 |
|  |  |  |  |  |  | 0.5464 |
| Total | 10938.3954 | 199 | 54.9668111 |  |  | 4.9935 |
| yield | Coefficient | Std. err. | t | $P>\|t\|$ | [95\% conf. interval] |  |
| fertilizer |  |  |  | 0.235 | -1.232505 | 4.997016 |
| 10-08-22 | 1.882256 | 1.57907 | 1.19 |  |  |  |
| 16-04-08 | -. 5687418 | 1.57907 | -0.36 | 0.719 | -3.683502 | 2.546019 |
| 18-24-06 | 4.904999 | 1.57907 | 3.11 | 0.002 | 1.790239 | 8.01976 |
| 29-03-04 | -1.217496 | 1.57907 | -0.77 | 0.442 | -4.332257 | 1.897264 |
| 1.irrigation | 8.899721 | 1.57907 | 5.64 | 0.000 | 5.784961 | 12.01448 |
| fertilizer\# irrigation |  |  |  |  |  |  |
| 10-08-22\#1 | 3.480928 | 2.233143 | 1.56 | 0.121 | -. 9240084 | 7.885865 |
| 16-04-08\#1 | 2.118743 | 2.233143 | 0.95 | 0.344 | -2.286193 | 6.52368 |
| 18-24-06\#1 | . 0356082 | 2.233143 | 0.02 | 0.987 | -4.369328 | 4.440545 |
| 29-03-04\#1 | -. 0416636 | 2.233143 | -0.02 | 0.985 | -4.4466 | 4.363273 |
| _cons | 36.91257 | 1.116571 | 33.06 | 0.000 | 34.7101 | 39.11504 |

We can perform pairwise comparisons of the cell means defined by the fertilizer and irrigation interaction.

```
. pwcompare fertilizer#irrigation, sort groups mcompare(tukey)
Pairwise comparisons of marginal linear predictions
Margins: asbalanced
```

|  | Number of <br> comparisons |
| :--- | ---: |
| fertilizer\#irrigation | 45 |


|  | Margin | Std. err. | Tukey <br> groups |
| ---: | ---: | ---: | :---: |
| fertilizer\#irrigation |  |  |  |
| $29-03-04 \# 0$ | 35.69507 | 1.116571 | A |
| $16-04-08 \# 0$ | 36.34383 | 1.116571 | A |
| $10-10-10 \# 0$ | 36.91257 | 1.116571 | AB |
| $10-08-22 \# 0$ | 38.79482 | 1.116571 | AB |
| $18-24-06 \# 0$ | 41.81757 | 1.116571 | BC |
| $29-03-04 \# 1$ | 44.55313 | 1.116571 | CD |
| $10-10-10 \# 1$ | 45.81229 | 1.116571 | CDE |
| $16-04-08 \# 1$ | 47.36229 | 1.116571 | DEF |
| $18-24-06 \# 1$ | 50.7529 | 1.116571 | EF |
| $10-08-22 \# 1$ | 51.17547 | 1.116571 | F |
|  |  |  |  |

```
Note: Margins sharing a letter in the group label are
    not significantly different at the 5% level.
```

Based on Tukey's HSD and a 5\% significance level, we would conclude that the mean yield for fertilizer 29-03-04 without irrigation is not significantly different from the mean yields for fertilizers $10-10-10,10-08-22$, and 16-04-08 when used without irrigation but is significantly different from the remaining means.

Up to this point, most of the pairwise comparisons that we have performed could have also been obtained with pwmean (see [R] pwmean) if we had not been interested in examining the results from the estimation command before making pairwise comparisons of the means. For instance, we could reproduce the results from the above pwcompare command by typing

```
. pwmean yield, over(fertilizer irrigation) sort group mcompare(tukey)
```

However, pwcompare extends the capabilities of pwmean in many ways. For instance, pwmean only allows for pairwise comparisons of the cell means determined by the highest-level interaction of the variables specified in the over () option. However, pwcompare allows us to fit a single model, such as the two-way model that we fit above,

```
. regress yield fertilizer##irrigation
```

and compute pairwise comparisons of the marginal means for only one of the variables in the model:


Here the standard errors for the differences in marginal means and the residual degrees of freedom are based on the full model. Therefore, the results will differ from those obtained from pwcompare after fitting the one-way model with only fertilizer (or equivalently using pwmean).

## Pairwise comparisons of slopes

If we fit a model with a factor variable that is interacted with a continuous variable, pwcompare will even allow us to make pairwise comparisons of the slopes of the continuous variable for the levels of the factor variable.

In this case, we have a continuous variable, NO3_N, indicating the amount of nitrate nitrogen already existing in the soil, based on a sample taken from each field.
. regress yield fertilizer\#\#c.NO3_N

| Source | SS | df | MS |
| ---: | :---: | ---: | :---: |
| Model | 7005.69932 | 9 | 778.411035 |
| Residual | 3932.69609 | 190 | 20.6984005 |
| Total | 10938.3954 | 199 | 54.9668111 |


| Number of obs | $=$ | 200 |
| :--- | :--- | ---: |
| $\mathrm{~F}(9,190)$ | $=$ | 37.61 |
| Prob > F | $=$ | 0.0000 |
| R-squared | $=$ | 0.6405 |
| Adj R-squared | $=$ | 0.6234 |
| Root MSE | $=$ | 4.5495 |


| yield | Coefficient | Std. err. | t | $P>\|t\|$ | [95\% con | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| fertilizer |  |  |  |  |  |  |
| 10-08-22 | 18.65019 | 8.452061 | 2.21 | 0.029 | 1.97826 | 35.32212 |
| 16-04-08 | -13.34076 | 10.07595 | -1.32 | 0.187 | -33.21585 | 6.534327 |
| 18-24-06 | 24.35061 | 9.911463 | 2.46 | 0.015 | 4.799973 | 43.90125 |
| 29-03-04 | 17.58529 | 8.446736 | 2.08 | 0.039 | . 9238646 | 34.24671 |
| N03_N | 4.915653 | . 7983509 | 6.16 | 0.000 | 3.340884 | 6.490423 |
| $\begin{gathered} \text { fertilizer\# } \\ \text { c.N03_N } \end{gathered}$ |  |  |  |  |  |  |
| 10-08-22 | -1.282039 | . 8953419 | -1.43 | 0.154 | -3.048126 | . 4840487 |
| 16-04-08 | -1.00571 | . 9025862 | -1.11 | 0.267 | -2.786087 | . 7746662 |
| 18-24-06 | -2.97627 | . 9136338 | -3.26 | 0.001 | -4.778438 | -1.174102 |
| 29-03-04 | -3.275947 | . 8247385 | -3.97 | 0.000 | -4.902767 | -1.649127 |
| _cons | -5.459168 | 7.638241 | -0.71 | 0.476 | -20.52581 | 9.607477 |

These are the pairwise differences of the slopes of NO3_N for each pair of fertilizers:
. pwcompare fertilizer\#c.N03_N, pveffects sort mcompare(scheffe)
Pairwise comparisons of marginal linear predictions
Margins: asbalanced

|  | Number of <br> comparisons |
| :--- | ---: |
| fertilizer\#C.NO3_N | 10 |


|  |  |  | Scheffe |  |
| ---: | ---: | ---: | ---: | ---: |
|  |  |  |  |  |
|  | Contrast | Std. err. | t | P>\|t| |
| fertilizer\#c.N03_N |  |  |  |  |
| $29-03-04$ vs 10-10-10 | -3.275947 | .8247385 | -3.97 | 0.004 |
| $18-24-06$ vs 10-10-10 | -2.97627 | .9136338 | -3.26 | 0.034 |
| $29-03-04$ vs 16-04-08 | -2.270237 | .4691771 | -4.84 | 0.000 |
| $29-03-04$ vs 10-08-22 | -1.993909 | .4550851 | -4.38 | 0.001 |
| $18-24-06$ vs 16-04-08 | -1.97056 | .612095 | -3.22 | 0.038 |
| $18-24-06$ vs 10-08-22 | -1.694232 | .6013615 | -2.82 | 0.099 |
| $10-08-22$ vs $10-10-10$ | -1.282039 | .8953419 | -1.43 | 0.727 |
| $16-04-08$ vs 10-10-10 | -1.00571 | .9025862 | -1.11 | 0.871 |
| $29-03-04$ vs $18-24-06$ | -.2996772 | .4900939 | -0.61 | 0.984 |
| $16-04-08$ vs $10-08-22$ | .276328 | .5844405 | 0.47 | 0.994 |

Using Scheffés adjustment, we find that five of the pairs have significantly different slopes at the 5\% level.

## Nonlinear models

pwcompare can also perform pairwise comparisons of the marginal linear predictions after fitting a nonlinear model. For instance, we can use the dataset from Beyond linear models in $[R]$ contrast and fit the following logistic regression model of patient satisfaction on hospital:

```
. use https://www.stata-press.com/data/r18/hospital
(Artificial hospital satisfaction data)
. logit satisfied i.hospital
Iteration 0: Log likelihood = -393.72216
Iteration 1: Log likelihood = -387.55736
Iteration 2: Log likelihood = -387.4768
Iteration 3: Log likelihood = -387.47679
Logistic regression Number of obs = 802
```

LR chi2(2) $=12.49$
Prob > chi2 $=0.0019$
Pseudo R2 $=0.0159$

| satisfied | Coefficient | Std. err. | $z$ | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | :---: | :---: | :---: | :---: | :---: | :---: |
| hospital |  |  |  |  |  |  |
| 2 | .5348129 | .2136021 | 2.50 | 0.012 | .1161604 | .9534654 |
| 3 | .7354519 | .2221929 | 3.31 | 0.001 | .2999618 | 1.170942 |
| _cons | 1.034708 | .1391469 | 7.44 | 0.000 | .7619855 | 1.307431 |

For this model, the marginal linear predictions are the predicted log odds for each hospital and can be obtained with the cimargins option:

| . pwcompare hospital, cimargins |
| :--- |
| Pairwise comparisons of marginal linear predictions |
| Margins: asbalanced |$\quad$|  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: |
|  | Margin | Std. err. | Unadjusted <br> [95\% conf. interval] |  |
| hospital |  |  |  |  |
| 1 | 1.034708 | .1391469 | .7619855 | 1.307431 |
| 2 | 1.569521 | .1620618 | 1.251886 | 1.887157 |
| 3 | 1.77016 | .1732277 | 1.43064 | 2.10968 |

The pairwise comparisons are, therefore, differences in the log odds. We can specify mcompare(bonferroni) and effects to request Bonferroni-adjusted $p$-values and confidence intervals.

| . pwcompare hospital, effect |
| :--- |
| Pairwise comparisons of marg |
| Margins: asbalanced |
| Number of <br> comparisons |
| satisfied <br> hospital |


|  |  |  | Bonferroni |  | Bonferroni <br> [95\% conf. interval] |  |
| :---: | ---: | :---: | :---: | :---: | :---: | :---: |
|  | Contrast | Std. err. | $\mathbf{z}$ | P>\|z| | [ |  |
| satisfied |  |  |  |  |  |  |
| hospital |  |  |  |  |  |  |
| 2 vs 1 | .5348129 | .2136021 | 2.50 | 0.037 | .0234537 | 1.046172 |
| 3 vs 1 | .7354519 | .2221929 | 3.31 | 0.003 | .2035265 | 1.267377 |
| 3 vs 2 | .200639 | .2372169 | 0.85 | 1.000 | -.3672535 | .7685314 |

For nonlinear models, only Bonferroni's adjustment, Šidák's adjustment, and Scheffé's adjustment are available.

If we want pairwise comparisons reported as odds ratios, we can specify the or option.

```
. pwcompare hospital, effects mcompare(bonferroni) or
Pairwise comparisons of marginal linear predictions
Margins: asbalanced
```

|  | Number of <br> comparisons |
| :---: | ---: |
| satisfied <br> hospital | 3 |


|  |  |  | Bonferroni |  | Bonferroni |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | Odds ratio | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% con | interval] |
| satisfied hospital |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
| 2 vs 1 | 1.707129 | . 3646464 | 2.50 | 0.037 | 1.023731 | 2.846733 |
| 3 vs 1 | 2.086425 | . 4635888 | 3.31 | 0.003 | 1.225718 | 3.551525 |
| 3 vs 2 | 1.222183 | . 2899226 | 0.85 | 1.000 | . 6926341 | 2.156597 |

Notice that these tests are still performed on the marginal linear predictions. The odds ratios reported here are the exponentiated versions of the pairwise differences of $\log$ odds in the previous output. For further discussion, see $[R]$ contrast.

## Multiple-equation models

pwcompare works with models containing multiple equations. Commands such as intreg and gnbreg allow their ancillary parameters to be modeled as a function of independent variables, and pwcompare can compare the margins within these equations. The equation() option can be used to specify the equation for which pairwise comparisons of the margins should be made. The atequations option specifies that pairwise comparisons be computed for each equation. In addition, pwcompare allows a special pseudofactor for equation-called _eqns-when working with results from manova, mvreg, mlogit, and mprobit.

Here we use the jaw fracture dataset described in example 4 of [MV] manova. We fit a multivariate regression model including one independent factor variable, fracture.
. use https://www.stata-press.com/data/r18/jaw
(Table 4.6. Two-way unbalanced data for fractures of the jaw, Rencher (1998)) . mvreg y1 y2 y3 = i.fracture

| Equation | Obs | Parms | RMSE | "R-sq" | F | $\mathrm{P}>\mathrm{F}$ |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| y1 | 27 | 3 | 10.42366 | 0.2966 | 5.060804 | 0.0147 |
| y2 | 27 | 3 | 6.325398 | 0.1341 | 1.858342 | 0.1777 |
| y3 | 27 | 3 | 5.976973 | 0.1024 | 1.368879 | 0.2735 |
|  | Coefficient | Std. err. | . $\quad \mathrm{t}$ | $P>\|t\|$ | [95\% conf. interval] |  |
| y1 |  |  |  |  |  |  |
| fracture <br> Two compo.. | -8.833333 | 4.957441 | -1.78 | 0.087 | -19.06499 | 1.398322 |
| One simpl.. | 6 | 5.394759 | 1.11 | 0.277 | -5.134235 | 17.13423 |
| _cons | 37 | 3.939775 | 9.39 | 0.000 | 28.8687 | 45.1313 |
| y2 |  |  |  |  |  |  |
| fracture <br> Two compo.. | -5.761905 | 3.008327 | -1.92 | 0.067 | -11.97079 | . 446977 |
| One simpl.. | -3.053571 | 3.273705 | -0.93 | 0.360 | -9.810166 | 3.703023 |
| _cons | 38.42857 | 2.390776 | 16.07 | 0.000 | 33.49425 | 43.36289 |
| y3 |  |  |  |  |  |  |
| fracture <br> Two compo.. | 4.261905 | 2.842618 | 1.50 | 0.147 | -1.60497 | 10.12878 |
| One simpl.. | . 9285714 | 3.093377 | 0.30 | 0.767 | -5.455846 | 7.312989 |
| _cons | 58.57143 | 2.259083 | 25.93 | 0.000 | 53.90891 | 63.23395 |

pwcompare performs pairwise comparisons of the margins using the coefficients from the first equation by default:


We can use the equation() option to get pwcompare to perform comparisons in the y2 equation:

```
. pwcompare fracture, equation(y2) mcompare(bonferroni)
Pairwise comparisons of marginal linear predictions
Margins: asbalanced
```

|  |  | Number of <br> comparisons |
| :--- | ---: | ---: |
| y2 | fracture | 3 |


|  | Contrast | Std. err. | $\begin{array}{ll}  & \text { Bonf } \\ {[95 \%} & \text { conf } \end{array}$ | $\begin{aligned} & \text { roni } \\ & \text { interval] } \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: |
| y2 fracture | -5.761905 | 3.008327 | -13.50426 | 1.980449 |
|  |  |  |  |  |
| Two compound fractures |  |  |  |  |
| vs |  |  |  |  |
| One compound fracture One simple fracture |  |  |  |  |
| vs | -3.053571 | 3.273705 | -11.47891 | 5.371769 |
| One compound fracture One simple fracture |  |  |  |  |
| vs |  |  |  |  |
| Two compound fractures | 2.708333 | 2.887136 | -4.722119 | 10.13879 |

Because we are working with mvreg results, we can use the _eqns pseudofactor to compare the margins between the three dependent variables. The levels of _eqns index the equations: 1 for the first equation, 2 for the second, and 3 for the third.

```
. pwcompare _eqns, mcompare(bonferroni)
Pairwise comparisons of marginal linear predictions
Margins: asbalanced
```

|  | Number of <br> comparisons |
| ---: | ---: |
| _eqns | 3 |


|  | Contrast | Std. err. | Bonferroni <br> [95\% conf. interval] |  |
| :--- | ---: | ---: | ---: | ---: |
| _ _eqns |  |  |  |  |
| 2 vs 1 | -.5654762 | 2.545923 | -7.117768 | 5.986815 |
| 3 vs 1 | 24.24603 | 2.320677 | 18.27344 | 30.21862 |
| 3 vs 2 | 24.81151 | 2.368188 | 18.71664 | 30.90637 |

For the previous command, the only methods available are mcompare(bonferroni), mcompare(sidak), or mcompare(scheffe). Methods that use the Studentized range are not appropriate for making comparisons across equations.

## Unbalanced data

pwcompare treats all factors as balanced when it computes the marginal means. By "balanced", we mean that the number of observations in each combination of factor levels (in each cell mean) is equal. We can alternatively specify the asobserved option when we have unbalanced data to obtain marginal means that are based on the observed cell frequencies from the model fit. For more details on the difference in these two types of marginal means and a discussion of when each may be appropriate, see $[R]$ margins and $[R]$ contrast.

In addition, when our data are not balanced, some of the multiple-comparison adjustments are no longer appropriate. Student-Newman-Keuls's method, Duncan's method, and Dunnett's method assume equal numbers of observations per group.

Here we use an unbalanced dataset and fit a two-way ANOVA model for cholesterol levels on race and age group. Then we perform pairwise comparisons of the mean cholesterol levels for each race, requesting Šidák's adjustment as well as marginal means that are computed using the observed cell frequencies.

```
. use https://www.stata-press.com/data/r18/cholesterol3
(Artificial cholesterol data, unbalanced)
```

. anova chol race\#\#agegrp

|  | $\begin{aligned} \text { Number of obs } & = \\ \text { Root MSE } & = \end{aligned}$ | $\begin{array}{r} 67 \\ 8.37496 \end{array}$ | R -squared $=$ <br> Adj R-squared $=$ |  | $\begin{aligned} & 0.8179 \\ & 0.7689 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | 16379.993 | 14 | 1169.9995 | 16.68 | 0.0000 |
| race | 230.7544 | 2 | 115.3772 | 1.64 | 0.2029 |
| agegrp | 13857.988 | 4 | 3464.4969 | 49.39 | 0.0000 |
| race\#agegrp | 857.81521 | 8 | 107.2269 | 1.53 | 0.1701 |
| Residual | 3647.2774 | 52 | 70.13995 |  |  |
| Total | 20027.27 | 66 | 303.44349 |  |  |

. pwcompare race, asobserved mcompare(sidak)
Pairwise comparisons of marginal linear predictions
Margins: asobserved

|  | Number of <br> comparisons |
| ---: | ---: |
| race | 3 |


|  |  |  | Sidak <br> [95\% conf. interval] |  |
| :--- | ---: | ---: | ---: | ---: |
| Contrast | Std. err. |  |  |  |
| White vs Black | -7.232433 | 2.686089 | -13.85924 | -.6056277 |
| Other vs Black | -5.231198 | 2.651203 | -11.77194 | 1.309541 |
| Other vs White | 2.001235 | 2.414964 | -3.956682 | 7.959152 |

## Empty cells

An empty cell is a combination of the levels of factor variables that is not observed in the estimation sample. When we have empty cells in our data, the marginal means involving those empty cells are not estimable as described in [R] margins. In addition, all pairwise comparisons involving a marginal mean that is not estimable are themselves not estimable. Here we use a dataset where we do not have any observations for white individuals in the $20-29$ age group. We can use the emptycells (reweight) option to reweight the nonempty cells so that we can estimate the marginal mean for whites and compute pairwise comparisons involving that marginal mean.
. use https://www.stata-press.com/data/r18/cholesterol2
(Artificial cholesterol data, empty cells)
. tabulate race agegrp

| Race | Age group |  |  |  |  |  |
| :---: | ---: | ---: | ---: | ---: | ---: | ---: |
| $30-39$ | $40-59$ | $60-79$ | Total |  |  |  |
| Black | 5 | $20-29$ | 5 | 5 | 5 | 5 |
| White | 5 | 0 | 5 | 5 | 5 | 25 |
| Other | 5 | 5 | 5 | 5 | 5 | 20 |
| Total | 15 | 10 | 15 | 15 | 15 | 70 |

. anova chol race\#\#agegrp

|  | $\begin{aligned} \text { Number of obs } & = \\ \text { Root MSE } & = \end{aligned}$ | $\begin{array}{r} 70 \\ 9.47055 \end{array}$ | R-squared $=$ <br> Adj R-squared $=$ |  | $\begin{aligned} & 0.7582 \\ & 0.7021 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: |
| Source | Partial SS | df | MS | F | Prob>F |
| Model | 15751.611 | 13 | 1211.6624 | 13.51 | 0.0000 |
| race | 305.49046 | 2 | 152.74523 | 1.70 | 0.1914 |
| agegrp | 14387.856 | 4 | 3596.964 | 40.10 | 0.0000 |
| race\#agegrp | 795.80757 | 7 | 113.6868 | 1.27 | 0.2831 |
| Residual | 5022.7156 | 56 | 89.69135 |  |  |
| Total | 20774.327 | 69 | 301.0772 |  |  |

. pwcompare race, emptycells(reweight)
Pairwise comparisons of marginal linear predictions
Margins: asbalanced
Empty cells: reweight

|  |  |  | Unadjusted <br> [95\% conf. interval] |  |
| :--- | ---: | :---: | :---: | :---: |
|  | Contrast | Std. err. |  |  |
| race |  |  |  |  |
| White vs Black | 2.922769 | 2.841166 | -2.768769 | 8.614308 |
| Other vs Black | -4.12621 | 2.678677 | -9.492244 | 1.239824 |
| Other vs White | -7.048979 | 2.841166 | -12.74052 | -1.35744 |

For further details on the emptycells (reweight) option, see $[R]$ margins and $[R]$ contrast.

## Stored results

pwcompare stores the following in $r()$ :
Scalars

```
    r(df_r)
    r(k_terms)
    r(level)
    r(balanced)
```

Macros

```
    r(cmd)
    r(cmdline)
    r(est_cmd)
    r(est_cmdline)
    r(title)
    r(emptycells)
    r(groups#)
    r(mcmethod_vs)
    r(mctitle_vs)
    r(mcadjustall_vs)
    r(margin_method)
    r(vce)
```

Matrices
r(b)
r(V)
$r$ (error)
$r$ (table)
r(M)
r(b_vs)
r(V_vs)
r(error_vs)
r(table_vs)
$r(\mathrm{~L})$
r(k_groups)
variance degrees of freedom
number of terms in marginlist
confidence level of confidence intervals
1 if fully balanced data, 0 otherwise
pwcompare
command as typed
$e(\mathrm{cmd})$ from original estimation results
$e$ (cmdline) from original estimation results
title in output
empspec from emptycells()
group codes for the \#th margin in $r(b)$
method from mcompare()
title for method from mcompare()
adjustall or empty
asbalanced or asobserved
vcetype specified in vce() in original estimation command
margin estimates
variance-covariance matrix of the margin estimates
margin estimability codes;
0 means estimable,
8 means not estimable
matrix containing the margins with their standard errors, test statistics, $p$-values, and confidence intervals
matrix that produces the margins from the model coefficients
margin difference estimates
variance-covariance matrix of the margin difference estimates
margin difference estimability codes;
0 means estimable,
8 means not estimable
matrix containing the margin differences with their standard errors, test statistics, $p$-values, and confidence intervals
matrix that produces the margin differences from the model coefficients
number of significance groups for each term
pwcompare with the post option also stores the following in e():
Scalars
$e\left(d f \_r\right) \quad$ variance degrees of freedom
e(k_terms)
e(balanced)
Macros

```
        e(cmd)
    e(cmdline)
    e(properties)
    e(est_cmd)
    e(est_cmdline)
    e(title)
    e(emptycells)
    e(margin_method)
    e(asbalanced)
    e(asobserved)
    e(vce)
```

Matrices
e(b)
e(V)
e(error)
e(M)
e(b_vs)
e(V_vs)
e(error_vs)
e(L)

## Methods and formulas

Methods and formulas are presented under the following headings:

Notation<br>Unadjusted comparisons<br>Bonferroni's method<br>Šidák's method<br>Scheffé's method<br>Tukey's method<br>Student-Newman-Keuls's method<br>Duncan's method<br>Dunnett's method

## Notation

pwcompare performs comparisons of margins; see Methods and formulas in [R] contrast.
If there are $k$ margins for a given factor term, then there are

$$
m=\binom{k}{2}=\frac{k(k-1)}{2}
$$

unique pairwise comparisons. Let the $i$ th pairwise comparison be denoted by

$$
\widehat{\delta}_{i}=l_{i}^{\prime} \mathbf{b}
$$

where $\mathbf{b}$ is a column vector of coefficients from the fitted model and $l_{i}$ is a column vector that forms the corresponding linear combination. If $\widehat{\mathbf{V}}$ denotes the estimated variance matrix for $\mathbf{b}$, then the standard error for $\widehat{\delta}_{i}$ is given by

$$
\widehat{\operatorname{se}}\left(\widehat{\delta}_{i}\right)=\sqrt{l_{i}^{\prime} \widehat{\mathbf{V}} l_{i}}
$$

The corresponding test statistic is then

$$
t_{i}=\frac{\widehat{\delta}_{i}}{\widehat{\operatorname{se}\left(\widehat{\delta}_{i}\right)}}
$$

and the limits for a $100(1-\alpha) \%$ confidence interval for the expected value of $\widehat{\delta}_{i}$ are

$$
\widehat{\delta}_{i} \pm c_{i}(\alpha) \widehat{\operatorname{se}}\left(\widehat{\delta}_{i}\right)
$$

where $c_{i}(\alpha)$ is the critical value corresponding to the chosen multiple-comparison method.

## Unadjusted comparisons

pwcompare computes unadjusted $p$-values and confidence intervals by default. pwcompare uses the $t$ distribution with $\nu=\mathrm{e}\left(\mathrm{df} \_\mathrm{r}\right)$ degrees of freedom when $\mathrm{e}\left(\mathrm{df} \_\mathrm{r}\right)$ is posted by the estimation command. The unadjusted two-sided $p$-value is

$$
{ }_{u} p_{i}=2 \operatorname{Pr}\left(t_{\nu}>\left|t_{i}\right|\right)
$$

and the unadjusted critical value ${ }_{u} c_{i}(\alpha)$ satisfies the following probability statement:

$$
\alpha=2 \operatorname{Pr}\left\{t_{\nu}>{ }_{u} c_{i}(\alpha)\right\}
$$

pwcompare uses the standard normal distribution when $e\left(d f \_r\right)$ is not posted.

## Bonferroni's method

For mcompare(bonferroni), the adjusted $p$-value is

$$
{ }_{b} p_{i}=\min \left(1, m_{u} p_{i}\right)
$$

and the adjusted critical value is

$$
{ }_{b} c_{i}(\alpha)={ }_{u} c_{i}(\alpha / m)
$$

## Šidák's method

For mcompare(sidak), the adjusted $p$-value is

$$
{ }_{\mathrm{si}} p_{i}=1-\left(1-{ }_{u} p_{i}\right)^{m}
$$

and the adjusted critical value is

$$
{ }_{\text {si }} c_{i}(\alpha)={ }_{u} c_{i}\left\{1-(1-\alpha)^{1 / m}\right\}
$$

## Scheffés method

For mcompare(scheffe), the adjusted $p$-value is

$$
{ }_{\mathrm{sc}} p_{i}=\operatorname{Pr}\left(F_{d, \nu}>t_{i}^{2} / d\right)
$$

where $F_{d, \nu}$ is distributed as an $F$ with $d$ numerator and $\nu$ denominator degrees of freedom and $d$ is the rank of the VCE for the term. The adjusted critical value satisfies the following probability statement:

$$
\alpha=\operatorname{Pr}\left[F_{d, \nu}>\left\{_{\mathrm{sc}} c_{i}(\alpha)\right\}^{2} / d\right]
$$

pwcompare uses the $\chi^{2}$ distribution when $e\left(d f \_r\right)$ is not posted.

## Tukey's method

For mcompare(tukey), the adjusted $p$-value is

$$
{ }_{t} p_{i}=\operatorname{Pr}\left(q_{k, \nu}>\left|t_{i}\right| \sqrt{2}\right)
$$

where $q_{k, \nu}$ is distributed as the Studentized range statistic for $k$ means and $\nu$ residual degrees of freedom (Miller 1981). The adjusted critical value satisfies the following probability statement:

$$
\alpha=\operatorname{Pr}\left\{q_{k, \nu}>{ }_{t} c_{i}(\alpha) \sqrt{2}\right\}
$$

## Student-Newman-Keuls's method

For mcompare (snk), suppose $t_{i}$ is comparing two margins that have $r$ other margins between them. Then the adjusted $p$-value is

$$
{ }_{\mathrm{snk}} p_{i}=\operatorname{Pr}\left(q_{r+2, \nu}>\left|t_{i}\right| \sqrt{2}\right)
$$

where $r$ ranges from 0 to $k-2$. The adjusted critical value ${ }_{\mathrm{snk}} c_{i}(\alpha)$ satisfies the following probability statement:

$$
\alpha=\operatorname{Pr}\left\{q_{r+2, \nu}>_{\mathrm{snk}} c_{i}(\alpha) \sqrt{2}\right\}
$$

## Duncan's method

For mcompare (duncan), the adjusted $p$-value is

$$
{ }_{\mathrm{dunc}} p_{i}=1-\left(1-\mathrm{snk} p_{i}\right)^{1 /(r+1)}
$$

and the adjusted critical value is

$$
\operatorname{dunc} c_{i}(\alpha)={ }_{\operatorname{snk}} c_{i}\left\{1-(1-\alpha)^{r+1}\right\}
$$

## Dunnett's method

For mcompare (dunnett), the margins are compared with a reference category, resulting in only $k-1$ pairwise comparisons. The adjusted $p$-value is

$$
\operatorname{dunn} p_{i}=\operatorname{Pr}\left(d_{k-1, \nu}>\left|t_{i}\right|\right)
$$

where $d_{k-1, \nu}$ is distributed as the many-one $t$ statistic (Miller 1981, 76). The adjusted critical value dunn $c_{i}(\alpha)$ satisfies the following probability statement:

$$
\alpha=\operatorname{Pr}\left\{d_{k-1, \nu}>_{\text {dunn }} c_{i}(\alpha)\right\}
$$

The multiple-comparison methods for mcompare(tukey), mcompare(snk), mcompare(duncan), and mcompare (dunnett) assume the normal distribution with equal variance; thus, these methods are allowed only with results from anova, regress, manova, and mvreg. mcompare(snk), mcompare (duncan), and mcompare (dunnett) assume equal sample size for each marginal mean. These options will cause pwcompare to report a footnote if unbalanced factors are detected.

## References

Baldwin, S. 2019. Psychological Statistics and Psychometrics Using Stata. College Station, TX: Stata Press.
Dinno, A. 2015. Nonparametric pairwise multiple comparisons in independent groups using Dunn's test. Stata Journal 15: 292-300.

Dunnett, C. W. 1955. A multiple comparison for comparing several treatments with a control. Journal of the American Statistical Association 50: 1096-1121. https://doi.org/10.2307/2281208.
-_ 1964. New tables for multiple comparisons with a control. Biometrics 20: 482-491. https://doi.org/10.2307/2528490.

Kramer, C. Y. 1956. Extension of multiple range tests to group means with unequal numbers of replications. Biometrics 12: 307-310. https://doi.org/10.2307/3001469.

Miller, R. G., Jr. 1981. Simultaneous Statistical Inference. 2nd ed. New York: Springer.
Milliken, G. A., and D. E. Johnson. 2009. Analysis of Messy Data, Volume 1: Designed Experiments. 2nd ed. Boca Raton, FL: CRC Press.

Mitchell, M. N. 2015. Stata for the Behavioral Sciences. College Station, TX: Stata Press.
——. 2021. Interpreting and Visualizing Regression Models Using Stata. 2nd ed. College Station, TX: Stata Press.
Searle, S. R. 1997. Linear Models for Unbalanced Data. New York: Wiley.
Taffé, P., M. Peng, V. Stagg, and T. Williamson. 2017. biasplot: A package to effective plots to assess bias and precision in method comparison studies. Stata Journal 17: 208-221.
2023. Extended biasplot command to assess bias, precision, and agreement in method comparison studies. Stata Journal 23: 97-118.
Tukey, J. W. 1953. The problem of multiple comparisons. Unpublished manuscript, Princeton University.
Winer, B. J., D. R. Brown, and K. M. Michels. 1991. Statistical Principles in Experimental Design. 3rd ed. New York: McGraw-Hill.

## Also see

[R] pwcompare postestimation - Postestimation tools for pwcompare
[R] contrast - Contrasts and linear hypothesis tests after estimation
[ $R$ ] lincom - Linear combinations of parameters
[R] margins - Marginal means, predictive margins, and marginal effects
[R] margins, pwcompare - Pairwise comparisons of margins
[R] pwmean - Pairwise comparisons of means
[R] test - Test linear hypotheses after estimation
[U] 20 Estimation and postestimation commands

## Title

pwcompare postestimation - Postestimation tools for pwcompare
Postestimation commands Remarks and examples Also see

## Postestimation commands

The following postestimation commands are available after pwcompare, post:
Command Description
estat vce variance-covariance matrix of the estimators (VCE)
estat (svy) postestimation statistics for survey data
estimates cataloging estimation results
etable table of estimation results
lincom point estimates, standard errors, testing, and inference for linear combinations of coefficients
nlcom point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients
test Wald tests of simple and composite linear hypotheses
testnl Wald tests of nonlinear hypotheses

## Remarks and examples

When we use the post option with pwcompare, the marginal linear predictions are posted as estimation results, and we can use postestimation commands to perform further analysis on them.

In Pairwise comparisons of means of [R] pwcompare, we fit a regression of wheat yield on types of fertilizers.

```
. use https://www.stata-press.com/data/r18/yield
(Artificial wheat yield dataset)
. regress yield i.fertilizer
    (output omitted)
```

We also used pwcompare with the cimargins option to obtain the marginal mean yield for each fertilizer. We can add the post option to this command to post these marginal means and their VCEs as estimation results.
. pwcompare fertilizer, cimargins post
Pairwise comparisons of marginal linear predictions
Margins: asbalanced

|  | Margin | Std. err. | Unadjusted <br> [95\% conf. interval] |  |
| :---: | ---: | :---: | ---: | ---: |
| fertilizer |  |  |  |  |
| $10-10-10$ | 41.36243 | 1.124298 | 39.14509 | 43.57977 |
| $10-08-22$ | 44.98515 | 1.124298 | 42.7678 | 47.20249 |
| $16-04-08$ | 41.85306 | 1.124298 | 39.63571 | 44.0704 |
| $18-24-06$ | 46.28523 | 1.124298 | 44.06789 | 48.50258 |
| $29-03-04$ | 40.1241 | 1.124298 | 37.90676 | 42.34145 |

Now, we can use nlcom to compute a percentage improvement in the mean yield for fertilizer 2 when compared with fertilizer 1.

```
. nlcom (pct_chg: 100*(_b[2.fertilizer] - _b[1.fertilizer])/_b[1.fertilizer])
```

    pct_chg: 100*(_b[2.fertilizer] - _b[1.fertilizer])/_b[1.fertilizer]
    |  | Coefficient | Std. err. | $z$ | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| :--- | ---: | :--- | :---: | :---: | :---: | :---: |
| pct_chg | 8.758479 | 4.015932 | 2.18 | 0.029 | .8873982 | 16.62956 |

The mean yield for fertilizer 2 is about $9 \%$ higher than that of fertilizer 1, with a standard error of $4 \%$.

## Also see

[R] pwcompare - Pairwise comparisons
[U] 20 Estimation and postestimation commands

## Title

pwmean - Pairwise comparisons of means

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| Reference | Also see |  |  |

## Description

pwmean performs pairwise comparisons of means. It computes all pairwise differences of the means of varname over the combination of the levels of the variables in varlist. The tests and confidence intervals for the pairwise comparisons assume equal variances across groups. pwmean also allows for adjusting the confidence intervals and $p$-values to account for multiple comparisons using Bonferroni's method, Scheffé's method, Tukey's method, Dunnett's method, and others.

See $[\mathrm{R}]$ pwcompare for performing pairwise comparisons of means, estimated marginal means, and other types of marginal linear predictions after anova, regress, and most other estimation commands.

## Quick start

All pairwise differences in the means of y over levels of categorical variable catvar pwmean y, over(catvar)

Same as above, and report test statistics and $p$-values for tests that differences equal zero pwmean $y$, over(catvar) effects

Adjust $p$-values and confidence intervals for multiple comparisons using Tukey's method pwmean y, over(catvar) effects mcompare(tukey)

Same as above, but adjust for multiple comparisons using Bonferroni's method pwmean y, over(catvar) effects mcompare(bonferroni)

Report mean of $y$ for each level of catvar, grouping means that are not significantly different pwmean y, over(catvar) groups

## Menu

Statistics $>$ Summaries, tables, and tests $>$ Summary and descriptive statistics $>$ Pairwise comparisons of means

## Syntax

pwmean varname [if] [in], over(varlist) [options]
options
Main

* over (varlist) mcompare (method)

Reporting
level(\#)
cieffects
pveffects
effects
cimeans
groups
sort
display_options

Description
compare means across each combination of the levels in varlist adjust for multiple comparisons; default is mcompare (noadjust)
*over(varlist) is required.
collect is allowed; see [U] 11.1.10 Prefix commands.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.
method
noadjust
bonferroni
sidak
scheffe
tukey
snk
duncan
dunnett
confidence level; default is level (95)
display a table of mean differences and confidence intervals; the default
display a table of mean differences and $p$-values
display a table of mean differences with $p$-values and confidence intervals
display a table of means and confidence intervals
display a table of means with codes that group them with other means that are not significantly different
sort results tables by displayed mean or difference
control column formats, line width, and factor-variable labeling

Description
do not adjust for multiple comparisons; the default
Bonferroni's method
Šidák's method
Scheffé's method
Tukey's method
Student-Newman-Keuls's method
Duncan's method
Dunnett's method

## Options

$\qquad$ Main
over (varlist) is required and specifies that means are computed for each combination of the levels of the variables in varlist.
mcompare (method) specifies the method for computing $p$-values and confidence intervals that account for multiple comparisons.
Most methods adjust the comparisonwise error rate, $\alpha_{c}$, to achieve a prespecified experimentwise error rate, $\alpha_{e}$.
mcompare(noadjust) is the default; it specifies no adjustment.

$$
\alpha_{c}=\alpha_{e}
$$

mcompare(bonferroni) adjusts the comparisonwise error rate based on the upper limit of the Bonferroni inequality:

$$
\alpha_{e} \leq m \alpha_{c}
$$

where $m$ is the number of comparisons within the term.
The adjusted comparisonwise error rate is

$$
\alpha_{c}=\alpha_{e} / m
$$

mcompare (sidak) adjusts the comparisonwise error rate based on the upper limit of the probability inequality

$$
\alpha_{e} \leq 1-\left(1-\alpha_{c}\right)^{m}
$$

where $m$ is the number of comparisons within the term.
The adjusted comparisonwise error rate is

$$
\alpha_{c}=1-\left(1-\alpha_{e}\right)^{1 / m}
$$

This adjustment is exact when the $m$ comparisons are independent.
mcompare (scheffe) controls the experimentwise error rate using the $F$ (or $\chi^{2}$ ) distribution with degrees of freedom equal to $k-1$ where $k$ is the number of means being compared.
mcompare (tukey) uses what is commonly referred to as Tukey's honestly significant difference. This method uses the Studentized range distribution instead of the $t$ distribution.
mcompare (snk) is a variation on mcompare (tukey) that counts only the number of means participating in the range for a given comparison instead of the full number of means.
mcompare (duncan) is a variation on mcompare (snk) with additional adjustment to the significance probabilities.
mcompare (dunnett) uses Dunnett's method for making comparisons with a reference category.

## Reporting

level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals. The significance level used by the groups option is $100-\#$, expressed as a percentage.
cieffects specifies that a table of the pairwise comparisons of means with their standard errors and confidence intervals be reported. This is the default.
pveffects specifies that a table of the pairwise comparisons of means with their standard errors, test statistics, and $p$-values be reported.
effects specifies that a table of the pairwise comparisons of means with their standard errors, test statistics, $p$-values, and confidence intervals be reported.
cimeans specifies that a table of the means with their standard errors and confidence intervals be reported.
groups specifies that a table of the means with their standard errors and group codes be reported. Means with the same letter in the group code are not significantly different at the specified significance level.
sort specifies that the reported tables be sorted by the mean or difference that is displayed in the table.
display_options: nofvlabel, fvwrap(\#), fvwrapon(style), cformat(\%fmt), pformat (\%fmt), sformat (\% fmt), and nolstretch.
nofvlabel displays factor-variable level values rather than attached value labels. This option overrides the fvlabel setting; see [R] set showbaselevels.
fvwrap(\#) specifies how many lines to allow when long value labels must be wrapped. Labels requiring more than \# lines are truncated. This option overrides the fvwrap setting; see [R] set showbaselevels.
fvwrapon(style) specifies whether value labels that wrap will break at word boundaries or break based on available space.
fvwrapon(word), the default, specifies that value labels break at word boundaries.
fvwrapon(width) specifies that value labels break based on available space.
This option overrides the fvwrapon setting; see [R] set showbaselevels.
cformat (\% fint) specifies how to format means, standard errors, and confidence limits in the table of pairwise comparison of means.
pformat (\%fmt) specifies how to format $p$-values in the table of pairwise comparison of means.
sformat (\%fmt) specifies how to format test statistics in the table of pairwise comparison of means.
nolstretch specifies that the width of the table of pairwise comparisons not be automatically widened to accommodate longer variable names. The default, lstretch, is to automatically widen the table of pairwise comparisons up to the width of the Results window. Specifying lstretch or nolstretch overrides the setting given by set lstretch. If set lstretch has not been set, the default is lstretch. nolstretch is not shown in the dialog box.

## Remarks and examples

pwmean performs pairwise comparisons (differences) of means, assuming a common variance among groups. It can easily adjust the $p$-values and confidence intervals for the differences to account for the elevated type I error rate due to multiple comparisons. Adjustments for multiple comparisons can be made using Bonferroni's method, Scheffé's method, Tukey's method, Dunnett's method, and others.

See $[R]$ margins, pwcompare for performing pairwise comparisons of marginal probabilities and other linear and nonlinear predictions after estimation commands.

Remarks are presented under the following headings:
Group means
Pairwise differences of means
Group output
Adjusting for multiple comparisons
Tukey's method
Dunnett's method
Multiple over() variables
Equal variance assumption

## Group means

Suppose we have data on the wheat yield of fields that were each randomly assigned an application of one of five types of fertilizers. Let's first look at the mean yield for each type of fertilizer.

```
. use https://www.stata-press.com/data/r18/yield
(Artificial wheat yield dataset)
. pwmean yield, over(fertilizer) cimeans
Pairwise comparisons of means with equal variances
Over: fertilizer
```

| yield | Mean | Std. err. | Unadjusted <br> [95\% <br> conf. interval] |  |
| :---: | ---: | :---: | ---: | ---: |
| fertilizer |  |  |  |  |
| $10-10-10$ | 41.36243 | 1.124298 | 39.14509 | 43.57977 |
| $10-08-22$ | 44.98515 | 1.124298 | 42.7678 | 47.20249 |
| $16-04-08$ | 41.85306 | 1.124298 | 39.63571 | 44.0704 |
| $18-24-06$ | 46.28523 | 1.124298 | 44.06789 | 48.50258 |
| $29-03-04$ | 40.1241 | 1.124298 | 37.90676 | 42.34145 |

## Pairwise differences of means

We can compute all pairwise differences in mean wheat yields for the types of fertilizers.

```
. pwmean yield, over(fertilizer) effects
Pairwise comparisons of means with equal variances
Over: fertilizer
```

| yield | Contrast | Std. err. | Unadjusted |  | Unadjusted |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  | t | $P>\|t\|$ | [95\% conf | interval] |
| fertilizer |  |  |  |  |  |  |
| 10-08-22 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | 3.62272 | 1.589997 | 2.28 | 0.024 | . 4869212 | 6.758518 |
| 16-04-08 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | . 4906299 | 1.589997 | 0.31 | 0.758 | $-2.645169$ | 3.626428 |
| 18-24-06 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | 4.922803 | 1.589997 | 3.10 | 0.002 | 1.787005 | 8.058602 |
| 29-03-04 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | -1.238328 | 1.589997 | -0.78 | 0.437 | $-4.374127$ | 1.89747 |
| 16-04-08 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-08-22 | -3.13209 | 1.589997 | -1.97 | 0.050 | -6.267889 | . 0037086 |
| 18-24-06 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-08-22 | 1.300083 | 1.589997 | 0.82 | 0.415 | -1.835715 | 4.435882 |
| 29-03-04 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-08-22 | -4.861048 | 1.589997 | -3.06 | 0.003 | -7.996847 | -1.725249 |
| 18-24-06 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 16-04-08 | 4.432173 | 1.589997 | 2.79 | 0.006 | 1.296375 | 7.567972 |
| 29-03-04 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 16-04-08 | -1.728958 | 1.589997 | -1.09 | 0.278 | $-4.864757$ | 1.406841 |
| 29-03-04 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 18-24-06 | -6.161132 | 1.589997 | -3.87 | 0.000 | -9.29693 | -3.025333 |

The contrast in the row labeled ( $10-08-22$ vs $10-10-10$ ) is the difference in the mean wheat yield for fertilizer 10-08-22 and fertilizer 10-10-10. At a $5 \%$ significance level, we conclude that there is a difference in the means for these two fertilizers. Likewise, the rows labeled (18-24-06 vs $10-10-10$ ), ( $29-03-04$ vs $10-08-22$ ), ( $18-24-06$ vs $16-04-08$ ) and ( $29-03-04$ vs $18-24-06$ ) show differences in these pairs of means. In all, we find that 5 of the 10 mean differences are significantly different from zero at a 5\% significance level.

We can specify the sort option to order the differences from smallest to largest in the table.

| yield | Contrast | Std. err. | Unad <br> t | usted $P>\|t\|$ | Unad [95\% conf | usted <br> interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| $\begin{gathered} \text { fertilizer } \\ 29-03-04 \end{gathered}$ |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| $\begin{aligned} & 18-24-06 \\ & 29-03-04 \end{aligned}$ | -6.161132 | 1.589997 | $-3.87$ | 0.000 | -9.29693 | -3.025333 |
| vs |  |  |  |  |  |  |
| 10-08-22 | -4.861048 | 1.589997 | -3.06 | 0.003 | -7.996847 | -1.725249 |
| 16-04-08 |  |  |  |  |  |  |
| 10-08-22 | -3.13209 | 1.589997 | -1.97 | 0.050 | -6.267889 | . 0037086 |
| 29-03-04 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 16-04-08 | -1.728958 | 1.589997 | -1.09 | 0.278 | -4.864757 | 1.406841 |
| 29-03-04 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | $-1.238328$ | 1.589997 | -0.78 | 0.437 | -4.374127 | 1.89747 |
| 16-04-08 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | . 4906299 | 1.589997 | 0.31 | 0.758 | -2.645169 | 3.626428 |
| 18-24-06 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-08-22 | 1.300083 | 1.589997 | 0.82 | 0.415 | -1.835715 | 4.435882 |
| 10-08-22 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | 3.62272 | 1.589997 | 2.28 | 0.024 | . 4869212 | 6.758518 |
| 18-24-06 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 16-04-08 | 4.432173 | 1.589997 | 2.79 | 0.006 | 1.296375 | 7.567972 |
| 18-24-06 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | 4.922803 | 1.589997 | 3.10 | 0.002 | 1.787005 | 8.058602 |

Ordering the pairwise differences is particularly convenient when we are comparing means for a large number of groups.

## Group output

We can use the group option to see the mean of each group and a visual representation of the tests for differences.

```
. pwmean yield, over(fertilizer) group sort
Pairwise comparisons of means with equal variances
Over: fertilizer
```

| yield | Mean | Std. err. | Unadjusted <br> groups |
| :---: | ---: | :---: | ---: |
| fertilizer |  |  |  |
| $29-03-04$ | 40.1241 | 1.124298 | A |
| $10-10-10$ | 41.36243 | 1.124298 | A |
| $16-04-08$ | 41.85306 | 1.124298 | AB |
| $10-08-22$ | 44.98515 | 1.124298 | BC |
| $18-24-06$ | 46.28523 | 1.124298 | C |

Note: Means sharing a letter in the group label are not significantly different at the $5 \%$ level.

Fertilizers 29-03-04, 10-10-10, and 16-04-08 are all in group A. This means that at our 5\% level of significance, we have insufficient information to distinguish their means. Likewise, fertilizers 16-0408 and 10-08-22 are in group B and cannot be distinguished at the $5 \%$ level. The same is true for fertilizers 10-08-22 and 18-24-06 in group C.

Fertilizer 29-03-04 and fertilizer 10-08-22 have no letters in common, indicating that the mean yields of these two groups are significantly different at the $5 \%$ level. We can conclude that any other fertilizers without a letter in common have significantly different means as well.

## Adjusting for multiple comparisons

The statistics in the examples above take no account that we are performing 10 comparisons. With our $5 \%$ significance level and assuming the comparisons are independent, we expect 1 in 20 tests of comparisons to be significant, even if all the population means are truly the same. If we are performing many comparisons, then we should account for the fact that some tests will be found significant by chance alone. More formally, the test for each pairwise comparison is made without adjusting for the elevated type I experimentwise error rate that is introduced when performing multiple tests. We can use the mcompare () option to adjust the confidence intervals and $p$-values for multiple comparisons.

## Tukey's method

Of the available adjustments for multiple comparisons, Tukey's honestly significant difference, Student-Newman-Keuls's method, and Duncan's method are most often used when performing all pairwise comparisons of means. Of these, Tukey's method is the most conservative and Duncan's method is the least conservative. For further discussion of each of the multiple-comparison adjustments, see [R] pwcompare.

Here we use Tukey's adjustment to compute $p$-values and confidence intervals for the pairwise differences.
. pwmean yield, over(fertilizer) effects sort mcompare(tukey)
Pairwise comparisons of means with equal variances
Over: fertilizer

|  | Number of <br> comparisons |
| ---: | ---: |
| fertilizer | 10 |


| yield | Contrast | Std. err. | Tukey |  | Tukey |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  | t | $P>\|t\|$ | [95\% con | interval] |
| $\begin{gathered} \text { fertilizer } \\ 29-03-04 \end{gathered}$ |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 18-24-06 | -6.161132 | 1.589997 | -3.87 | 0.001 | -10.53914 | -1.78312 |
| 29-03-04 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-08-22 | -4.861048 | 1.589997 | -3.06 | 0.021 | -9.239059 | -. 4830368 |
| 16-04-08 |  |  |  |  |  |  |
| vs |  |  |  |  | -7.510101 | 1.245921 |
| 10-08-22 | -3.13209 | 1.589997 | -1.97 | 0.285 |  |  |
| 29-03-04 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 16-04-08 | -1.728958 | 1.589997 | -1.09 | 0.813 | -6.106969 | 2.649053 |
| 29-03-04 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | -1.238328 | 1.589997 | -0.78 | 0.936 | -5.616339 | 3.139683 |
| 16-04-08 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | . 4906299 | 1.589997 | 0.31 | 0.998 | -3.887381 | 4.868641 |
| 18-24-06 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-08-22 | 1.300083 | 1.589997 | 0.82 | 0.925 | -3.077928 | 5.678095 |
| 10-08-22 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | 3.62272 | 1.589997 | 2.28 | 0.156 | -. 7552913 | 8.000731 |
| 18-24-06 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 16-04-08 | 4.432173 | 1.589997 | 2.79 | 0.046 | . 0541623 | 8.810185 |
| 18-24-06 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | 4.922803 | 1.589997 | 3.10 | 0.019 | . 5447922 | 9.300815 |

When using a 5\% significance level, Tukey's adjustment indicates that four pairs of means are different. With the adjustment, we no longer conclude that the difference in the mean yields for fertilizers $10-08-22$ and $10-10-10$ is significantly different from zero.

## Dunnett's method

Now, let's suppose that fertilizer 10-10-10 actually represents fields on which no fertilizer was applied. In this case, we can use Dunnett's method for comparing each of the fertilizers with the control.

```
. pwmean yield, over(fertilizer) effects mcompare(dunnett)
Pairwise comparisons of means with equal variances
Over: fertilizer
```

|  | Number of <br> comparisons |
| :--- | ---: |
| fertilizer | 4 |


| yield | Contrast | Std. err. | Dunnett |  | Dunnett |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  | t | $P>\|t\|$ | [95\% conf | interval] |
| fertilizer |  |  |  |  |  |  |
| 10-08-22 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | 3.62272 | 1.589997 | 2.28 | 0.079 | -. 2918331 | 7.537273 |
| 16-04-08 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | . 4906299 | 1.589997 | 0.31 | 0.994 | $-3.423923$ | 4.405183 |
| 18-24-06 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | 4.922803 | 1.589997 | 3.10 | 0.008 | 1.00825 | 8.837356 |
| 29-03-04 |  |  |  |  |  |  |
| vs |  |  |  |  |  |  |
| 10-10-10 | -1.238328 | 1.589997 | -0.78 | 0.852 | -5.152881 | 2.676225 |

Using Dunnett's adjustment, we conclude that only fertilizer 4 (18-24-06) produces a mean yield that is significantly different from the mean yield of the field with no fertilizer applied.

By default, pwmean treats the lowest level of the group variable as the control. If, for instance, fertilizer 3 (16-04-08) was our control group, we could type

```
. pwmean yield, over(b3.fertilizer) effects mcompare(dunnett)
```

using the b3. factor-variable operator to specify this level as the reference level.

## Multiple over() variables

When we specify more than one variable in the over () option, pairwise comparisons are performed for the means defined by each combination of levels of these variables.

```
. pwmean yield, over(fertilizer irrigation) group
Pairwise comparisons of means with equal variances
Over: fertilizer irrigation
```

| yield | Mean | Std. err. | Unadjusted groups |
| :---: | :---: | :---: | :---: |
| fertilizer\#irrigation |  |  |  |
| 10-10-10\#0 | 36.91257 | 1.116571 | A |
| 10-10-10\#1 | 45.81229 | 1.116571 | B |
| 10-08-22\#0 | 38.79482 | 1.116571 | A C |
| 10-08-22\#1 | 51.17547 | 1.116571 | E |
| 16-04-08\#0 | 36.34383 | 1.116571 | A |
| 16-04-08\#1 | 47.36229 | 1.116571 | B |
| 18-24-06\#0 | 41.81757 | 1.116571 | CD |
| 18-24-06\#1 | 50.7529 | 1.116571 | E |
| 29-03-04\#0 | 35.69507 | 1.116571 | A |
| 29-03-04\#1 | 44.55313 | 1.116571 | B D |

Note: Means sharing a letter in the group label are not significantly different at the $5 \%$ level.

Here the row labeled $10-10-10 \# 0$ is the mean for the fields treated with fertilizer 10-10-10 and without irrigation. This mean is significantly different from the mean of all fertilizer/irrigation pairings that do not have an A in the "Unadjusted groups" column. These include all pairings where the fields were irrigated as well as the fields treated with fertilizer 18-24-06 but without irrigation.

## Equal variance assumption

pwmean performs multiple comparisons assuming that there is a common variance for all groups. In the case of two groups, this is equivalent to performing the familiar two-sample $t$ test when equal variances are assumed.
. ttest yield, by(irrigation)
Two-sample t test with equal variances

. pwmean yield, over(irrigation) effects
Pairwise comparisons of means with equal variances
Over: irrigation

| yield | Contrast | Std. err. | Unadjusted |  | Unadjusted |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| irrigation |  |  |  |  |  |  |
| 1 vs 0 | 10.01844 | . 7732872 | 12.96 | 0.000 | 8.493509 | 11.54338 |

The signs for the difference, the test statistic, and the confidence intervals are reversed because the difference is taken in the opposite direction. The $p$-value from pwmean is equivalent to the one for the two-sided test in the ttest output.
pwmean extends the capabilities of ttest to allow for simultaneously comparing all pairs of means and to allow for using one common variance estimate for all the tests instead of computing a separate pooled variance for each pair of means when using multiple ttest commands. In addition, pwmean allows adjustments for multiple comparisons, many of which rely on an assumption of equal variances among groups.

## Stored results

pwmean stores the following in e() :
Scalars
e(df_r)
e(balanced)
Macros
e(cmd)
e(cmdline)
e(title)
e(depvar)
e(over)
e(properties)
Matrices
e(b)
e(V)
e(error)
e(b_vs)
e(V_vs)
e(error_vs)
variance degrees of freedom
1 if fully balanced data, 0 otherwise
pwmean
command as typed
title in output
name of variable from which the means are computed
varlist from over()
b V
mean estimates
variance-covariance matrix of the mean estimates
mean estimability codes;
0 means estimable,
8 means not estimable
mean difference estimates
variance-covariance matrix of the mean difference estimates
mean difference estimability codes;
0 means estimable,
8 means not estimable

In addition to the above, the following is stored in r() :

Scalars
$r$ (level) confidence level of confidence intervals
Macros
r(groups\#)
r(mcmethod_vs)
r(mctitle_vs)
Matrices
$r$ (table)
r(table_vs)
r(k_groups)
group codes for the \#th margin in $r(b)$
method from mcompare()
title for method from mcompare() and confidence intervals. $p$-values, and confidence intervals
number of significance groups for each term
matrix containing the coefficients with their standard errors, test statistics, $p$-values,
matrix containing the margin differences with their standard errors, test statistics,

Note that results stored in r() are updated when the command is replayed and will be replaced when any r -class command is run after the estimation command.

## Methods and formulas

pwmean is a convenience command that uses pwcompare after fitting a fully factorial linear model.
See Methods and formulas described in [R] pwcompare.

## Reference

Searle, S. R. 1997. Linear Models for Unbalanced Data. New York: Wiley.

## Also see

[R] pwmean postestimation - Postestimation tools for pwmean
[R] contrast - Contrasts and linear hypothesis tests after estimation
[R] margins - Marginal means, predictive margins, and marginal effects
[R] margins, pwcompare - Pairwise comparisons of margins
[R] pwcompare - Pairwise comparisons
[R] ttest - $t$ tests (mean-comparison tests)
[U] 20 Estimation and postestimation commands

## Title

pwmean postestimation - Postestimation tools for pwmean

Postestimation commands Remarks and examples Also see

## Postestimation commands

The following postestimation commands are available after pwmean:

| Command | Description |
| :--- | :--- |
| estat vce | variance-covariance matrix of the estimators (VCE) <br> estimates <br> cataloging estimation results <br> etable |
| lincom | point estimates, standard errors, testing, and inference for linear combinations of <br> coefficients |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations <br> of coefficients |
| test | Wald tests of simple and composite linear hypotheses <br> testnl |

## Remarks and examples

In Pairwise differences of means of [R] pwmean, we computed all pairwise differences in mean wheat yields for five fertilizers.

```
. use https://www.stata-press.com/data/r18/yield
(Artificial wheat yield dataset)
. pwmean yield, over(fertilizer)
Pairwise comparisons of means with equal variances
Over: fertilizer
```

| yield | Contrast | Std. err. | $\begin{array}{r} \text { Una } \\ {[95 \% \text { con }} \end{array}$ | usted interval] |
| :---: | :---: | :---: | :---: | :---: |
| fertilizer |  |  |  |  |
| $10-08-22$ vs $10-10-10$ | 3.62272 | 1.589997 | . 4869212 | 6.758518 |
| $16-04-08$ vs $10-10-10$ | . 4906299 | 1.589997 | -2.645169 | 3.626428 |
| $18-24-06$ vs $10-10-10$ | 4.922803 | 1.589997 | 1.787005 | 8.058602 |
| $29-03-04$ vs $10-10-10$ | -1.238328 | 1.589997 | -4.374127 | 1.89747 |
| $16-04-08$ vs $10-08-22$ | -3.13209 | 1.589997 | -6.267889 | . 0037086 |
| $18-24-06$ vs $10-08-22$ | 1.300083 | 1.589997 | -1.835715 | 4.435882 |
| $29-03-04$ vs $10-08-22$ | -4.861048 | 1.589997 | -7.996847 | -1.725249 |
| $18-24-06$ vs 16-04-08 | 4.432173 | 1.589997 | 1.296375 | 7.567972 |
| $29-03-04$ vs 16-04-08 | -1.728958 | 1.589997 | -4.864757 | 1.406841 |
| $29-03-04$ vs 18-24-06 | -6.161132 | 1.589997 | -9.29693 | -3.025333 |

After pwmean, we can use testnl to test whether the improvement in mean wheat yield when using fertilizer 18-24-06 instead of fertilizer 29-03-04 is significantly different from $10 \%$.
. testnl (_b[4.fertilizer] - _b[5.fertilizer])/_b[5.fertilizer] = 0.1
(1) (_b[4.fertilizer] - _b[5.fertilizer])/_b[5.fertilizer] = 0.1 chi2(1) = 1.57
Prob $>$ chi2 $=0.2106$
The improvement is not significantly different from $10 \%$.

## Also see

[R] pwmean - Pairwise comparisons of means
[U] 20 Estimation and postestimation commands

## Title

QC - Quality control charts

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

These commands provide standard quality-control charts. cchart draws a c chart; pchart, a p (fraction-defective) chart; rchart, an $R$ (range or dispersion) chart; xchart, an $\overline{\mathrm{X}}$ (control line) chart; and shewhart, vertically aligned $\bar{X}$ and $R$ charts.

## Quick start

c chart for dvar defects per unit identified by uvar cchart dvar uvar
p chart for dvar defective items out of nvar items inspected from each unit identified by uvar pchart dvar uvar nvar

Same as above, but stabilize the p chart for unequal numbers of items inspected per unit pchart dvar uvar nvar, stabilized
$R$ chart for the range of measurements $m 1, m 2, m 3$, and $m 4$ rchart m1 m2 m3 m4, connect(1)

Same as above, but use known process standard deviation of 0.5 for control limits rchart m 1 m 2 m 3 m 4 , connect (1) $\operatorname{std}(.5)$
$\bar{X}$ chart for measurements $m 1, m 2, m 3$, and $m 4$
xchart m1 m2 m3 m4, connect(1)
Same as above, but use known process standard deviation of 0.5 and grand mean of 10 for control limits
xchart m1 m2 m3 m4, connect(1) std(.5) mean(10)
Shewhart chart with vertically aligned $R$ and $\bar{X}$ charts
shewhart m 1 m 2 m 3 m 4 , connect(1) std(.5) mean(10)

## Menu

## cchart

Statistics $>$ Other $>$ Quality control $>\mathrm{C}$ chart
pchart
Statistics $>$ Other $>$ Quality control $>\mathrm{P}$ chart
rchart
Statistics $>$ Other $>$ Quality control $>\mathrm{R}$ chart

## xchart

Statistics $>$ Other $>$ Quality control $>$ X-bar chart

## shewhart

Statistics $>$ Other $>$ Quality control $>$ Vertically aligned X-bar and R chart

## Syntax

Draw a c chart
cchart defect_var unit_var [, cchart_options]

Draw a $p$ (fraction-defective) chart
pchart reject_var unit_var ssize_var [, pchart_options]

Draw an $R$ (range or dispersion) chart rchart varlist [if] [in] [, rchart_options]

Draw an $\overline{\mathrm{X}}$ (control line) chart
xchart varlist [if] [in] [, xchart_options $]$

Draw vertically aligned $\overline{\mathrm{X}}$ and $R$ charts
shewhart varlist [if] [in] [, shewhart_options]
cchart_options
Description
Main
nograph
Plot
connect_options
marker_options
marker_label_options
Control limits
clopts (cline_options)
Add plots
addplot (plot)
Y axis, X axis, Titles, Legend, Overall
twoway_options
suppress graph
affect rendition of the plotted points change look of markers (color, size, etc.) add marker labels; change look or position
affect rendition of the control limits
add other plots to the generated graph
any options other than by () documented in [G-3] twoway_options
pchart_options
Description

## Main

stabilized
nograph


Plot
connect_options
marker_options
marker_label_options

## Control limits

clopts(cline_options)
Add plots
addplot (plot)
Y axis, X axis, Titles, Legend, Overall
twoway_options
stabilize the p chart when sample sizes are unequal suppress graph
store the fractions of defective elements and the lower and upper control limits
affect rendition of the plotted points
change look of markers (color, size, etc.)
add marker labels; change look or position
affect rendition of the control limits
add other plots to the generated graph
any options other than by () documented in [G-3] twoway_options
rchart_options
Main
std (\#)
nograph
Plot
connect_options
marker_options
marker_label_options
Control limits
clopts(cline_options)
Add plots
addplot (plot)
Y axis, X axis, Titles, Legend, Overall
twoway_options

Description
user-specified standard deviation
suppress graph
affect rendition of the plotted points change look of markers (color, size, etc.) add marker labels; change look or position
affect rendition of the control limits
add other plots to the generated graph
any options other than by () documented in [G-3] twoway_options
xchart_options
Main

```
    std(#)
```

    mean (\#)
    lower (\#) upper (\#)
    nograph
    Plot
connect_options
marker_options
marker_label_options
Control limits
clopts (cline_options)
Add plots
addplot (plot)
Y axis, X axis, Titles, Legend, Overall
twoway_options

Description
user-specified standard deviation user-specified mean
lower and upper limits of the X-bar limits suppress graph
affect rendition of the plotted points change look of markers (color, size, etc.) add marker labels; change look or position
affect rendition of the control limits
add other plots to the generated graph
any options other than by () documented in [G-3] twoway_options
shewhart_options
Main
std (\#)
mean (\#)
nograph
Plot
connect_options
marker_options
marker_label_options
Control limits
clopts(cline_options)
Y axis, X axis, Titles, Legend, Overall combine_options

Description
user-specified standard deviation
user-specified mean
suppress graph
affect rendition of the plotted points
change look of markers (color, size, etc.)
add marker labels; change look or position
affect rendition of the control limits
any options documented in [G-2] graph combine
collect is allowed with all QC commands; see [U] 11.1.10 Prefix commands.

## Options

Main
stabilized stabilizes the p chart when sample sizes are unequal.
std(\#) specifies the standard deviation of the process. The R chart is calculated (based on the range) if this option is not specified.
mean (\#) specifies the grand mean, which is calculated if not specified.
lower (\#) and upper (\#) must be specified together or not at all. They specify the lower and upper limits of the $\overline{\mathrm{X}}$ chart. Calculations based on the mean and standard deviation (whether specified by option or calculated) are used otherwise.
nograph suppresses the graph.
generate (newvar ${ }_{f}$ newvar $_{\mathrm{lcl}}$ newvar $_{\mathrm{ucl}}$ ) stores the plotted values in the p chart. newvar ${ }_{f}$ will contain the fractions of defective elements; newvar ${ }_{\mathrm{lcl}}$ and $n e w v a r ~_{\text {ucl }}$ will contain the lower and upper control limits, respectively.
$\qquad$ Plot
connect_options affect whether lines connect the plotted points and the rendition of those lines; see [G-3] connect_options.
marker_options affect the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G-3] marker_options.
marker_label_options specify if and how the markers are to be labeled; see [G-3] marker_label_options.

Control limits
clopts(cline_options) affects the rendition of the control limits; see [G-3] cline_options.

Add plots
addplot (plot) provides a way to add other plots to the generated graph. See [G-3] addplot_option.

Y axis, X axis, Titles, Legend, Overall
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).
combine_options (shewhart only) are any of the options documented in [G-2] graph combine. These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

## Remarks and examples

Control charts may be used to define the goal of a repetitive process, to control that process, and to determine if the goal has been achieved. Walter A. Shewhart of Bell Telephone Laboratories devised the first control chart in 1924. In 1931, Shewhart published Economic Control of Quality of Manufactured Product. According to Burr, "Few fields of knowledge have ever been so completely explored and charted in the first exposition" (1976, 29). Shewhart states that "a phenomenon will be said to be controlled when, through the use of past experience, we can predict, at least within limits, how the phenomenon may be expected to vary in the future. Here it is understood that prediction within limits means that we can state, at least approximately, the probability that the observed phenomenon will fall within given limits" $(1931,6)$.

For more information on quality-control charts, see Burr (1976), Duncan (1986), Harris (1999), or Ryan (2011).

## > Example 1: cchart

cchart graphs a c chart showing the number of nonconformities in a unit, where defect_var records the number of defects in each inspection unit and unit_var records the unit number. The unit numbers need not be in order. For instance, consider the following example dataset from Ryan (2011, 186):


Sorted by:
. list in $1 / 5$

|  | day | defects |
| :--- | ---: | ---: |
|  |  |  |
| 1. | 1 | 7 |
| 2. | 2 | 5 |
| 3. | 3 | 11 |
| 4. | 4 | 13 |
| 5. | 5 | 9 |
|  |  |  |

cchart defects day, title(c chart for nonconforming transistors)
c chart for nonconforming transistors


The expected number of defects is 10.6 , with lower and upper control limits of 0.8327 and 20.37, respectively. No units are out of control.

## > Example 2: pchart

pchart graphs a p chart, which shows the fraction of nonconforming items in a subgroup, where reject_var records the number rejected in each inspection unit, unit_var records the inspection unit number, and ssize_var records the number inspected in each unit.

Consider the example dataset from Ryan $(2011,186)$ of the number of nonconforming transistors out of 1,000 inspected each day during the month of April:


Sorted by:
. list in $1 / 5$

|  | day | rejects | ssize |
| :--- | ---: | ---: | ---: |
| 1. | 1 | 7 | 1000 |
| 2. | 2 | 5 | 1000 |
| 3. | 3 | 11 | 1000 |
| 4. | 4 | 13 | 1000 |
| 5. | 5 | 9 | 1000 |

pchart rejects day ssize


All the points are within the control limits, which are 0.0009 for the lower limit and 0.0203 for the upper limit.

Here the sample sizes are fixed at 1,000 , so the ssize variable contains 1,000 for each observation. Sample sizes need not be fixed, however. Say that our data were slightly different:
. use https://www.stata-press.com/data/r18/ncu3
. list in $1 / 5$

|  | day | rejects | ssize |
| :--- | ---: | ---: | ---: |
| 1. | 1 | 7 | 920 |
| 2. | 2 | 5 | 920 |
| 3. | 3 | 11 | 920 |
| 4. | 4 | 13 | 950 |
| 5. | 5 | 9 | 950 |
|  |  |  |  |

pchart rejects day ssize


Here the control limits are, like the sample size, no longer constant. The stabilize option will stabilize the control chart:
. pchart rejects day ssize, stabilize


0 units are out of control.
Stabilized p Chart, average number of defects $=.0119$.
> Example 3: rchart
rchart displays an R chart showing the range for repeated measurements at various times. Variables within observations record measurements. Observations represent different samples.

For instance, say that we take five samples of 5 observations each. In our first sample, our measurements are $10,11,10,11$, and 12 . The data are
. list

|  | m 1 | m 2 | m 3 | m 4 | m 5 |
| :--- | ---: | ---: | ---: | ---: | ---: |
|  | 10 | 11 | 10 | 11 | 12 |
| 2. | 12 | 10 | 9 | 10 | 9 |
| 3. | 10 | 11 | 10 | 12 | 10 |
| 4. | 9 | 9 | 9 | 10 | 11 |
| 5. | 12 | 12 | 12 | 12 | 13 |

. rchart m1-m5, connect(1)


The expected range in each sample is 2 with lower and upper control limits of 0 and 4.23 , respectively. If we know that the process standard deviation is 0.3 , we could specify
. rchart m1-m5, connect(1) std(.3)

> Example 4: xchart
xchart graphs an $\bar{X}$ chart for repeated measurements at various times. Variables within observations record measurements, and observations represent different samples. Using the same data as in the previous example, we type . xchart m1-m5, connect(1)


The average measurement in the sample is 10.64 , and the lower and upper control limits are 9.486 and 11.794 , respectively. Suppose that we knew from prior information that the mean of the process is 11 . Then, we would type
. xchart m1-m5, connect(1) mean(11)


If we also know that the standard deviation of the process is 0.3 , we could type
. xchart m1-m5, connect(1) mean(11) std(.3)


Finally, xchart allows us to specify our own control limits:

```
. xchart m1-m5, connect(1) mean(11) lower(10) upper(12)
```


> Example 5: shewhart
shewhart displays a vertically aligned $\bar{X}$ and $R$ chart in the same image. To produce the bestlooking combined image possible, you will want to use the xchart and rchart commands separately and then combine the graphs. shewhart, however, is more convenient.

Using the same data as previously, but realizing that the standard deviation should have been 0.4 , we type

```
. shewhart m1-m5, connect(1) mean(11) std(.4)
```



Walter Andrew Shewhart (1891-1967) was born in Illinois and educated as a physicist, with degrees from the Universities of Illinois and California. After a brief period teaching physics, he worked for the Western Electric Company and (from 1925) the Bell Telephone Laboratories. His name is most associated with control charts used in quality controls, but his many other interests ranged generally from quality assurance to the philosophy of science.

## Stored results

cchart stores the following in $r()$ :
Scalars
$r$ (cbar) expected number of nonconformities
$r\left(l c l \_c\right) \quad$ lower control limit
$r$ (ucl_c) upper control limit
$r(N) \quad$ number of observations
$r$ (out_c) number of units out of control
$r$ (below_c) number of units below the lower limit
$r$ (above_c) number of units above the upper limit
pchart stores the following in $r()$ :
Scalars
$r$ (pbar) average fraction of nonconformities
$r\left(l c l \_p\right) \quad$ lower control limit
$r$ (ucl_p) upper control limit
$r(N) \quad$ number of observations
$r$ (out_p) number of units out of control
$r$ (below_p) number of units below the lower limit
$r$ (above_p) number of units above the upper limit
rchart stores the following in $r()$ :
Scalars
$r$ (central_line) ordinate of the central line
r(lcl_r) lower control limit
$r$ (ucl_r) upper control limit
$r(N) \quad$ number of observations
$r$ (out_r) number of units out of control
$r$ (below_r) number of units below the lower limit
$r$ (above_r) number of units above the upper limit
xchart stores the following in $r()$ :
Scalars
$r$ (xbar) grand mean
$r\left(1 c l \_x\right) \quad$ lower control limit
$r$ (ucl_x) upper control limit
$r(N) \quad$ number of observations
$r$ (out_x) number of units out of control
$r$ (below_x) number of units below the lower limit
$r$ (above_x) number of units above the upper limit
shewhart stores in $r()$ the combination of stored results from xchart and rchart.

## Methods and formulas

For the c chart, the number of defects per unit, $C$, is taken to be a value of a random variable having a Poisson distribution. If $k$ is the number of units available for estimating $\lambda$, the parameter of the Poisson distribution, and if $C_{i}$ is the number of defects in the $i$ th unit, then $\lambda$ is estimated by $\bar{C}=\sum_{i} C_{i} / k$. Then

$$
\begin{aligned}
\text { central line } & =\bar{C} \\
\mathrm{UCL} & =\bar{C}+3 \sqrt{\bar{C}} \\
\mathrm{LCL} & =\bar{C}-3 \sqrt{\bar{C}}
\end{aligned}
$$

Control limits for the p chart are based on the sampling theory for proportions, using the normal approximation to the binomial. If $k$ samples are taken, the estimator of $p$ is given by $\bar{p}=\sum_{i} \widehat{p}_{i} / k$, where $\widehat{p_{i}}=x_{i} / n_{i}$, and $x_{i}$ is the number of defects in the $i$ th sample of size $n_{i}$. The central line and the control limits are given by

$$
\begin{aligned}
\text { central line } & =\bar{p} \\
\mathrm{UCL} & =\bar{p}+3 \sqrt{\bar{p}(1-\bar{p}) / n_{i}} \\
\mathrm{LCL} & =\bar{p}-3 \sqrt{\bar{p}(1-\bar{p}) / n_{i}}
\end{aligned}
$$

Control limits for the R chart are based on the distribution of the range of samples of size $n$ from a normal population. If the standard deviation of the process, $\sigma$, is known,

$$
\begin{aligned}
\text { central line } & =d_{2} \sigma \\
\mathrm{UCL} & =D_{2} \sigma \\
\mathrm{LCL} & =D_{1} \sigma
\end{aligned}
$$

where $d_{2}, D_{1}$, and $D_{2}$ are functions of the number of observations in the sample and are obtained from the table published in Beyer (1976).

When $\sigma$ is unknown,

$$
\begin{aligned}
\text { central line } & =\bar{R} \\
\mathrm{UCL} & =\left(D_{2} / d_{2}\right) \bar{R} \\
\mathrm{LCL} & =\left(D_{1} / d_{2}\right) \bar{R}
\end{aligned}
$$

where $\bar{R}=\sum_{i} R_{i} / k$ is the range of the $k$ sample ranges $R_{i}$.
Control limits for the $\overline{\mathrm{X}}$ chart are given by

$$
\begin{aligned}
\text { central line } & =\bar{x} \\
\qquad \mathrm{UCL} & =\bar{x}+(3 / \sqrt{n}) \sigma \\
\mathrm{LCL} & =\bar{x}-(3 / \sqrt{n}) \sigma
\end{aligned}
$$

if $\sigma$ is known. If $\sigma$ is unknown,

$$
\begin{aligned}
\text { central line } & =\bar{x} \\
\text { UCL } & =\bar{x}+A_{2} \bar{R} \\
\mathrm{LCL} & =\bar{x}-A_{2} \bar{R}
\end{aligned}
$$

where $\bar{R}$ is the average range as defined above and $A_{2}$ is a function (op. cit.) of the number of observations in the sample.

Isobel Loutit (1909-2009) is known for her work during World War II to improve the accuracy of targeting for anti-aircraft guns and as a contributor to the field of quality control. Loutit was born in Selkirk Manitoba, Canada. She graduated from the University of Manitoba in 1929 with a degree in mathematics and was one of the first women to work as a professional statistician in Canada. After graduation, she obtained a job teaching French. However, because of her training, she served as a substitute math teacher when needed.

When World War II started, Loutit took a job as a quality control statistician at Northern Electric. Shortly after that, the Canadian government advertised for women with technical training to fill jobs that had been vacated by men who had gone to war. She took a position testing equipment for the military. This job eventually returned her to Northern Electric, this time as a government employee verifying the accuracy of Northern Electric's Vickers anti-aircraft gun predictor, which was used to aim artillery at incoming planes. Recognizing the quality of her work, the CEO of Northern Electric rehired Loutit as an engineer, the only position for which her pay would not be capped as a woman. She later became the first female manager at Northern Electric and the first woman to chair the Montreal Section of the American Society for Quality Control.

## References

Alejo, J., A. K. Bera, A. F. Galvao, G. Montes-Rojas, and Z. Xiao. 2016. Tests for normality based on the quantile-mean covariance. Stata Journal 16: 1039-1057.
Bayart, D. 2001. Walter Andrew Shewhart. In Statisticians of the Centuries, ed. C. C. Heyde and E. Seneta, 398-401. New York: Springer.
Beyer, W. H. 1976. Factors for computing control limits. In Vol. 2 of Handbook of Tables for Probability and Statistics, ed. W. H. Beyer, 451-465. Cleveland, OH: The Chemical Rubber Company.
Burr, I. W. 1976. Statistical Quality Control Methods. New York: Dekker.
Caulcutt, R. 2004. Control charts in practice. Significance 1: 81-84. https://doi.org/10.1111/j.1740-9713.2004.024.x.
Duncan, A. J. 1986. Quality Control and Industrial Statistics. 5th ed. Homewood, IL: Irwin.
Harris, R. L. 1999. Information Graphics: A Comprehensive Illustrated Reference. New York: Oxford University Press.
Ryan, T. P. 2011. Statistical Methods for Quality Improvement. 3rd ed. Hoboken, NJ: Wiley.
Shewhart, W. A. 1931. Economic Control of Quality of Manufactured Product. New York: Van Nostrand.

## Also see <br> [R] serrbar - Graph standard error bar chart

## Title

qreg - Quantile regression

Description
Options for qreg
Remarks and examples
Also see

## Quick start

Options for iqreg
Stored results

## Menu

Options for sqreg
Methods and formulas

## Syntax

Options for bsqreg
References

## Description

qreg fits quantile (including median) regression models, also known as least absolute value, minimum absolute deviation, or minimum L1-norm value. The quantile regression models fit by qreg express the quantiles of the conditional distribution as linear functions of the independent variables.
iqreg estimates interquantile range regressions, regressions of the difference in quantiles. The estimated variance-covariance matrix of the estimators (VCE) is obtained via bootstrapping.
sqreg estimates simultaneous-quantile regression. It produces the same coefficients as qreg for each quantile. Reported standard errors will be similar, but sqreg obtains an estimate of the VCE via bootstrapping, and the VCE includes between-quantile blocks. Thus, you can test and construct confidence intervals comparing coefficients describing different quantiles.
bsqreg is equivalent to sqreg with one quantile.

## Quick start

Quantile regression
Median regression of $y$ on $x 1$ and $x 2$
qreg y x1 x2
Add categorical covariate a using factor-variable syntax
qreg y x1 x2 i.a
Same as above, but with standard errors using a biweight kernel for the nonparametric density estimator
qreg y x1 x2 i.a, vce(, kernel(biweight))
Quantile regression of the 75 th percentile of $y$ on $x 1, x 2$, and a
qreg $y \mathrm{x} 1 \mathrm{x} 2$ i.a, quantile(.75)

## Interquantile range regression

Difference between the 90th and 10th quantiles of y on x 1 , x 2 , and a with bootstrap standard errors iqreg $y$ x1 x2 i.a, quantiles(.1 .9)

Simultaneous-quantile regression
Simultaneous estimation of quantile regressions for the 10th and 90th quantiles of y with bootstrap standard errors
sqreg y x1 x2 i.a, quantiles(.1 .9)
Same as above, but for the 25th, 50th, and 75th quantiles of $y$ sqreg y x1 x2 i.a, quantiles(. 25 . 5 .75)

Same as above, but increase the number of bootstrap replications to 500
sqreg y x1 x2 i.a, quantiles(. 25 . 5 . 75) reps (500)

Bootstrapped quantile regression
Single quantile regression for the 25th quantile with bootstrap standard errors
bsqreg y x1 x2 i.a, quantile(.25)

## Menu

## qreg

Statistics $>$ Nonparametric analysis $>$ Quantile regression

## iqreg

Statistics $>$ Nonparametric analysis $>$ Interquantile regression

## sqreg

Statistics $>$ Nonparametric analysis $>$ Simultaneous-quantile regression

## bsqreg

Statistics $>$ Nonparametric analysis $>$ Bootstrapped quantile regression

## Syntax

Quantile regression

$$
\text { qreg depvar }[\text { indepvars }][\text { if }][\text { in }][\text { weight }][\text {, qreg_options }]
$$

Interquantile range regression

```
iqreg depvar [indepvars] [if] [in] [, iqreg_options]
```

Simultaneous-quantile regression

```
sqreg depvar [indepvars] [if] [in] [, sqreg_options]
```

Bootstrapped quantile regression
bsqreg depvar [indepvars] [if] [in] [, bsqreg_options]
qreg_options Description

Model
quantile(\#) estimate \# quantile; default is quantile(.5)
SE/Robust
$\operatorname{vce}([$ vcetype $],[$ vceopts $]$ ) technique used to estimate standard errors
Reporting
level(\#)
display_options

Optimization
optimization_options
wlsiter(\#)
set confidence level; default is level (95)
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
control the optimization process; seldom used attempt \# weighted least-squares iterations before doing linear programming iterations
vcetype
iid
robust

Description
compute the VCE assuming the residuals are i.i.d. compute the robust VCE

| denmethod | Description |
| :--- | :--- |
| $\underline{\text { fitted }}$ | use the empirical quantile function using fitted values; the default <br> use the empirical residual quantile function |
| $\underline{\text { kernel }[(\text { kernel })]}$ | use a nonparametric kernel density estimator; default is <br> epanechnikov |

bwidth
hsheather
bofinger
chamberlain

Description
Hall-Sheather's bandwidth; the default
Bofinger's bandwidth
Chamberlain's bandwidth

Description
Epanechnikov kernel function; the default
alternative Epanechnikov kernel function
biweight kernel function
cosine trace kernel function
Gaussian kernel function
Parzen kernel function
rectangle kernel function
triangle kernel function
iqreg_options
Model
quantiles(\# \#)
reps(\#)
Reporting
level(\#)
nodots
display_options

Description
interquantile range; default is quantiles(.25 .75)
perform \# bootstrap replications; default is reps (20)
set confidence level; default is level(95)
suppress display of the replication dots
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
sqreg_options
Description
Model
quantiles (\# [\# [\#...]]) estimate \# quantiles; default is quantiles(.5)
reps(\#)
perform \# bootstrap replications; default is reps (20)
Reporting
level(\#)
nodots
display_options
set confidence level; default is level (95)
suppress display of the replication dots
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

## Model

quantile(\#)
reps(\#)
Reporting
level(\#)
display_options
estimate \# quantile; default is quantile(.5) perform \# bootstrap replications; default is reps (20)
set confidence level; default is level (95)
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

[^48]
## Options for qreg

Model
quantile(\#) specifies the quantile to be estimated and should be a number between 0 and 1 , exclusive. Numbers larger than 1 are interpreted as percentages. The default value of 0.5 corresponds to the median.

SE/Robust
$\operatorname{vce}([$ vcetype $],[$ vceopts $])$ specifies the type of VCE to compute and the density estimation method to use in computing the VCE.
vcetype specifies the type of VCE to compute. Available types are iid and robust.
vce (iid), the default, computes the VCE under the assumption that the residuals are independent and identically distributed (i.i.d.).
vce (robust) computes the robust VCE under the assumption that the residual density is continuous and bounded away from 0 and infinity at the specified quantile(); see Koenker (2005, sec. 4.2).
vceopts consists of available denmethod and bwidth options.
denmethod specifies the method to use for the nonparametric density estimator. Available methods are fitted, residual, or kernel [ (kernel)], where the optional kernel must be one of the kernel choices listed below.
fitted and residual specify that the nonparametric density estimator use some of the structure imposed by quantile regression. The default fitted uses a function of the fitted values and residual uses a function of the residuals. vce(robust, residual) is not allowed.
kernel() specifies that the nonparametric density estimator use a kernel method. The available kernel functions are epanechnikov, epan2, biweight, cosine, gaussian, parzen, rectangle, and triangle. The default is epanechnikov. See [R] kdensity for the kernel function forms.
bwidth specifies the bandwidth method to use by the nonparametric density estimator. Available methods are hsheather for the Hall-Sheather bandwidth, bofinger for the Bofinger bandwidth, and chamberlain for the Chamberlain bandwidth.

See Koenker (2005, sec. 3.4 and 4.10) for a description of the sparsity estimation techniques and the Hall-Sheather and Bofinger bandwidth formulas. See Chamberlain (1994, eq. 2.2) for the Chamberlain bandwidth.

```
Reporting
level (\#); see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fmt), and nolstretch; see [R] Estimation options.
```


## Optimization

optimization_options: iterate (\#), [no]log, trace. iterate() specifies the maximum number of iterations; log/nolog specifies whether to show the iteration $\log$ (see set iterlog in $[R]$ set iter); and trace specifies that the iteration log should include the current parameter vector. These options are seldom used.
wlsiter(\#) specifies the number of weighted least-squares iterations that will be attempted before the linear programming iterations are started. The default is wlsiter(1). If there are convergence problems, increasing this number should help.

## Options for iqreg

$\qquad$ Model
quantiles(\# \#) specifies the quantiles to be compared. The first number must be less than the second, and both should be between 0 and 1, exclusive. Numbers larger than 1 are interpreted as percentages. Not specifying this option is equivalent to specifying quantiles(.25 .75), meaning the interquantile range.
reps(\#) specifies the number of bootstrap replications to be used to obtain an estimate of the variance-covariance matrix of the estimators (standard errors). reps (20) is the default and is arguably too small. reps (100) would perform 100 bootstrap replications. reps (1000) would perform 1,000 replications.
level (\#); see [R] Estimation options.
nodots suppresses display of the replication dots.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fmt), and nolstretch; see [R] Estimation options.

## Options for sqreg

$\qquad$ Model
quantiles (\# [\# [\#...]]) specifies the quantiles to be estimated and should contain numbers between 0 and 1, exclusive. Numbers larger than 1 are interpreted as percentages. The default value of 0.5 corresponds to the median.
reps(\#) specifies the number of bootstrap replications to be used to obtain an estimate of the variance-covariance matrix of the estimators (standard errors). reps (20) is the default and is arguably too small. reps (100) would perform 100 bootstrap replications. reps (1000) would perform 1,000 replications.

Reporting
level(\#); see [R] Estimation options.
nodots suppresses display of the replication dots.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap (\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.

## Options for bsqreg

$\qquad$ Model
quantile(\#) specifies the quantile to be estimated and should be a number between 0 and 1 , exclusive. Numbers larger than 1 are interpreted as percentages. The default value of 0.5 corresponds to the median.
reps(\#) specifies the number of bootstrap replications to be used to obtain an estimate of the variance-covariance matrix of the estimators (standard errors). reps(20) is the default and is arguably too small. reps (100) would perform 100 bootstrap replications. reps (1000) would perform 1,000 replications.
$\qquad$ Reporting
level (\#); see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap (\#), fvwrapon (style), cformat ( $\%$ fint), pformat ( $\% \mathrm{fmt}$ ), sformat ( $\% f m t$ ), and nolstretch; see [R] Estimation options.

## Remarks and examples

Remarks are presented under the following headings:

[^49]
## Median regression

qreg fits quantile regression models. The default form is median regression, where the objective is to estimate the median of the dependent variable, conditional on the values of the independent variables. This method is similar to ordinary regression, where the objective is to estimate the conditional mean of the dependent variable. Simply put, median regression finds a line through the data that minimizes the sum of the absolute residuals rather than the sum of the squares of the residuals, as in ordinary regression. Equivalently, median regression expresses the median of the conditional distribution of the dependent variable as a linear function of the conditioning (independent) variables. Cameron and Trivedi (2022, chap. 15) provide a nice introduction to quantile regression using Stata.

## > Example 1: Estimating the conditional median

Consider a two-group experimental design with 5 observations per group:

```
. use https://www.stata-press.com/data/r18/twogrp
. list
```

| $x$ | $y$ |
| :---: | ---: |
| 0 | 0 |
| 0 | 1 |
| 0 | 3 |
| 0 | 4 |
| 0 | 95 |
| 1 | 14 |
| 1 | 19 |
| 1 | 20 |
| 1 | 22 |
| 1 | 23 |

- qreg $\mathrm{y} x$

Iteration 1: WLS sum of weighted deviations $=60.941342$
Iteration 1: Sum of abs. weighted deviations $=\quad 55.5$

Iteration 2: Sum of abs. weighted deviations $=55$
$\begin{array}{lll}\text { Median regression } & & \text { Number of obs }= \\ \text { Raw sum of deviations } & 78.5 \text { (about 14) } & \end{array}$
Min sum of deviations 55 Pseudo R2 $=0.2994$

| y | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :--- | :---: | :---: | :---: | :---: |
| x | 17 | 18.23213 | 0.93 | 0.378 | -25.04338 | 59.04338 |
| _cons | 3 | 12.89207 | 0.23 | 0.822 | -26.72916 | 32.72916 |

We have estimated the equation

$$
\mathrm{y}_{\text {median }}=3+17 \mathrm{x}
$$

We look back at our data. x takes on the values 0 and 1 , so the median for the $\mathrm{x}=0$ group is 3 , whereas for $\mathrm{x}=1$ it is $3+17=20$. The output reports that the raw sum of absolute deviations about 14 is 78.5 ; that is, the sum of $|y-14|$ is 78.5 . Fourteen is the unconditional median of $y$, although in these data, any value between 14 and 19 could also be considered an unconditional median (we have an even number of observations, so the median is bracketed by those two values). In any case, the raw sum of deviations of y about the median would be the same no matter what number we choose between 14 and 19. (With a "median" of 14 , the raw sum of deviations is 78.5 . Now think
of choosing a slightly larger number for the median and recalculating the sum. Half the observations will have larger negative residuals, but the other half will have smaller positive residuals, resulting in no net change.)

We turn now to the actual estimated equation. The sum of the absolute deviations about the solution $\mathrm{y}_{\text {median }}=3+17 \mathrm{x}$ is 55. The pseudo- $R^{2}$ is calculated as $1-55 / 78.5 \approx 0.2994$. This result is based on the idea that the median regression is the maximum likelihood estimate for the double-exponential distribution.

## - Technical note

qreg is an alternative to regular regression or robust regression-see $[R]$ regress and $[R]$ rreg. Let's compare the results:


Unlike qreg, regress fits ordinary linear regression and is concerned with predicting the mean rather than the median, so both results are, in a technical sense, correct. Putting aside those technicalities, however, we tend to use either regression to describe the central tendency of the data, of which the mean is one measure and the median another. Thus, we can ask, "which method better describes the central tendency of these data?"

Means-and therefore ordinary linear regression-are sensitive to outliers, and our data were purposely designed to contain two such outliers: 95 for $\mathrm{x}=0$ and 14 for $\mathrm{x}=1$. These two outliers dominated the ordinary regression and produced results that do not reflect the central tendency well-you are invited to enter the data and graph y against x .

Robust regression attempts to correct the outlier-sensitivity deficiency in ordinary regression:

| Huber iteration 1: Maximum difference in weights $=.7311828$ |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Huber iteration 2: Maximum difference in weights $=.17695779$ |  |  |  |  |  |  |
| Huber iteration 3: Maximum difference in weights $=.03149585$ |  |  |  |  |  |  |
| Biweight iteration 4: Maximum difference in weights $=.1979335$ |  |  |  |  |  |  |
| Biweight iteration 5: Maximum difference in weights $=.23332905$ |  |  |  |  |  |  |
| Biweight iteration 6: Maximum difference in weights $=.09960067$ |  |  |  |  |  |  |
| Biweight iteration 7: Maximum difference in weights $=.02691458$ |  |  |  |  |  |  |
| Biweight iteration 8: Maximum difference in weights $=.0009113$ |  |  |  |  |  |  |
| Robust regression |  |  |  | Number of obs $=10$ |  |  |
|  |  |  |  | F( | 8) | 80.63 |
|  |  |  |  | Prob > |  | 0.0000 |
| y | Coefficient | Std. err. | t | $P>\|t\|$ | [95\% conf | interval] |
| x | 18.16597 | 2.023114 | 8.98 | 0.000 | 13.50066 | 22.83128 |
| _cons | 2.000003 | 1.430558 | 1.40 | 0.200 | -1.298869 | 5.298875 |

Here rreg discarded the first outlier completely. (We know this because we included the genwt () option on rreg and, after fitting the robust regression, examined the weights.) For the other "outlier", rreg produced a weight of 0.47 .

In any case, the answers produced by qreg and rreg to describe the central tendency are similar, but the standard errors are different. In general, robust regression will have smaller standard errors because it is not as sensitive to the exact placement of observations near the median. You are welcome to try removing the first outlier in the qreg estimation to observe an improvement in the standard errors by typing

```
. qreg y x if _n!=5
```

Also, some authors (Rousseeuw and Leroy 1987, 11) have noted that quantile regression, unlike the unconditional median, may be sensitive to even one outlier if its leverage is high enough. Rousseeuw and Leroy (1987) discuss estimators that are more robust to perturbations to the data than either mean regression or quantile regression.

In the end, quantile regression may be more useful for the interpretation of the parameters that it estimates than for its robustness to perturbations to the data.

## > Example 2: Median regression

Let's now consider a less artificial example using the automobile data described in [U] 1.2.2 Example datasets. Using median regression, we will regress each car's price on its weight and length and whether it is of foreign manufacture:

| Iteration 1: WLS sum of weighted deviations $=56397.829$ |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Iteration 1: Sum of abs. weighted deviations $=55950.5$ |  |  |  |  |  |  |
| Iteration 2: Sum of abs. weighted deviations $=55264.718$ |  |  |  |  |  |  |
| Iteration 3: Sum of abs. weighted deviations $=54762.283$ |  |  |  |  |  |  |
| Iteration 4: Sum of abs. weighted deviations $=54734.152$ |  |  |  |  |  |  |
| Iteration 5: Sum of abs. weighted deviations $=54552.638$ note: alternate solutions exist. |  |  |  |  |  |  |
| Iteration 6: Sum of abs. weighted deviations $=54465.511$ |  |  |  |  |  |  |
| Iteration 7: Sum of abs. weighted deviations $=54443.699$ |  |  |  |  |  |  |
| Iteration 8: Sum of abs. weighted deviations $=54411.294$ |  |  |  |  |  |  |
| Median regression |  |  |  | Number of obs $=\quad 74$ |  |  |
| Raw sum of deviations 71102.5 (about 4934) |  |  |  |  |  |  |
| Min sum of | eviations 54 | 1.29 |  |  | do R2 | 0.2347 |
| price | Coefficient | Std. err. | t | $P>\|t\|$ | [95\% con | interval] |
| weight | 3.933588 | 1.328718 | 2.96 | 0.004 | 1.283543 | 6.583632 |
| length | -41.25191 | 45.46469 | -0.91 | 0.367 | -131.9284 | 49.42456 |
| foreign | 3377.771 | 885.4198 | 3.81 | 0.000 | 1611.857 | 5143.685 |
| _cons | 344.6489 | 5182.394 | 0.07 | 0.947 | -9991.31 | 10680.61 |

The estimated equation is

$$
\text { price }_{\text {median }}=3.93 \text { weight }-41.25 \text { length }+3377.8 \text { foreign }+344.65
$$

The output may be interpreted in the same way as linear regression output; see [R] regress. The variables weight and foreign are significant, but length is not significant. The median price of the cars in these data is $\$ 4,934$. This value is a median (one of the two center observations), not the median, which would typically be defined as the midpoint of the two center observations.

## Quantile regression

Quantile regression is similar to median regression in that it estimates an equation expressing a quantile of the conditional distribution, albeit one that generally differs from the 0.5 quantile that is the median. For example, specifying quantile(.25) estimates the parameters that describe the 25th percentile (first quartile) of the conditional distribution.

Quantile regression allows for effects of the independent variables to differ over the quantiles. For example, Chamberlain (1994) finds that union membership has a larger effect on the lower quantiles than on the higher quantiles of the conditional distribution of US wages. That the effects of the independent variables may vary over quantiles of the conditional distribution is an important advantage of quantile regression over mean regression.

## > Example 3: Estimating quantiles other than the median

Returning to real data, the equation for the 25 th percentile of price conditional on weight, length, and foreign in our automobile data is

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. qreg price weight length foreign, quantile(.25)
Iteration 1: WLS sum of weighted deviations = 49469.235
Iteration 1: Sum of abs. weighted deviations = 49728.883
Iteration 2: Sum of abs. weighted deviations = 45669.89
Iteration 3: Sum of abs. weighted deviations = 43416.646
Iteration 4: Sum of abs. weighted deviations = 41947.221
Iteration 5: Sum of abs. weighted deviations = 41093.025
Iteration 6: Sum of abs. weighted deviations = 37623.424
Iteration 7: Sum of abs. weighted deviations = 35721.453
Iteration 8: Sum of abs. weighted deviations = 35226.308
Iteration 9: Sum of abs. weighted deviations = 34823.319
Iteration 10: Sum of abs. weighted deviations = 34801.777
```

.25 Quantile regression $\quad$ Number of obs $=\quad 74$
Raw sum of deviations 41912.75 (about 4187)
Min sum of deviations 34801.78 Pseudo R2 $=0.1697$

| price | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | :---: | :---: | ---: | ---: |
| weight | 1.831789 | .6328903 | 2.89 | 0.005 | .5695289 | 3.094049 |
| length | 2.84556 | 21.65558 | 0.13 | 0.896 | -40.34514 | 46.03626 |
| foreign | 2209.925 | 421.7401 | 5.24 | 0.000 | 1368.791 | 3051.059 |
| _cons | -1879.775 | 2468.46 | -0.76 | 0.449 | -6802.963 | 3043.413 |

Compared with our previous median regression, the coefficient on length now has a positive sign, and the coefficients on foreign and weight are reduced. The actual lower quartile is $\$ 4,187$, substantially less than the median $\$ 4,934$.

We can also estimate the upper quartile as a function of the same three variables:

| Iteration 1: WLS sum of weighted deviations $=55465.741$ |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Iteration 1: Sum of abs. weighted deviations $=55652.957$ |  |  |  |  |  |  |
| Iteration 2: | um of abs. weighted deviations $=52994.785$ |  |  |  |  |  |
| Iteration 3: | Sum of abs. weighted deviations $=50189.446$ |  |  |  |  |  |
| Iteration 4: | Sum of abs. weighted deviations $=49898.245$ |  |  |  |  |  |
| Iteration 5: | Sum of abs. weighted deviations $=49398.106$ |  |  |  |  |  |
| Iteration 6: | Sum of abs. weighted deviations $=49241.835$ |  |  |  |  |  |
| Iteration 7: | Sum of abs. weighted deviations $=49197.967$ |  |  |  |  |  |
| . 75 Quantile regression <br> Raw sum of deviations 79860.75 (about 6342) <br> Min sum of deviations 49197.97 |  |  |  | Number of obs $=\quad 74$ |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  | do R2 | 0.3840 |
| price | Coefficient | Std. err. | t | $P>\|t\|$ | [95\% con | interval] |
| weight | 9.22291 | 1.785767 | 5.16 | 0.000 | 5.66131 | 12.78451 |
| length | -220.7833 | 61.10352 | -3.61 | 0.001 | -342.6504 | -98.91616 |
| foreign | 3595.133 | 1189.984 | 3.02 | 0.004 | 1221.785 | 5968.482 |
| _cons | 20242.9 | 6965.02 | 2.91 | 0.005 | 6351.61 | 34134.2 |

This result tells a different story: weight is much more important, and length is now significant - with a negative coefficient! The prices of high-priced cars seem to be determined by factors different from those affecting the prices of low-priced cars.

## - Technical note

One explanation for having substantially different regression functions for different quantiles is that the data are heteroskedastic, as we will demonstrate below. The following statements create a sharply heteroskedastic set of data:

```
. drop _all
. set obs 10000
Number of observations (_N) was 0, now 10,000.
set seed 50550
. generate x = .1 + .9 * runiform()
. generate y = x * runiform()^2
```

Let's now fit the regressions for the 5th and 95th quantiles:

```
. qreg y x, quantile(.05)
Iteration 1: WLS sum of weighted deviations = 555.44181
Iteration 1: Sum of abs. weighted deviations = 555.25622
Iteration 2: Sum of abs. weighted deviations = 115.02628
Iteration 3: Sum of abs. weighted deviations = 89.617883
Iteration 4: Sum of abs. weighted deviations = 89.61679
```



| y | Coefficient | Std. err. | t | P>\|t| | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| x | .0028667 | .0004395 | 6.52 | 0.000 | .0020052 | .0037283 |
| _cons | -.0001135 | .0002661 | -0.43 | 0.670 | -.0006352 | .0004081 |

```
. qreg y x, quantile(.95)
Iteration 1: WLS sum of weighted deviations = 624.91903
Iteration 1: Sum of abs. weighted deviations = 621.88928
Iteration 2: Sum of abs. weighted deviations = 182.03243
Iteration 3: Sum of abs. weighted deviations = 170.42588
Iteration 4: Sum of abs. weighted deviations = 169.05915
Iteration 5: Sum of abs. weighted deviations = 169.05911
```

| .95 Quantile regression | Number of obs $=$ | 10,000 |
| :--- | :--- | :--- | :--- |
| Raw sum of deviations 275.9779 (about .60579139) |  |  |
| Min sum of deviations 169.0591 | Pseudo R2 | $=10.3874$ |


| y | Coefficient | Std. err. | t | $\mathrm{P}>\mid \mathrm{tl}$ | $[95 \%$ conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| x | .9010814 | .008758 | 102.89 | 0.000 | .883914 | .9182488 |
| _cons | -.0004053 | .0053028 | -0.08 | 0.939 | -.0107999 | .0099893 |

The coefficient on x , in particular, differs markedly between the two estimates. For the mathematically inclined, it is not too difficult to show that the theoretical lines are $\mathrm{y}=0.0025 \mathrm{x}$ for the 5 th percentile and $\mathrm{y}=0.9025 \mathrm{x}$ for the 95 th, numbers in close agreement with our numerical results.

The estimator for the standard errors computed by qreg assumes that the sample is independent and identically distributed (i.i.d.); see Estimated standard errors and Methods and formulas for details. Because the data are conditionally heteroskedastic, we should have used bsqreg to consistently estimate the standard errors using a bootstrap method.

## Estimated standard errors

The variance-covariance matrix of the estimator (VCE) depends on the reciprocal of the density of the dependent variable evaluated at the quantile of interest. This function, known as the "sparsity function", is hard to estimate.

The default method, which uses the fitted values for the predicted quantiles, generally performs well, but other methods may be preferred in larger samples. The vce() suboptions denmethod and bwidth provide other estimators of the sparsity function, the details of which are described in Methods and formulas.

For models with heteroskedastic errors, option vce(robust) computes a Huber (1967) form of sandwich estimate (Koenker 2005). Alternatively, Gould (1992, 1997b) introduced generalized versions of qreg that obtain estimates of the standard errors by using bootstrap resampling (see Efron and Tibshirani [1993] or Wu [1986] for an introduction to bootstrap standard errors). The iqreg, sqreg, and bsqreg commands provide a bootstrapped estimate of the entire variance-covariance matrix of the estimators.

## > Example 4: Obtaining robust standard errors

Example 2 of qreg on real data above was a median regression of price on weight, length, and foreign using auto.dta. Suppose, after investigation, we are convinced that car price observations are not independent. We decide that standard errors robust to non-i.i.d. errors would be appropriate and use the option vce (robust).

| Iteration 1: WLS sum of weighted deviations $=56397.829$ |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Iteration 1: Sum of abs. weighted deviations $=55950.5$ |  |  |  |  |  |  |
| Iteration 2: Sum of abs. weighted deviations $=55264.718$ |  |  |  |  |  |  |
| Iteration 3: Sum of abs. weighted deviations $=54762.283$ |  |  |  |  |  |  |
| Iteration 4: Sum of abs. weighted deviations $=54734.152$ |  |  |  |  |  |  |
| Iteration 5: Sum of abs. weighted deviations $=54552.638$ note: alternate solutions exist. |  |  |  |  |  |  |
| Iteration 6: Sum of abs. weighted deviations $=54465.511$ |  |  |  |  |  |  |
| Iteration 7: Sum of abs. weighted deviations $=54443.699$ |  |  |  |  |  |  |
| Iteration 8: Sum of abs. weighted deviations $=54411.294$ |  |  |  |  |  |  |
| Median regression |  |  |  | Number of obs $=\quad 74$ |  |  |
| Raw sum of deviations 71102.5 (about 4934) |  |  |  |  |  |  |
| Min sum of | eviations 544 | 11.29 |  |  | do R2 | 0.2347 |
| price | Coefficient | Robust std. err. | t | $p>\|t\|$ | [95\% con | interval] |
| weight | 3.933588 | 1.694477 | 2.32 | 0.023 | . 55406 | 7.313116 |
| length | -41.25191 | 51.73571 | -0.80 | 0.428 | -144.4355 | 61.93171 |
| foreign | 3377.771 | 728.5115 | 4.64 | 0.000 | 1924.801 | 4830.741 |
| _cons | 344.6489 | 5096.528 | 0.07 | 0.946 | -9820.055 | 10509.35 |

We see that the robust standard error for weight increases, making it less significant in modifying the median automobile price. The standard error for length also increases, but the standard error for the foreign indicator decreases.

For comparison, we repeat the estimation using bootstrap standard errors:

| . use https://www.stata-press.com/dat (1978 automobile data) <br> . set seed 1001 <br> . bsqreg price weight length foreign (fitting base model) |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Bootstrap replications (20): .........10......... 20 done |  |  |  |  |  |  |
| Median regression, bootstrap(20) SEs <br> Raw sum of deviations 71102.5 (about 4934) Min sum of deviations 54411.29 |  |  |  | Number of obs $=$ $\qquad$ |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  | do R2 | 0.2347 |
| price | Coefficient | Std. err. | t | $P>\|t\|$ | [95\% con | interval] |
| weight | 3.933588 | 2.941839 | 1.34 | 0.186 | -1.933726 | 9.800901 |
| length | -41.25191 | 73.47105 | -0.56 | 0.576 | -187.7853 | 105.2815 |
| foreign | 3377.771 | 1352.518 | 2.50 | 0.015 | 680.2582 | 6075.284 |
| _cons | 344.6489 | 5927.045 | 0.06 | 0.954 | -11476.47 | 12165.77 |

The coefficient estimates are the same-indeed, they are obtained using the same technique. Only the standard errors differ. Therefore, the $t$ statistics, significance levels, and confidence intervals also differ.

Because bsqreg (as well as sqreg and iqreg) obtains standard errors by randomly resampling the data, the standard errors it produces will not be the same from run to run unless we first set the random-number seed to the same number; see $[R]$ set seed.

By default, bsqreg, sqreg, and iqreg use 20 replications. We can control the number of replications by specifying the reps () option:

| . bsqreg price weight length i.foreign, reps(1000) (fitting base model) |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Bootstrap replications ( 1,000 ) |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
| > . . . . . . 120......... 130.... . . . . 140. . . . . . . . 150. . . . . . . . 160. . . . . . . . 170 . . |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
| > . 500.........510.........520........ . 530. . . . . . . . 540 . . . . . . . . . 550. . . . . . . . . 560 |  |  |  |  |  |  |
| > ........570.........580.........590......... 600..........610..... . . . . $620 . .$. . |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
| > 90.........700.........710.........720..........730..........740... . . . . . 750. |  |  |  |  |  |  |
| > . . . . .760.... . . . .770. . . . . . . .780. . . . . . . .790. . . . . . . . 800 . . . . . . . . 810. . . . . . |  |  |  |  |  |  |
| > .820.........830.........840.........850..........860..........870......... . 880. |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
| Median regression, bootstrap(1000) SEs <br> Raw sum of deviations 71102.5 (about 4934) Min sum of deviations 54411.29 |  |  |  | Number of obs $=\quad 74$ |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  | do R2 | 0.2347 |
| price | Coefficient | Std. err. | t | $P>\|t\|$ | [95\% conf | interval] |
| weight | 3.933588 | 2.58771 | 1.52 | 0.133 | -1.227437 | 9.094613 |
| length | -41.25191 | 68.02626 | -0.61 | 0.546 | -176.926 | 94.42219 |
| foreign |  |  |  |  |  |  |
| Foreign | 3377.771 | 1070.777 | 3.15 | 0.002 | 1242.174 | 5513.368 |
| _cons | 344.6489 | 5862.991 | 0.06 | 0.953 | -11348.72 | 12038.02 |

A comparison of the standard errors is informative.

| Variable | qreg | qreg <br> vce(robust) | bsqreg <br> reps(20) | bsqreg <br> reps (1000) |
| :--- | :---: | :---: | :---: | :---: |
| weight | 1.329 | 1.694 | 2.942 | 2.588 |
| length | 45.46 | 51.74 | 73.47 | 68.03 |
| 1.foreign | 885.4 | 728.5 | 1353. | 1071. |
| _cons | 5182. | 5097. | 5927. | 5863. |

The results shown above are typical for models with heteroskedastic errors. (Our dependent variable is price; if our model had been in terms of $\ln$ (price), the standard errors estimated by qreg and bsqreg would have been nearly identical.) Also, even for heteroskedastic errors, 20 replications is generally sufficient for hypothesis tests against 0 .

## Interquantile and simultaneous-quantile regression

Consider a quantile regression model where the $q$ th quantile is given by

$$
Q_{q}(y)=a_{q}+b_{q, 1} x_{1}+b_{q, 2} x_{2}
$$

For instance, the 75th and 25th quantiles are given by

$$
\begin{aligned}
Q_{0.75}(y) & =a_{0.75}+b_{0.75,1} x_{1}+b_{0.75,2} x_{2} \\
Q_{0.25}(y) & =a_{0.25}+b_{0.25,1} x_{1}+b_{0.25,2} x_{2}
\end{aligned}
$$

The difference in the quantiles is then

$$
Q_{0.75}(y)-Q_{0.25}(y)=\left(a_{0.75}-a_{0.25}\right)+\left(b_{0.75,1}-b_{0.25,1}\right) x_{1}+\left(b_{0.75,2}-b_{0.25,2}\right) x_{2}
$$

qreg fits models such as $Q_{0.75}(y)$ and $Q_{0.25}(y)$. iqreg fits interquantile models, such as $Q_{0.75}(y)-$ $Q_{0.25}(y)$. The relationships of the coefficients estimated by qreg and iqreg are exactly as shown: iqreg reports coefficients that are the difference in coefficients of two qreg models, and, of course, iqreg reports the appropriate standard errors, which it obtains by bootstrapping.
sqreg is like qreg in that it estimates the equations for the quantiles

$$
\begin{aligned}
Q_{0.75}(y) & =a_{0.75}+b_{0.75,1} x_{1}+b_{0.75,2} x_{2} \\
Q_{0.25}(y) & =a_{0.25}+b_{0.25,1} x_{1}+b_{0.25,2} x_{2}
\end{aligned}
$$

The coefficients it obtains are the same that would be obtained by estimating each equation separately using qreg. sqreg differs from qreg in that it estimates the equations simultaneously and obtains an estimate of the entire variance-covariance matrix of the estimators by bootstrapping. Thus, you can perform hypothesis tests concerning coefficients both within and across equations.

For example, to fit the above model, you could type

```
. qreg y x1 x2, quantile(.25)
. qreg y x1 x2, quantile(.75)
```

By doing this, you would obtain estimates of the parameters, but you could not test whether $b_{0.25,1}=b_{0.75,1}$ or, equivalently, $b_{0.75,1}-b_{0.25,1}=0$. If your interest really is in the difference of coefficients, you could type

```
. iqreg y x1 x2, quantiles(.25 .75)
```

The "coefficients" reported would be the difference in quantile coefficients. You could also estimate both quantiles simultaneously and then test the equality of the coefficients:

```
. sqreg y x1 x2, quantiles(.25 .75)
. test [q25]x1 = [q75]x1
```

Whether you use iqreg or sqreg makes no difference for this test. sqreg, however, because it estimates the quantiles simultaneously, allows you to test other hypotheses. iqreg, by focusing on quantile differences, presents results in a way that is easier to read.

Finally, sqreg can estimate quantiles singly,

```
. sqreg y x1 x2, quantiles(.5)
```

and can thereby be used as a substitute for the slower bsqreg. (Gould [1997b] presents timings demonstrating that sqreg is faster than bsqreg.) sqreg can also estimate more than two quantiles simultaneously:

```
. sqreg y x1 x2, quantiles(.25 .5 .75)
```


## > Example 5: Simultaneous quantile estimation

In demonstrating qreg, we performed quantile regressions using auto.dta. We discovered that the regression of price on weight, length, and foreign produced vastly different coefficients for the $0.25,0.5$, and 0.75 quantile regressions. Here are the coefficients that we obtained:

| Variable | 25th <br> percentile | 50th <br> percentile | 75th <br> percentile |
| :--- | :---: | :---: | :---: |
| weight | 1.83 | 3.93 | 9.22 |
| length | 2.85 | -41.25 | -220.8 |
| foreign | 2209.9 | 3377.8 | 3595.1 |
| _cons | -1879.8 | 344.6 | 20242.9 |

All we can say, having estimated these equations separately, is that price seems to depend differently on the weight, length, and foreign variables depending on the portion of the price distribution we examine. We cannot be more precise because the estimates have been made separately. With sqreg, however, we can estimate all the effects simultaneously:

| ```(1978 automobile data) . set seed 1001 . sqreg price weight length foreign, q(.25 .5 .75) reps(100) (fitting base model)``` |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  |  |  |  |
| Simultaneous quantile regression bootstrap(100) SEs |  |  |  | Number of obs $=$ . 25 Pseudo R2 = <br> . 50 Pseudo R2 = <br> . 75 Pseudo R2 = |  | $\begin{array}{r} 74 \\ 0.1697 \\ 0.2347 \\ 0.3840 \end{array}$ |
| price | Coefficient | Bootstrap std. err. | t | $p>\|t\|$ | [95\% con | interval] |
| q25 |  |  |  |  |  |  |
| weight | 1.831789 | 1.206151 | 1.52 | 0.133 | -. 573803 | 4.237381 |
| length | 2.84556 | 26.775 | 0.11 | 0.916 | -50.55549 | 56.24661 |
| foreign | 2209.925 | 946.9585 | 2.33 | 0.022 | 321.2762 | 4098.575 |
| _cons | -1879.775 | 3243.182 | -0.58 | 0.564 | -8348.097 | 4588.548 |
| q50 |  |  |  |  |  |  |
| weight | 3.933588 | 2.34028 | 1.68 | 0.097 | -. 7339531 | 8.601129 |
| length | -41.25191 | 59.58361 | -0.69 | 0.491 | -160.0877 | 77.58386 |
| foreign | 3377.771 | 1081.475 | 3.12 | 0.003 | 1220.836 | 5534.706 |
| _cons | 344.6489 | 5062.804 | 0.07 | 0.946 | -9752.795 | 10442.09 |
| q75 |  |  |  |  |  |  |
| weight | 9.22291 | 2.631084 | 3.51 | 0.001 | 3.975378 | 14.47044 |
| length | -220.7833 | 89.54754 | -2.47 | 0.016 | -399.3802 | -42.18635 |
| foreign | 3595.133 | 1208.769 | 2.97 | 0.004 | 1184.319 | 6005.947 |
| _cons | 20242.9 | 9682.24 | 2.09 | 0.040 | 932.2846 | 39553.52 |

The coefficient estimates above are the same as those previously estimated, although the standard error estimates are a little different. sqreg obtains estimates of variance by bootstrapping. The important thing here, however, is that the full covariance matrix of the estimators has been estimated and stored, and thus it is now possible to perform hypothesis tests. Are the effects of weight the same at the 25th and 75 th percentiles?

```
. test [q25]weight = [q75]weight
( 1) [q25]weight - [q75]weight = 0
    F( 1, 70) = 8.57
        Prob > F = 0.0046
```

It appears that they are not. We can obtain a confidence interval for the difference by using lincom:

| price | Coefficient | Std. err. | t | $P>\|t\|$ | [95\% conf | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| (1) | 7.391121 | 2.524626 | 2.93 | 0.005 | 2.355914 | 12.42633 |

Indeed, we could test whether the weight and length sets of coefficients are equal at the three quantiles estimated:

```
. quietly test [q25]weight = [q50]weight
. quietly test [q25]weight = [q75]weight, accumulate
. quietly test [q25]length = [q50]length, accumulate
. test [q25]length = [q75]length, accumulate
( 1) [q25]weight - [q50]weight = 0
( 2) [q25]weight - [q75]weight = 0
( 3) [q25]length - [q50]length = 0
(4) [q25]length - [q75]length = 0
    F(4, 70) = 2.25
        Prob > F = 0.0727
```

iqreg focuses on one quantile comparison but presents results that are more easily interpreted:

| .75-. 25 Interquantile regression bootstrap(100) SEs |  |  |  | $\begin{array}{rrr} \text { Number of obs }= & 74 \\ .75 \text { Pseudo R2 }= & 0.3840 \\ .25 \text { Pseudo R2 }= & 0.1697 \end{array}$ |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| price | Coefficient | Bootstrap std. err. | t | $P>\|t\|$ | [95\% con | interval] |
| weight | 7.391121 | 2.524626 | 2.93 | 0.005 | 2.355914 | 12.42633 |
| length | -223.6288 | 84.21504 | -2.66 | 0.010 | -391.5904 | -55.66724 |
| foreign | 1385.208 | 1321.832 | 1.05 | 0.298 | -1251.103 | 4021.519 |
| _cons | 22122.68 | 9329.178 | 2.37 | 0.020 | 3516.219 | 40729.14 |

Looking only at the 0.25 and 0.75 quantiles (the interquartile range), the iqreg command output is easily interpreted. Increases in weight correspond significantly to increases in price dispersion. Increases in length correspond to decreases in price dispersion. The foreign variable does not significantly change price dispersion.

Do not make too much of these results; the purpose of this example is simply to illustrate the sqreg and iqreg commands and to do so in a context that suggests why analyzing dispersion might be of interest.
lincom after sqreg produced the same $t$ statistic for the interquartile range of weight, as did the iqreg command above. In general, they will not agree exactly because of the randomness of bootstrapping, unless the random-number seed is set to the same value before estimation (as was done here).

Gould (1997a) presents simulation results showing that the coverage-the actual percentage of confidence intervals containing the true value-for iqreg is appropriate.

## What are the parameters?

In this section, we use a specific data-generating process (DGP) to illustrate the interpretation of the parameters estimated by qreg. If simulation experiments are not intuitive to you, skip this section.

In general, quantile regression parameterizes the quantiles of the distribution of $y$ conditional on the independent variables $\mathbf{x}$ as $\mathbf{x} \boldsymbol{\beta}$, where $\boldsymbol{\beta}$ is a vector of estimated parameters. In our example, we include a constant term and a single independent variable, and we express quantiles of the distribution of $y$ conditional on $x$ as $\beta_{0}+\beta_{1} x$.

We use simulated data to illustrate what we mean by a conditional distribution and how to interpret the parameters $\boldsymbol{\beta}$ estimated by qreg. We also note how we could change our example to illustrate a DGP for which the estimator in qreg would be misspecified.

We suppose that the distribution of $y$ conditional on $x$ has a Weibull form. If $y$ has a Weibull distribution, the distribution function is $F(y)=1-\exp \left\{-(y / \lambda)^{k}\right\}$, where the scale parameter $\lambda>0$ and the shape parameter $k>0$. We can make $y$ have a Weibull distribution function conditional on $x$ by making the scale parameter or the shape parameter functions of $x$. In our example, we specify a particular DGP by supposing that $\lambda=(1+\alpha x), \alpha=1.5, x=1+\sqrt{\nu}$, and that $\nu$ has a $\chi^{2}(1)$ distribution. For the moment, we leave the parameter $k$ as is so that we can discuss how this decision relates to model specification.

Plugging in for $\lambda$ yields the functional form for the distribution of $y$ conditional on $x$, which is known as the conditional distribution function and is denoted $F(y \mid x) . F(y \mid x)$ is the distribution for $y$ for each given value of $x$.

Some algebra yields that $F(y \mid x)=1-\exp \left[-\{y /(1+\alpha x)\}^{k}\right]$. Letting $\tau=F(y \mid x)$ implies that $0 \leq \tau \leq 1$, because probabilities must be between 0 and 1 .

To obtain the $\tau$ quantile of the distribution of $y$ conditional on $x$, we solve

$$
\tau=1-\exp \left[-\{y /(1+\alpha x)\}^{k}\right]
$$

for $y$ as a function of $\tau, x, \alpha$, and $k$. The solution is

$$
\begin{equation*}
y=(1+\alpha x)\{-\ln (1-\tau)\}^{(1 / k)} \tag{1}
\end{equation*}
$$

For any value of $\tau \in(0,1)$, expression (1) gives the $\tau$ quantile of the distribution of $y$ conditional on $x$. To use qreg, we must rewrite (1) as a function of $x, \beta_{0}$, and $\beta_{1}$. Some algebra yields that (1) can be rewritten as

$$
y=\beta_{0}+\beta_{1} * x
$$

where $\beta_{0}=\{-\ln (1-\tau)\}^{(1 / k)}$ and $\beta_{1}=\alpha\{-\ln (1-\tau)\}^{(1 / k)}$. We can express the conditional quantiles as linear combinations of $x$, which is a property of the estimator implemented in qreg.

If we parameterize $k$ as a nontrivial function of $x$, the conditional quantiles will not be linear in $x$. If the conditional quantiles cannot be represented as linear functions of $x$, we cannot estimate the true parameters of the DGP. This restriction illustrates the limits of the estimator implemented in qreg.

We set $k=2$ for our example.
Conditional quantile regression allows the coefficients to change with the specified quantile. For our DGP, the coefficients $\beta_{0}$ and $\beta_{1}$ increase as $\tau$ gets larger. Substituting in for $\alpha$ and $k$ yields that $\beta_{0}=\sqrt{-\ln (1-\tau)}$ and $\beta_{1}=1.5 \sqrt{-\ln (1-\tau)}$. Table 1 presents the true values for $\beta_{0}$ and $\beta_{1}$ implied by our DGP when $\tau \in\{0.25,0.5,0.8\}$.

Table 1: True values for $\beta_{0}$ and $\beta_{1}$

| $\tau$ | $\beta_{0}$ | $\beta_{1}$ |
| :--- | :--- | :--- |
| 0.25 | 0.53636 | 0.80454 |
| 0.5 | 0.8325546 | 1.248832 |
| 0.8 | 1.268636 | 1.902954 |

We can also use (1) to generate data from the specified distribution of $y$ conditional on $x$ by plugging in random uniform numbers for $\tau$. Each random uniform number substituted in for $\tau$ in (1) yields a draw from the conditional distribution of $y$ given $x$.

## > Example 6

In this example, we generate 100,000 observations from our specified DGP by substituting random uniform numbers for $\tau$ in (1), with $\alpha=1.5, k=2, x=1+\sqrt{\nu}$, and $\nu$ coming from a $\chi^{2}(1)$ distribution.

We begin by executing the code that implements this method; below, we discuss each line of the output produced.

```
. clear // drop existing variables
. set seed 1234571 // set random-number seed
. set obs 100000 // set number of observations
Number of observations (_N) was 0, now 100,000.
. generate double tau = runiform() // generate uniform variate
. generate double x = 1 + sqrt(rchi2(1)) // generate values for x
. generate double lambda = 1 + 1.5*x // lambda is 1 + alpha*x
. generate double k = 2
    // fix value of k
    // generate random values for y
    // given x
. generate double y = lambda*((-ln}(1-tau))^(1/k)
```

Although the comments at the end of each line briefly describe what each line is doing, we provide a more careful description. The first line drops any variables in memory. The second sets the seed of the random-number generator so that we will always get the same sequence of random uniform numbers. The third line sets the sample size to 100,000 observations, and the fourth line reports the change in sample size.

The fifth line substitutes random uniform numbers for $\tau$. This line is the key to the algorithm. This standard method, known as inverse-probability transforms, for computing random numbers is discussed by Cameron and Trivedi (2022, 220-221), among others.

Lines 6-8 generate $x, \lambda$, and $k$ per our specified DGP. Lines $9-11$ implement (1) using the previously generated $\lambda, x$, and $k$.

At the end, we have 100,000 observations on $y$ and $x$, with $y$ coming from the conditional distribution that we specified above.

## > Example 7

In the example below, we use qreg to estimate $\beta_{1}$ and $\beta_{0}$, the parameters from the conditional quantile function, for the 0.5 quantile from our simulated data.

```
. qreg y x, quantile(.5)
Iteration 1: WLS sum of weighted deviations = 68573.243
Iteration 1: Sum of abs. weighted deviations = 68571.918
Iteration 2: Sum of abs. weighted deviations = 68308.342
Iteration 3: Sum of abs. weighted deviations = 68241.17
Iteration 4: Sum of abs. weighted deviations = 68232.043
Iteration 5: Sum of abs. weighted deviations = 68230.304
Iteration 6: Sum of abs. weighted deviations = 68229.643
Iteration 7: Sum of abs. weighted deviations = 68229.532
Iteration 8: Sum of abs. weighted deviations = 68229.514
Iteration 9: Sum of abs. weighted deviations = 68229.508
Iteration 10: Sum of abs. weighted deviations = 68229.506
Iteration 11: Sum of abs. weighted deviations = 68229.505
```

Median regression $\quad$ Number of obs $=100,000$
Raw sum of deviations 73861.64 (about 2.9443724)
Min sum of deviations 68229.51 Pseudo R2 $=0.0763$

| y | Coefficient | Std. err. | t | $\mathrm{P}>\mid \mathrm{tI}$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| x | 1.266062 | .0117759 | 107.51 | 0.000 | 1.242981 | 1.289143 |
| _cons | .8083315 | .0222972 | 36.25 | 0.000 | .7646291 | .8520338 |

In the qreg output, the results for x correspond to the estimate of $\beta_{1}$, and the results for $\quad$ cons correspond to the estimate of $\beta_{0}$. The reported estimates are close to their true values of 1.248832 and 0.8325546 , which are given in table 1 .

The intuition in this example comes from the ability of qreg to recover the true parameters of our specified DGP. As we increase the number of observations in our sample size, the qreg estimates will get closer to the true values.

## > Example 8

In the example below, we estimate the parameters of the conditional quantile function for the 0.25 quantile and compare them with the true values.

```
. qreg y x, quantile(.25)
Iteration 1: WLS sum of weighted deviations = 65395.359
Iteration 1: Sum of abs. weighted deviations = 65397.892
Iteration 2: Sum of abs. weighted deviations = 52640.481
Iteration 3: Sum of abs. weighted deviations = 50706.508
Iteration 4: Sum of abs. weighted deviations = 49767.356
Iteration 5: Sum of abs. weighted deviations = 49766.98
Iteration 6: Sum of abs. weighted deviations = 49765.818
Iteration 7: Sum of abs. weighted deviations = 49765.589
Iteration 8: Sum of abs. weighted deviations = 49765.549
Iteration 9: Sum of abs. weighted deviations = 49765.533
Iteration 10: Sum of abs. weighted deviations = 49765.528
Iteration 11: Sum of abs. weighted deviations = 49765.527
Iteration 12: Sum of abs. weighted deviations = 49765.527
Iteration 13: Sum of abs. weighted deviations = 49765.527
.25 Quantile regression 100, Number of obs = 100,000
\begin{tabular}{r|rccccr}
\hline y & Coefficient & Std. err. & t & \(\mathrm{P}>|\mathrm{tl}|\) & [95\% conf. interval] \\
\hline x & .8207143 & .0106425 & 77.12 & 0.000 & .799855 & .8415735 \\
_cons & .5075988 & .0201512 & 25.19 & 0.000 & .4681026 & .547095 \\
\hline
\end{tabular}
```

As above, qreg reports the estimates of $\beta_{1}$ and $\beta_{0}$ in the output table for x and _cons, respectively. The reported estimates are close to their true values of 0.80454 and 0.53636 , which are given in table 1. As expected, the estimates are close to their true values. Also as expected, the estimates for the 0.25 quantile are smaller than the estimates for the 0.5 quantile.

## > Example 9

We finish this section by estimating the parameters of the conditional quantile function for the 0.8 quantile and comparing them with the true values.


As above, qreg reports the estimates of $\beta_{1}$ and $\beta_{0}$ in the output table for x and _cons, respectively. The reported estimates are close to their true values of 1.902954 and 1.268636 , which are given in table 1. As expected, the estimates are close to their true values. Also as expected, the estimates for the 0.8 quantile are larger than the estimates for the 0.5 quantile.

## Stored results

qreg stores the following in e():
Scalars

```
e(N)
    e(df_m)
    e(df_r)
    e(q)
    e(q-v)
    e(r2_p)
    e(sum_adev)
    e(sum_rdev)
    e(sum_w)
    e(f_r)
    e(sparsity)
    e(bwidth)
    e(kbwidth)
    e(rank)
    e(convcode)
```

Macros

Matrices
e(b)
number of observations model degrees of freedom residual degrees of freedom quantile requested value of the quantile pseudo- $R^{2}$ sum of absolute deviations sum of raw deviations sum of weights density estimate sparsity estimate bandwidth
kernel bandwidth
rank of e(V)
0 if converged; otherwise, return code for why nonconvergence
Macros

```
e(cmd)
e(cmdline)
e(depvar)
e(bwmethod)
e(denmethod)
e(kernel)
e(wtype)
e(wexp)
e(vce)
e(vcetype)
e(properties)
e(predict)
e(marginsnotok)
e(asbalanced)
e(asobserved)
```


## qreg

command as typed
name of dependent variable
bandwidth method; hsheather, bofinger, or chamberlain
density estimation method; fitted, residual, or kernel
kernel function
weight type
weight expression
vcetype specified in vce()
title used to label Std. err.
b V
program used to implement predict
predictions disallowed by margins
factor variables fvset as asbalanced
factor variables fvset as asobserved

```
    e(V)
e(V)
```

coefficient vector
variance-covariance matrix of the estimators
Functions
e(sample) marks estimation sample
In addition to the above, the following is stored in r() :

```
Matrices
    r(table)
```

matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.
iqreg stores the following in e():
Scalars
e(N)
e(df_r)
e(q0)
e(q1)
e(reps)
e(r2_p_q0)
e(r2_p_q1)
e(sumrdev0)
e(sumrdev1)
e(sumadev0)
e(sumadev1)
e(rank)
e(convcode)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(vcetype)
e(properties)
e(predict)
e(marginsnotok)
e(asbalanced)
e(asobserved)
Matrices
e(b)
e(V)
Functions
e(sample) marks estimation sample

In addition to the above, the following is stored in r() :

Matrices $r$ (table)
matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.
sqreg stores the following in e():
Scalars
e(N)
e(df_r)
$e\left(n \_q\right)$
e(q\#)
e(reps)
e(r2_p_q\#)
e(sumrdv\#)
e(sumadv\#)
e(rank)
e(convcode)
Macros

```
e(cmd)
    e(cmdline)
    e(depvar)
    e(eqnames)
    e(vcetype)
    e(properties)
    e(predict)
```

number of observations
residual degrees of freedom
number of quantiles requested
the quantiles requested
number of replications
pseudo- $R^{2}$ for q\#
sum of raw deviations for $q \#$
sum of absolute deviations for $\mathrm{q} \#$
rank of $e(V)$
0 if converged; otherwise, return code for why nonconvergence

## sqreg

command as typed
name of dependent variable
names of equations
title used to label Std. err.
b V
program used to implement predict

```
e(marginsnotok) predictions disallowed by margins
e(asbalanced) factor variables fvset as asbalanced
e(asobserved) factor variables fvset as asobserved
Matrices
    e(b) coefficient vector
    e(V) variance-covariance matrix of the estimators
Functions
    e(sample) marks estimation sample
```

In addition to the above, the following is stored in r() :

Matrices
$r$ (table) matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r -class command is run after the estimation command.
bsqreg stores the following in e():
Scalars
e(N)
e(df_r)
$e(q)$
e(q_v)
e(reps)
e(r2_p)
e(sum_adev)
e(sum_rdev)
e(rank)
e(convcode)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(properties)
e(predict)
e(marginsnotok)
e(asbalanced)
e(asobserved)
Matrices
e(b)
e(V)
Functions
$e$ (sample) marks estimation sample

In addition to the above, the following is stored in r() :

Matrices
$r(t a b l e) \quad$ matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

Methods and formulas are presented under the following headings:
Introduction
Linear programming formulation of quantile regression
Standard errors when residuals are i.i.d.
Pseudo- $R^{2}$

## Introduction

According to Stuart and Ord (1991, 1084), the method of minimum absolute deviations was first proposed by Boscovich in 1757 and was later developed by Laplace; Stigler (1986, 39-55) and Hald (1998, 97-103, 112-116) provide historical details. According to Bloomfield and Steiger (1980), Harris (1950) later observed that the problem of minimum absolute deviations could be turned into the linear programming problem that was first implemented by Wagner (1959). Interest has grown in this method because robust methods and extreme value modeling have become more popular. Statistical and computational properties of minimum absolute deviation estimators are surveyed by Narula and Wellington (1982). Cameron and Trivedi (2005), Hao and Naiman (2007), and Wooldridge (2010) provide excellent introductions to quantile regression methods, while Koenker (2005) gives an in-depth review of the topic.

## Linear programming formulation of quantile regression

Define $\tau$ as the quantile to be estimated; the median is $\tau=0.5$. For each observation $i$, let $\varepsilon_{i}$ be the residual

$$
\varepsilon_{i}=y_{i}-\mathbf{x}_{i}^{\prime} \widehat{\boldsymbol{\beta}}_{\tau}
$$

The objective function to be minimized is

$$
\begin{align*}
c_{\tau}\left(\varepsilon_{i}\right) & =\left(\tau \mathbf{1}\left\{\varepsilon_{i} \geq 0\right\}+(1-\tau) \mathbf{1}\left\{\varepsilon_{i}<0\right\}\right)\left|\varepsilon_{i}\right| \\
& =\left(\tau \mathbf{1}\left\{\varepsilon_{i} \geq 0\right\}-(1-\tau) \mathbf{1}\left\{\varepsilon_{i}<0\right\}\right) \varepsilon_{i}  \tag{2}\\
& =\left(\tau-\mathbf{1}\left\{\varepsilon_{i}<0\right\}\right) \varepsilon_{i}
\end{align*}
$$

where $1\{\cdot\}$ is the indicator function. This function is sometimes referred to as the check function because it resembles a check mark (Wooldridge 2010, 450); the slope of $c_{\tau}\left(\varepsilon_{i}\right)$ is $\tau$ when $\varepsilon_{i}>0$ and is $\tau-1$ when $\varepsilon_{i}<0$, but is undefined for $\varepsilon_{i}=0$. Choosing the $\widehat{\boldsymbol{\beta}}_{\tau}$ that minimize $c_{\tau}\left(\varepsilon_{i}\right)$ is equivalent to finding the $\widehat{\boldsymbol{\beta}}_{\tau}$ that make $\mathbf{x} \widehat{\boldsymbol{\beta}}_{\tau}$ best fit the quantiles of the distribution of $y$ conditional on $\mathbf{x}$.

This minimization problem is set up as a linear programming problem and is solved with linear programming techniques, as suggested by Armstrong, Frome, and Kung (1979) and described in detail by Koenker (2005). Here $2 n$ slack variables, $\mathbf{u}_{n \times 1}$ and $\mathbf{v}_{n \times 1}$, are introduced, where $u_{i} \geq 0, v_{i} \geq 0$, and $u_{i} \times v_{i}=0$, reformulating the problem as

$$
\min _{\boldsymbol{\beta}_{\tau}, \mathbf{u}, \mathbf{v}}\left\{\tau \mathbf{1}_{n}^{\prime} \mathbf{u}+(1-\tau) \mathbf{1}_{n}^{\prime} \mathbf{v} \mid \mathbf{y}-\mathbf{X} \boldsymbol{\beta}_{\tau}=\mathbf{u}-\mathbf{v}\right\}
$$

where $1_{n}$ is a vector of 1 s . This is a linear objective function on a polyhedral constraint set with $\binom{n}{k}$ vertices, and our goal is to find the vertex that minimizes (2). Each step in the search is described by a set of $k$ observations through which the regression plane passes, called the basis. A step is taken by replacing a point in the basis if the linear objective function can be improved. If this occurs, a line is printed in the iteration log. The definition of convergence is exact in the sense that no amount of added iterations could improve the objective function.

A series of weighted least-squares (WLS) regressions is used to identify a set of observations as a starting basis. The WLS algorithm for $\tau=0.5$ is taken from Schlossmacher (1973) with a generalization for $0<\tau<1$ implied from Hunter and Lange (2000).

## Standard errors when residuals are i.i.d.

The estimator for the VCE implemented in qreg assumes that the errors of the model are independent and identically distributed (i.i.d.). When the errors are i.i.d., the large-sample VCE is

$$
\begin{equation*}
\operatorname{cov}\left(\boldsymbol{\beta}_{\tau}\right)=\frac{\tau(1-\tau)}{f_{Y}^{2}\left(\xi_{\tau}\right)}\left\{E\left(\mathbf{x}_{i} \mathbf{x}_{i}^{\prime}\right)\right\}^{-1} \tag{3}
\end{equation*}
$$

where $\xi_{\tau}=F_{Y}^{-1}(\tau)$ and $F_{Y}(y)$ is the distribution function of $Y$ with density $f_{Y}(y)$. See Koenker (2005, 73) for this result. From (3), we see that the regression precision depends on the inverse of the density function, termed the sparsity function, $s_{\tau}=1 / f_{Y}\left(\xi_{\tau}\right)$.

While $1 / n \sum_{i=1}^{n} \mathbf{x}_{i} \mathbf{x}_{i}^{\prime}$ estimates $E\left(\mathbf{x}_{i} \mathbf{x}_{i}^{\prime}\right)$, estimating the sparsity function is more difficult. qreg provides several methods to estimate the sparsity function. The different estimators are specified through the suboptions of vce(iid, denmethod bwidth). The suboption denmethod specifies the functional form for the sparsity estimator. The default is fitted.

Here we outline the logic underlying the fitted estimator. Because $F_{Y}(y)$ is the distribution function for $Y$, we have $f_{Y}(y)=\left\{d F_{y}(y)\right\} / d y, \tau=F_{Y}\left(\xi_{\tau}\right)$, and $\xi_{\tau}=F_{Y}^{-1}(\tau)$. When differentiating the identity $F_{Y}\left\{F_{Y}^{-1}(\tau)\right\}=\tau$, the sparsity function can be written as $s_{\tau}=\left\{F_{Y}^{-1}(\tau)\right\} / d t$. Numerically, we can approximate the derivative using the centered difference,

$$
\begin{equation*}
\frac{F_{Y}^{-1}(\tau)}{d t} \approx \frac{F_{Y}^{-1}(\tau+h)-F_{Y}^{-1}(\tau-h)}{2 h}=\frac{\xi_{\tau+h}-\xi_{\tau-h}}{2 h}=\widehat{s}_{\tau} \tag{4}
\end{equation*}
$$

where $h$ is the bandwidth.
The empirical quantile function is computed by first estimating $\boldsymbol{\beta}_{\tau+h}$ and $\boldsymbol{\beta}_{\tau-h}$, and then computing $\widehat{F}_{Y}^{-1}(\tau+h)=\overline{\mathbf{x}}^{\prime} \widehat{\boldsymbol{\beta}}_{\tau+h}$ and $\widehat{F}_{Y}^{-1}(\tau-h)=\overline{\mathbf{x}}^{\prime} \widehat{\boldsymbol{\beta}}_{\tau-h}$, where $\overline{\mathbf{x}}$ is the sample mean of the independent variables $\mathbf{x}$. These quantities are then substituted into (4).

Alternatively, as the option suggests, vce(iid, residual) specifies that qreg use the empirical quantile function of the residuals to estimate the sparsity. Here we substitute $F_{\epsilon}$, the distribution of the residuals, for $F_{Y}$, which only differ by their first moments.

The $k$ residuals associated with the linear programming basis will be zero, where $k$ is the number of regression coefficients. These zero residuals are removed before computing the $\tau+h$ and $\tau-h$ quantiles, $\varepsilon_{(\tau+h)}=\widehat{F}_{\epsilon}^{-1}(\tau+h)$ and $\varepsilon_{(\tau-h)}=\widehat{F}_{\epsilon}^{-1}(\tau-h)$. The $\widehat{F}_{\epsilon}^{-1}$ estimates are then substituted for $F_{Y}^{-1}$ in (4).

Each of the estimators for the sparsity function depends on a bandwidth. The vce () suboption bwidth specifies the bandwidth method to use. The three bandwidth options and their citations are hsheather (Hall and Sheather 1988), bofinger (Bofinger 1975), and chamberlain (Chamberlain 1994).

Their formulas are

$$
\begin{aligned}
& h_{s}=n^{-1 / 3} \Phi^{-1}\left(1-\frac{\alpha}{2}\right)^{2 / 3}\left[\frac{3}{2} \times \frac{\phi\left\{\Phi^{-1}(\tau)\right\}^{2}}{2 \Phi^{-1}(\tau)^{2}+1}\right]^{1 / 3} \\
& h_{b}=n^{-1 / 5}\left[\frac{\frac{9}{2} \phi\left\{\Phi^{-1}(\tau)\right\}^{4}}{\left\{2 \Phi^{-1}(\tau)^{2}+1\right\}^{2}}\right]^{1 / 5} \\
& h_{c}=\Phi^{-1}\left(1-\frac{\alpha}{2}\right) \sqrt{\frac{\tau(1-\tau)}{n}}
\end{aligned}
$$

where $h_{s}$ is the Hall-Sheather bandwidth, $h_{b}$ is the Bofinger bandwidth, $h_{c}$ is the Chamberlain bandwidth, $\Phi()$ and $\phi()$ are the standard normal distribution and density functions, $n$ is the sample size, and $100(1-\alpha)$ is the confidence level set by the level () option. Koenker (2005) discusses the derivation of the Hall-Sheather and the Bofinger bandwidth formulas. You should avoid modifying the confidence level when replaying estimates that use the Hall-Sheather or Chamberlain bandwidths because these methods use the confidence level to estimate the coefficient standard errors.

Finally, the vce() suboption kernel (kernel) specifies that qreg use one of several kernel density estimators to estimate the sparsity function. kernel allows you to choose which kernel function to use, where the default is the Epanechnikov kernel. See $[R]$ kdensity for the functional form of the eight kernels.

The kernel bandwidth is computed using an adaptive estimate of scale

$$
h_{k}=\min \left(\hat{\sigma}, \frac{r_{q}}{1.34}\right) \times\left\{\Phi^{-1}(\tau+h)-\Phi^{-1}(\tau-h)\right\}
$$

where $h$ is one of $h_{s}, h_{b}$, or $h_{c} ; r_{q}$ is the interquartile range; and $\widehat{\sigma}$ is the standard deviation of $\mathbf{y}$; see Silverman $(1986,47)$ and Koenker $(2005,81)$ for discussions. Let $\widehat{f}_{\epsilon}\left(\varepsilon_{i}\right)$ be the kernel density estimate for the $i$ th residual, and then the kernel estimator for the sparsity function is

$$
\widehat{s}_{\tau}=\frac{n h_{k}}{\sum_{i=1}^{n} \widehat{f}_{\epsilon}\left(\varepsilon_{i}\right)}
$$

Finally, substituting your choice of sparsity estimate into (3) results in the i.i.d. variance-covariance matrix

$$
\mathbf{V}_{n}=\widehat{s}_{\tau}^{2} \tau(1-\tau)\left(\sum_{i=1}^{n} \mathbf{x}_{i} \mathbf{x}_{i}^{\prime}\right)^{-1}
$$

## Pseudo-R ${ }^{2}$

The pseudo- $R^{2}$ is calculated as

$$
1-\frac{\text { sum of weighted deviations about estimated quantile }}{\text { sum of weighted deviations about raw quantile }}
$$

This is based on the likelihood for a double-exponential distribution $e^{v_{i}\left|\varepsilon_{i}\right|}$, where $v_{i}$ are multipliers

$$
v_{i}= \begin{cases}\tau & \text { if } \varepsilon_{i}>0 \\ (1-\tau) & \text { otherwise }\end{cases}
$$

Minimizing the objective function (2) with respect to $\boldsymbol{\beta}_{\tau}$ also minimizes $\sum_{i}\left|\varepsilon_{i}\right| v_{i}$, the sum of weighted least absolute deviations. For example, for the 50 th percentile $v_{i}=1$, for all $i$, and we have median regression. If we want to estimate the 75th percentile, we weight the negative residuals by 0.25 and the positive residuals by 0.75 . It can be shown that the criterion is minimized when $75 \%$ of the residuals are negative.

## References

Angrist, J. D., and J.-S. Pischke. 2009. Mostly Harmless Econometrics: An Empiricist's Companion. Princeton, NJ: Princeton University Press.
Armstrong, R. D., E. L. Frome, and D. S. Kung. 1979. Algorithm 79-01: A revised simplex algorithm for the absolute deviation curve fitting problem. Communications in Statistics—Simulation and Computation 8: 175-190. https://doi.org/10.1080/03610917908812113.
Biewen, M., and P. Erhardt. 2021. arhomme: An implementation of the Arellano and Bonhomme (2017) estimator for quantile regression with selection correction. Stata Journal 21: 602-625.

Bloomfield, P., and W. Steiger. 1980. Least absolute deviations curve-fitting. SIAM Journal on Scientific Computing 1: 290-301. https://doi.org/10.1137/0901019.
Bofinger, E. 1975. Estimation of a density function using order statistics. Australian Journal of Statistics 17: 1-7. https://doi.org/10.1111/j.1467-842X.1975.tb01366.x.

Bottai, M., and N. Orsini. 2019. qmodel: A command for fitting parametric quantile models. Stata Journal 19: 261-293.
Cameron, A. C., and P. K. Trivedi. 2005. Microeconometrics: Methods and Applications. New York: Cambridge University Press.
——. 2022. Microeconometrics Using Stata. 2nd ed. College Station, TX: Stata Press.
Chamberlain, G. 1994. Quantile regression, censoring, and the structure of wages. In Advances in Econometrics, Vol. 1: Sixth World Congress, ed. C. A. Sims, 171-210. Cambridge: Cambridge University Press.
Chernozhukov, V., I. Fernández-Val, S. Han, and A. Kowalski. 2019. Censored quantile instrumental-variable estimation with Stata. Stata Journal 19: 768-781.

Deb, P., E. C. Norton, and W. G. Manning. 2017. Health Econometrics Using Stata. College Station, TX: Stata Press.
Drukker, D. M. 2016. Quantile regression allows covariate effects to differ by quantile. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2016/09/27/quantile-regression-allows-covariate-effects-to-differ-by-quantile/.

Efron, B., and R. J. Tibshirani. 1993. An Introduction to the Bootstrap. New York: Chapman and Hall/CRC.
Frölich, M., and B. Melly. 2010. Estimation of quantile treatment effects with Stata. Stata Journal 10: 423-457.
Gould, W. W. 1992. sg11.1: Quantile regression with bootstrapped standard errors. Stata Technical Bulletin 9: 19-21. Reprinted in Stata Technical Bulletin Reprints, vol. 2, pp. 137-139. College Station, TX: Stata Press.
—_. 1997a. cre46: Better numerical derivatives and integrals. Stata Technical Bulletin 35: 3-5. Reprinted in Stata Technical Bulletin Reprints, vol. 6, pp. 8-12. College Station, TX: Stata Press.
__. 1997b. sg70: Interquantile and simultaneous-quantile regression. Stata Technical Bulletin 38: 14-22. Reprinted in Stata Technical Bulletin Reprints, vol. 7, pp. 167-176. College Station, TX: Stata Press.
Gould, W. W., and W. H. Rogers. 1994. Quantile regression as an alternative to robust regression. In 1994 Proceedings of the Statistical Computing Section. Alexandria, VA: American Statistical Association.
Hald, A. 1998. A History of Mathematical Statistics from 1750 to 1930. New York: Wiley.
Hall, P., and S. J. Sheather. 1988. On the distribution of a Studentized quantile. Journal of the Royal Statistical Society, Series B 50: 381-391. https://doi.org/10.1111/j.2517-6161.1988.tb01735.x.
Hao, L., and D. Q. Naiman. 2007. Quantile Regression. Thousand Oaks, CA: Sage.
Harris, T. 1950. Regression using minimum absolute deviations. American Statistician 4: 14-15. https://doi.org/10.1080/00031305.1950.10501620.

Huber, P. J. 1967. The behavior of maximum likelihood estimates under nonstandard conditions. In Vol. 1 of Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, 221-233. Berkeley: University of California Press.

Huber, P. J., and E. M. Ronchetti. 2009. Robust Statistics. 2nd ed. New York: Wiley.
Hunter, D. R., and K. Lange. 2000. Quantile regression via an MM algorithm. Journal of Computational and Graphical Statistics 9: 60-77. https://doi.org/10.2307/1390613.
Kaplan, D. M. 2022. Smoothed instrumental variables quantile regression. Stata Journal 22: 379-403.
Koenker, R. 2005. Quantile Regression. New York: Cambridge University Press.
Koenker, R., and K. Hallock. 2001. Quantile regression. Journal of Economic Perspectives 15: 143-156. https://doi.org/10.1257/jep.15.4.143.
Muñoz, E., and M. Siravegna. 2021. Implementing quantile selection models in Stata. Stata Journal 21: 952-971.
Narula, S. C., and J. F. Wellington. 1982. The minimum sum of absolute errors regression: A state of the art survey. International Statistical Review 50: 317-326. https://doi.org/10.2307/1402501.
Orsini, N., and M. Bottai. 2011. Logistic quantile regression in Stata. Stata Journal 11: 327-344.
Rios-Avila, F. 2020. Recentered influence functions (RIFs) in Stata: RIF regression and RIF decomposition. Stata Journal 20: 51-94.

Rousseeuw, P. J., and A. M. Leroy. 1987. Robust Regression and Outlier Detection. New York: Wiley.
Schlossmacher, E. J. 1973. An iterative technique for absolute deviations curve fitting. Journal of the American Statistical Association 68: 857-859. https://doi.org/10.2307/2284512.

Silverman, B. W. 1986. Density Estimation for Statistics and Data Analysis. London: Chapman and Hall.
Stigler, S. M. 1986. The History of Statistics: The Measurement of Uncertainty before 1900. Cambridge, MA: Belknap Press.

Stuart, A., and J. K. Ord. 1991. Kendall's Advanced Theory of Statistics: Distribution Theory, Vol. 1. 5th ed. New York: Oxford University Press.

Wagner, H. M. 1959. Linear programming techniques for regression analysis. Journal of the American Statistical Association 54: 206-212. https://doi.org/10.2307/2282146.

Wooldridge, J. M. 2010. Econometric Analysis of Cross Section and Panel Data. 2nd ed. Cambridge, MA: MIT Press.
Wu, C. F. J. 1986. Jackknife, bootstrap and other resampling methods in regression analysis. Annals of Statistics 14: 1261-1350 (including discussions and rejoinder). https://doi.org/10.1214/aos/1176350142.

## Also see

[R] qreg postestimation - Postestimation tools for qreg, iqreg, sqreg, and bsqreg
[R] bootstrap - Bootstrap sampling and estimation
$[R]$ ivqregress - Instrumental-variables quantile regression
$[R]$ regress - Linear regression
[R] rreg - Robust regression
[BAYES] bayes: qreg - Bayesian quantile regression ${ }^{+}$
[MI] Estimation - Estimation commands for use with mi estimate
[U] 20 Estimation and postestimation commands

## Title

qreg postestimation - Postestimation tools for qreg, iqreg, sqreg, and bsqreg

| Postestimation commands | predict | margins estat |
| :--- | :--- | :--- |
| Remarks and examples | Also see |  |

## Postestimation commands

The following postestimation command is of special interest after sqreg:

| Command | Description |
| :--- | :--- |
| estat coefplot | plot coefficients and their confidence intervals at different quantiles |

The following postestimation commands are available after qreg, iqreg, bsqreg, and sqreg:

| Command | Description |
| :---: | :---: |
| contrast | contrasts and ANOVA-style joint tests of estimates |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estimates | cataloging estimation results |
| etable | table of estimation results |
| $\dagger$ forecast | dynamic forecasts and simulations |
| * hausman | Hausman's specification test |
| lincom | point estimates, standard errors, testing, and inference for linear combinations of coefficients |
| linktest | link test for model specification |
| margins | marginal means, predictive margins, marginal effects, and average marginal effects |
| marginsplot | graph the results from margins (profile plots, interaction plots, etc.) |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients |
| predict | predictions and their SEs, residuals, etc. |
| predictnl | point estimates, standard errors, testing, and inference for generalized predictions |
| pwcompare | pairwise comparisons of estimates |
| test | Wald tests of simple and composite linear hypotheses |
| testnl | Wald tests of nonlinear hypotheses |

[^50]
## predict

## Description for predict

predict creates a new variable containing predictions such as linear predictions, standard errors, and residuals.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

For qreg, iqreg, and bsqreg

```
predict [type] newvar [if] [in] [, [xb|stdp|residuals]]
```

For sqreg
predict [type] newvar [if] [in] [, equation(eqno[,eqno]) statistic]
statistic Description
Main
xb linear prediction; the default
stdp standard error of the linear prediction
stddp standard error of the difference in linear predictions
residuals
residuals
These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

## Options for predict

$\qquad$ Main
xb , the default, calculates the linear prediction.
stdp calculates the standard error of the linear prediction.
stddp is allowed only after you have fit a model using sqreg. The standard error of the difference in linear predictions ( $\mathbf{x}_{1 j} \mathbf{b}-\mathbf{x}_{2 j} \mathbf{b}$ ) between equations 1 and 2 is calculated. residuals calculates the residuals, that is, $y_{j}-\mathbf{x}_{j} \mathbf{b}$. equation (eqno [,eqno]) specifies the equation to which you are making the calculation.
equation() is filled in with one eqno for the xb , stdp, and residuals options. equation(\#1) would mean that the calculation is to be made for the first equation, equation(\#2) would mean the second, and so on. You could also refer to the equations by their names. equation (income) would refer to the equation named income and equation(hours) to the equation named hours. If you do not specify equation(), results are the same as if you had specified equation(\#1). To use stddp, you must specify two equations. You might specify equation(\#1, \#2) or equation (q80, q20) to indicate the 80th and 20th quantiles.

## margins

## Description for margins

margins estimates margins of response for linear predictions.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [options]
```

statistic Description
$\mathrm{xb} \quad$ linear prediction; the default
stdp not allowed with margins
stddp not allowed with margins
residuals not allowed with margins

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

## estat

## Description for estat

estat coefplot plots the estimated coefficients and their confidence intervals (CIs) after sqreg.

## Menu for estat

Statistics > Postestimation

## Syntax for estat

```
estat coefplot [varname] [, options]
```

varname is one of the variables specified when fitting sqreg; the default is the first endogenous variable.
options Description
noci do not plot the CIs
nools do not plot the ordinary least-squares (OLS) estimates
Plot
connect_options
marker_options
change look of lines or connecting method
change look of markers (color, size, etc.)
CI plot
ciopts(area_options) affect rendition of the pointwise CIs
Line options
lineopts(cline_options) affect rendition of reference line identifying the OLS estimates
Y axis, X axis, Titles, Legend, Overall
twoway_options any options other than by () documented in [G-3] twoway_options

## Options for estat

noci removes plots of the pointwise CIs. The default is to plot the CIs. nools removes the plot of the OLS estimates. The default is to plot the OLS reference line.
$\qquad$ Plot
connect_options specify how points on a graph are to be connected; [G-3] connect_options. marker_options affect the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G-3] marker_options.
$\qquad$ Cl plot
ciopts(area_options) affects rendition of the pointwise CIs; see [G-3] area_options.
$\qquad$ Line options
lineopts(cline_options) affects rendition of reference line identifying the OLS estimates; see [G-3] cline_options.

Y axis, X axis, Titles, Legend, Overall
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

## Remarks and examples

## > Example 1: Predictions after qreg and iqreg

In example 4 of $[R]$ qreg, we fit regressions for the lower and the upper quartile of the price variable. The predict command can be used to obtain the linear prediction after each regression.

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. qreg price weight length foreign, quantile(.25)
    (output omitted)
. predict q25
(option xb assumed; fitted values)
. qreg price weight length foreign, quantile(.75)
    (output omitted)
- predict q75
(option xb assumed; fitted values)
```

We can use the variables generated by predict to compute the predicted interquartile range, that is,

```
. generate iqr1 = q75 - q25
```

If we directly perform the interquartile range regression with the iqreg command, we can predict the interquartile range and also the standard error for the prediction.

```
. iqreg price weight length foreign, quantile(.25 .75)
    (output omitted)
- predict iqr2
(option xb assumed; fitted values)
predict stdp, stdp
```

We now plot the predicted interquartile range versus variable length:

```
scatter iqr2 length
```



As stated in example 5 of [R] qreg, the negative coefficient for the length variable means that increases in length imply decreases in the interquartile range and therefore in price dispersion. Consequently, we could have expected a downward trend in the plot, but there is not. This is because the regression output indicates that when we hold the rest of the variables constant, an increase in length leads to a decrease in iqr2. However, there is a high correlation between weight and length, which could be masking the effect of length on iqr2. We can achieve a better visualization by using a contour plot.
. twoway contour iqr2 weight length, level(10)


We can see the effect by setting a fixed value of length on the vertical axis, say, 3,000 lbs. When we move from left to right on the horizontal axis, we see that for small values of length, iqr2 values are shown in red, meaning high values, and when we move toward the right, the graph indicates transition into increasingly smaller values.

## Example 2: Coefficients plot after sqreg

In example 5 of $[R]$ qreg, we simultaneously estimated the quantile regressions at different quantile indexes.

```
. use https://www.stata-press.com/data/r18/auto, clear
(1978 automobile data)
. set seed 1001
. sqreg price weight length foreign, q(.25 .5 .75) reps(100)
    (output omitted)
```

We can now use estat coefplot to visualize the coefficients at different quantiles. For example, we can type estat coefplot weight to see the trend of the effects of weight on price across quantiles.

```
. estat coefplot weight
```



The resulting graph shows that there is an upward trend in the effects. For reference, the red line shows the OLS estimates.

## Also see

[R] qreg - Quantile regression
[U] 20 Estimation and postestimation commands

## Title

query - Display system parameters

Description Syntax Remarks and examples Also see

## Description

query displays the settings of various Stata parameters.

## Syntax

```
query [ memory | output | interface | graphics | network | update |
    trace | mata | java | lapack | putdocx | putpdf | python | rng |
    sort | unicode | other ]
```


## Remarks and examples

query provides more system information than you will ever want to know. You do not need to understand every line of output that query produces if all you need is one piece of information. Here is what happens when you type query:
. query

| Memory settings |  |  |
| :--- | :--- | :--- |
| set maxvar | 5000 | $2048-120000$; max. vars allowed |
| set niceness | 5 | $0-10$ |
| set min_memory | 0 | $0-1600 \mathrm{~g}$ |
| set max_memory | . | $32 \mathrm{~m}-1600 \mathrm{~g}$ or. |
| set segmentsize | 32 m | $1 \mathrm{~m}-32 \mathrm{~g}$ |
| set adosize | 1000 | kilobytes |
| set max_preservemem | 1 g | $0-1600 \mathrm{~g}$ |

Output settings
set more off

| set more | off |  |
| :---: | :---: | :---: |
| set rmsg | off |  |
| set dp | period | may be period or comma |
| set linesize | 80 | characters |
| set pagesize | 32 | lines |
| set iterlog | on |  |
| set level | 95 | percent confidence intervals |
| set clevel | 95 | percent credible intervals |
| set showbaselevels |  | may be empty, off, on, or all |
| set showemptycells |  | may be empty, off, or on |
| set showomitted |  | may be empty, off, or on |
| set fvlabel | on |  |
| set fvwrap | 1 |  |
| set fvwrapon | word | may be word or width |


| set lstretch | may be empty, off, or on |  |
| :--- | :--- | :--- |
| set cformat <br> set pformat <br> set sformat | may be empty or a numerical format <br> may be empty or a numerical format <br> may be empty or a numerical format |  |
| set coeftabresults <br> set dots | on <br> on |  |
| set logtype smcl <br> set logmsg  | may be smcl or text |  |
| set collect_label <br> set collect_style <br> set table_style <br> set etable_style <br> set dtable_style <br> set collect_warn | default <br> default <br> table <br> etable <br> dtable <br> on |  |

Interface settings

| set dockable | on |
| :--- | :--- |
| set floatwindows | off |
| set locksplitters | off |
| set pinnable | on |
| set taskbargroups | on |
| set doublebuffer | on |


| set linegap | 1 | pixels |
| :--- | :--- | :--- |
| set scrollbufsize | 204800 | characters |
| set fastscroll | on |  |
| set reventries | 5000 | lines |
| set maxdb | 50 | dialog boxes |

Graphics settings
set graphics on
set autotabgraphs on
set scheme stcolor
set printcolor asis may be automatic, asis, gs1, gs2, gs3
set copycolor asis may be automatic, asis, gs1, gs2, gs3
Network settings
set httpproxy off
set httpproxyhost
set httpproxyport 80
set httpproxyauth off
set httpproxyuser
set httpproxypw

```
Update settings
    set update_query off
    set update_interval 7
    set update_prompt on
```

| Trace (programming debugging) | settings |
| :---: | :---: |
| set trace | off |
| set tracedepth | 32000 |
| set traceexpand | on |
| set tracesep | on |
| set traceindent | on |
| set tracenumber | off |
| set tracehilite |  |

Mata settings
set matastrict off
set matalnum off
set mataoptimize on
set matafavor space may be space or speed
set matacache 2000 kilobytes
set matalibs lmatabase;lmataado;lmatabma;lmatacollect;lmataerm;lm
> atafc;lmatagsem;lmatalasso;lmatamcmc;lmatameta;lmatami;lmatamixlog;lmatanumli
> b;lmataopt;lmatapath;lmatapostest;lmatapss;lmatasem;lmatasp;lmatasvy;lmatatab set matamofirst off
set matasolvetol . may be . or any double-precision number

| ```Java settings set java_heapmax set java_home > x64\zulu-jdk17.0.6\``` | 4096m <br> C: \Program Files \Stata18\utilities $\backslash j a v a \backslash w i n d o w s-~$ |
| :---: | :---: |
| LAPACK settings set lapack_mkl set lapack_mkl_cnr | ```on default``` |
| putdocx settings <br> set docx_hardbreak <br> set docx_paramode <br> set docx_maxtable | $\begin{array}{ll} \text { off } \\ \text { off } & \\ 500 & 1-10000 \end{array}$ |
| putpdf settings <br> set pdf_maxtable | 500 1-10000 |

Python settings
set python_exec
set python_userpath
RNG settings
set rng default may be default, mt64, mt64s, or kiss32
set rngstate XAA00000000000000000000000000000000000000...
set rngstream
1
rng stream number

```
sort settings
    set sortmethod default may be default, fsort, or qsort
    fsort current sort method
    set sortrngstate 1001XZA112210f4b16c1cb10507a1f38cb440c400...
```

Unicode settings

| set locale_ui |  |
| :--- | :--- |
| set locale_functions | en_US |

```
Other settings
    set type
    set maxiter
    set searchdefault
    set varabbrev
    set emptycells
    set fvtrack
    set fvbase
    set processors
    set kmp_blocktime
    set odbcdriver
    set haverdir
    set fredkey
    set collect_double on
    set dtascomplevel 1 dtas default compression level; 0-9
    set reshape_favor default
    may be default, memory, or speed
    set doeditbackup on
```

The output is broken into several divisions: memory, output, interface, graphics, network, update, trace, Mata, Java, LAPACK, putdocx, putpdf, Python, RNG, sort, Unicode, and other settings. We will discuss each one in turn.

We generated the output above using Stata/MP for Windows. Here is what happens when we type query and we are running Stata/SE for Mac:
. query

| Memory settings |  |  |
| :---: | :---: | :---: |
| set maxvar | 5000 | 2048-32767; max. vars allowed |
| set niceness | 5 | 0-10 |
| set min_memory | 0 | 0-1600g |
| set max_memory |  | $32 \mathrm{~m}-1600 \mathrm{~g}$ or . |
| set segmentsize | 32m | $1 \mathrm{~m}-32 \mathrm{~g}$ |
| set adosize | 1000 | kilobytes |
| Output settings |  |  |
| set more | off |  |
| set rmsg | off |  |
| set dp | period | may be period or comma |
| set linesize | 80 | characters |
| set pagesize | 23 | lines |
| set iterlog | on |  |
| set level | 95 | percent confidence intervals |
| set clevel | 95 | percent credible intervals |
| set showbaselevels |  | may be empty, off, on, or all |
| set showemptycells |  | may be empty, off, or on |
| set showomitted |  | may be empty, off, or on |
| set fvlabel | on |  |
| set fvwrap | 1 |  |
| set fvwrapon | word | may be word or width |
| set lstretch |  | may be empty, off, or on |


| set cformat <br> set pformat <br> set sformat | may be empty or a numerical format <br> may be empty or a numerical format <br> may be empty or a numerical format |  |
| :--- | :--- | :--- |
| set coeftabresults <br> set dots | on <br> on | may be smcl or text |
| set logtype <br> set logmsg | smcl <br> on |  |
| set collect_label <br> set collect_style <br> set table_style <br> set etable_style <br> set dtable_style <br> set collect_warn | default <br> default <br> table <br> etable <br> dtable <br> on |  |
| set notifyuser <br> set playsnd <br> set include_bitmap | on <br> off <br> on |  |

Interface settings

| set revkeyboard | on |
| :--- | :--- |
| set varkeyboard | on |
| set smoothfonts | on |


| set linegap | 1 | pixels |
| :--- | :--- | :--- |
| set scrollbufsize <br> set reventries | 204800 <br> characters <br> lines |  |
| set maxdb | 5000 | dialog boxes |

Graphics settings
set graphics on
set scheme stcolor
set printcolor asis may be automatic, asis, gs1, gs2, gs3
set copycolor asis may be automatic, asis, gs1, gs2, gs3
set maxbezierpath 0 bezier path elements
Network settings
set httpproxy off
set httpproxyhost
set httpproxyport 80
set httpproxyauth off
set httpproxyuser
set httpproxypw
Update settings
set update_query off
set update_interval 7
set update_prompt on

| Trace (programming debugging) settings |  |
| :---: | :---: |
| set trace | off |
| set tracedepth | 32000 |
| set traceexpand | on |
| set tracesep | on |
| set traceindent | on |
| set tracenumber | off |
| set tracehilite |  |



```
Other settings 
Other settings 
Other settings 
Other settings 
Other settings 
Other settings 
Other settings 
Other settings 
Other settings 
Other settings 
Other settings 
Other settings 
Other settings 
Other settings 
Other settings 
Other settings 
Other settings 
Other settings 
Other settings 
Other settings 
Other settings 
Other settings 
Other settings 
Other settings 
Other settings 
Other settings 
Other settings 
Other settings 
Other settings 
```


## Memory settings

Memory settings indicate how memory is allocated, the maximum number of variables, and the maximum size of a matrix.

For more information, see

```
            maxvar [D] memory
            niceness [D] memory
    min_memory [D] memory
    max_memory [D] memory
segmentsize [D] memory
    adosize [P] sysdir
max_preservemem [P] preserve
```


## Output settings

Output settings show how Stata displays output on the screen and in $\log$ files.
For more information, see

```
            more [R] more
            rmsg [P] rmsg
            dp [D] format
linesize [R] log
pagesize [R] more
    iterlog [R] set iter
            level [R] level
            clevel [BAYES] set clevel
showbaselevels [R] set showbaselevels
showemptycells [R] set showbaselevels
    showomitted [R] set showbaselevels
            fvlabel [R] set showbaselevels
                    fvwrap [R] set showbaselevels
            fvwrapon [R] set showbaselevels
            lstretch [R] set
            cformat [R] set cformat
            pformat [R] set cformat
            sformat [R] set cformat
coeftabresults [R] set
            dots [R] set
            logtype [R] log
            logmsg [R] log
    collect_label [TABLES] set collect_label
    collect_style [TABLES] set collect_style
        table_style [TABLES] set table_style
    etable_style [TABLES] set etable_style
    dtable_style [TABLES] set dtable_style
    collect_warn [TABLES] set collect_warn
        notifyuser [R] set
            playsnd [R] set
include_bitmap [R] set
```


## Interface settings

Interface settings control how Stata's interface works.
For more information, see

| dockable | $[R]$ set |
| ---: | :--- |
| floatwindows | $[R]$ set |
| locksplitters | $[R]$ set |
| pinnable | $[R]$ set |
| taskbargroups | $[R]$ set |
| doublebuffer | $[R]$ set |
| revkeyboard | $[R]$ set |
| varkeyboard | $[R]$ set |
| smoothfonts | $[R]$ set |
| linegap | $[R]$ set |
| scrollbufsize | $[R]$ set |
| fastscroll | $[R]$ set |
| reventries | $[R]$ set |
| maxdb | $[R]$ db |

## Graphics settings

Graphics settings indicate how Stata's graphics are displayed.
For more information, see

```
            graphics [G-2] set graphics
autotabgraphs [R] set
            scheme [G-2] set scheme
    printcolor [G-2] set printcolor
    copycolor [G-2] set printcolor
maxbezierpath [R] set
```


## Network settings

Network settings determine how Stata interacts with the Internet.
For more information, see $[\mathrm{R}]$ netio.

## Update settings

Update settings determine how Stata performs updates.
For more information, see [R] update.

## Trace settings

Trace settings adjust Stata's behavior and are particularly useful in debugging code.
For more information, see [P] trace.

## Mata settings

Mata settings affect Mata's system parameters.
For more information, see [M-3] mata set.

## Java settings

Java settings control the Java Runtime Environment.
For more information, see [P] Java utilities.

## LAPACK settings

LAPACK settings control the Intel MKL LAPACK routines.
For more information, see [M-1] LAPACK.

## putdocx settings

putdocx settings control how spacing is handled in blocks of text and the number of tables allowed.

For more information, see [RPT] set docx and [RPT] putdocx table.

## putpdf settings

putpdf settings control the number of tables allowed.
For more information, see [RPT] putpdf table.

## Python settings

Python settings control the version and search paths of Python.
For more information, see [P] PyStata integration.

## RNG settings

The RNG settings set Stata's random-number generators.
For more information, see [R] set rng, [R] set seed (set rngstate), and [R] set rngstream.

## sort settings

The sort settings affect sort, gsort, and other commands that use sorting as part of their computation.

For more information, see $[\mathrm{P}]$ set sortmethod and $[\mathrm{P}]$ set sortrngstate.

## Unicode settings

Unicode settings affect Stata's localization package for the user interface.
For more information, see [P] set locale_ui and [P] set locale_functions.

## Other settings

The other settings are a miscellaneous collection.
For more information, see

```
            type [D] generate
            maxiter [R] set iter
searchdefault [R] search
            varabbrev [R] set
    emptycells [R] set
            fvtrack [R] set
            fvbase [R] set
    processors [R] set
kmp_blocktime [R] set
            odbcmgr [D] odbc
    odbcdriver [D] odbc
            haverdir [D] import haver
            fredkey [D] import fred
collect_double [TABLES] set collect_double
dtascomplevel [R] set
reshape_favor [D] reshape
    doeditbackup [R] set
```

In general, the parameters displayed by query can be changed by set; see [R] set.

## Also see

[R] set - Overview of system parameters
[M-3] mata set - Set and display Mata system parameters
[P] creturn - Return c-class values

## Title

ranksum - Equality tests on unmatched data

Description
Options for ranksum
Methods and formulas

## Quick start <br> Options for median <br> References

Menu
Remarks and examples Also see

Syntax
Stored results

## Description

ranksum tests the hypothesis that two independent samples (that is, unmatched data) are from populations with the same distribution by using the Wilcoxon rank-sum test, which is also known as the Mann-Whitney two-sample statistic (Wilcoxon 1945; Mann and Whitney 1947).
median performs a nonparametric $k$-sample test on the equality of medians. It tests the null hypothesis that the $k$ samples were drawn from populations with the same median. For two samples, the $\chi^{2}$ test statistic is computed both with and without a continuity correction.
ranksum and median are for use with unmatched data. For equality tests on matched data, see [ R ] signrank.

## Quick start

Wilcoxon rank-sum test
Test for equality of distributions of v over two groups defined by the levels of catvar1
ranksum v, by (catvar1)
Compute an exact $p$-value for the Wilcoxon rank-sum test
ranksum $v$, by (catvar1) exact
Estimate the probability that a case from the first level of catvar1 has a greater value of v than a case from the second level of catvar1
ranksum v, by (catvar1) porder

Nonparametric equality-of-medians test
Equality of medians test for v over two or more groups defined by the levels of catvar2
median $v$, by (catvar2)
Also report Fisher's exact test
median $v$, by (catvar2) exact
Same as above, but split cases at the median evenly between the above and below groups median $v$, by (catvar2) exact medianties(split)

## Menu

ranksum
Statistics $>$ Nonparametric analysis $>$ Tests of hypotheses $>$ Wilcoxon rank-sum test

## median

Statistics $>$ Nonparametric analysis $>$ Tests of hypotheses $>$ K-sample equality-of-medians test

## Syntax

Wilcoxon rank-sum test
ranksum varname [if] [in], by (groupvar) [exact porder]

Nonparametric equality-of-medians test
median varname [if] [in] [weight], by (groupvar) [median_options]
ranksum_options
Description

## Main

* by (groupvar)
exact
median_options


## Main

* by (groupvar)
exact
medianties (below) medianties(above) medianties(drop)
medianties(split)
porder probability that variable for first group is larger than variable for second group
grouping variable report exact $p$-value for rank-sum test; by default, exact $p$-value is computed when total sample size $\leq 200$


## Options for ranksum


#### Abstract

Main by (groupvar) is required. It specifies the name of the grouping variable. exact specifies that the exact $p$-value be computed in addition to the approximate $p$-value. The exact $p$-value is based on the actual randomization distribution of the test statistic. The approximate $p$-value is based on a normal approximation to the randomization distribution. By default, the exact $p$-value is computed for sample sizes $n=n_{1}+n_{2} \leq 200$ because the normal approximation may not be precise in small samples. The exact computation can be suppressed by specifying noexact. For sample sizes larger than 200 , you must specify exact to compute the exact $p$-value. The exact computation is available for sample sizes $n \leq 1000$. As the sample size approaches 1,000 , the computation takes significantly longer. porder displays an estimate of the probability that a random draw from the first population is larger than a random draw from the second population.


## Options for median

$\qquad$ Main
by (groupvar) is required. It specifies the name of the grouping variable.
exact displays the $p$-value calculated by Fisher's exact test. For two samples, both one- and two-sided $p$-values are displayed.
medianties (below $\mid$ above $\mid$ drop $\mid$ split) specifies how values equal to the overall median are to be handled. The median test computes the median for varname by using all observations and then divides the observations into those falling above the median and those falling below the median. When values for an observation are equal to the sample median, they can be dropped from the analysis by specifying medianties (drop); added to the group above or below the median by specifying medianties (above) or medianties (below), respectively; or if there is more than 1 observation with values equal to the median, they can be equally divided into the two groups by specifying medianties (split). If this option is not specified, medianties (below) is assumed.

## Remarks and examples

$>$ Example 1
We are testing the effectiveness of a new fuel additive. We run an experiment with 24 cars: 12 cars with the fuel treatment and 12 cars without. We input these data by creating a dataset with 24 observations. mpg records the mileage rating, and treat records 0 if the mileage corresponds to untreated fuel and 1 if it corresponds to treated fuel.
. use https://www.stata-press.com/data/r18/fuel2
. ranksum mpg, by(treat)
Two-sample Wilcoxon rank-sum (Mann-Whitney) test

| treat | Obs | Rank sum | Expected |
| ---: | ---: | ---: | ---: |
| Untreated | 12 | 128 | 150 |
| Treated | 12 | 172 | 150 |
| Combined | 24 | 300 | 300 |

Unadjusted variance
300.00

Adjustment for ties -4.04

Adjusted variance
295.96

H0: $\operatorname{mpg}($ treat $==$ Untreated $)=\operatorname{mpg}($ treat $==$ Treated $)$
$z=-1.279$
Prob $>|z|=0.2010$
Exact prob $=0.2117$
Because the total sample is only 24 cars, the exact $p$-value is computed by default. If the sample size were greater than 200 , we would have to specify the exact option if we wanted the exact $p$-value computed.

Despite the small sample size, the $p$-value computed using a normal approximation, 0.2010 , is similar to the exact $p$-value, 0.2117 . These results indicate that the distributions are not statistically different at a 0.05 significance level.

Similarly, the median test,

| Median test |  |  |  |
| :---: | :---: | :---: | :---: |
| Greater than the | Whether car received fuel additive |  |  |
| median | Untreated | Treated | Total |
| no | 7 | 5 | 12 |
| yes | 5 | 7 | 12 |
| Total | 12 | 12 | 24 |
| Pearson chi2 $(1)=0.6667 \quad \operatorname{Pr}=0.414$ |  |  |  |
| Fisher's exact $=\quad 0.684$ |  |  |  |
| 1 -sided Fisher's exact $=\quad 0.342$ |  |  |  |
| Continuity corrected: |  |  |  |
| Pearson chi2 $(1)=0.1667 \quad \operatorname{Pr}=0.683$ |  |  |  |

fails to reject the null hypothesis that there is no difference between the fuel with the additive and the fuel without the additive.

Compare these results from these two tests with those obtained from the signrank and signtest where we found significant differences; see [R] signrank. An experiment run on 24 different cars is not as powerful as a before-and-after comparison using the same 12 cars.

## Stored results

ranksum stores the following in $r()$ :
Scalars

```
r(N) sample size
    r(N_1) sample size of first group
    r(N_2) sample size of second group
    r(z) z
    r(Var_a) adjusted variance
    r(group1) value of variable for first group
    r(sum_obs) observed sum of ranks for first group
    r(sum_exp) expected sum of ranks for first group
    r(p) two-sided p-value from normal approximation
    r(p_l) lower one-sided p-value from normal approximation
    r(p_u) upper one-sided p-value from normal approximation
    r(p_exact) two-sided exact p-value
    r(p_l_exact) lower one-sided exact p-value
    r(p_u_exact) upper one-sided exact p-value
    r(porder) probability that draw from first population is larger than draw from second population
```

median stores the following in r() :

Scalars

| $\mathrm{r}(\mathrm{N})$ | sample size |
| :--- | :--- |
| $\mathrm{r}(\mathrm{chi2})$ | Pearson's $\chi^{2}$ |
| $\mathrm{r}(\mathrm{chi2}$ _cc) | continuity-corrected Pearson's $\chi^{2}$ |
| $\mathrm{r}($ groups $)$ | number of groups compared |
| $\mathrm{r}(\mathrm{p})$ | $p$-value for Pearson's $\chi^{2}$ test |
| $\mathrm{r}(\mathrm{p}-\mathrm{cc})$ | continuity-corrected $p$-value |
| $\mathrm{r}(\mathrm{p}$ eexact) | Fisher's exact $p$-value |
| $\mathrm{r}\left(\mathrm{p} 1 \_\right.$exact $)$ | one-sided Fisher's exact $p$-value |

## Methods and formulas

For a practical introduction to these techniques with an emphasis on examples rather than theory, see Acock (2023), Bland (2015), or Sprent and Smeeton (2007). For a summary of these tests, see Snedecor and Cochran (1989).

Methods and formulas are presented under the following headings:

```
ranksum
median
```


## ranksum

For the Wilcoxon rank-sum test, there are two independent random variables, $X_{1}$ and $X_{2}$, and we test the null hypothesis that $X_{1} \sim X_{2}$. We have a sample of size $n_{1}$ from $X_{1}$ and another of size $n_{2}$ from $X_{2}$.

The data are then ranked without regard to the sample to which they belong. If the data are tied, averaged ranks are used. Wilcoxon's test statistic (1945) is the sum of the ranks for the observations in the first sample:

$$
T=\sum_{i=1}^{n_{1}} R_{1 i}
$$

Mann and Whitney's $U$ statistic (1947) is the number of pairs $\left(X_{1 i}, X_{2 j}\right)$ such that $X_{1 i}>X_{2 j}$. These statistics differ only by a constant:

$$
U=T-\frac{n_{1}\left(n_{1}+1\right)}{2}
$$

Fisher's principle of randomization provides a method for calculating the distribution of the test statistic. The randomization distribution consists of all the possible values of $T$ resulting from the $\binom{n}{n_{1}}$ ways to choose $n_{1}$ ranks from the set of all $n=n_{1}+n_{2}$ observed ranks (untied or tied) and assign them to the first sample. When the exact option is specified (or implied for $n \leq 200$ ), this distribution is computed using a recursive algorithm whose computational time is proportional to $n^{4}$. (See Fisher [1935] for the principle of randomization; Wilcoxon, Katti, and Wilcox [1970] for the computation with untied ranks; and Hill and Peto [1971] for the general recursive algorithm.)
$p$-values can also be computed using a normal approximation to the randomization distribution. It is a straightforward exercise to verify that

$$
E(T)=\frac{n_{1}(n+1)}{2} \quad \text { and } \quad \operatorname{Var}(T)=\frac{n_{1} n_{2} s^{2}}{n}
$$

where $s$ is the standard deviation of the combined ranks, $r_{i}$, for both groups:

$$
s^{2}=\frac{1}{n-1} \sum_{i=1}^{n}\left(r_{i}-\bar{r}\right)^{2}
$$

This formula for the variance is exact and holds both when there are no ties and when there are ties and we use averaged ranks. (Indeed, the variance formula holds for the randomization distribution of choosing $n_{1}$ numbers from any set of $n$ numbers.)

For the normal approximation, we calculate

$$
z=\frac{T-E(T)}{\sqrt{\operatorname{Var}(T)}}
$$

When the porder option is specified, the probability

$$
p=\frac{U}{n_{1} n_{2}}
$$

is computed.

## Technical note

We follow the great majority of the literature in naming these tests for Wilcoxon, Mann, and Whitney. However, they were independently developed by several other researchers in the late 1940s and early 1950s. In addition to Wilcoxon, Mann, and Whitney, credit is due to Festinger (1946), Whitfield (1947), Haldane and Smith (1947), and Van der Reyden (1952). Leon Festinger (1919-1989), John Burdon Sanderson Haldane (1892-1964), and Cedric Austen Bardell Smith (1917-2002) are well known for other work, but little seems to be known about Whitfield or van der Reyden. For a detailed study, including information on these researchers, see Berry, Mielke, and Johnston (2012).

## median

The median test examines whether it is likely that two or more samples came from populations with the same median. The null hypothesis is that the samples were drawn from populations with the same median. The alternative hypothesis is that at least one sample was drawn from a population with a different median. The test should be used only with ordinal or interval data.

Assume that there are score values for $k$ independent samples to be compared. The median test is performed by first computing the median score for all observations combined, regardless of the sample group. Each score is compared with this computed grand median and is classified as being above the grand median, below the grand median, or equal to the grand median. Observations with scores equal to the grand median can be dropped, added to the "above" group, added to the "below" group, or split between the two groups.

Once all observations are classified, the data are cast into a $2 \times k$ contingency table, and a Pearson's $\chi^{2}$ test or Fisher's exact test is performed.

Henry Berthold Mann (1905-2000) was born in Vienna, Austria, where he completed a doctorate in algebraic number theory. He moved to the United States in 1938 and for several years made his livelihood by tutoring in New York. During this time, he proved a celebrated conjecture in number theory and studied statistics at Columbia with Abraham Wald, with whom he wrote three papers. After the war, he taught at Ohio State and the Universities of Wisconsin and Arizona. In addition to his work in number theory and statistics, he made major contributions to algebra and combinatorics.

Donald Ransom Whitney (1915-2007) studied at Oberlin, Princeton, and Ohio State Universities and worked at the latter throughout his career. His PhD thesis under Henry Mann was on nonparametric statistics. It was this work that produced the test that bears their names.

## References

Acock, A. C. 2023. A Gentle Introduction to Stata. Rev. 6th ed. College Station, TX: Stata Press.
Berry, K. J., P. W. Mielke, Jr., and J. E. Johnston. 2012. The two-sample rank-sum test: Early development. Electronic Journal for History of Probability and Statistics 8: 1-26.
https://www.jehps.net/decembre2012/BerryMielkeJohnston.pdf.
Bland, M. 2015. An Introduction to Medical Statistics. 4th ed. Oxford: Oxford University Press.
Conroy, R. M. 2012. What hypotheses do "nonparametric" two-group tests actually test? Stata Journal 12: 182-190.
Feiveson, A. H. 2002. Power by simulation. Stata Journal 2: 107-124.
Festinger, L. 1946. The significance of difference between means without reference to the frequency distribution function. Psychometrika 11: 97-105. https://doi.org/10.1007/BF02288926.
Fisher, R. A. 1935. The Design of Experiments. Edinburgh: Oliver and Boyd.
Haldane, J. B. S., and C. A. B. Smith. 1947. A simple exact test for birth-order effect. Annals of Human Genetics 14: 117-124. https://doi.org/10.1111/j.1469-1809.1947.tb02383.x.

Harris, T., and J. W. Hardin. 2013. Exact Wilcoxon signed-rank and Wilcoxon Mann-Whitney ranksum tests. Stata Journal 13: 337-343.
Hill, I. D., and R. Peto. 1971. Algorithm AS 35: Probabilities derived from finite populations. Journal of the Royal Statistical Society, Series C 20: 99-105. https://doi.org/10.2307/2346642.
Kruskal, W. H. 1957. Historical notes on the Wilcoxon unpaired two-sample test. Journal of the American Statistical Association 52: 356-360. https://doi.org/10.2307/2280906.
Mann, H. B., and D. R. Whitney. 1947. On a test of whether one of two random variables is stochastically larger than the other. Annals of Mathematical Statistics 18: 50-60. https://doi.org/10.1214/aoms/1177730491.

Snedecor, G. W., and W. G. Cochran. 1989. Statistical Methods. 8th ed. Ames, IA: Iowa State University Press.
Sprent, P., and N. C. Smeeton. 2007. Applied Nonparametric Statistical Methods. 4th ed. Boca Raton, FL: Chapman and Hall/CRC.

Van der Reyden, D. 1952. A simple statistical significance test. Rhodesia Agricultural Journal 49: 96-104.
Whitfield, J. W. 1947. Rank correlation between two variables, one of which is ranked, the other dichotomous. Biometrika 34: 292-296. https://doi.org/10.2307/2332439.

Wilcoxon, F. 1945. Individual comparisons by ranking methods. Biometrics 1: 80-83. https://doi.org/10.2307/3001968.
Wilcoxon, F., S. K. Katti, and R. A. Wilcox. 1970. Critical values and probability levels for the Wilcoxon rank sum test and the Wilcoxon signed rank test. In Selected Tables in Mathematical Statistics, ed. I. of Mathematical Statistics, vol. 1, 171-259. Providence, RI: American Mathematical Society.

## Also see

[R] signrank - Equality tests on matched data
$[\mathrm{R}]$ ttest $-t$ tests (mean-comparison tests)

## Title

ratio - Estimate ratios

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

ratio produces estimates of ratios, along with standard errors.

## Quick start

Estimate, standard error, and $95 \%$ confidence interval for the ratio of v 1 to v 2 ratio v1/v2

With bootstrap standard errors

```
ratio v1/v2, vce(bootstrap)
```

Ratios of v1 to v2 and v3 to v2
ratio (v1/v2) (v3/v2)
Same as above, but name the ratios ratio1 and ratio2
ratio (ratio1: v1/v2) (ratio2: v3/v2)
Test that ratio1 is equal to ratio2
test ratio1 = ratio2
Ratio of v1 to v2 over strata defined by levels of svar
ratio v1/v2, over(svar)
Direct standardization across categories cvar, weighting by standardization weight wvar ratio v1/v2, stdize(cvar) stdweight(wvar)

## Menu

Statistics $>$ Summaries, tables, and tests $>$ Summary and descriptive statistics $>$ Ratios

## Syntax

## Basic syntax

```
ratio [name:] varname [/] varname
```

Full syntax

```
ratio ([name:] varname [/] varname)
    [([name:] varname [/] varname) ...][if ][in][weight ][,options }
```

options
Description
Model
stdize (varname) stdweight(varname) nostdrescale
if/in/over
over (varlist) group over subpopulations defined by varlist

## SE/Cluster

vce (vcetype)
variable identifying strata for standardization
weight variable for standardization
do not rescale the standard weight variable

Reporting
level(\#)
noheader
nolegend
display_options
coeflegend
vcetype may be linearized, cluster clustvar, bootstrap, or jackknife
bootstrap, collect, jackknife, mi estimate, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands.
vce (bootstrap) and vce (jackknife) are not allowed with the mi estimate prefix; see [MI] mi estimate.
Weights are not allowed with the bootstrap prefix; see $[R]$ bootstrap.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
coeflegend does not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

Model
stdize (varname) specifies that the point estimates be adjusted by direct standardization across the strata identified by varname. This option requires the stdweight() option.
stdweight (varname) specifies the weight variable associated with the standard strata identified in the stdize() option. The standardization weights must be constant within the standard strata. nostdrescale prevents the standardization weights from being rescaled within the over() groups. This option requires stdize() but is ignored if the over() option is not specified.

## if/in/over

over (varlist) specifies that estimates be computed for multiple subpopulations, which are identified by the different values of the variables in varlist. Only numeric, nonnegative, integer-valued variables are allowed in over (varlist).

## SE/Cluster

vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (linearized), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
vce(linearized), the default, uses the linearized or sandwich estimator of variance.

## Reporting

level(\#); see [R] Estimation options.
noheader prevents the table header from being displayed. This option implies nolegend.
nolegend prevents the table legend identifying the ratios from being displayed.
display_options: vsquish, noemptycells, nofvlabel, fvwrap(\#), fvwrapon(style),
cformat (\% fint), and nolstretch; see [R] Estimation options.
The following option is available with ratio but is not shown in the dialog box:
coeflegend; see [R] Estimation options.

## Remarks and examples

## > Example 1

Using the fuel data from example 3 of $[\mathrm{R}]$ ttest, we estimate the ratio of mileage for the cars without the fuel treatment (mpg1) to those with the fuel treatment (mpg2).

```
. use https://www.stata-press.com/data/r18/fuel
. ratio myratio: mpg1/mpg2
Ratio estimation Number of obs = 12
    myratio: mpg1/mpg2
\begin{tabular}{r|rccc}
\hline & Ratio & \begin{tabular}{c} 
Linearized \\
std. err.
\end{tabular} & {\([95 \%\) conf. interval] } \\
\hline myratio & .9230769 & .032493 & .8515603 & .9945936
\end{tabular}
```

Using these results, we can test to see if this ratio is significantly different from one.

```
. test myratio = 1
( 1) myratio = 1
    F( 1, 11) = 5.60
        Prob > F = 0.0373
```

We find that the ratio is different from one at the $5 \%$ significance level but not at the $1 \%$ significance level.

## > Example 2

Using state-level census data, we want to test whether the marriage rate is equal to the deathrate.

```
. use https://www.stata-press.com/data/r18/census2
(1980 Census data by state)
. ratio (deathrate: death/pop) (marrate: marriage/pop)
Ratio estimation Number of obs = 50
    deathrate: death/pop
        marrate: marriage/pop
\begin{tabular}{r|rcrr}
\hline & Ratio & \begin{tabular}{c} 
Linearized \\
std. err.
\end{tabular} & [95\% conf. interval] \\
\hline deathrate & .0087368 & .0002052 & .0083244 & .0091492 \\
marrate & .0105577 & .0006184 & .009315 & .0118005 \\
\hline
\end{tabular}
```

```
. test deathrate = marrate
```

. test deathrate = marrate
( 1) deathrate - marrate = 0
( 1) deathrate - marrate = 0
F( 1, 49) = 6.93
F( 1, 49) = 6.93
Prob > F = 0.0113

```
        Prob > F = 0.0113
```


## Stored results

ratio stores the following in e():
Scalars

```
e(N) number of observations
    e(N_over) number of subpopulations
    e(N_stdize) number of standard strata
    e(N_clust) number of clusters
    e(k_eq) number of equations in e(b)
    e(df_r) sample degrees of freedom
    e(rank) rank of e(V)
```

Macros
$e(\mathrm{cmd})$
e(cmdline) command as typed
e(varlist) varlist
e(stdize) varname from stdize()
$e$ (stdweight) varname from stdweight()
e(wtype) weight type
$e$ (wexp) weight expression
$e$ (title) title in estimation output
e(clustvar) name of cluster variable
e(over)
$e$ (namelist)
varlist from over()
e(namelist) ratio identifiers
e(vce)
vcetype specified in vce()
e(vcetype) title used to label Std. err.
e(properties) b V
$e$ (estat_cmd) program used to implement estat
$e$ (marginsnotok) predictions disallowed by margins
Matrices
e(b)
e(V)
e(_N)
e(_N_stdsum)
e(_p_stdize)
e(error)
vector of ratio estimates
(co)variance estimates
vector of numbers of nonmissing observations
number of nonmissing observations within the standard strata
standardizing proportions
error code corresponding to $e(b)$
Functions
e(sample) marks estimation sample

In addition to the above, the following is stored in r() :

Matrices
$r$ (table) matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

Methods and formulas are presented under the following headings:
The ratio estimator
Survey data
The survey ratio estimator
The standardized ratio estimator
The poststratified ratio estimator
The standardized poststratified ratio estimator
Subpopulation estimation

## The ratio estimator

Let $R=Y / X$ be the ratio to be estimated, where $Y$ and $X$ are totals; see [ R$]$ total. The estimate for $R$ is $\widehat{R}=\widehat{Y} / \widehat{X}$ (the ratio of the sample totals). From the delta method (that is, a first-order Taylor expansion), the approximate variance of the sampling distribution of the linearized $\widehat{R}$ is

$$
V(\widehat{R}) \approx \frac{1}{X^{2}}\left\{V(\widehat{Y})-2 R \operatorname{Cov}(\widehat{Y}, \widehat{X})+R^{2} V(\widehat{X})\right\}
$$

Direct substitution of $\widehat{X}, \widehat{R}$, and the estimated variances and covariance of $\widehat{X}$ and $\widehat{Y}$ leads to the following variance estimator:

$$
\begin{equation*}
\widehat{V}(\widehat{R})=\frac{1}{\widehat{X}^{2}}\left\{\widehat{V}(\widehat{Y})-2 \widehat{R} \widehat{\operatorname{Cov}}(\widehat{Y}, \widehat{X})+\widehat{R}^{2} \widehat{V}(\widehat{X})\right\} \tag{1}
\end{equation*}
$$

## Survey data

See [SVY] Variance estimation, [SVY] Direct standardization, and [SVY] Poststratification for discussions that provide background information for the following formulas.

## The survey ratio estimator

Let $Y_{j}$ and $X_{j}$ be survey items for the $j$ th individual in the population, where $j=1, \ldots, M$ and $M$ is the size of the population. The associated population ratio for the items of interest is $R=Y / X$ where

$$
Y=\sum_{j=1}^{M} Y_{j} \quad \text { and } \quad X=\sum_{j=1}^{M} X_{j}
$$

Let $y_{j}$ and $x_{j}$ be the corresponding survey items for the $j$ th sampled individual from the population, where $j=1, \ldots, m$ and $m$ is the number of observations in the sample.

The estimator $\widehat{R}$ for the population ratio $R$ is $\widehat{R}=\widehat{Y} / \widehat{X}$, where

$$
\widehat{Y}=\sum_{j=1}^{m} w_{j} y_{j} \quad \text { and } \quad \widehat{X}=\sum_{j=1}^{m} w_{j} x_{j}
$$

and $w_{j}$ is a sampling weight. The score variable for the ratio estimator is

$$
z_{j}(\widehat{R})=\frac{y_{j}-\widehat{R} x_{j}}{\widehat{X}}=\frac{\widehat{X} y_{j}-\widehat{Y} x_{j}}{\widehat{X}^{2}}
$$

## The standardized ratio estimator

Let $D_{g}$ denote the set of sampled observations that belong to the $g$ th standard stratum and define $I_{D_{g}}(j)$ to indicate if the $j$ th observation is a member of the $g$ th standard stratum; where $g=1$, $\ldots, L_{D}$ and $L_{D}$ is the number of standard strata. Also, let $\pi_{g}$ denote the fraction of the population that belongs to the $g$ th standard stratum, thus $\pi_{1}+\cdots+\pi_{L_{D}}=1$. Note that $\pi_{g}$ is derived from the stdweight() option.

The estimator for the standardized ratio is

$$
\widehat{R}^{D}=\sum_{g=1}^{L_{D}} \pi_{g} \frac{\widehat{Y}_{g}}{\widehat{X}_{g}}
$$

where

$$
\widehat{Y}_{g}=\sum_{j=1}^{m} I_{D_{g}}(j) w_{j} y_{j}
$$

and $\widehat{X}_{g}$ is similarly defined. The score variable for the standardized ratio is

$$
z_{j}\left(\widehat{R}^{D}\right)=\sum_{g=1}^{L_{D}} \pi_{g} I_{D_{g}}(j) \frac{\widehat{X}_{g} y_{j}-\widehat{Y}_{g} x_{j}}{\widehat{X}_{g}^{2}}
$$

## The poststratified ratio estimator

Let $P_{k}$ denote the set of sampled observations that belong to poststratum $k$, and define $I_{P_{k}}(j)$ to indicate if the $j$ th observation is a member of poststratum $k$, where $k=1, \ldots, L_{P}$ and $L_{P}$ is the number of poststrata. Also, let $M_{k}$ denote the population size for poststratum $k . P_{k}$ and $M_{k}$ are identified by specifying the poststrata() and postweight () options on svyset; see [SVY] svyset.

The estimator for the poststratified ratio is

$$
\widehat{R}^{P}=\frac{\widehat{Y}^{P}}{\widehat{X}^{P}}
$$

where

$$
\widehat{Y}^{P}=\sum_{k=1}^{L_{P}} \frac{M_{k}}{\widehat{M}_{k}} \widehat{Y}_{k}=\sum_{k=1}^{L_{P}} \frac{M_{k}}{\widehat{M}_{k}} \sum_{j=1}^{m} I_{P_{k}}(j) w_{j} y_{j}
$$

and $\widehat{X}^{P}$ is similarly defined. The score variable for the poststratified ratio is

$$
z_{j}\left(\widehat{R}^{P}\right)=\frac{z_{j}\left(\widehat{Y}^{P}\right)-\widehat{R}^{P} z_{j}\left(\widehat{X}^{P}\right)}{\widehat{X}^{P}}=\frac{\widehat{X}^{P} z_{j}\left(\widehat{Y}^{P}\right)-\widehat{Y}^{P} z_{j}\left(\widehat{X}^{P}\right)}{\left(\widehat{X}^{P}\right)^{2}}
$$

where

$$
z_{j}\left(\widehat{Y}^{P}\right)=\sum_{k=1}^{L_{P}} I_{P_{k}}(j) \frac{M_{k}}{\widehat{M}_{k}}\left(y_{j}-\frac{\widehat{Y}_{k}}{\widehat{M}_{k}}\right)
$$

and $z_{j}\left(\widehat{X}^{P}\right)$ is similarly defined.

## The standardized poststratified ratio estimator

The estimator for the standardized poststratified ratio is

$$
\widehat{R}^{D P}=\sum_{g=1}^{L_{D}} \pi_{g} \frac{\widehat{Y}_{g}^{P}}{\widehat{X}_{g}^{P}}
$$

where

$$
\widehat{Y}_{g}^{P}=\sum_{k=1}^{L_{p}} \frac{M_{k}}{\widehat{M}_{k}} \widehat{Y}_{g, k}=\sum_{k=1}^{L_{p}} \frac{M_{k}}{\widehat{M}_{k}} \sum_{j=1}^{m} I_{D_{g}}(j) I_{P_{k}}(j) w_{j} y_{j}
$$

and $\widehat{X}_{g}^{P}$ is similarly defined. The score variable for the standardized poststratified ratio is

$$
z_{j}\left(\widehat{R}^{D P}\right)=\sum_{g=1}^{L_{D}} \pi_{g} \frac{\widehat{X}_{g}^{P} z_{j}\left(\widehat{Y}_{g}^{P}\right)-\widehat{Y}_{g}^{P} z_{j}\left(\widehat{X}_{g}^{P}\right)}{\left(\widehat{X}_{g}^{P}\right)^{2}}
$$

where

$$
z_{j}\left(\widehat{Y}_{g}^{P}\right)=\sum_{k=1}^{L_{P}} I_{P_{k}}(j) \frac{M_{k}}{\widehat{M}_{k}}\left\{I_{D_{g}}(j) y_{j}-\frac{\widehat{Y}_{g, k}}{\widehat{M}_{k}}\right\}
$$

and $z_{j}\left(\widehat{X}_{g}^{P}\right)$ is similarly defined.

## Subpopulation estimation

Let $S$ denote the set of sampled observations that belong to the subpopulation of interest, and define $I_{S}(j)$ to indicate if the $j$ th observation falls within the subpopulation.

The estimator for the subpopulation ratio is $\widehat{R}^{S}=\widehat{Y}^{S} / \widehat{X}^{S}$, where

$$
\widehat{Y}^{S}=\sum_{j=1}^{m} I_{S}(j) w_{j} y_{j} \quad \text { and } \quad \widehat{X}^{S}=\sum_{j=1}^{m} I_{S}(j) w_{j} x_{j}
$$

Its score variable is

$$
z_{j}\left(\widehat{R}^{S}\right)=I_{S}(j) \frac{y_{j}-\widehat{R}^{S} x_{j}}{\widehat{X}^{S}}=I_{S}(j) \frac{\widehat{X}^{S} y_{j}-\widehat{Y}^{S} x_{j}}{\left(\widehat{X}^{S}\right)^{2}}
$$

The estimator for the standardized subpopulation ratio is

$$
\widehat{R}^{D S}=\sum_{g=1}^{L_{D}} \pi_{g} \frac{\widehat{Y}_{g}^{S}}{\widehat{X}_{g}^{S}}
$$

where

$$
\widehat{Y}_{g}^{S}=\sum_{j=1}^{m} I_{D_{g}}(j) I_{S}(j) w_{j} y_{j}
$$

and $\widehat{X}_{g}^{S}$ is similarly defined. Its score variable is

$$
z_{j}\left(\widehat{R}^{D S}\right)=\sum_{g=1}^{L_{D}} \pi_{g} I_{D_{g}}(j) I_{S}(j) \frac{\widehat{X}_{g}^{S} y_{j}-\widehat{Y}_{g}^{S} x_{j}}{\left(\widehat{X}_{g}^{S}\right)^{2}}
$$

The estimator for the poststratified subpopulation ratio is

$$
\widehat{R}^{P S}=\frac{\widehat{Y}^{P S}}{\widehat{X}^{P S}}
$$

where

$$
\widehat{Y}^{P S}=\sum_{k=1}^{L_{P}} \frac{M_{k}}{\widehat{M}_{k}} \widehat{Y}_{k}^{S}=\sum_{k=1}^{L_{P}} \frac{M_{k}}{\widehat{M}_{k}} \sum_{j=1}^{m} I_{P_{k}}(j) I_{S}(j) w_{j} y_{j}
$$

and $\widehat{X}^{P S}$ is similarly defined. Its score variable is

$$
z_{j}\left(\widehat{R}^{P S}\right)=\frac{\widehat{X}^{P S} z_{j}\left(\widehat{Y}^{P S}\right)-\widehat{Y}^{P S} z_{j}\left(\widehat{X}^{P S}\right)}{\left(\widehat{X}^{P S}\right)^{2}}
$$

where

$$
z_{j}\left(\widehat{Y}^{P S}\right)=\sum_{k=1}^{L_{P}} I_{P_{k}}(j) \frac{M_{k}}{\widehat{M}_{k}}\left\{I_{S}(j) y_{j}-\frac{\widehat{Y}_{k}^{S}}{\widehat{M}_{k}}\right\}
$$

and $z_{j}\left(\widehat{X}^{P S}\right)$ is similarly defined.
The estimator for the standardized poststratified subpopulation ratio is

$$
\widehat{R}^{D P S}=\sum_{g=1}^{L_{D}} \pi_{g} \frac{\widehat{Y}_{g}^{P S}}{\widehat{X}_{g}^{P S}}
$$

where

$$
\widehat{Y}_{g}^{P S}=\sum_{k=1}^{L_{p}} \frac{M_{k}}{\widehat{M}_{k}} \widehat{Y}_{g, k}^{S}=\sum_{k=1}^{L_{p}} \frac{M_{k}}{\widehat{M}_{k}} \sum_{j=1}^{m} I_{D_{g}}(j) I_{P_{k}}(j) I_{S}(j) w_{j} y_{j}
$$

and $\widehat{X}_{g}^{P S}$ is similarly defined. Its score variable is

$$
z_{j}\left(\widehat{R}^{D P S}\right)=\sum_{g=1}^{L_{D}} \pi_{g} \frac{\widehat{X}_{g}^{P S} z_{j}\left(\widehat{Y}_{g}^{P S}\right)-\widehat{Y}_{g}^{P S} z_{j}\left(\widehat{X}_{g}^{P S}\right)}{\left(\widehat{X}_{g}^{P S}\right)^{2}}
$$

where

$$
z_{j}\left(\widehat{Y}_{g}^{P S}\right)=\sum_{k=1}^{L_{P}} I_{P_{k}}(j) \frac{M_{k}}{\widehat{M}_{k}}\left\{I_{D_{g}}(j) I_{S}(j) y_{j}-\frac{\widehat{Y}_{g, k}^{S}}{\widehat{M}_{k}}\right\}
$$

and $z_{j}\left(\widehat{X}_{g}^{P S}\right)$ is similarly defined.

## References

Cochran, W. G. 1977. Sampling Techniques. 3rd ed. New York: Wiley.
Stuart, A., and J. K. Ord. 1994. Kendall's Advanced Theory of Statistics: Distribution Theory, Vol. 1. 6th ed. London: Arnold.

## Also see

$[R]$ ratio postestimation - Postestimation tools for ratio
[R] mean - Estimate means
[R] proportion - Estimate proportions
[R] total - Estimate totals
[MI] Estimation - Estimation commands for use with mi estimate
[SVY] Direct standardization - Direct standardization of means, proportions, and ratios
[SVY] Poststratification - Poststratification for survey data
[SVY] Subpopulation estimation - Subpopulation estimation for survey data
[SVY] svy estimation - Estimation commands for survey data
[SVY] Variance estimation - Variance estimation for survey data
[U] 20 Estimation and postestimation commands

## Title

ratio postestimation - Postestimation tools for ratio

Postestimation commands Remarks and examples Also see

## Postestimation commands

The following postestimation commands are available after ratio:

| Command | Description |
| :--- | :--- |
| contrast | contrasts and ANOVA-style joint tests of estimates |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estat (svy) | postestimation statistics for survey data <br> estimates <br> etable |
| lincoming estimation results |  |
| table of estimation results |  |$\quad$| point estimates, standard errors, testing, and inference for linear combinations of |
| :--- |
| coefficients |

## Remarks and examples

For examples of the use of test after ratio, see [ R ] ratio.

## Also see

$[\mathrm{R}]$ ratio - Estimate ratios
[U] 20 Estimation and postestimation commands

## Title

reg3 - Three-stage estimation for systems of simultaneous equations

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

reg3 estimates a system of structural equations, where some equations contain endogenous variables among the explanatory variables. Estimation is via three-stage least squares (3SLS); see Zellner and Theil (1962). Typically, the endogenous explanatory variables are dependent variables from other equations in the system. reg3 supports iterated GLS estimation and linear constraints.
reg3 can also estimate systems of equations by seemingly unrelated regression estimation (SURE), multivariate regression (MVREG), and equation-by-equation ordinary least squares (OLS) or two-stage least squares (2SLS).

## Nomenclature

Under 3SLS or 2SLS estimation, a structural equation is defined as one of the equations specified in the system. A dependent variable will have its usual interpretation as the left-hand-side variable in an equation with an associated disturbance term. All dependent variables are explicitly taken to be endogenous to the system and are treated as correlated with the disturbances in the system's equations. Unless specified in an endog() option, all other variables in the system are treated as exogenous to the system and uncorrelated with the disturbances. The exogenous variables are taken to be instruments for the endogenous variables.

## Quick start

System of equations regressing y1 on $\mathrm{x} 1, \mathrm{x} 2$, and y 2 and y 2 on $\mathrm{x} 1, \mathrm{x} 3$, and y 1

```
reg3 (y1 x1 x2 y2) (y2 x1 x3 y1)
```

Same as above, but name equations eq1 and eq2
reg3 (eq1: y1 x1 x2 y2) (eq2: y2 x1 x3 y1)
Same as above, but specify that x 1 is endogenous and add instrumental variable v 1

```
reg3 (eq1: y1 x1 x2 y2) (eq2: y2 x1 x3 y1), endog(x1) exog(v1)
```

Same as above, but with iterated estimation

```
reg3 (eq1: y1 x1 x2 y2) (eq2: y2 x1 x3 y1), endog(x1) exog(v1) ireg3
```


## Menu

Statistics $>$ Endogenous covariates $>$ Three-stage least squares

## Syntax

Basic syntax


Full syntax

```
reg3 ( eqname \(\left._{1}:\right]\) depvar \(_{1 a}\left[\right.\) depvar \(\left._{1 b} \ldots=\right]\) varlist \(_{1}[\), noconstant \(\left.]\right)\)
    ([eqname \(\left.{ }_{2}:\right]\) depvar \(_{2 a}\left[\right.\) depvar \(\left._{2 b} \ldots=\right]\) varlist \(_{2}[\), noconstant \(\left.]\right)\)
    \(\left(\left[\right.\right.\) eqname \(\left._{N}:\right]\) depvar \(_{N a}\left[\right.\) depvar \(\left._{N b} \ldots=\right]\) varlist \(_{N}[\), noconstant \(\left.]\right)\)
    \([\) if \(][\) in \(][\) weight \(][\), options \(]\)
```

options
Description
Model
ireg3 iterate until estimates converge
3sls three-stage least squares; the default
2sls two-stage least squares
ols ordinary least squares (OLS)
sure
mvreg
corr (correlation)
exog (varlist)
endog (varlist)
inst (varlist)
allexog
noconstant
constraints (constraints)
seemingly unrelated regression estimation (SURE)
sure with OLS degrees-of-freedom adjustment
unstructured or independent correlation structure; default is unstructured
exogenous variables not specified in system equations
additional right-hand-side endogenous variables
full list of exogenous variables
all right-hand-side variables are exogenous
suppress constant from instrument list
apply specified linear constraints
SE/Robust
vce(vcetype)
vcetype may be unadjusted, robust, or cluster clustvar
df adj.
small
dfk
dfk2
report small-sample statistics
use small-sample adjustment
use alternate adjustment
Reporting
level(\#)
first
nocnsreport
display_options
set confidence level; default is level (95)
report first-stage regression
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

Optimization
optimization_options
noheader
notable
nofooter
coeflegend
control the optimization process; seldom used
suppress display of header
suppress display of coefficient table
suppress display of footer
display legend instead of statistics
varlist $_{1}, \ldots$, varlist $_{N}$ and the $\operatorname{exog}()$ and the inst() varlist may contain factor variables; see [U] 11.4.3 Factor variables. You must have the same levels of factor variables in all equations that have factor variables.
depvar and varlist may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bootstrap, by, collect, fp, jackknife, rolling, and statsby are allowed; see [U] 11.1.10 Prefix commands.
Weights are not allowed with the bootstrap prefix; see [R] bootstrap.
aweights are not allowed with the jackknife prefix; see [R] jackknife.
aweights and fweights are allowed; see [U] 11.1.6 weight.
noheader, notable, nofooter, and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.
Explicit equation naming (eqname:) cannot be combined with multiple dependent variables in an equation specification.

## Options

## Model

ireg3 causes reg3 to iterate over the estimated disturbance covariance matrix and parameter estimates until the parameter estimates converge. Although the iteration is usually successful, there is no guarantee that it will converge to a stable point. Under SURE, this iteration converges to the maximum likelihood estimates.

3sls specifies the full 3SLS estimation of the system and is the default for reg3.
2sls causes reg3 to perform equation-by-equation 2SLS on the full system of equations. This option implies dfk, small, and corr(independent).
Cross-equation testing should not be performed after estimation with this option. With 2 sls , no covariance is estimated between the parameters of the equations. For cross-equation testing, use 3sls.
ols causes reg3 to perform equation-by-equation OLS on the system-even if dependent variables appear as regressors or the regressors differ for each equation; see [MV] mvreg. ols implies allexog, dfk, small, and corr (independent); nodfk and nosmall may be specified to override dfk and small.

The covariance of the coefficients between equations is not estimated under this option, and cross-equation tests should not be performed after estimation with ols. For cross-equation testing, use sure or 3sls (the default).
sure causes reg3 to perform a SURE of the system-even if dependent variables from some equations appear as regressors in other equations; see $[R]$ sureg. sure is a synonym for allexog.
mvreg is identical to sure, except that the disturbance covariance matrix is estimated with an OLS degrees-of-freedom adjustment-the dfk option. If the regressors are identical for all equations, the parameter point estimates will be the standard MVREG results. If any of the regressors differ, the point estimates are those for SURE with an OLS degrees-of-freedom adjustment in computing the covariance matrix. nodfk and nosmall may be specified to override dfk and small.
corr (correlation) specifies the assumed form of the correlation structure of the equation disturbances and is rarely requested explicitly. For the family of models fit by reg3, the only two allowable correlation structures are unstructured and independent. The default is unstructured.
This option is used almost exclusively to estimate a system of equations by 2SLS or to perform OLS regression with reg3 on multiple equations. In these cases, the correlation is set to independent, forcing reg3 to treat the covariance matrix of equation disturbances as diagonal in estimating model parameters. Thus, a set of two-stage coefficient estimates can be obtained if the system contains endogenous right-hand-side variables, or OLS regression can be imposed, even if the regressors differ across equations. Without imposing independent disturbances, reg3 would estimate the former by 3SLS and the latter by SURE.
Any tests performed after estimation with the independent option will treat coefficients in different equations as having no covariance; cross-equation tests should not be used after specifying corr (independent).
exog (varlist) specifies additional exogenous variables that are included in none of the system equations.
This can occur when the system contains identities that are not estimated. If implicitly exogenous variables from the equations are listed here, reg3 will just ignore the additional information. Specified variables will be added to the exogenous variables in the system and used in the first stage as instruments for the endogenous variables. By specifying dependent variables from the structural equations, you can use $\operatorname{exog}()$ to override their endogeneity.
endog (varlist) identifies variables in the system that are not dependent variables but are endogenous to the system. These variables must appear in the variable list of at least one equation in the system. Again, the need for this identification often occurs when the system contains identities. For example, a variable that is the sum of an exogenous variable and a dependent variable may appear as an explanatory variable in some equations.
inst (varlist) specifies a full list of all exogenous variables and may not be used with the endog() or $\operatorname{exog}()$ options. It must contain a full list of variables to be used as instruments for the endogenous regressors. Like exog(), the list may contain variables not specified in the system of equations. This option can be used to achieve the same results as the endog() and exog() options, and the choice is a matter of convenience. Any variable not specified in the varlist of the inst () option is assumed to be endogenous to the system. As with $\operatorname{exog}()$, including the dependent variables from the structural equations will override their endogeneity.
allexog indicates that all right-hand-side variables are to be treated as exogenous-even if they appear as the dependent variable of another equation in the system. This option can be used to enforce a SURE or MVREG estimation even when some dependent variables appear as regressors.
noconstant, constraints (constraints); see [R] Estimation options.

SE/Robust
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (unadjusted), that are robust to some kinds of misspecification (robust), and that allow for intragroup correlation (cluster clustvar); see [R] vce_option.
vce (unadjusted), the default, specifies that an unadjusted (nonrobust) VCE matrix be used; this results in efficient estimates when assuming homoskedasticity.
df adj.
small specifies that small-sample statistics be computed. It shifts the test statistics from $\chi^{2}$ and $z$ statistics to $F$ statistics and $t$ statistics. This option is intended primarily to support MVREG. Although the standard errors from each equation are computed using the degrees of freedom for
the equation, the degrees of freedom for the $t$ statistics are all taken to be those for the first equation. This approach poses no problem under MVREG because the regressors are the same across equations.
dfk specifies the use of an alternative divisor in computing the covariance matrix for the equation residuals. As an asymptotically justified estimator, reg3 by default uses the number of sample observations $n$ as a divisor. When the dfk option is set, a small-sample adjustment is made, and the divisor is taken to be $\sqrt{\left(n-k_{i}\right)\left(n-k_{j}\right)}$, where $k_{i}$ and $k_{j}$ are the number of parameters in equations $i$ and $j$, respectively.
dfk2 specifies the use of an alternative divisor in computing the covariance matrix for the equation errors. When the dfk2 option is set, the divisor is taken to be the mean of the residual degrees of freedom from the individual equations.

Reporting
level (\#); see [R] Estimation options.
first requests that the first-stage regression results be displayed during estimation.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap (\#), fvwrapon (style), cformat ( $\% \overline{f m t}$ ), pformat ( $\%$ fint), sformat ( $\% f m t$ ), and nolstretch; see [R] Estimation options.
$\sqrt{\text { Optimization }}$
optimization_options control the iterative process that minimizes the sum of squared errors when ireg3 is specified. These options are seldom used.
iterate(\#) specifies the maximum number of iterations. When the number of iterations equals \#, the optimizer stops and presents the current results, even if the convergence tolerance has not been reached. The default is the number set using set maxiter, which is 300 by default.
trace adds to the iteration $\log$ a display of the current parameter vector.
log and nolog specify whether to display the iteration log. The iteration log is displayed by default unless you used set iterlog off to suppress it; see set iterlog in [R] set iter.
tolerance (\#) specifies the tolerance for the coefficient vector. When the relative change in the coefficient vector from one iteration to the next is less than or equal to \#, the optimization process is stopped. tolerance $(1 e-6)$ is the default.

The following options are available with reg3 but are not shown in the dialog box:
noheader suppresses display of the header reporting the estimation method and the table of equation summary statistics.
notable suppresses display of the coefficient table.
nofooter suppresses display of the footer reporting the list of endogenous and exogenous variables in the model.
coeflegend; see [R] Estimation options.

## Remarks and examples

reg3 estimates systems of structural equations where some equations contain endogenous variables among the explanatory variables. Generally, these endogenous variables are the dependent variables of other equations in the system, though not always. The disturbance is correlated with the endogenous variables-violating the assumptions of OLS. Further, because some of the explanatory variables are the dependent variables of other equations in the system, the error terms among the equations are expected to be correlated. reg3 uses an instrumental-variables approach to produce consistent estimates and generalized least squares (GLS) to account for the correlation structure in the disturbances across the equations. Good general references on three-stage estimation include Davidson and MacKinnon (1993, 651-661) and Greene (2018, 363-365).

Three-stage least squares can be thought of as producing estimates from a three-step process.
Step 1. Develop instrumented values for all endogenous variables. These instrumented values can simply be considered as the predicted values resulting from a regression of each endogenous variable on all exogenous variables in the system. This stage is identical to the first step in 2SLS and is critical for the consistency of the parameter estimates.

Step 2. Obtain a consistent estimate for the covariance matrix of the equation disturbances. These estimates are based on the residuals from a 2SLS estimation of each structural equation.
Step 3. Perform a GLS-type estimation using the covariance matrix estimated in the second stage and with the instrumented values in place of the right-hand-side endogenous variables.

## - Technical note

The estimation and use of the covariance matrix of disturbances in three-stage estimation is almost identical to the SURE method-sureg. As with SURE, using this covariance matrix improves the efficiency of the three-stage estimator. Even without the covariance matrix, the estimates would be consistent. (They would be 2SLS estimates.) This improvement in efficiency comes with a caveat. All the parameter estimates now depend on the consistency of the covariance matrix estimates. If one equation in the system is misspecified, the disturbance covariance estimates will be inconsistent, and the resulting coefficients will be biased and inconsistent. Alternatively, if each equation is estimated separately by 2 SLS ( $[R]$ regress), only the coefficients in the misspecified equation are affected.

## Technical note

If an equation is just identified, the 3SLS point estimates for that equation are identical to the 2SLS estimates. However, as with sureg, even if all equations are just identified, fitting the model via reg3 has at least one advantage over fitting each equation separately via ivregress; by using reg3, tests involving coefficients in different equations can be performed easily using test or testnl.

## > Example 1

A simple macroeconomic model relates consumption (consump) to private and government wages paid (wagepriv and wagegovt). Simultaneously, private wages depend on consumption, total government expenditures (govt), and the lagged stock of capital in the economy (capital1). Although this is not a plausible model, it does meet the criterion of being simple. This model could be written as

$$
\begin{aligned}
\text { consump } & =\beta_{0}+\beta_{1} \text { wagepriv }+\beta_{2} \text { wagegovt }+\epsilon_{1} \\
\text { wagepriv } & =\beta_{3}+\beta_{4} \text { consump }+\beta_{5} \text { govt }+\beta_{6} \text { capital1 }+\epsilon_{2}
\end{aligned}
$$

If we assume that this is the full system, consump and wagepriv will be endogenous variables, with wagegovt, govt, and capital1 exogenous. Data for the US economy on these variables are taken from Klein (1950). This model can be fit with reg3 by typing

```
. use https://www.stata-press.com/data/r18/klein
. reg3 (consump wagepriv wagegovt) (wagepriv consump govt capital1)
Three-stage least-squares regression
```

| Equation | Obs | Params | RMSE | "R-squared" | chi2 | P>chi2 |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: |
| consump | 22 | 2 | 1.776297 | 0.9388 | 208.02 | 0.0000 |
| wagepriv | 22 | 3 | 2.372443 | 0.8542 | 80.04 | 0.0000 |


|  | Coefficient | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :--- | ---: | ---: | ---: | ---: |
| consump |  |  |  |  |  |  |
| wagepriv | .8012754 | .1279329 | 6.26 | 0.000 | .5505314 | 1.052019 |
| wagegovt | 1.029531 | .3048424 | 3.38 | 0.001 | .432051 | 1.627011 |
| _cons | 19.3559 | 3.583772 | 5.40 | 0.000 | 12.33184 | 26.37996 |
| wagepriv |  |  |  |  |  |  |
| consump | .4026076 | .2567312 | 1.57 | 0.117 | -.1005764 | .9057916 |
| govt | 1.177792 | .5421253 | 2.17 | 0.030 | .1152461 | 2.240338 |
| capital1 | -.0281145 | .0572111 | -0.49 | 0.623 | -.1402462 | .0840173 |
| _cons | 14.63026 | 10.26693 | 1.42 | 0.154 | -5.492552 | 34.75306 |

Endogenous: consump wagepriv
Exogenous: wagegovt govt capital1
Without showing the 2SLS results, we note that the consumption function in this system falls under the conditions noted earlier. That is, the 2SLS and 3SLS coefficients for the equation are identical.
> Example 2
Some of the most common simultaneous systems encountered are supply-and-demand models. A simple system could be specified as

$$
\begin{aligned}
& \text { qDemand }=\beta_{0}+\beta_{1} \text { price }+\beta_{2} \text { pcompete }+\beta_{3} \text { income }+\epsilon_{1} \\
& \text { qSupply }=\beta_{4}+\beta_{5} \text { price }+\beta_{6} \text { praw }+\epsilon_{2} \\
& \text { Equilibrium condition: quantity }=\text { qDemand }=\text { qSupply }
\end{aligned}
$$

where

> quantity is the quantity of a product produced and sold,
price is the price of the product,
pcompete is the price of a competing product,
income is the average income level of consumers, and
praw is the price of raw materials used to produce the product.
In this system, price is assumed to be determined simultaneously with demand. The important statistical implications are that price is not a predetermined variable and that it is correlated with the disturbances of both equations. The system is somewhat unusual: quantity is associated with two disturbances. This fact really poses no problem because the disturbances are specified on the behavioral demand and supply equations-two separate entities. Often one of the two equations is rewritten to place price on the left-hand side, making this endogeneity explicit in the specification.

To provide a concrete illustration of the effects of simultaneous equations, we can simulate data for the above system by using known coefficients and disturbance properties. Specifically, we will simulate the data as

$$
\begin{aligned}
& \text { qDemand }=40-1.0 \text { price }+0.25 \text { pcompete }+0.5 \text { income }+\epsilon_{1} \\
& \text { qSupply }=0.5 \text { price }-0.75 \text { praw }+\epsilon_{2}
\end{aligned}
$$

where

$$
\begin{aligned}
& \epsilon_{1} \sim N(0,3.8) \\
& \epsilon_{2} \sim N(0,2.4)
\end{aligned}
$$

For comparison, we can estimate the supply and demand equations separately by OLS. The estimates for the demand equation are
. use https://www.stata-press.com/data/r18/supDem
. regress quantity price pcompete income

| Source | SS | df | MS | Number of obs | = | 49 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | F (3, 45) | = | 1.00 |
| Model | 23.1579302 | 3 | 7.71931008 | Prob > F | = | 0.4004 |
| Residual | 346.459313 | 45 | 7.69909584 | R -squared | = | 0.0627 |
|  |  |  |  | Adj R-squared | = | 0.0002 |
| Total | 369.617243 | 48 | 7.70035923 | Root MSE | = | 2.7747 |


| quantity | Coefficient | Std. err. | t | $\mathrm{P}>\mid \mathrm{tI}$ | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | ---: | ---: |
| price | .1186265 | .1716014 | 0.69 | 0.493 | -.2269965 | .4642496 |
| pcompete | .0946416 | .1200815 | 0.79 | 0.435 | -.1472149 | .3364981 |
| income | .0785339 | .1159867 | 0.68 | 0.502 | -.1550754 | .3121432 |
| _cons | 7.563261 | 5.019479 | 1.51 | 0.139 | -2.54649 | 17.67301 |

The OLS estimates for the supply equation are


Examining the coefficients from these regressions, we note that they are not close to the known parameters used to generate the simulated data. In particular, the positive coefficient on price in the demand equation stands out. We constructed our simulated data to be consistent with economic theory-people demand less of a product if its price rises and more if their personal income rises. Although the price coefficient is statistically insignificant, the positive value contrasts starkly with what is predicted from economic price theory and the -1.0 value that we used in the simulation. Likewise, we are disappointed with the insignificance and level of the coefficient on average income. The supply equation has correct signs on the two main parameters, but their levels are different from the known values. In fact, the coefficient on price (0.724675) is different from the simulated parameter (0.5) at the $5 \%$ level of significance.

All of these problems are to be expected. We explicitly constructed a simultaneous system of equations that violated one of the assumptions of least squares. Specifically, the disturbances were correlated with one of the regressors-price.

Two-stage least squares can be used to address the correlation between regressors and disturbances. Using instruments for the endogenous variable, price, 2SLS will produce consistent estimates of the parameters in the system. Let's use ivregress (see [R] ivregress) to see how our simulated system behaves when fit using 2SLS.
. ivregress 2sls quantity (price = praw) pcompete income
Instrumental variables 2SLS regression

Endogenous: price
Exogenous: pcompete income praw
. ivregress 2sls quantity (price = pcompete income) praw

| Instrumental variables 2SLS regression | Number of obs | $=$ |
| :--- | :--- | :--- |
| Wald chi2(2) | $=$ | 39.25 |
|  | Prob > chi2 | $=$ |
|  | R-squared | $=$ |
|  | Root MSE | $=$ |


| quantity | Coefficient | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| price | .5773133 | .1749974 | 3.30 | 0.001 | .2343247 | .9203019 |
| praw | -.7835496 | .1312414 | -5.97 | 0.000 | -1.040778 | -.5263213 |
| _cons | -2.550694 | 5.273067 | -0.48 | 0.629 | -12.88571 | 7.784327 |

Endogenous: price
Exogenous: praw pcompete income
We are now much happier with the estimation results. All the coefficients from both equations are close to the true parameter values for the system. In particular, the coefficients are all well within $95 \%$ confidence intervals for the parameters. The missing $R^{2}$ in the demand equation seems unusual; we will discuss that more later.

Finally, this system could be estimated using 3SLS. To demonstrate how large systems might be handled and to avoid multiline commands, we will use global macros (see [P] macro) to hold the specifications for our equations.
. global demand "(qDemand: quantity price pcompete income)"
. global supply "(qSupply: quantity price praw)"
. reg3 \$demand \$supply, endog(price)
We must specify price as endogenous because it does not appear as a dependent variable in either equation. Without this option, reg3 would assume that there are no endogenous variables in the system and produce seemingly unrelated regression (sureg) estimates. The reg3 output from our series of commands is

Three-stage least-squares regression

| Equation | Obs | Params | RMSE | "R-squared" |  | P>chi2 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| qDemand | 49 | 3 | 3.739686 | -0.8540 |  | 0.0338 |
| qSupply | 49 | 2 | 1.752501 | $0.5928 \quad 39.25$ |  | 0.0000 |
|  | Coefficient | Std. err | z | $P>\|z\|$ | [95\% conf. interval] |  |
| qDemand |  |  |  |  |  |  |
| price | -1.014345 | . 3742036 | -2.71 | 0.007 | -1.74777 | -. 2809194 |
| pcompete | . 2647206 | . 1464194 | 1.81 | 0.071 | -. 0222561 | . 5516973 |
| income | . 5299146 | . 1898161 | 2.79 | 0.005 | . 1578819 | . 9019472 |
| _cons | 40.08749 | 10.77072 | 3.72 | 0.000 | 18.97726 | 61.19772 |
| qSupply |  |  |  |  |  |  |
| price | . 5773133 | . 1749974 | 3.30 | 0.001 | . 2343247 | . 9203019 |
| praw | -. 7835496 | . 1312414 | -5.97 | 0.000 | -1.040778 | -. 5263213 |
| _cons | -2.550694 | 5.273067 | -0.48 | 0.629 | -12.88571 | 7.784327 |

Endogenous: quantity price
Exogenous: pcompete income praw
The use of 3SLS over 2SLS is essentially an efficiency issue. The coefficients of the demand equation from 3SLS are close to the coefficients from two-stage least squares, and those of the supply equation are identical. The latter case was mentioned earlier for systems with some exactly identified equations. However, even for the demand equation, we do not expect the coefficients to change systematically. What we do expect from three-stage least squares are more precise estimates of the parameters given the validity of our specification and reg3's use of the covariances among the disturbances.

Let's summarize the results. With OLS, we got obviously biased estimates of the parameters. No amount of data would have improved the OLS estimates-they are inconsistent in the face of the violated OLS assumptions. With 2SLS, we obtained consistent estimates of the parameters, and these would have improved with more data. With 3SLS, we obtained consistent estimates of the parameters that are more efficient than those obtained by 2SLS.

## Technical note

We noted earlier that the $R^{2}$ was missing from the two-stage estimates of the demand equation. Now we see that the $R^{2}$ is negative for the three-stage estimates of the same equation. How can we have a negative $R^{2}$ ?

In most estimators, other than least squares, the $R^{2}$ is no more than a summary measure of the overall in-sample predictive power of the estimator. The computational formula for $R^{2}$ is $R^{2}=1-\mathrm{RSS} / \mathrm{TSS}$, where RSS is the residual sum of squares (sum of squared residuals) and TSS is the total sum of squared deviations about the mean of the dependent variable. In a standard linear model with a constant, the model from which the TSS is computed is nested within the full model from which RSS is computed-they both have a constant term based on the same data. Thus, it must be that TSS $\geq$ RSS and $R^{2}$ is constrained between 0 and 1 .

For 2SLS and 3SLS, some of the regressors enter the model as instruments when the parameters are estimated. However, because our goal is to fit the structural model, the actual values, not the instruments for the endogenous right-hand-side variables, are used to determine $R^{2}$. The model residuals are computed over a different set of regressors from those used to fit the model. The two-
or three-stage estimates are no longer nested within a constant-only model of the dependent variable, and the residual sum of squares is no longer constrained to be smaller than the total sum of squares.

A negative $R^{2}$ in 3SLS should be taken for exactly what it is-an indication that the structural model predicts the dependent variable worse than a constant-only model. Is this a problem? It depends on the application. Three-stage least squares applied to our contrived supply-and-demand example produced good estimates of the known true parameters. Still, the demand equation produced an $R^{2}$ of -0.854 . How do we feel about our parameter estimates? This should be determined by the estimates themselves, their associated standard errors, and the overall model significance. On this basis, negative $R^{2}$ and all, we feel pretty good about all the parameter estimates for both the supply and demand equations. Would we want to make predictions about equilibrium quantity by using the demand equation alone? Probably not. Would we want to make these quantity predictions by using the supply equation? Possibly, because based on in-sample predictions, they seem better than those from the demand equations. However, both the supply and demand estimates are based on limited information. If we are interested in predicting quantity, a reduced-form equation containing all our independent variables would usually be preferred.

## Technical note

As a matter of syntax, we could have specified the supply-and-demand model on one line without using global macros.

- reg3 (quantity price pcompete income) (quantity price praw), endog(price)
Three-stage least-squares regression


Endogenous: quantity price
Exogenous: pcompete income praw
However, here reg3 has been forced to create a unique equation name for the supply equation2quantity. Both the supply and demand equations could not be designated as quantity, so a number was prefixed to the name for the supply equation.

We could have specified

```
. reg3 (qDemand: quantity price pcompete income) (qSupply: quantity price praw),
> endog(price)
```

and obtained the same results and equation labeling as when we used global macros to hold the equation specifications.

Without explicit equation names, reg3 always assumes that the dependent variable should be used to name equations. When each equation has a different dependent variable, this rule causes no problems and produces easily interpreted result tables. If the same dependent variable appears in more than one equation, however, reg3 will create a unique equation name based on the dependent variable name. Because equation names must be used for cross-equation tests, you have more control in this situation if explicit names are placed on the equations.
> Example 3: Using the full syntax of reg3
Klein's (1950) model of the US economy is often used to demonstrate system estimators. It contains several common features that will serve to demonstrate the full syntax of reg3. The Klein model is defined by the following seven relationships:

$$
\begin{align*}
\mathrm{c} & =\beta_{0}+\beta_{1} \mathrm{p}+\beta_{2} \mathrm{~L} \cdot \mathrm{p}+\beta_{3} \mathrm{w}+\epsilon_{1}  \tag{1}\\
\mathrm{i} & =\beta_{4}+\beta_{5} \mathrm{p}+\beta_{6} \mathrm{~L} \cdot \mathrm{p}+\beta_{7} \mathrm{~L} \cdot \mathrm{k}+\epsilon_{2}  \tag{2}\\
\mathrm{wp} & =\beta_{8}+\beta_{9} \mathrm{y}+\beta_{10} \mathrm{~L} \cdot \mathrm{y}+\beta_{11} \mathrm{yr}+\epsilon_{3}  \tag{3}\\
\mathrm{y} & =\mathrm{c}+\mathrm{i}+\mathrm{g}  \tag{4}\\
\mathrm{p} & =\mathrm{y}-\mathrm{t}-\mathrm{wp}  \tag{5}\\
\mathrm{k} & =\mathrm{L} \cdot \mathrm{k}+\mathrm{i}  \tag{6}\\
\mathrm{w} & =\mathrm{wg}+\mathrm{wp} \tag{7}
\end{align*}
$$

Here we have used Stata's lag operator L. to represent variables that appear with a one-period lag in our model; see [U] 13.10 Time-series operators.

The variables in the model are listed below. Two sets of variable names are shown. The concise first name uses traditional economics mnemonics, whereas the second name provides more guidance for everyone else. The concise names serve to keep the specification of the model small (and quite understandable to economists).

| Short name | Long name | Variable definition | Type |
| :--- | :--- | :--- | :--- |
| c | consump | Consumption | endogenous |
| p | profits | Private industry profits | endogenous |
| wp | wagepriv | Private wage bill | endogenous |
| wg | wagegovt | Government wage bill | exogenous |
| w | wagetot | Total wage bill | endogenous |
| i | invest | Investment | endogenous |
| k | capital | Capital stock | endogenous |
| y | totinc | Total income/demand | endogenous |
| g | govt | Government spending | exogenous |
| t | taxnetx | Indirect bus. taxes + net exports | exogenous |
| yr | year | Year-1931 | exogenous |

Equations (1)-(3) are behavioral and contain explicit disturbances ( $\epsilon_{1}, \epsilon_{2}$, and $\epsilon_{3}$ ). The remaining equations are identities that specify additional variables in the system and their accounting relationships with the variables in the behavioral equations. Some variables are explicitly endogenous by appearing as dependent variables in (1)-(3). Others are implicitly endogenous as linear combinations that contain other endogenous variables (for example, w and p). Still other variables are implicitly exogenous by appearing in the identities but not in the behavioral equations (for example, wg and g ).

Using the concise names, we can fit Klein's model with the following command:

```
. use https://www.stata-press.com/data/r18/klein2
. reg3 (c p L.p w) (i p L.p L.k) (wp y L.y yr), endog(w p y) exog(t wg g)
```

Three-stage least-squares regression

| Equation | Obs | Params | RMSE | "R-squared" | chi2 | P>chi2 |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: |
| c | 21 | 3 | .9443305 | 0.9801 | 864.59 | 0.0000 |
| i | 21 | 3 | 1.446736 | 0.8258 | 162.98 | 0.0000 |
| wp | 21 | 3 | .7211282 | 0.9863 | 1594.75 | 0.0000 |


|  | Coefficient | Std. err. | z | $P>\|z\|$ | [95\% conf | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| c |  |  |  |  |  |  |
| p |  |  |  |  |  |  |
| --. | . 1248904 | . 1081291 | 1.16 | 0.248 | -. 0870387 | . 3368194 |
| L1. | . 1631439 | . 1004382 | 1.62 | 0.104 | -. 0337113 | . 3599992 |
| w | . 790081 | . 0379379 | 20.83 | 0.000 | . 715724 | . 8644379 |
| _cons | 16.44079 | 1.304549 | 12.60 | 0.000 | 13.88392 | 18.99766 |
| i |  |  |  |  |  |  |
| p |  |  |  |  |  |  |
| --. | -. 0130791 | . 1618962 | -0.08 | 0.936 | -. 3303898 | . 3042316 |
| L1. | . 7557238 | . 1529331 | 4.94 | 0.000 | . 4559805 | 1.055467 |
| k |  |  |  |  |  |  |
| L1. | -. 1948482 | . 0325307 | -5.99 | 0.000 | -. 2586072 | -. 1310893 |
| _cons | 28.17785 | 6.793768 | 4.15 | 0.000 | 14.86231 | 41.49339 |
| wp |  |  |  |  |  |  |
| y |  |  |  |  |  |  |
| -- | . 4004919 | . 0318134 | 12.59 | 0.000 | . 3381388 | . 462845 |
| L1. | . 181291 | . 0341588 | 5.31 | 0.000 | . 1143411 | . 2482409 |
| yr | . 149674 | . 0279352 | 5.36 | 0.000 | . 094922 | . 2044261 |
| _cons | 1.797216 | 1.115854 | 1.61 | 0.107 | -. 3898181 | 3.984251 |

Endogenous: c i wp w p y
Exogenous: L.p L.k L.y yr t wg g
We used the exog() option to identify $t$, wg , and g as exogenous variables in the system. These variables must be identified because they are part of the system but appear directly in none of the behavioral equations. Without this option, reg3 would not know they were part of the system and could be used as instrumental variables. The endog() option specifying w, p, and y is also required. Without this information, reg3 would be unaware that these variables are linear combinations that include endogenous variables. We did not include k in the endog() option because only its lagged value appears in the behavioral equations.

## - Technical note

Rather than listing additional endogenous and exogenous variables, we could specify the full list of exogenous variables in an inst() option,

```
. reg3 (c p L.p w) (i p L.p L.k) (wp y L.y yr), inst(g t wg yr L.p L.k L.y)
```

or equivalently,

```
. global conseqn "(c p L.p w)"
. global inveqn "(i p L.p L.k)"
. global wageqn "(wp y L.y yr)"
. global inlist "g t wg yr L.p L.k L.y"
. reg3 $conseqn $inveqn $wageqn, inst($inlist)
```

Macros and explicit equations can also be mixed in the specification

```
. reg3 $conseqn (i p L.p L.k) $wageqn, endog(w p y) exog(t wg g)
```

or

```
. reg3 (c p L.p w) $inveqn (wp y L.y yr), endog(w p y) exog(t wg g)
```

Placing the equation-binding parentheses in the global macros was also arbitrary. We could have used

```
.global consump "c p L.p w"
. global invest "i p L.p L.k"
. global wagepriv "wp y L.y yr"
. reg3 ($consump) ($invest) ($wagepriv), endog(w p y) exog(t wg g)
```

reg3 is tolerant of all combinations, and these commands will produce identical output.

Switching to the full variable names, we can fit Klein's model with the commands below. We will use global macros to store the lists of endogenous and exogenous variables. Again, this is not necessary: these lists could have been typed directly on the command line. However, assigning the lists to local macros makes additional processing easier if alternative models are to be fit. We will also use the ireg3 option to produce the iterated estimates.

```
. use https://www.stata-press.com/data/r18/kleinfull
. global conseqn "(consump profits L.profits wagetot)"
. global inveqn "(invest profits L.profits L.capital)"
. global wageqn "(wagepriv totinc L.totinc year)"
. global enlist "wagetot profits totinc"
. global exlist "taxnetx wagegovt govt"
. reg3 $conseqn $inveqn $wageqn, endog($enlist) exog($exlist) ireg3
Iteration 1: Tolerance = . }371254
Iteration 2: Tolerance = . }189471
Iteration 3: Tolerance = . }107640
    (output omitted)
Iteration 24: Tolerance = 7.049e-07
Three-stage least-squares regression, iterated
```

| Equation | Obs | Params | RMSE | "R-squared" | chi2 | P>chi2 |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: |
| consump | 21 | 3 | .9565088 | 0.9796 | 970.31 | 0.0000 |
| invest | 21 | 3 | 2.134327 | 0.6209 | 56.78 | 0.0000 |
| wagepriv | 21 | 3 | .7782334 | 0.9840 | 1312.19 | 0.0000 |


|  | Coefficient | Std. err. | z | $P>\|z\|$ | [95\% conf | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| consump profits |  |  |  |  |  |  |
| -- | . 1645096 | . 0961979 | 1.71 | 0.087 | -. 0240348 | . 3530539 |
| L1. | . 1765639 | . 0901001 | 1.96 | 0.050 | -. 0000291 | . 3531569 |
| wagetot | . 7658011 | . 0347599 | 22.03 | 0.000 | . 6976729 | . 8339294 |
| _cons | 16.55899 | 1.224401 | 13.52 | 0.000 | 14.15921 | 18.95877 |
| invest |  |  |  |  |  |  |
| profits |  |  |  |  |  |  |
| --. | -. 3565316 | . 2601568 | -1.37 | 0.171 | -. 8664296 | . 1533664 |
| L1. | 1.011299 | . 2487745 | 4.07 | 0.000 | . 5237098 | 1.498888 |
| capital |  |  |  |  |  |  |
| L1. | -. 2602 | . 0508694 | -5.12 | 0.000 | -. 3599022 | -. 1604978 |
| _cons | 42.89629 | 10.59386 | 4.05 | 0.000 | 22.13271 | 63.65987 |
| $\begin{aligned} & \text { wagepriv } \\ & \text { totinc } \end{aligned}$ |  |  |  |  |  |  |
| -- | . 3747792 | . 0311027 | 12.05 | 0.000 | . 3138191 | . 4357394 |
| L1. | . 1936506 | . 0324018 | 5.98 | 0.000 | . 1301443 | . 257157 |
| year | . 1679262 | . 0289291 | 5.80 | 0.000 | . 1112263 | . 2246261 |
| _cons | 2.624766 | 1.195559 | 2.20 | 0.028 | . 2815124 | 4.968019 |

Endogenous: consump invest wagepriv wagetot profits totinc
Exogenous: L.profits L.capital L.totinc year taxnetx wagegovt govt

## > Example 4: Constraints with reg3

As a simple example of constraints, (1) above may be rewritten with both wages explicitly appearing (rather than as a variable containing the sum). Using the longer variable names, we have

$$
\text { consump }=\beta_{0}+\beta_{1} \text { profits }+\beta_{2} \text { L.profits }+\beta_{3} \text { wagepriv }+\beta_{12} \text { wagegovt }+\epsilon_{1}
$$

To retain the effect of the identity in (7), we need $\beta_{3}=\beta_{12}$ as a constraint on the system. We obtain this result by defining the constraint in the usual way and then specifying its use in reg3. Because reg3 is a system estimator, we will need to use the full equation syntax of constraint. The assumption that the following commands are entered after the model above has been estimated. We are simply changing the definition of the consumption equation (consump) and adding a constraint on two of its parameters. The rest of the model definition is carried forward.

```
. global conseqn "(consump profits L.profits wagepriv wagegovt)"
. constraint define 1 [consump]wagepriv = [consump]wagegovt
. reg3 $conseqn $inveqn $wageqn, endog($enlist) exog($exlist) constr(1) ireg3
note: additional endogenous variables not in the system have no effect and are
    ignored (wagetot).
Iteration 1: Tolerance = . 3712547
Iteration 2: Tolerance = . }18947
Iteration 3: Tolerance = . }1076
    (output omitted)
Iteration 24: Tolerance = 7.049e-07
```

Three-stage least-squares regression, iterated

| Equation | Obs | Params | RMSE | "R-squared" | chi2 | P>chi2 |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: |
| consump | 21 | 3 | .9565086 | 0.9796 | 970.31 | 0.0000 |
| invest | 21 | 3 | 2.134326 | 0.6209 | 56.78 | 0.0000 |
| wagepriv | 21 | 3 | .7782334 | 0.9840 | 1312.19 | 0.0000 |

( 1) [consump]wagepriv - [consump] wagegovt $=0$

|  | Coefficient | Std. err. | z | $P>\|z\|$ | [95\% con | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| consump profits |  |  |  |  |  |  |
| -- | . 1645097 | . 0961978 | 1.71 | 0.087 | -. 0240346 | . 353054 |
| L1. | . 1765639 | . 0901001 | 1.96 | 0.050 | -. 0000291 | . 3531568 |
| wagepriv | . 7658012 | . 0347599 | 22.03 | 0.000 | . 6976729 | . 8339294 |
| wagegovt | . 7658012 | . 0347599 | 22.03 | 0.000 | . 6976729 | . 8339294 |
| _cons | 16.55899 | 1.224401 | 13.52 | 0.000 | 14.1592 | 18.95877 |
| invest |  |  |  |  |  |  |
| - | -. 3565311 | . 2601567 | -1.37 | 0.171 | -. 8664288 | . 1533666 |
| L1. | 1.011298 | . 2487744 | 4.07 | 0.000 | . 5237096 | 1.498887 |
| capital |  |  |  |  |  |  |
| L1. | -. 2601999 | . 0508694 | -5.12 | 0.000 | -. 359902 | -. 1604977 |
| _cons | 42.89626 | 10.59386 | 4.05 | 0.000 | 22.13269 | 63.65984 |
| $\begin{aligned} & \text { wagepriv } \\ & \text { totinc } \end{aligned}$ |  |  |  |  |  |  |
| --. | . 3747792 | . 0311027 | 12.05 | 0.000 | . 313819 | . 4357394 |
| L1. | . 1936506 | . 0324018 | 5.98 | 0.000 | . 1301443 | . 257157 |
| year | . 1679262 | . 0289291 | 5.80 | 0.000 | . 1112263 | . 2246261 |
| _cons | 2.624766 | 1.195559 | 2.20 | 0.028 | . 281512 | 4.968019 |

Endogenous: consump invest wagepriv wagetot profits totinc
Exogenous: L.profits wagegovt L.capital L.totinc year taxnetx govt
As expected, none of the parameter or standard error estimates has changed from the previous estimates (before the seventh significant digit). We have simply decomposed the total wage variable into its two parts and constrained the coefficients on these parts. The warning about additional endogenous variables was just reg3's way of letting us know that we had specified some information that was irrelevant to the estimation of the system. We had left the wagetot variable in our endog macro. It does not mean anything to the system to specify wagetot as endogenous because it is no longer in the system. That's fine with reg3 and fine for our current purposes.

We can also impose constraints across the equations. For example, the admittedly meaningless constraint of requiring profits to have the same effect in both the consumption and investment equations could be imposed. Retaining the constraint on the wage coefficients, we would estimate this constrained system.
. constraint define 2 [consump]profits = [invest]profits
. reg3 \$conseqn \$inveqn \$wageqn, endog(\$enlist) exog(\$exlist) constr(1 2) ireg3
note: additional endogenous variables not in the system have no effect and are ignored (wagetot).

Iteration 1: Tolerance = . 1427927
Iteration 2: Tolerance $=$. 032539
Iteration 3: Tolerance $=.00307811$
Iteration 4: Tolerance $=.00016903$
Iteration 5: Tolerance $=.00003409$
Iteration 6: Tolerance $=7.763 \mathrm{e}-06$
Iteration 7: Tolerance $=9.240 \mathrm{e}-07$
Three-stage least-squares regression, iterated

| Equation | Obs | Params | RMSE | "R-squared" |  | chi2 | P>chi2 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| consump | 21 | 3 | . 9504669 | 0.9798 |  | 1019.54 | 0.0000 |
| invest | 21 | 3 | 1.247066 | 0.8706 |  | 144.57 | 0.0000 |
| wagepriv | 21 | 3 | . 7225276 | 0.9862 |  | 1537.45 | 0.0000 |
| [consump] wagepriv - [consump] wagegovt $=0$ <br> [consump] profits - [invest]profits = 0 |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  | Coefficient Std. err. |  | z | $P>\|z\|$ | [95\% conf. interval] |  |  |
| consump profits |  |  |  |  |  |  |  |
| -- | . 1075413 | . 0957767 | 1.12 | 0.262 | -. 0801777 |  | . 2952602 |
| L1. | . 1712756 | . 0912613 | 1.88 | 0.061 | -. 0075932 |  | . 3501444 |
| wagepriv | . 798484 | . 0340876 | 23.42 | 0.000 |  | 16734 | . 8652946 |
| wagegovt | . 798484 | . 0340876 | 23.42 | 0.000 |  | 16734 | . 8652946 |
| _cons | 16.2521 | 1.212157 | 13.41 | 0.000 |  | 87631 | 18.62788 |
| invest |  |  |  |  |  |  |  |
| profits |  |  |  |  |  |  |  |
|  | . 1075413 | . 0957767 | 1.12 | 0.262 | -. 0801777 |  | . 2952602 |
| L1. | . 6443378 | . 1058682 | 6.09 | 0.000 | . 43684 |  | . 8518356 |
| capital |  |  |  |  |  |  |  |
| L1. | -. 1766669 | . 0261889 | -6.75 | 0.000 | -. 2279962 |  | -. 1253375 |
| _cons | 24.31931 | 5.284325 | 4.60 | 0.000 | 13.96222 |  | 34.6764 |
| wagepriv totinc |  |  |  |  |  |  |  |
| -- | . 4014106 | . 0300552 | 13.36 | 0.000 | . 3425035 |  | . 4603177 |
| L1. | . 1775359 | . 0321583 | 5.52 | 0.000 | . 1145068 |  | . 240565 |
| year | . 1549211 | . 0282291 | 5.49 | 0.000 | $\begin{array}{r} .099593 \\ -.2837242 \end{array}$ |  | $\begin{array}{r} .2102492 \\ 4.203299 \end{array}$ |
| _cons | 1.959788 | 1.14467 | 1.71 | 0.087 |  |  |  |

Endogenous: consump invest wagepriv wagetot profits totinc
Exogenous: L.profits wagegovt L.capital L.totinc year taxnetx govt

## - Technical note

Identification in a system of simultaneous equations involves the notion that there is enough information to estimate the parameters of the model given the specified functional form. Underidentification usually manifests itself as one matrix in the 3SLS computations. The most commonly
violated order condition for 2SLS or 3SLS involves the number of endogenous and exogenous variables. There must be at least as many noncollinear exogenous variables in the remaining system as there are endogenous right-hand-side variables in an equation. This condition must hold for each structural equation in the system.

Put as a set of rules the following:

1. Count the number of right-hand-side endogenous variables in an equation and call this $m_{i}$.
2. Count the number of exogenous variables in the same equation and call this $k_{i}$.
3. Count the total number of exogenous variables in all the structural equations plus any additional variables specified in an $\operatorname{exog}()$ or inst() option and call this $K$.
4. If $m_{i}>\left(K-k_{i}\right)$ for any structural equation $(i)$, then the system is underidentified and cannot be estimated by 3SLS.
We are also possibly in trouble if any of the exogenous variables are linearly dependent. We must have $m_{i}$ linearly independent variables among the exogenous variables represented by ( $K-k_{i}$ ).

The complete conditions for identification involve rank-order conditions on several matrices. For a full treatment, see Theil (1971) or Greene (2018, 363-365).

Henri Theil (1924-2000) was born in Amsterdam and awarded a PhD in 1951 by the University of Amsterdam. He researched and taught econometric theory, statistics, microeconomics, macroeconomic modeling, and economic forecasting, and policy at (now) Erasmus University Rotterdam, the University of Chicago, and the University of Florida. Theil's many specific contributions include work on 2SLS and 3SLS, inequality and concentration, and consumer demand.

## Stored results

reg3 stores the following in e() :

```
Scalars
    e(N) number of observations
    e(k) number of parameters
    e(k_eq) number of equations in e(b)
    e(mss_#)
    e(df_m#)
    e(rss_#)
    e(df_r)
    e(r2_#)
    e(F_#)
    e(rmse_#)
    e(dfk2_adj)
    e(ll)
    e(N_clust)
    e(chi2_#)
    e(p_#)
    e(cons_#)
    e(rank)
    e(ic)
        model sum of squares for equation #
        model degrees of freedom for equation #
        residual sum of squares for equation #
        residual degrees of freedom (small)
        R
        F statistic for equation # (small)
        root mean squared error for equation #
        divisor used with VCE when dfk2 specified
        log likelihood
        number of clusters
        \chi}\mp@subsup{\chi}{}{2}\mathrm{ for equation #
        p-value for model test for equation #
        1 when equation # has a constant, 0 otherwise
        rank of e(V)
                                number of iterations
```

```
Macros
    e(cmd) reg3
    e(cmdline) command as typed
    e(depvar) names of dependent variables
    e(exog) names of exogenous variables
    e(endog) names of endogenous variables
    e(eqnames) names of equations
    e(corr) correlation structure
    e(wtype)
    e(wexp)
    e(method)
    e(small)
    e(dfk)
    e(clustvar)
    e(vce)
    e(vcetype)
    e(properties)
    e(predict)
    e(marginsok)
    e(marginsnotok)
    e(marginsdefault)
    e(asbalanced)
    e(asobserved)
Matrices
    e(b)
    e(Cns)
    e(Sigma)
    e(V)
    weight type
    weight expression
    3sls, 2sls, ols, sure, or mvreg
    small, if specified
    dfk, if specified
    name of cluster variable
    vcetype specified in vce()
    title used to label Std. err.
    b V
    program used to implement predict
    predictions allowed by margins
    predictions disallowed by margins
    default predict() specification for margins
    factor variables fvset as asbalanced
    factor variables fvset as asobserved
    coefficient vector
    constraints matrix
    \boldsymbol{\Sigma}}\mathrm{ matrix
variance-covariance matrix of the estimators
Functions
    e(sample) marks estimation sample
```

In addition to the above, the following is stored in r() :
Matrices
$r$ (table) matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

The most concise way to represent a system of equations for 3SLS requires thinking of the individual equations and their associated data as being stacked. reg3 does not expect the data in this format, but it is a convenient shorthand. The system could then be formulated as

$$
\left[\begin{array}{c}
\mathbf{y}_{1} \\
\mathbf{y}_{2} \\
\vdots \\
\mathbf{y}_{M}
\end{array}\right]=\left[\begin{array}{cccc}
\mathbf{Z}_{1} & 0 & \ldots & 0 \\
0 & \mathbf{Z}_{2} & \ldots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \ldots & \mathbf{Z}_{M}
\end{array}\right]\left[\begin{array}{c}
\boldsymbol{\beta}_{1} \\
\boldsymbol{\beta}_{2} \\
\vdots \\
\boldsymbol{\beta}_{M}
\end{array}\right]+\left[\begin{array}{c}
\boldsymbol{\epsilon}_{1} \\
\boldsymbol{\epsilon}_{2} \\
\vdots \\
\boldsymbol{\epsilon}_{M}
\end{array}\right]
$$

In full matrix notation, this is just

$$
\mathbf{y}=\mathbf{Z} \mathbf{B}+\boldsymbol{\epsilon}
$$

The $\mathbf{Z}$ elements in these matrices represent both the endogenous and the exogenous right-hand-side variables in the equations.

Also assume that there will be correlation between the disturbances of the equations so that

$$
E\left(\boldsymbol{\epsilon} \boldsymbol{\epsilon}^{\prime}\right)=\boldsymbol{\Sigma}
$$

where the disturbances are further assumed to have an expected value of $0 ; E(\boldsymbol{\epsilon})=0$.
The first stage of 3SLS regression requires developing instrumented values for the endogenous variables in the system. These values can be derived as the predictions from a linear regression of each endogenous regressor on all exogenous variables in the system or, more succinctly, as the projection of each regressor through the projection matrix of all exogenous variables onto the regressors. Designating the set of all exogenous variables as $\mathbf{X}$ results in

$$
\widehat{\mathbf{z}}_{i}=\mathbf{X}\left(\mathbf{X}^{\prime} \mathbf{X}\right)^{-1} \mathbf{X}^{\prime} \mathbf{z}_{i} \quad \text { for each } i
$$

Taken collectively, these $\widehat{\mathbf{Z}}$ contain the instrumented values for all the regressors. They take on the actual values for the exogenous variables and first-stage predictions for the endogenous variables. Given these instrumented variables, a generalized least squares (GLS) or Aitken (1935) estimator can be formed for the parameters of the system

$$
\widehat{\mathbf{B}}=\left\{\widehat{\mathbf{Z}}^{\prime}\left(\boldsymbol{\Sigma}^{-1} \otimes \mathbf{I}\right) \widehat{\mathbf{Z}}\right\}^{-1} \widehat{\mathbf{Z}}^{\prime}\left(\boldsymbol{\Sigma}^{-1} \otimes \mathbf{I}\right) \mathbf{y}
$$

All that remains is to obtain a consistent estimator for $\boldsymbol{\Sigma}$. This estimate can be formed from the residuals of 2SLS estimates of each equation in the system. Alternately, and identically, the residuals can be computed from the estimates formed by taking $\Sigma$ to be an identity matrix. This maintains the full system of coefficients and allows constraints to be applied when the residuals are computed.

If we take $\mathbf{E}$ to be the matrix of residuals from these estimates, a consistent estimate of $\boldsymbol{\Sigma}$ is

$$
\widehat{\boldsymbol{\Sigma}}=\frac{\mathbf{E}^{\prime} \mathbf{E}}{n}
$$

where $n$ is the number of observations in the sample. An alternative divisor for this estimate can be obtained with the dfk option as outlined under options.

With the estimate of $\widehat{\boldsymbol{\Sigma}}$ placed into the GLS estimating equation,

$$
\widehat{\mathbf{B}}=\left\{\widehat{\mathbf{Z}}^{\prime}\left(\widehat{\boldsymbol{\Sigma}}^{-1} \otimes \mathbf{I}\right) \widehat{\mathbf{Z}}\right\}^{-1} \widehat{\mathbf{Z}}^{\prime}\left(\widehat{\boldsymbol{\Sigma}}^{-1} \otimes \mathbf{I}\right) \mathbf{y}
$$

is the 3SLS estimates of the system parameters.
The asymptotic variance-covariance matrix of the estimator is just the standard formulation for a GLS estimator

$$
\mathbf{V}_{\widehat{\mathbf{B}}}=\left\{\widehat{\mathbf{Z}}^{\prime}\left(\widehat{\boldsymbol{\Sigma}}^{-1} \otimes \mathbf{I}\right) \widehat{\mathbf{Z}}\right\}^{-1}
$$

Iterated 3SLS estimates can be obtained by computing the residuals from the three-stage parameter estimates, using these to formulate a new $\widehat{\boldsymbol{\Sigma}}$, and recomputing the parameter estimates. This process is repeated until the estimates $\widehat{\mathbf{B}}$ converge-if they converge. Convergence is not guaranteed. When estimating a system by SURE, these iterated estimates will be the maximum likelihood estimates for the system. The iterated solution can also be used to produce estimates that are invariant to choice of system and restriction parameterization for many linear systems under full 3SLS.

The exposition above follows the parallel developments in Greene (2018) and Davidson and MacKinnon (1993).

This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce(robust) and vce(cluster clustvar), respectively. See [P] _robust, particularly Introduction and Methods and formulas.

Alexander Craig Aitken (1895-1967) was born in Dunedin, New Zealand. He attended the University of Otago on a full scholarship but left the university to enlist in the New Zealand Expeditionary Force during World War I. He would later reflect on the time he spent in active service and publish two war memoirs.

After returning from the war, he completed his studies at the University of Otago and then worked on his PhD at the University of Edinburgh under E. T. Whittaker. Although he suffered weeks of illness during the course of his postgraduate studies, he managed to write an impressive thesis on data smoothing. He then became a professor of mathematics, and later the Chair of Pure Mathematics, a post he would hold until his retirement in 1965.
Among his many accolades, he was elected a Fellow of the Royal Society of Edinburgh and a Fellow of the Royal Society of Literature, following the publication of his second war memoir. He published many papers on numerical analysis and statistics as well as a couple of books on matrices. Aitken is credited with deriving the generalized least squares estimator, which has been referred to as Aitken's generalized least squares. Aside from his many academic contributions, he had mental calculating abilities like no other. He is known to have multiplied two 9-digit numbers in less than a minute and could recite up to 707 digits of $\pi$.

## References

Aitken, A. C. 1935. On least squares and linear combination of observations. Proceedings of the Royal Society of Edinburgh 55: 42-48. https://doi.org/10.1017/S0370164600014346.

Bewley, R. 2000. Mr. Henri Theil: An interview with the International Journal of Forecasting. International Journal of Forecasting 16: 1-16. https://doi.org/10.1016/S0169-2070(99)00031-X.
Davidson, R., and J. G. MacKinnon. 1993. Estimation and Inference in Econometrics. New York: Oxford University Press.

Greene, W. H. 2018. Econometric Analysis. 8th ed. New York: Pearson.
Klein, L. R. 1950. Economic Fluctuations in the United States 1921-1941. New York: Wiley.
Nichols, A. 2007. Causal inference with observational data. Stata Journal 7: 507-541.
Poi, B. P. 2006. Jackknife instrumental variables estimation in Stata. Stata Journal 6: 364-376.
Theil, H. 1971. Principles of Econometrics. New York: Wiley.
Zellner, A., and H. Theil. 1962. Three stage least squares: Simultaneous estimate of simultaneous equations. Econometrica 29: 54-78. https://doi.org/10.2307/1911287.

## Also see

[R] reg3 postestimation - Postestimation tools for reg3
$[R]$ ivregress - Single-equation instrumental-variables regression
[R] nlsur - Estimation of nonlinear systems of equations
$[R]$ regress - Linear regression
[R] sureg - Zellner's seemingly unrelated regression
[MV] mvreg - Multivariate regression
[SEM] Example 7 - Nonrecursive structural model
[SEM] Intro 5 - Tour of models
[TS] forecast - Econometric model forecasting
[U] 20 Estimation and postestimation commands

## Title

reg3 postestimation - Postestimation tools for reg3

Postestimation commands
Remarks and examples
Also see
predict margins
Methods and formulas Reference

## Postestimation commands

The following postestimation commands are available after reg3:

| Command | Description |
| :---: | :---: |
| contrast | contrasts and ANOVA-style joint tests of estimates |
| *estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estimates | cataloging estimation results |
| etable | table of estimation results |
| forecast | dynamic forecasts and simulations |
| hausman | Hausman's specification test |
| lincom | point estimates, standard errors, testing, and inference for linear combinations of coefficients |
| margins | marginal means, predictive margins, marginal effects, and average marginal effects |
| marginsplot | graph the results from margins (profile plots, interaction plots, etc.) |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients |
| predict | predictions and their SEs, residuals, etc. |
| predictnl | point estimates, standard errors, testing, and inference for generalized predictions |
| pwcompare | pairwise comparisons of estimates |
| test | Wald tests of simple and composite linear hypotheses |
| testnl | Wald tests of nonlinear hypotheses |

[^51]
## predict

## Description for predict

predict creates a new variable containing predictions such as linear predictions, standard errors, residuals, and differences between the linear predictions of two equations.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

```
    predict [type] newvar [if] [in] [, equation(eqno[,eqno]) statistic]
statistic Description
```

Main
$\mathrm{xb} \quad$ linear prediction; the default
stdp standard error of the linear prediction
residuals residuals
difference difference between the linear predictions of two equations
stddp standard error of the difference in linear predictions

These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

## Options for predict

$\underset{\text { equation (eqno }[\text {, eqno }] \text { ) specifies to which equation you are referring. }}{\sqrt{\text { Main }} \text {. }}$
equation() is filled in with one eqno for the xb , stdp , and residuals options. equation(\#1) would mean the calculation is to be made for the first equation, equation(\#2) would mean the second, and so on. You could also refer to the equations by their names. equation(income) would refer to the equation named income and equation(hours) to the equation named hours. If you do not specify equation(), results are the same as if you specified equation(\#1).
difference and stddp refer to between-equation concepts. To use these options, you must specify two equations, for example, equation(\#1,\#2) or equation(income,hours). When two equations must be specified, equation() is required.
xb , the default, calculates the linear prediction (fitted values) - the prediction of $\mathbf{x}_{j} \mathbf{b}$ for the specified equation.
stdp calculates the standard error of the prediction for the specified equation. It can be thought of as the standard error of the predicted expected value or mean for the observation's covariate pattern. The standard error of the prediction is also referred to as the standard error of the fitted value.
residuals calculates the residuals.
difference calculates the difference between the linear predictions of two equations in the system. With equation(\#1,\#2), difference computes the prediction of equation(\#1) minus the prediction of equation(\#2).
stddp is allowed only after you have previously fit a multiple-equation model. The standard error of the difference in linear predictions ( $\mathbf{x}_{1 j} \mathbf{b}-\mathbf{x}_{2 j} \mathbf{b}$ ) between equations 1 and 2 is calculated.

For more information on using predict after multiple-equation estimation commands, see $[R]$ predict.

## margins

## Description for margins

margins estimates margins of response for linear predictions and differences between the linear predictions of two equations.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...)...] [options]
```

statistic Description

| default | linear predictions for each equation |
| :--- | :--- |
| xb | linear prediction for a specified equation |
| difference | difference between the linear predictions of two equations |
| stdp | not allowed with margins |
| residuals | not allowed with margins |
| stddp | not allowed with margins |

xb defaults to the first equation.
Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [ R$]$ margins.

## Remarks and examples

## > Example 1: Using predict

In example 2 of $[R]$ reg3, we fit a simple supply-and-demand model. Here we obtain the fitted supply and demand curves assuming that the exogenous regressors equal their sample means. We first replace each of the three exogenous regressors with their sample means, then we call predict to obtain the predictions.

```
. use https://www.stata-press.com/data/r18/supDem
. global demand "(qDemand: quantity price pcompete income)"
. global supply "(qSupply: quantity price praw)"
. reg3 $demand $supply, endog(price)
(output omitted)
. summarize pcompete, meanonly
. replace pcompete = r(mean)
(49 real changes made)
```

. summarize income, meanonly
. replace income $=r(m e a n)$
(49 real changes made)
. summarize praw, meanonly
. replace praw = r(mean)
(49 real changes made)

- predict demand, equation(qDemand)
(option $\mathbf{x b}$ assumed; fitted values)
. predict supply, equation(qSupply)
(option xb assumed; fitted values)
. graph twoway line demand price, sort || line supply price, ytitle(" ")
> legend(label(1 "Fitted values: qDemand") label(2 "Fitted values: qSupply"))


As we would expect based on economic theory, the demand curve slopes downward while the supply curve slopes upward. With the exogenous variables at their mean levels, the equilibrium price and quantity are slightly less than 33 and 13 , respectively.
> Example 2: Obtaining forecasts
In example 3 of [R] reg3, we fit Klein's (1950) model of the US economy. That model includes three stochastic equations we fit using reg3 as well as four identities. Here we briefly illustrate how the forecast command can be used to obtain forecasts for all the endogenous variables in the model. For a more detailed discussion of how to forecast with this model, see [TS] forecast.

In Stata, we type

```
. use https://www.stata-press.com/data/r18/klein2, clear
. reg3 (c p L.p w) (i p L.p L.k) (wp y L.y yr), endog(w p y) exog(t wg g)
(output omitted)
estimates store kleineqs
. forecast create kleinmodel
    Forecast model kleinmodel started.
. forecast estimates kleineqs
    Added estimation results from reg3.
    Forecast model kleinmodel now contains 3 endogenous variables.
    forecast identity y = c + i + g
    Forecast model kleinmodel now contains 4 endogenous variables.
. forecast identity p = y - t - wp
    Forecast model kleinmodel now contains 5 endogenous variables.
    forecast identity k = L.k + i
    Forecast model kleinmodel now contains 6 endogenous variables.
    forecast identity w = wg + wp
    Forecast model kleinmodel now contains 7 endogenous variables.
. forecast solve, begin(1937)
Computing dynamic forecasts for model kleinmodel.
Starting period: }193
Ending period: }194
Forecast prefix: f_
1937:
1938: ..............................................................
1939: .......................................................
1940: ....................................................
1941: ...........................................................
Forecast }7\mathrm{ variables spanning 5 periods.
```

Here we have obtained dynamic forecasts for our 7 endogenous variables beginning in 1937. By default, the variables containing the forecasts begin with the prefix $f_{\text {_ }}$. Next we plot the forecast and actual values of consumption:

```
. tsline c f_c
```



For more information about producing forecasts, see [TS] forecast.

## Methods and formulas

The computational formulas for the statistics produced by predict can be found in $[R]$ predict and $[R]$ regress postestimation.

## Reference

Klein, L. R. 1950. Economic Fluctuations in the United States 1921-1941. New York: Wiley.

## Also see

[R] reg3 - Three-stage estimation for systems of simultaneous equations
[U] 20 Estimation and postestimation commands

## Title

regress - Linear regression

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| Acknowledgments | References | Also see |  |

## Description

regress performs ordinary least-squares linear regression. regress can also perform weighted estimation, compute robust and cluster-robust standard errors, and adjust results for complex survey designs.

## Quick start

Simple linear regression of y on x 1
regress y x1
Regression of $y$ on $x 1$, $x 2$, and indicators for categorical variable a
regress y x1 x2 i.a
Add the interaction between continuous variable x 2 and a
regress y x1 c.x2\#\#i.a
Fit model for observations where v 1 is greater than zero
regress $y$ x1 $x 2$ i.a if $v 1>0$
With cluster-robust standard errors for clustering by levels of cvar
regress y x1 x2 i.a, vce(cluster cvar)
With cluster-robust standard errors for clustering by levels of cvar1 and cvar2
regress y x1 x2 i.a, vce(cluster cvar1 cvar2)
With bootstrap standard errors
regress y x1 x2 i.a, vce(bootstrap)
Report standardized coefficients
regress y x1 x2 i.a, beta
Adjust for complex survey design using svyset data
svy: regress y x1 x2 i.a
Use sampling weight wvar
regress y x1 x2 i.a [pweight=wvar]

## Menu

Statistics $>$ Linear models and related $>$ Linear regression

## Syntax

regress depvar [indepvars] [if] [in] [weight] [, options]
options Description

## Model

noconstant
hascons
tsscons
SE/Robust
vce (vcetype)

Reporting
level(\#)
beta
eform(string)
depname (varname)
clustertable
display_options
suppress constant term
has user-supplied constant
compute total sum of squares with constant; seldom used
vcetype may be ols, robust, cluster clustvarlist, bootstrap,
$\underline{\text { jackknife, hc2 }[\text { clustvar }], \text { or hc3 }}$
set confidence level; default is level (95)
report standardized beta coefficients
report exponentiated coefficients and label as string
substitute dependent variable name; programmer's option
display table of multiway cluster combinations
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
noheader
notable
plus
mse1
coeflegend
> suppress output header
> suppress coefficient table
> make table extendable
> force mean squared error to 1
> display legend instead of statistics
indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
depvar and indepvars may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bayes, bootstrap, by, collect, fmm, fp, jackknife, mfp, mi estimate, nestreg, rolling, statsby, stepwise, and svy are allowed; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: regress and [FMM] fmm: regress.
vce(bootstrap) and vce(jackknife) are not allowed with the mi estimate prefix; see [MI] mi estimate.
Weights are not allowed with the bootstrap prefix; see $[R]$ bootstrap.
aweights are not allowed with the jackknife prefix; see [R] jackknife.
hascons, tsscons, vce(), beta, noheader, notable, plus, depname(), mse1, and weights are not allowed with the svy prefix; see [SVY] svy.
aweights, fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
noheader, notable, plus, mse1, and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

noconstant; see [R] Estimation options.
hascons indicates that a user-defined constant or its equivalent is specified among the independent variables in indepvars. Some caution is recommended when specifying this option, as resulting estimates may not be as accurate as they otherwise would be. Use of this option requires "sweeping" the constant last, so the moment matrix must be accumulated in absolute rather than deviation form. This option may be safely specified when the means of the dependent and independent variables are all reasonable and there is not much collinearity between the independent variables. The best procedure is to view hascons as a reporting option-estimate with and without hascons and verify that the coefficients and standard errors of the variables not affected by the identity of the constant are unchanged.
tsscons forces the total sum of squares to be computed as though the model has a constant, that is, as deviations from the mean of the dependent variable. This is a rarely used option that has an effect only when specified with noconstant. It affects the total sum of squares and all results derived from the total sum of squares.
vce(vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (ols), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvarlist), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
vce(ols), the default, uses the standard variance estimator for ordinary least-squares regression.
vce(cluster clustvarlist) specifies that standard errors allow for intragroup correlation within groups defined by one or more variables in clustvarlist, relaxing the usual requirement that the observations be independent. For example, vce(cluster clustvar1) produces clusterrobust standard errors that allow for observations that are independent across groups defined by clustvar1 but not necessarily independent within groups. You could also type vce(cluster clustvar1 clustvar2 ...clusterp) to account for correlation within groups formed by $p$ variables (multiway clustering).
regress also allows the following:
vce(hc2 [clustvar] [, dfadjust]) and vce(hc3) specify alternative bias corrections for the robust variance calculation. vce (hc2) and vce (hc3) may not be specified with the svy prefix. In the unclustered case, vce (robust) uses $\widehat{\sigma}_{j}^{2}=\{n /(n-k)\} u_{j}^{2}$ as an estimate of the variance of the $j$ th observation, where $n$ is the number of observations, $k$ is the number of regressors, $u_{j}$ is the calculated residual, and $n /(n-k)$ is included to improve the overall estimate's small-sample properties.
vce (hc2) instead uses $u_{j}^{2} /\left(1-h_{j j}\right)$ as the observation's variance estimate, where $h_{j j}$ is the diagonal element of the hat (projection) matrix. This estimate is unbiased if the model really is homoskedastic. vce (hc2) tends to produce slightly more conservative confidence intervals. vce (hc2 clustvar) produces estimates that allow for intragroup correlation within groups defined by clustvar. dfadjust computes the Bell and McCaffrey (2002) adjusted degrees of freedom based on clustvar. Note that dfadjust does not affect multiple-imputation results when the command is used with mi estimate. See Methods and formulas for a description of the computation when clustvar is specified.
vce (hc3) uses $u_{j}^{2} /\left(1-h_{j j}\right)^{2}$ as suggested by Davidson and MacKinnon (1993), who report that this method tends to produce better results when the model really is heteroskedastic. vce(hc3) produces confidence intervals that tend to be even more conservative.
See Davidson and MacKinnon (1993, 554-556) and Angrist and Pischke (2009, 294-308) for more discussion on these two bias corrections.

Reporting
level (\#); see [R] Estimation options.
beta asks that standardized beta coefficients be reported instead of confidence intervals. The beta coefficients are the regression coefficients obtained by first standardizing all variables to have a mean of 0 and a standard deviation of 1 . beta may not be specified with vce(cluster clustvarlist) or the svy prefix.
eform(string) is used only in programs and ado-files that use regress to fit models other than linear regression. eform() specifies that the coefficient table be displayed in exponentiated form as defined in [R] Maximize and that string be used to label the exponentiated coefficients in the table.
depname (varname) is used only in programs and ado-files that use regress to fit models other than linear regression. depname() may be specified only at estimation time. varname is recorded as the identity of the dependent variable, even though the estimates are calculated using depvar. This method affects the labeling of the output-not the results calculated - but could affect subsequent calculations made by predict, where the residual would be calculated as deviations from varname rather than depvar. depname() is most typically used when depvar is a temporary variable (see [P] macro) used as a proxy for varname.
depname() is not allowed with the svy prefix.
clustertable displays a table reporting cluster combinations and the number of clusters per combination. This option is available only when vce (cluster clustvarlist) is specified with more than one variable in clustvarlist to compute multiway cluster-robust standard errors.
display_options: noci, nopvalues, dfci, dfpvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\% fimt), pformat ( $\% f m t$ ), sformat ( $\% f m t$ ), and nolstretch; see [R] Estimation options.
dfci specifies that parameter degrees of freedom and confidence intervals be reported in the coefficient table.
dfpvalues specifies that parameter degrees of freedom and $p$-values be reported in the coefficient table.

The following options are available with regress but are not shown in the dialog box:
noheader suppresses the display of the ANOVA table and summary statistics at the top of the output; only the coefficient table is displayed. This option is often used in programs and ado-files.
notable suppresses display of the coefficient table.
plus specifies that the output table be made extendable. This option is often used in programs and ado-files.
mse1 is used only in programs and ado-files that use regress to fit models other than linear regression and is not allowed with the svy prefix. mse1 sets the mean squared error to 1 , forcing the variance-covariance matrix of the estimators to be $\left(\mathbf{X}^{\prime} \mathbf{X}\right)^{-1}$ (see Methods and formulas below) and affecting calculated standard errors. Degrees of freedom for $t$ statistics is calculated as $n$ rather than $n-k$.
coeflegend; see [R] Estimation options.

## Remarks and examples

Remarks are presented under the following headings:

Ordinary least squares<br>Treatment of the constant<br>Robust standard errors<br>Weighted regression<br>Video examples

regress performs linear regression, including ordinary least squares and weighted least squares. See [U] 27 Overview of Stata estimation commands for a list of other regression commands that may be of interest. For a general discussion of linear regression, see Kutner et al. (2005).

See Stock and Watson (2019) and Wooldridge (2020) for an excellent treatment of estimation, inference, interpretation, and specification testing in linear regression models. See Wooldridge (2010, chap. 4) for a more advanced discussion along the same lines.

See Hamilton (2013, chap. 7) and Cameron and Trivedi (2022, chap. 3) for an introduction to linear regression using Stata. Dohoo, Martin, and Stryhn (2012, 2010) discuss linear regression using examples from epidemiology, and Stata datasets and do-files used in the text are available. Cameron and Trivedi (2022) discuss linear regression using econometric examples with Stata. Mitchell (2021) shows how to use graphics and postestimation commands to understand a fitted regression model.

Chatterjee and Hadi (2012) explain regression analysis by using examples containing typical problems that you might encounter when performing exploratory data analysis. We also recommend Weisberg (2014), who emphasizes the importance of the assumptions of linear regression and problems resulting from these assumptions. Becketti (2020) discusses regression analysis with an emphasis on time-series data. Angrist and Pischke (2009) approach regression as a tool for exploring relationships, estimating treatment effects, and providing answers to public policy questions. For a mathematically rigorous treatment, see Peracchi (2001, chap. 6). Finally, see Plackett (1972) if you are interested in the history of regression. Least squares, which dates back to the 1790s, was discovered independently by Legendre and Gauss.

## Ordinary least squares

## > Example 1: Basic linear regression

Suppose that we have data on the mileage rating and weight of 74 automobiles. The variables in our data are mpg, weight, and foreign. The last variable assumes the value 1 for foreign and 0 for domestic automobiles. We wish to fit the model

$$
\mathrm{mpg}=\beta_{0}+\beta_{1} \text { weight }+\beta_{2} \text { foreign }+\epsilon
$$

This model can be fit with regress by typing

regress produces a variety of summary statistics along with the table of regression coefficients. At the upper left, regress reports an analysis-of-variance (ANOVA) table. The column headings SS, df, and MS stand for "sum of squares", "degrees of freedom", and "mean square", respectively. In this example, the total sum of squares is 2,443.5: 1,619.3 accounted for by the model and 824.2 left unexplained. Because the regression included a constant, the total sum reflects the sum after removal of means, as does the sum of squares due to the model. The table also reveals that there are 73 total degrees of freedom (counted as 74 observations less 1 for the mean removal), of which 2 are consumed by the model, leaving 71 for the residual.

To the right of the ANOVA table are presented other summary statistics. The $F$ statistic associated with the ANOVA table is 69.75 . The statistic has 2 numerator and 71 denominator degrees of freedom. The $F$ statistic tests the hypothesis that all coefficients excluding the constant are zero. The chance of observing an $F$ statistic that large or larger is reported as 0.0000 , which is Stata's way of indicating a number smaller than 0.00005 . The $R^{2}$ for the regression is 0.6627 , and the $R^{2}$ adjusted for degrees of freedom $\left(R_{a}^{2}\right)$ is 0.6532 . The root mean squared error, labeled Root MSE, is 3.4071 . It is the square root of the mean squared error reported for the residual in the ANOVA table.

Finally, Stata produces a table of the estimated coefficients. The first line of the table indicates that the left-hand-side variable is mpg. Thereafter follow the estimated coefficients. Our fitted model is

$$
\text { mpg_hat }=41.68-0.0066 \text { weight }-1.65 \text { foreign }
$$

Reported to the right of the coefficients in the output are the standard errors. For instance, the standard error for the coefficient on weight is 0.0006371 . The corresponding $t$ statistic is -10.34 , which has a two-sided significance level of 0.000 . This number indicates that the significance is less than 0.0005 . The $95 \%$ confidence interval for the coefficient is $[-0.0079,-0.0053]$.
> Example 2: Transforming the dependent variable
If we had a graph comparing mpg with weight, we would notice that the relationship is distinctly nonlinear. This is to be expected because energy usage per distance should increase linearly with weight, but mpg is measuring distance per energy used. We could obtain a better model by generating a new variable measuring the number of gallons used per 100 miles (gp100m) and then using this new variable in our model:

$$
\operatorname{gp} 100 \mathrm{~m}=\beta_{0}+\beta_{1} \text { weight }+\beta_{2} \text { foreign }+\epsilon
$$

We can now fit this model:

```
. generate gp100m = 100/mpg
```

. regress gp100m weight foreign

| Source | SS | df | MS | Number of obs | = | 74 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | $\mathrm{F}(2,71)$ | = | 113.97 |
| Model | 91.1761694 | 2 | 45.5880847 | Prob > F | = | 0.0000 |
| Residual | 28.4000913 | 71 | . 400001287 | R -squared | = | 0.7625 |
|  |  |  |  | Adj R-squared | = | 0.7558 |
| Total | 119.576261 | 73 | 1.63803097 | Root MSE | = | . 63246 |


| gp100m | Coefficient | Std. err. | t | P>\|t| | [95\% conf. interval] |  |
| ---: | ---: | :--- | :---: | :---: | ---: | ---: |
| weight | .0016254 | .0001183 | 13.74 | 0.000 | .0013896 | .0018612 |
| foreign | .6220535 | .1997381 | 3.11 | 0.003 | .2237871 | 1.02032 |
| _cons | -.0734839 | .4019932 | -0.18 | 0.855 | -.8750354 | .7280677 |

Fitting the physically reasonable model increases our $R^{2}$ to 0.7625 .

## > Example 3: Obtaining beta coefficients

regress shares the features of all estimation commands. Among other things, this means that after running a regression, we can use test to test hypotheses about the coefficients, estat vce to examine the covariance matrix of the estimators, and predict to obtain predicted values, residuals, and influence statistics. See [U] 20 Estimation and postestimation commands. Options that affect how estimates are displayed, such as beta or level(), can be used when replaying results.

Suppose that we meant to specify the beta option to obtain beta coefficients (regression coefficients normalized by the ratio of the standard deviation of the regressor to the standard deviation of the dependent variable). Even though we forgot, we can specify the option now:

| Source | SS | df | MS | Number of obs | = | 74 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | $\mathrm{F}(2,71)$ | = | 113.97 |
| Model | 91.1761694 | 2 | 45.5880847 | Prob > F | = | 0.0000 |
| Residual | 28.4000913 | 71 | . 400001287 | R -squared | = | 0.7625 |
|  |  |  |  | Adj R-squared | = | 0.7558 |
| Total | 119.576261 | 73 | 1.63803097 | Root MSE | = | . 63246 |
| gp100m | Coefficient | Std. err. | t | $p>\|t\|$ |  | Beta |
| weight | . 0016254 | . 0001183 | 13.74 | 0.000 |  | . 9870255 |
| foreign | . 6220535 | . 1997381 | 3.11 | 0.003 |  | . 2236673 |
| _cons | -. 0734839 | . 4019932 | -0.18 | 0.855 |  | . |

## Treatment of the constant

By default, regress includes an intercept (constant) term in the model. The noconstant option suppresses it, and the hascons option tells regress that the model already has one.

## > Example 4: Suppressing the constant term

We wish to fit a regression of the weight of an automobile against its length, and we wish to impose the constraint that the weight is zero when the length is zero.

If we simply type regress weight length, we are fitting the model

$$
\text { weight }=\beta_{0}+\beta_{1} \text { length }+\epsilon
$$

Here a length of zero corresponds to a weight of $\beta_{0}$. We want to force $\beta_{0}$ to be zero or, equivalently, estimate an equation that does not include an intercept:

$$
\text { weight }=\beta_{1} \text { length }+\epsilon
$$

We do this by specifying the noconstant option:

| Source | SS | df | MS | Number of obs | = | 74 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | F (1, 73) | = | 3450.13 |
| Model | 703869302 | 1 | 703869302 | Prob > F | = | 0.0000 |
| Residual | 14892897.8 | 73 | 204012. 299 | R-squared | = | 0.9793 |
|  |  |  |  | Adj R-squared | = | 0.9790 |
| Total | 718762200 | 74 | 9713002.7 | Root MSE | = | 451.68 |


| weight | Coefficient | Std. err. | $t$ | P>\|t| | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | :---: | :---: |
| length | 16.29829 | .2774752 | 58.74 | 0.000 | 15.74528 | 16.8513 |

In our data, length is measured in inches and weight in pounds. We discover that each inch of length adds 16 pounds to the weight.

Sometimes there is no need for Stata to include a constant term in the model. Most commonly, this occurs when the model contains a set of mutually exclusive indicator variables. hascons is a variation of the noconstant option-it tells Stata not to add a constant to the regression because the regression specification already has one, either directly or indirectly.

For instance, we now refit our model of weight as a function of length and include separate constants for foreign and domestic cars by specifying bn.foreign. bn.foreign is factor-variable notation for "no base for foreign" or "include all levels of variable foreign in the model"; see [U] 11.4.3 Factor variables.


## Technical note

There is a subtle distinction between the hascons and noconstant options. We can most easily reveal it by refitting the last regression, specifying noconstant rather than hascons:
. regress weight length bn.foreign, noconstant

| Source | SS | df | MS |
| ---: | ---: | ---: | ---: |
| Model | 714315766 | 3 | 238105255 |
| Residual | 4446433.7 | 71 | 62625.8268 |
| Total | 718762200 | 74 | 9713002.7 |


| Number of obs | $=$ | 74 |
| :--- | :--- | ---: |
| F $(3,71)$ | $=$ | 3802.03 |
| Prob > F | $=$ | 0.0000 |
| R-squared | $=$ | 0.9938 |
| Adj R-squared | $=$ | 0.9936 |
| Root MSE | $=$ | 250.25 |


| weight | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| :---: | ---: | :--- | :---: | :---: | :---: | :---: |
| length | 31.44455 | 1.601234 | 19.64 | 0.000 | 28.25178 | 34.63732 |
| foreign |  |  |  |  |  |  |
| Domestic <br> Foreign | -2850.25 | 315.9691 | -9.02 | 0.000 | -3480.274 | -2220.225 |
| -2983.927 | 275.1041 | -10.85 | 0.000 | -3532.469 | -2435.385 |  |

Comparing this output with that produced by the previous regress command, we see that they are almost, but not quite, identical. The parameter estimates and their associated statistics-the second half of the output - are identical. The overall summary statistics and the ANOVA table - the first half of the output-are different, however.

In the first case, the $R^{2}$ is shown as 0.8992 ; here it is shown as 0.9938 . In the first case, the $F$ statistic is 316.54 ; now it is $3,802.03$. The numerator degrees of freedom is different as well. In the first case, the numerator degrees of freedom is 2 ; now the degrees of freedom is 3 . Which is correct?

Both are. Specifying the hascons option causes regress to adjust the ANOVA table and its associated statistics for the explanatory power of the constant. The regression in effect has a constant; it is just written in such a way that a separate constant is unnecessary. No such adjustment is made with the noconstant option.

## - Technical note

When the hascons option is specified, regress checks to make sure that the model does in fact have a constant term. If regress cannot find a constant term, it automatically adds one. Fitting a model of weight on length and specifying the hascons option, we obtain

| Source | SS | df | MS | Number of obs |  | 74 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Model | 39461306.8 | 1 | 39461306.8 | Pro | F | 0.0000 |
| Residual | 4632871.55 | 72 | 64345.4382 |  |  | 0.8949 |
|  |  |  |  |  | squared | 0.8935 |
| Total | 44094178.4 | 73 | 604029.841 | Roo |  | 253.66 |
| weight | Coefficient | Std. err. | t | $P>\|t\|$ | [95\% conf. interval] |  |
| length | 33.01988 | 1.333364 | 24.76 | 0.000 | 30.36187 | 35.67789 |
| _cons | -3186.047 | 252.3113 | -12.63 | 0.000 | -3689.02 | -2683.073 |

Even though we specified hascons, regress included a constant, anyway. It also added a note to our output: "note: option hascons false".

## - Technical note

Even if the model specification effectively includes a constant term, we need not specify the hascons option. regress is always on the lookout for collinear variables and omits them from the model. For instance,
. regress weight length bn.foreign
note: 1.foreign omitted because of collinearity.

| Source | SS | df | MS | Number of obs | = | 74 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | $\mathrm{F}(2,71)$ | = | 316.54 |
| Model | 39647744.7 | 2 | 19823872.3 | Prob > F | = | 0.0000 |
| Residual | 4446433.7 | 71 | 62625.8268 | R -squared | = | 0.8992 |
|  |  |  |  | Adj R-squared | = | 0.8963 |
| Total | 44094178.4 | 73 | 604029.841 | Root MSE | = | 250.25 |


| weight | Coefficient | Std. err. | t | $\mathrm{P}>\mid \mathrm{tI}$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| length | 31.44455 | 1.601234 | 19.64 | 0.000 | 28.25178 | 34.63732 |
| foreign <br> Domestic <br> Foreign | 133.6775 | 77.47615 <br> (omitted) | 1.73 | 0.089 | -20.80555 | 288.1605 |
| _cons | -2983.927 | 275.1041 | -10.85 | 0.000 | -3532.469 | -2435.385 |

## Robust standard errors

regress with the vce(robust) option substitutes a robust variance matrix calculation for the conventional calculation, or if vce (cluster clustvarlist) is specified, allows relaxing the assumption of independence within groups. How this method works is explained in [U] 20.22 Obtaining robust variance estimates. Below, we show how well this approach works.

## > Example 5: Heteroskedasticity and robust standard errors

Specifying the vce(robust) option is equivalent to requesting White-corrected standard errors in the presence of heteroskedasticity. We use the automobile data and, in the process of looking at the energy efficiency of cars, analyze a variable with considerable heteroskedasticity.

We will examine the amount of energy - measured in gallons of gasoline-that the cars in the data need to move 1,000 pounds of their weight 100 miles. We are going to examine the relative efficiency of foreign and domestic cars.

```
. generate gpmw = ((1/mpg)/weight)*100*1000
```

. summarize gpmw

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| gpmw | 74 | 1.682184 | .2426311 | 1.09553 | 2.30521 |

In these data, the engines consume between 1.10 and 2.31 gallons of gas to move 1,000 pounds of the car's weight 100 miles. If we ran a regression with conventional standard errors of gpmw on foreign, we would obtain
. regress gpmw foreign

| Source | SS | df | MS |
| ---: | :---: | ---: | :---: |
| Model | .936705572 | 1 | .936705572 |
| Residual | 3.36079459 | 72 | .046677703 |
| Total | 4.29750017 | 73 | .058869865 |


| Number of obs | $=$ | 74 |
| :--- | :--- | ---: |
| $\mathrm{~F}(1,72)$ | $=$ | 20.07 |
| Prob > F | $=$ | 0.0000 |
| R-squared | $=$ | 0.2180 |
| Adj R-squared | $=$ | 0.2071 |
| Root MSE | $=$ | .21605 |


| gpmw | Coefficient | Std. err. | $t$ | P>\|t| | [95\% conf. interval] |  |
| ---: | ---: | :---: | ---: | :---: | ---: | ---: |
| foreign | .2461526 | .0549487 | 4.48 | 0.000 | .1366143 | .3556909 |
| _cons | 1.609004 | .0299608 | 53.70 | 0.000 | 1.549278 | 1.66873 |

regress with the vce(robust) option, on the other hand, reports


The point estimates are the same (foreign cars need one-quarter gallon more gas), but the standard errors differ by roughly $20 \%$. Conventional regression reports the $95 \%$ confidence interval as $[0.14,0.36$ ], whereas the robust standard errors make the interval [0.11, 0.38].

Which is right? Notice that gpmw is a variable with considerable heteroskedasticity:

| Car origin | Summary of gpmw |  |  |
| :---: | :---: | :---: | :---: |
| Domestic | 1.6090039 | . 16845182 | 52 |
| Foreign | 1.8551565 | . 30186861 | 22 |
| Total | 1.6821844 | . 24263113 | 74 |

Thus, here we favor the robust standard errors. In [U] 20.22 Obtaining robust variance estimates, we show another example using linear regression where it makes little difference whether we specify vce(robust). The linear-regression assumptions were true, and we obtained nearly linear-regression results. The advantage of the robust estimate is that in neither case did we have to check assumptions.

## - Technical note

regress purposefully suppresses displaying the ANOVA table when vce (robust) is specified. This is done because the sums of squares are no longer appropriate for use in the usual hypothesis tests, even though computationally the sums of squares remain the same. In the nonrobust setting, the $F$ statistic reported by regress is defined in terms of the sums of squares, as in ANOVA. When vce (robust) is specified, the ANOVA test is not valid, and the $F$ statistic corresponds to a Wald test based on the robustly estimated variance matrix.

Some references give formulas for the $F$ statistic in terms of either $R^{2}$ or the root MSE. It is not appropriate to use those formulas for the $F$ statistic with robust standard errors because the $R^{2}$ and root MSE are calculated from the sums of squares. Moreover, the root MSE can no longer be used as an estimate for $\sigma$ because there is no longer a single $\sigma$ to estimate - the variance of the residual varies observation by observation. However, regress continues to report the $R^{2}$ and the root MSE in the robust setting because those statistics are still usable in other settings. In particular, $R^{2}$ remains valid as a goodness-of-fit statistic.
> Example 6: Alternative robust standard errors
The vce (hc2) and vce (hc3) options modify the robust variance calculation. In the context of linear regression without clustering, the idea behind the robust calculation is somehow to measure $\sigma_{j}^{2}$, the variance of the residual associated with the $j$ th observation, and then to use that estimate to improve the estimated variance of $\widehat{\boldsymbol{\beta}}$. Because residuals have (theoretically and practically) mean 0 , one estimate of $\sigma_{j}^{2}$ is the observation's squared residual itself $-u_{j}^{2}$. A finite-sample correction could improve that by multiplying $u_{j}^{2}$ by $n /(n-k)$, and, as a matter of fact, vce (robust) uses $\{n /(n-k)\} u_{j}^{2}$ as its estimate of the residual's variance.
vce(hc2) and vce(hc3) use alternative estimators of the observation-specific variances. For instance, if the residuals are homoskedastic, we can show that the expected value of $u_{j}^{2}$ is $\sigma^{2}\left(1-h_{j j}\right)$, where $h_{j j}$ is the $j$ th diagonal element of the projection (hat) matrix. $h_{j j}$ has average value $k / n$, so $1-h_{j j}$ has average value $1-k / n=(n-k) / n$. Thus, the default robust estimator $\widehat{\sigma}_{j}=\{n /(n-k)\} u_{j}^{2}$ amounts to dividing $u_{j}^{2}$ by the average of the expectation.
vce (hc2) divides $u_{j}^{2}$ by $1-h_{j j}$ itself, so it should yield better estimates if the residuals really are homoskedastic. vce (hc3) divides $u_{j}^{2}$ by $\left(1-h_{j j}\right)^{2}$ and has no such clean interpretation. Davidson and MacKinnon (1993) show that $u_{j}^{2} /\left(1-h_{j j}\right)^{2}$ approximates a more complicated estimator that they obtain by jackknifing (MacKinnon and White 1985). Angrist and Pischke (2009) also illustrate the relative merits of these adjustments.

Here are the results of refitting our efficiency model using vce(hc2) and vce(hc3):
. regress gpmw foreign, vce(hc2)


## Example 7: Standard errors for clustered data

The vce(cluster clustvarlist) and vce(hc2 clustvar) options relax the assumption of independence. Below, we have 28,534 observations on 4,711 women aged $14-46$ years. Data were collected on these women between 1968 and 1988. We are going to fit a classic earnings model, and we begin by ignoring that the majority of the women in the dataset have multiple observations.

```
. use https://www.stata-press.com/data/r18/regsmpl, clear
(NLS women 14-26 in 1968)
```

. regress ln_wage age c.age\#c.age tenure

| Source | SS | df | MS |
| ---: | ---: | ---: | ---: |
| Model | 1054.52501 | 3 | 351.508335 |
| Residual | 5360.43962 | 28,097 | .190783344 |
| Total | 6414.96462 | 28,100 | .228290556 |


| Number of obs | $=$ | 28,101 |
| :--- | :--- | ---: |
| $\mathrm{~F}(3,28097)$ | $=$ | 1842.45 |
| Prob $>$ F | $=$ | 0.0000 |
| R-squared | $=$ | 0.1644 |
| Adj R-squared | $=$ | 0.1643 |
| Root MSE | $=$ | .43679 |


| ln_wage | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| age | .0752172 | .0034736 | 21.65 | 0.000 | .0684088 | .0820257 |
| c.age\#c.age | -.0010851 | .0000575 | -18.86 | 0.000 | -.0011979 | -.0009724 |
| tenure | .0390877 | .0007743 | 50.48 | 0.000 | .0375699 | .0406054 |
| _cons | .3339821 | .0504413 | 6.62 | 0.000 | .2351148 | .4328495 |

The number of observations in our model is 28,101 because Stata drops observations that have a missing value for one or more of the variables in the model. We can be reasonably certain that the standard errors reported above are meaningless. Without a doubt, a woman with higher-than-average wages in one year typically has higher-than-average wages in other years, and so the residuals are not independent. One way to deal with this is to use cluster-robust standard errors. We do this by specifying vce(cluster id) or vce(hc2 id), which treat only observations with different person ids as truly independent:

| . regress ln_wage age c.age\#c.age tenure, vce(cluster id) |  |  |  |
| :--- | :--- | :--- | :--- |
| Linear regression | Number of obs | $=$ | 28,101 |
|  | F(3, 4698) | $=$ | 748.82 |
|  | Prob $>\mathrm{F}$ | $=$ | 0.0000 |
|  | R-squared | $=$ | 0.1644 |
|  | Root MSE | $=$ | .43679 |

(Std. err. adjusted for 4,699 clusters in idcode)

| ln_wage | Coefficient | Robust |  |  |  |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |  |  |
| age | .0752172 | .0045711 | 16.45 | 0.000 | .0662557 | .0841788 |
|  |  |  |  |  |  |  |
| c.age\#c.age | -.0010851 | .0000778 | -13.94 | 0.000 | -.0012377 | -.0009325 |
| tenure | .0390877 | .0014425 | 27.10 | 0.000 | .0362596 | .0419157 |
| _cons | .3339821 | .0641918 | 5.20 | 0.000 | .208136 | .4598282 |

For comparison, we focus on the tenure coefficient, which in economics jargon can be interpreted as the rate of return for keeping your job. The $95 \%$ confidence interval we previously estimated - an interval we do not believe-is [ $0.038,0.041$ ]. The robust interval is twice as wide, being [ $0.036,0.042$ ]. For this example, vce(hc2 id) gives standard errors similar to vce(cluster id).

Another possible way to account for the lack of independence is to fit a random-effects model. Here is the random-effects result:

| Random-effects GLS regression Group variable: idcode |  |  |  | Number | of obs | = | 28,101 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | Number | of groups | $=$ | 4,699 |
| R-squared: |  |  |  | Obs per group: |  |  |  |
| Within $=0.1370$ |  |  |  |  | min |  | 1 |
| Between $=0.2154$ |  |  |  |  | avg | $=$ | 6.0 |
| Overall $=0.1608$ |  |  |  |  | max | = | 15 |
|  |  |  |  | Wald ch | i2(3) | = | 4717.05 |
| corr (u_i, X) = 0 (assumed) |  |  |  | Prob > | chi2 | = | 0.0000 |
| $1 n$ wage | Coefficient | Std. err. | z | $\mathrm{P}>\|z\|$ | [95\% con | nf | interval] |
| age | . 0568296 | . 0026958 | 21.08 | 0.000 | . 0515459 |  | . 0621132 |
| c.age\#c.age | -. 0007566 | . 0000447 | -16.93 | 0.000 | -. 0008441 |  | -. 000669 |
| tenure | . 0260135 | . 0007477 | 34.79 | 0.000 | . 0245481 |  | . 0274789 |
| _cons | . 6136792 | . 0394611 | 15.55 | 0.000 | . 5363368 |  | . 6910216 |
| sigma_u | . 33542449 |  |  |  |  |  |  |
| sigma_e | . 29674679 | (fraction of variance due to $u_{-} \mathrm{i}$ ) |  |  |  |  |  |
| rho | . 56095413 |  |  |  |  |  |  |

Robust regression estimated the $95 \%$ interval [ $0.036,0.042$ ], and xtreg (see [XT] xtreg) estimates [ $0.025,0.027$ ]. Which is better? The random-effects regression estimator assumes a lot. We can check some of these assumptions by performing a Hausman test. Using estimates (see [R] estimates store), we store the random-effects estimation results, and then we run the required fixed-effects regression to perform the test.
. estimates store random
. xtreg ln_wage age c.age\#c.age tenure, fe
Fixed-effects (within) regression $\quad$ Number of obs $=28,101$
Group variable: idcode $\quad$ Number of groups $=4,699$
R-squared:
Obs per group:
Within $=0.1375$
$\min =\quad 1$
Between $=0.2066$
$\operatorname{avg}=\quad 6.0$
Overall $=0.1568 \quad \max =\quad 15$


| ln_wage | Coefficient | Std. err. | t | $P>\|t\|$ | [95\% con | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| age | . 0522751 | . 002783 | 18.78 | 0.000 | . 0468202 | . 05773 |
| c.age\#c.age | -. 0006717 | . 0000461 | -14.56 | 0.000 | -. 0007621 | -. 0005813 |
| tenure | . 021738 | . 000799 | 27.21 | 0.000 | . 020172 | . 023304 |
| _cons | . 687178 | . 0405944 | 16.93 | 0.000 | . 6076103 | . 7667456 |
| sigma_u | . 38743138 | (fraction of variance due to u_i) |  |  |  |  |
| sigma_e | . 29674679 |  |  |  |  |  |
| rho | . 6302569 |  |  |  |  |  |

. hausman . random

```
—— Coefficients ——_
```

|  | (b) | (B) random | $\begin{gathered} \text { (b-B) } \\ \text { Difference } \end{gathered}$ | $\begin{gathered} \text { sqrt (diag } \left.\left(V \_b-V \_B\right)\right) \\ \text { Std. err. } \end{gathered}$ |
| :---: | :---: | :---: | :---: | :---: |
| age | . 0522751 | . 0568296 | -. 0045545 | . 0006913 |
| c.age\#c.age | -. 0006717 | -. 0007566 | . 0000849 | . 0000115 |
| tenure | . 021738 | . 0260135 | -. 0042756 | . 0002816 |

```
                        b \(=\) Consistent under HO and Ha ; obtained from xtreg.
            \(B=\) Inconsistent under Ha, efficient under HO; obtained from xtreg.
Test of HO: Difference in coefficients not systematic
    chi2 (3) \(=(b-B) \cdot\left[\left(V \_b-V \_B\right)^{\wedge}(-1)\right](b-B)\)
        \(=336.62\)
Prob \(>\) chi2 \(=0.0000\)
```

The Hausman test casts grave suspicions on the random-effects model we just fit, so we should be careful in interpreting those results.

Meanwhile, our robust regression results still stand, as long as we are careful about the interpretation. The correct interpretation is that, if the data collection were repeated (on women sampled the same way as in the original sample), and if we were to refit the model, $95 \%$ of the time we would expect the estimated coefficient on tenure to be in the range [0.036, 0.042].

Even with robust regression, we must be careful about going beyond that statement. Here the Hausman test is probably picking up something that differs within and between person, which would cast doubt on our robust regression model in terms of interpreting [ $0.036,0.042$ ] to contain the rate of return for keeping a job, economywide, for all women, without exception.

Let's take this example a bit further by also recognizing workers with the same education level, grade, may be more alike than those with different education levels. Here we will use multiway clustering (Cameron, Gelbach, and Miller 2008) to account for correlations within individuals over years and within education levels, assuming observations from different people and different education level are independent.

```
. regress ln_wage age c.age#c.age tenure, vce(cluster idcode grade) clustertable
    min = 19
    avg = 3,138
    max = 4,697
```

Number of obs $=28,099$
Cluster comb $=3$
$\mathrm{F}(3,18)=247.23$
Prob > F $\quad=0.0000$
$R$-squared $\quad=0.1644$
Adj R-squared $=0.1644$
Root MSE $=0.4368$

Linear regression
Clusters per comb.:

| Cluster combination | Clusters <br> per comb. |
| ---: | ---: |
| idcode | 4,697 |
| grade | 19 |
| idcode\#grade | 4,697 |


| ln_wage | Coefficient | Robust |  |  |  |  |
| ---: | ---: | ---: | :---: | ---: | ---: | ---: | ---: |
| std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |  |  |
| age | .0751665 | .018361 | 4.09 | 0.001 | .0365914 | .1137415 |
|  |  |  |  |  |  |  |
| c.age\#c.age | -.0010842 | .0002504 | -4.33 | 0.000 | -.0016103 | -.0005581 |
| tenure | .0391104 | .0018302 | 21.37 | 0.000 | .0352653 | .0429554 |
| _cons | .334631 | .2769961 | 1.21 | 0.243 | -.2473162 | .9165782 |

Cluster combinations formed by idcode and grade.
With the clustertable option, the output includes a table that describes the cluster combinations and reports the number of levels for each cluster combination. We clustered on two variables, so there are three cluster combinations, and, in general, for $p$ cluster variables, there are $p^{2}-1$ cluster combinations. The $t$-statistic degrees of freedom is chosen from the cluster combination that has the smallest number of levels. In this case, we have $19-1=18$ degrees of freedom, and the Wald test $F$ statistic, computed from the cluster-robust VCE, has a denominator degrees of freedom of 18 . The $95 \%$ confidence interval for tenure increases slightly to [ $0.035,0.043$ ]. Yet the confidence intervals here are not strictly comparable; this model is fit to two fewer observations because of missing values in the cluster variable grade, and we conjecture that observations are correlated within grade.

## Weighted regression

regress can perform weighted and unweighted regression. We indicate the weight by specifying the [weight] qualifier.
$>$ Example 8: Using means as regression variables
We have census data recording the deathrate (drate) and median age (medage) for each state. The data also record the region of the country in which each state is located and the overall population of the state:


Sorted by:
We can use factor variables to include dummy variables for region. Because the variables in the regression reflect means rather than individual observations, the appropriate method of estimation is analytically weighted least squares (Davidson and MacKinnon 2004, 261-262), where the weight is total population:

```
. regress drate medage i.region [aweight=pop]
(sum of wgt is 225,907,472)
```

| Source | SS | df | MS | Number of obs | = | 50 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | F (4, 45) | = | 37.21 |
| Model | 4096.6093 | 4 | 1024.15232 | Prob > F | = | 0.0000 |
| Residual | 1238.40987 | 45 | 27.5202192 | R -squared | = | 0.7679 |
|  |  |  |  | Adj R-squared | = | 0.7472 |
| Total | 5335.01916 | 49 | 108.877942 | Root MSE | = | 5.246 |


| drate | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| :---: | ---: | :---: | :---: | :---: | :---: | ---: |
| medage | 4.283183 | .5393329 | 7.94 | 0.000 | 3.196911 | 5.369455 |
| region |  |  |  |  |  |  |
| N Cntrl | .3138738 | 2.456431 | 0.13 | 0.899 | -4.633632 | 5.26138 |
| South | -1.438452 | 2.320244 | -0.62 | 0.538 | -6.111663 | 3.234758 |
| West | -10.90629 | 2.681349 | -4.07 | 0.000 | -16.30681 | -5.505777 |
| _cons | -39.14727 | 17.23613 | -2.27 | 0.028 | -73.86262 | -4.431915 |

To weight the regression by population, we added the qualifier [aweight=pop] to the end of the regress command. Stata informed us that the sum of the weight is $2.2591 \times 10^{8}$; there were approximately 226 million people residing in the United States according to our 1980 data.

In the weighted regression, we see that the coefficient on West is statistically significant but that the coefficients on $N$ Cntrl and South are not. We use testparm to test the joint significance of the region variable. Because we fit a weighted regression, testparm uses the appropriately weighted variance-covariance matrix.

```
. testparm i.region
(1) 2.region = 0
(2) 3.region = 0
( 3) 4.region = 0
    F( 3, 45) = 9.84
        Prob > F = 0.0000
```

The results indicate that the region variables are jointly significant. Note that we could have performed this same test by typing contrast region. You may prefer to use the contrast command because, in addition to the joint test, you can perform other tests such as comparisons of each region's mean to the grand mean; see $[R]$ contrast for more information.
regress also accepts frequency weights (fweights). Frequency weights are appropriate when the data do not reflect cell means but instead represent replicated observations. Specifying aweights or fweights will not change the parameter estimates, but it will change the corresponding significance levels.

For instance, if we specified [fweight=pop] in the weighted regression example above-which would be statistically incorrect-Stata would treat the data as if the data represented 226 million independent observations on death rates and median age. The data most certainly do not represent that-they represent 50 observations on state averages.

With aweights, Stata treats the number of observations on the process as the number of observations in the data. When we specify fweights, Stata treats the number of observations as if it were equal to the sum of the weights; see Methods and formulas below.

## - Technical note

A frequent inquiry sent to StataCorp Technical Services is to describe the effect of specifying [aweight $=$ exp] with regress in terms of transformation of the dependent and independent variables. The mechanical answer is that typing

```
. regress y x1 x2 [aweight=n]
```

is equivalent to fitting the model

$$
y_{j} \sqrt{n_{j}}=\beta_{0} \sqrt{n_{j}}+\beta_{1} x_{1 j} \sqrt{n_{j}}+\beta_{2} x_{2 j} \sqrt{n_{j}}+u_{j} \sqrt{n_{j}}
$$

This regression will reproduce the coefficients and covariance matrix produced by the aweighted regression. The mean squared errors (estimates of the variance of the residuals) will, however, be different. The transformed regression reports $s_{t}^{2}$, an estimate of $\operatorname{Var}\left(u_{j} \sqrt{n_{j}}\right)$. The aweighted regression reports $s_{a}^{2}$, an estimate of $\operatorname{Var}\left(u_{j} \sqrt{n_{j}} \sqrt{N / \sum_{k} n_{k}}\right)$, where $N$ is the number of observations. Thus,

$$
\begin{equation*}
s_{a}^{2}=\frac{N}{\sum_{k} n_{k}} s_{t}^{2}=\frac{s_{t}^{2}}{\bar{n}} \tag{1}
\end{equation*}
$$

The logic for this adjustment is as follows: Consider the model

$$
y=\beta_{0}+\beta_{1} x_{1}+\beta_{2} x_{2}+u
$$

Assume that, were this model fit on individuals, $\operatorname{Var}(u)=\sigma_{u}^{2}$, a constant. Assume that individual data are not available; what is available are averages $\left(\bar{y}_{j}, \bar{x}_{1 j}, \bar{x}_{2 j}\right)$ for $j=1, \ldots, N$, and each average is calculated over $n_{j}$ observations. Then it is still true that

$$
\bar{y}_{j}=\beta_{0}+\beta_{1} \bar{x}_{1 j}+\beta_{2} \bar{x}_{2 j}+\bar{u}_{j}
$$

where $\bar{u}_{j}$ is the average of $n_{j}$ mean 0 , variance $\sigma_{u}^{2}$ deviates and has variance $\sigma_{\bar{u}}^{2}=\sigma_{u}^{2} / n_{j}$. Thus, multiplying through by $\sqrt{n_{j}}$ produces

$$
\bar{y}_{j} \sqrt{n_{j}}=\beta_{0} \sqrt{n_{j}}+\beta_{1} \bar{x}_{1 j} \sqrt{n_{j}}+\beta_{2} \bar{x}_{2 j} \sqrt{n_{j}}+\bar{u}_{j} \sqrt{n_{j}}
$$

and $\operatorname{Var}\left(\bar{u}_{j} \sqrt{n_{j}}\right)=\sigma_{u}^{2}$. The mean squared error, $s_{t}^{2}$, reported by fitting this transformed regression is an estimate of $\sigma_{u}^{2}$. The coefficients and covariance matrix could also be obtained by aweighted regress. The only difference would be in the reported mean squared error, which from (1) is $\sigma_{u}^{2} / \bar{n}$. On average, each observation in the data reflects the averages calculated over $\bar{n}=\sum_{k} n_{k} / N$ individuals, and thus this reported mean squared error is the average variance of an observation in the dataset. We can retrieve the estimate of $\sigma_{u}^{2}$ by multiplying the reported mean squared error by $\bar{n}$.

More generally, aweights are used to solve general heteroskedasticity problems. In these cases, we have the model

$$
y_{j}=\beta_{0}+\beta_{1} x_{1 j}+\beta_{2} x_{2 j}+u_{j}
$$

and the variance of $u_{j}$ is thought to be proportional to $a_{j}$. If the variance is proportional to $a_{j}$, it is also proportional to $\alpha a_{j}$, where $\alpha$ is any positive constant. Not quite arbitrarily, but with no loss of generality, we could choose $\alpha=\sum_{k}\left(1 / a_{k}\right) / N$, the average value of the inverse of $a_{j}$. We can then write $\operatorname{Var}\left(u_{j}\right)=k \alpha a_{j} \sigma^{2}$, where $k$ is the constant of proportionality that is no longer a function of the scale of the weights.

Dividing this regression through by the $\sqrt{a_{j}}$,

$$
y_{j} / \sqrt{a_{j}}=\beta_{0} / \sqrt{a_{j}}+\beta_{1} x_{1 j} / \sqrt{a_{j}}+\beta_{2} x_{2 j} / \sqrt{a_{j}}+u_{j} / \sqrt{a_{j}}
$$

produces a model with $\operatorname{Var}\left(u_{j} / \sqrt{a_{j}}\right)=k \alpha \sigma^{2}$, which is the constant part of $\operatorname{Var}\left(u_{j}\right)$. This variance is a function of $\alpha$, the average of the reciprocal weights; if the weights are scaled arbitrarily, then so is this variance.

We can also fit this model by typing

```
. regress y x1 x2 [aweight=1/a]
```

This input will produce the same estimates of the coefficients and covariance matrix; the reported mean squared error is, from (1), $\left\{N / \sum_{k}\left(1 / a_{k}\right)\right\} k \alpha \sigma^{2}=k \sigma^{2}$. This variance is independent of the scale of $a_{j}$.

## Video examples

## Simple linear regression in Stata

Fitting and interpreting regression models: Linear regression with categorical predictors
Fitting and interpreting regression models: Linear regression with continuous predictors
Fitting and interpreting regression models: Linear regression with continuous and categorical predictors

## Stored results

regress stores the following in e():
Scalars
e(N)
e(mss)
e(df_m)
e(rss)
e(df_r)
e(r2)
e(r2_a)
e(F)
e(rmse)
e(11)
e(11_0)
e(N_clust)
e(rank)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(wtype)
e(wexp)
e(model)
e(title)
e(clustvar)
e(cluster\#)
e(vce)
e(vcetype)
e(properties)
e(estat_cmd)
$e$ (predict)
e(marginsok)
e(asbalanced)
e(asobserved)
Matrices
e(b)
e(V)
e(beta)
e(V_modelbased)
e(adj_df) e(kcluster)
Functions

In addition to the above, the following is stored in r() :

Matrices
$r$ (table) matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r -class command is run after the estimation command.

## Methods and formulas

Methods and formulas are presented under the following headings:
Coefficient estimation and ANOVA table
Weighted regression
A general notation for the robust variance calculation
Robust calculation for regress

## Coefficient estimation and ANOVA table

Variables printed in lowercase and not boldfaced (for example, $x$ ) are scalars. Variables printed in lowercase and boldfaced (for example, $\mathbf{x}$ ) are column vectors. Variables printed in uppercase and boldfaced (for example, $\mathbf{X}$ ) are matrices.

Let $\mathbf{X}$ denote the matrix of observations on the right-hand-side variables, $\mathbf{y}$ the vector of observations on the left-hand-side variables. Define $\mathbf{A}$ as $\mathbf{X}^{\prime} \mathbf{X}$ and a as $\mathbf{X}^{\prime} \mathbf{y}$. The coefficient vector $\mathbf{b}$ is defined as $\mathbf{A}^{-1} \mathbf{a}$. Although not shown in the notation, unless hascons is specified, $\mathbf{A}$ and a are accumulated in deviation form and the constant is calculated separately. This comment applies to all statistics listed below.

The total sum of squares, TSS, equals $\mathbf{y}^{\prime} \mathbf{y}$ if there is no intercept and $\mathbf{y}^{\prime} \mathbf{y}-\left\{\left(\mathbf{1}^{\prime} \mathbf{y}\right)^{2} / n\right\}$ otherwise. The degrees of freedom is $n-c$, where $n$ is the number of observations and $c=1$ if there is a constant in the regression and 0 otherwise.

The residual sum of squares, RSS, is defined as $(\mathbf{y}-\mathbf{X b})^{\prime}(\mathbf{y}-\mathbf{X b})$. The degrees of freedom is $n-k$, where $n$ is the number of observations and $k$ is the number of right-hand-side variables (including the constant).

The model sum of squares, MSS, equals TSS - RSS. The degrees of freedom is $k-c$.
The mean squared error, $s^{2}$, is defined as $\operatorname{RSS} /(n-k)$. The root mean squared error is $s$, its square root.

The $F$ statistic with $k-c$ and $n-k$ degrees of freedom is defined as

$$
F=\frac{\mathrm{MSS}}{(k-c) s^{2}}
$$

The $R^{2}$ is defined as $R^{2}=1-\mathrm{RSS} / \mathrm{TSS}$.
The adjusted $R^{2}$ is defined as $R_{a}^{2}=1-\left(1-R^{2}\right)(n-c) /(n-k)$.
The conventional estimate of variance is $s^{2} \mathbf{A}^{-1}$. The calculation of variance estimates when robust variance estimates are specified is described below.

## Weighted regression

Let $\mathbf{v}$ be a column vector of weights specified by the user. Let $\mathbf{w}$ be a column vector of normalized weights, $\mathbf{w}=\left\{\mathbf{v} /\left(\mathbf{1}^{\prime} \mathbf{v}\right)\right\}\left(\mathbf{1}^{\prime} \mathbf{1}\right)$. For fweights, $\mathbf{w}=\mathbf{v}$. For historical reasons, iweights are treated like fweights when robust standard errors are not specified. Instead, when vce (robust), vce(cluster clustvarlist), vce(hc2), or vce(hc3) is specified, iweights are treated like aweights.

If the user specifies weights, the number of observations, $n$, in the above formulas is defined as $\mathbf{1}^{\prime} \mathbf{w}$. For iweights, this is truncated to an integer. The sum of the weights is $\mathbf{1}^{\prime} \mathbf{v} . \mathbf{X}^{\prime} \mathbf{X}, \mathbf{X}^{\prime} \mathbf{y}$, and $\mathbf{y}^{\prime} \mathbf{y}$ are replaced in the above formulas by $\mathbf{X}^{\prime} \mathbf{D X}, \mathbf{X}^{\prime} \mathbf{D} \mathbf{y}$, and $\mathbf{y}^{\prime} \mathbf{D} \mathbf{y}$, respectively, where $\mathbf{D}$ is a diagonal matrix whose diagonal elements are the elements of $\mathbf{w}$.

## A general notation for the robust variance calculation

Put aside all context of linear regression and the notation that goes with it - we will return to it. First, we are going to establish a notation for describing robust variance calculations.

The calculation formula for the robust variance calculation is

$$
\widehat{\mathcal{V}}=q_{c} \widehat{\mathbf{V}}\left(\sum_{k=1}^{M} \mathbf{u}_{k}^{(G) \prime} \mathbf{u}_{k}^{(G)}\right) \widehat{\mathbf{V}}
$$

where

$$
\mathbf{u}_{k}^{(G)}=\sum_{j \in G_{k}} w_{j} \mathbf{u}_{j}
$$

$G_{1}, G_{2}, \ldots, G_{M}$ are the clusters specified by vce (cluster clustvarlist) when clustvarlist contains only one variable, and $w_{j}$ are the user-specified weights, normalized if aweights or pweights are specified and equal to 1 if no weights are specified.

For fweights without clusters, the variance formula is

$$
\widehat{\mathcal{V}}=q_{c} \widehat{\mathbf{V}}\left(\sum_{j=1}^{N} w_{j} \mathbf{u}_{j}^{\prime} \mathbf{u}_{j}\right) \widehat{\mathbf{V}}
$$

which is the same as expanding the dataset and making the calculation on the unweighted data.
If vce(cluster clustvarlist) is not specified, $M=N$, and each cluster contains 1 observation. The inputs into this calculation are

- $\widehat{\mathbf{V}}$, which is typically a conventionally calculated variance matrix;
- $\mathbf{u}_{j}, j=1, \ldots, N$, a row vector of scores; and
- $q_{\mathrm{c}}$, a constant finite-sample adjustment.

Thus, we can now describe how estimators apply the robust calculation formula by defining $\widehat{\mathbf{V}}, \mathbf{u}_{j}$, and $q_{\mathrm{c}}$.

Two definitions are popular enough for $q_{\mathrm{c}}$ to deserve a name. The regression-like formula for $q_{\mathrm{c}}$ (Fuller et al. 1986) is

$$
q_{\mathrm{c}}=\frac{N-1}{N-k} \frac{M}{M-1}
$$

where $M$ is the number of clusters and $N$ is the number of observations. For weights, $N$ refers to the sum of the weights if weights are frequency weights and the number of observations in the dataset (ignoring weights) in all other cases. Also note that, weighted or not, $M=N$ when vce(cluster clustvarlist) is not specified, and then $q_{\mathrm{c}}=N /(N-k)$.

The asymptotic-like formula for $q_{\mathrm{c}}$ is

$$
q_{\mathrm{c}}=\frac{M}{M-1}
$$

where $M=N$ if vce(cluster clustvarlist) is not specified.
See [U] 20.22 Obtaining robust variance estimates and [P] _robust for a discussion of the robust variance estimator and a development of these formulas.

## Robust calculation for regress

For regress, $\widehat{\mathbf{V}}=\mathbf{A}^{-1}$. The other terms are vce(robust), but not vce(hc2) or vce(hc3),

$$
\mathbf{u}_{j}=\left(y_{j}-\mathbf{x}_{j} \mathbf{b}\right) \mathbf{x}_{j}
$$

and $q_{\mathrm{c}}$ is given by its regression-like definition. vce(hc2),

$$
\mathbf{u}_{j}=\frac{1}{\sqrt{1-h_{j j}}}\left(y_{j}-\mathbf{x}_{j} \mathbf{b}\right) \mathbf{x}_{j}
$$

where $q_{\mathrm{c}}=1$ and $h_{j j}=\mathbf{x}_{j}\left(\mathbf{X}^{\prime} \mathbf{X}\right)^{-1} \mathbf{x}_{j}{ }^{\prime}$. vce(hc3),

$$
\mathbf{u}_{j}=\frac{1}{1-h_{j j}}\left(y_{j}-\mathbf{x}_{j} \mathbf{b}\right) \mathbf{x}_{j}
$$

where $q_{\mathrm{c}}=1$ and $h_{j j}=\mathbf{x}_{j}\left(\mathbf{X}^{\prime} \mathbf{X}\right)^{-1} \mathbf{x}_{j}{ }^{\prime}$. vce(hc2 clustvar),

$$
\mathbf{u}_{j}=\left(\mathbf{y}_{j}-\mathbf{X}_{j} \mathbf{b}\right)^{\prime}\left(\mathbf{I}_{G_{j}}-\mathbf{H}_{j j}\right)^{-\frac{1}{2}} \mathbf{X}_{j}
$$

where $q_{\mathrm{c}}=1, \mathbf{H}_{j j}=\mathbf{X}_{j}\left(\mathbf{X}^{\prime} \mathbf{X}\right)^{-1} \mathbf{X}_{j}{ }^{\prime}$ for cluster $j$ of size $G_{j}, G_{j} \times k$ data matrix $\mathbf{X}_{j}$, and $G_{j} \times 1$ vector $\mathbf{y}_{j}$. $\left(\mathbf{I}_{G_{j}}-\mathbf{H}_{j j}\right)^{-\frac{1}{2}}$ is the inverse of the symmetric square root of $\left(\mathbf{I}_{G_{j}}-\mathbf{H}_{j j}\right)$ (Bell and McCaffrey 2002).
vce(hc2 [clustvar], dfadjust) directs regress to compute the adjusted degrees of freedom described by Imbens and Kolesár (2016). Define the $N \times M$ matrix G such that the $j$ th column is

$$
\left(\mathbf{I}_{N}-\mathbf{H}\right)_{j}{ }^{\prime}\left(\mathbf{I}_{G_{j}}-\mathbf{H}_{j j}\right)^{-\frac{1}{2}} \mathbf{X}_{j}\left(\mathbf{X}^{\prime} \mathbf{X}\right)^{-1} e_{l}
$$

where $j=1, \ldots, M, e_{l}$ is a unary vector for the $l$ th regressor, $l=1, \ldots, k$, and $\left(\mathbf{I}_{N}-\mathbf{H}\right)_{j}$ is the $G_{j} \times N$ subset of the $N \times N$ matrix $\left(\mathbf{I}_{N}-\mathbf{H}\right)$ for cluster $j$. The Bell and McCaffrey (2002) adjusted degrees of freedom for the $l$ th regressor is

$$
\begin{aligned}
K_{l} & =\frac{\operatorname{tr}\left(\mathbf{G}^{\prime} \mathbf{G}\right)^{2}}{\operatorname{tr}\left(\left(\mathbf{G}^{\prime} \mathbf{G}\right)^{2}\right)} \\
& =\frac{\left(\sum_{i=1}^{M} \lambda_{i}\right)^{2}}{\sum_{i=1}^{M} \lambda_{i}^{2}}
\end{aligned}
$$

where $\operatorname{tr}(\cdot)$ is the trace function and $\lambda_{i}$ are the eigenvalues of $\mathbf{G}^{\prime} \mathbf{G}$.

When $M$ is large, or when there are no clusters and $M=N$, computing the eigenvalues can be time consuming. We define

$$
\begin{aligned}
\mathbf{a}_{j} & =\left(\mathbf{I}_{G_{j}}-\mathbf{H}_{j j}\right)^{-\frac{1}{2}} \mathbf{X}_{j}\left(\mathbf{X}^{\prime} \mathbf{X}\right)^{-1} \mathbf{e}_{L, k} \\
\mathbf{b}_{j} & =\mathbf{H}_{j} \mathbf{a}_{j} \\
\mathbf{A} & =\left(\mathbf{a}_{1}^{\prime} \mathbf{a}_{1}, \ldots, \mathbf{a}_{M}^{\prime} \mathbf{a}_{M}\right)^{\prime}=\left(A_{1}, A_{2}, \ldots, A_{M}\right)^{\prime} \\
\mathbf{B} & =\left(\mathbf{b}_{1}, \ldots, \mathbf{b}_{M}\right)=\left(\begin{array}{cccc}
B_{1,1} & B_{1,2} & \cdots & B_{1, M} \\
\vdots & \vdots & \ddots & \vdots \\
B_{N, 1} & B_{N, 2} & \cdots & B_{N, M}
\end{array}\right)
\end{aligned}
$$

for $j=1, \ldots, M$ and $\mathbf{H}_{j}=\mathbf{X}\left(\mathbf{X}^{\prime} \mathbf{X}\right)^{-1} \mathbf{X}_{j}^{\prime}$. Then $\mathbf{G}^{\prime} \mathbf{G}=\operatorname{diag}(\mathbf{A})-\mathbf{B}^{\prime} \mathbf{B}$ (Kolesár 2021).
We now express the adjusted degrees of freedom as

$$
K_{l}=\frac{\left(\sum_{j=1}^{M} A_{j}-\sum_{i=1}^{N} \sum_{j=1}^{M} B_{i j}^{2}\right)^{2}}{\sum_{j=1}^{M} A_{j}^{2}-2 \sum_{j=1}^{M} A_{j} \sum_{i=1}^{N} B_{i j}^{2}+\sum_{j_{1}=1}^{M} \sum_{j_{2}=1}^{M}\left(\mathbf{b}_{j_{1}}^{\prime} \mathbf{b}_{j_{2}}\right)^{2}}
$$

which can be computed efficiently in Mata and using QR decomposition. For example, by decomposing $\mathbf{X}=\mathbf{Q R}$, where $\mathbf{Q}$ is $n \times k$ and orthonormal and $\mathbf{R}$ is $k \times k$ and upper triangular, we can rewrite the matrix $\mathbf{B}$ so that it has dimension $k \times M$ instead of $N \times M$.

When weights are specified, we use the weighted covariate matrix $\widetilde{\mathbf{X}}=\operatorname{diag}(\mathbf{w})^{\frac{1}{2}} \mathbf{X}$ and its corresponding projection matrix $\widetilde{\mathbf{H}}$, as well as the cluster covariance matrices $\widetilde{\mathbf{X}}_{j}$, their projection matrices $\widetilde{\mathbf{H}}_{j j}$, and weighted residuals $\widetilde{\varepsilon}_{j}=\operatorname{diag}\left(\mathbf{w}_{j}\right)^{\frac{1}{2}} \widehat{\varepsilon}_{j}$. When frequency weights are specified without clusters, we substitute $1 / \sqrt{1-h_{j j}}$ with $1 / \sqrt{w_{j}-\widetilde{h}_{j j}}$, where $j=1, \ldots, N$. Also, when there are no clusters, the weights are included in the degrees-of-freedom algebra

$$
K_{l}=\frac{\left(\sum_{j=1}^{N} w_{j} A_{j}-\sum_{i=1}^{k} \sum_{j=1}^{N} B_{i j}^{2}\right)^{2}}{\sum_{j=1}^{N} w_{j} A_{j}^{2}-2 \sum_{j=1}^{N} A_{j} \sum_{i=1}^{k} B_{i j}^{2}+\sum_{j_{1}=1}^{N} \sum_{j_{2}=1}^{N}\left(\mathbf{b}_{j_{1}}^{\prime} \mathbf{b}_{j_{2}}\right)^{2}}
$$

Here we substituted $k$ for $N$ in the row dimension of $\mathbf{B}$ as it is when using QR decomposition to perform the computations.

With weights, the vce(hc2 clustvar) computation discussed above is modified to

$$
\begin{aligned}
\mathbf{u}_{j} & =\left(\operatorname{diag}\left(\mathbf{w}_{j}\right)^{\frac{1}{2}}\left(\mathbf{y}_{j}-\mathbf{X}_{j} \mathbf{b}\right)\right)^{\prime}\left(\mathbf{I}_{G_{j}}-\widetilde{\mathbf{H}}_{j j}\right)^{-\frac{1}{2}} \widetilde{\mathbf{X}}_{j} \\
& =\left(\widetilde{\mathbf{y}}_{j}-\widetilde{\mathbf{X}}_{j} \mathbf{b}\right)^{\prime}\left(\mathbf{I}_{G_{j}}-\widetilde{\mathbf{H}}_{j j}\right)^{-\frac{1}{2}} \widetilde{\mathbf{X}}_{j}
\end{aligned}
$$

## Multiway clustering

When you type vce(cluster clustvarlist) with more than one variable, the variance-covariance estimator uses multiway cluster-robust variance estimation. This is carried out by estimating the robust VCE for all combinations of the specified cluster variables and summing. For $p$ cluster variables, there will be $P=2^{p}-1$ cluster variable combinations. Let $\mathbf{V}_{i}$ be the $i$ th robust VCE, $i=1, \ldots, P$. Define $S_{j}, j=1, \ldots, p$, as the set of indices $i$ involving $j$ cluster variables. The size, or cardinality, of $S_{j}$ is $\left|S_{j}\right|=\binom{p}{j}$ and $\sum_{j=1}^{p}\binom{p}{j}=2^{p}-1$. For example, for $p=4,\left|S_{1}\right|=\binom{4}{1}=4,\left|S_{2}\right|=\binom{4}{2}=6$, $\left|S_{3}\right|=\binom{4}{3}=4$, and $\left|S_{4}\right|=\binom{4}{4}=1$. The multiway cluster-robust VCE is then

$$
\mathbf{V}_{*}=\sum_{j=1}^{p}(-1)^{j-1} \sum_{i \in S_{j}} \mathbf{V}_{i}
$$

You are more likely to cluster on two or, maybe, three variables. In the case of two cluster variables, the computation would be

$$
\mathbf{V}_{*}=V_{1}+V_{2}-V_{12}
$$

where $V_{1}$ corresponds to the variance-covariance computation clustering at the level of the first cluster, $V_{2}$ corresponds to the second level, and $V_{12}$ corresponds to the variance-covariance computation for the group formed by the intersection of both clustering levels.

An eigendecomposition on $\mathbf{V}_{*}$ ensures the matrix to be positive semidefinite. Let the columns of matrix $\mathbf{U}$ contain the eigenvectors of $\mathbf{V}_{*}$ and the vector contain $\mathbf{u}$, its eigenvalues. If $\mathbf{V}_{*}$ is not positive definite, some of the elements of $\mathbf{u}$ will be less than 0 . Let $\mathbf{u}_{+}$contain all the nonnegative elements of $\mathbf{u}$ and zeros where $u_{i}<0$. The matrix $\mathbf{V}_{+}=\mathbf{U} \cdot \operatorname{diag}\left(\mathbf{u}_{+}\right) \cdot \mathbf{U}^{\prime}$ will then be positive semidefinite.

## Acknowledgments

The robust estimate of variance was first implemented in Stata by Mead Over of the Center for Global Development, Dean Jolliffe of the World Bank, and Andrew Foster of the Department of Economics at Brown University (Over, Jolliffe, and Foster 1996).

The history of regression is long and complicated: the books by Stigler (1986) and Hald (1998) are devoted largely to the story. Legendre published first on least squares in 1805. Gauss published later in 1809, but he had the idea earlier. Gauss, and especially Laplace, tied least squares to a normal errors assumption. The idea of the normal distribution can itself be traced back to De Moivre in 1733. Laplace discussed a variety of other estimation methods and error assumptions over his long career, while linear models long predate either innovation. Most of this work was linked to problems in astronomy and geodesy.

A second wave of ideas started when Galton used graphical and descriptive methods on data bearing on heredity to develop what he called regression. His term reflects the common phenomenon that characteristics of offspring are positively correlated with those of parents but with regression slope such that offspring "regress toward the mean". Galton's work was rather intuitive: contributions from Pearson, Edgeworth, Yule, and others introduced more formal machinery, developed related ideas on correlation, and extended application into the biological and social sciences. So most of the elements of regression as we know it were in place by 1900.

Pierre-Simon Laplace (1749-1827) was born in Normandy and was early recognized as a remarkable mathematician. He weathered a changing political climate well enough to rise to Minister of the Interior under Napoleon in 1799 (although only for 6 weeks) and to be made a Marquis by Louis XVIII in 1817. He made many contributions to mathematics and physics, his two main interests being theoretical astronomy and probability theory (including statistics). Laplace transforms are named for him.

Adrien-Marie Legendre (1752-1833) was born in Paris (or possibly in Toulouse) and educated in mathematics and physics. He worked in number theory, geometry, differential equations, calculus, function theory, applied mathematics, and geodesy. The Legendre polynomials are named for him. His main contribution to statistics is as one of the discoverers of least squares. He died in poverty, having refused to bow to political pressures.

Johann Carl Friedrich Gauss (1777-1855) was born in Braunschweig (Brunswick), now in Germany. He studied there and at Göttingen. His doctoral dissertation at the University of Helmstedt was a discussion of the fundamental theorem of algebra. He made many fundamental contributions to geometry, number theory, algebra, real analysis, differential equations, numerical analysis, statistics, astronomy, optics, geodesy, mechanics, and magnetism. An outstanding genius, Gauss worked mostly in isolation in Göttingen.

Francis Galton (1822-1911) was born in Birmingham, England, into a well-to-do family with many connections: he and Charles Darwin were first cousins. After an unsuccessful foray into medicine, he became independently wealthy at the death of his father. Galton traveled widely in Europe, the Middle East, and Africa, and became celebrated as an explorer and geographer. His pioneering work on weather maps helped in the identification of anticyclones, which he named. From about 1865, most of his work was centered on quantitative problems in biology, anthropology, and psychology. In a sense, Galton (re)invented regression, and he certainly named it. Galton also promoted the normal distribution, correlation approaches, and the use of median and selected quantiles as descriptive statistics. He was knighted in 1909.

## References

Adkins, L. C., and R. C. Hill. 2011. Using Stata for Principles of Econometrics. 4th ed. Hoboken, NJ: Wiley.
Angrist, J. D., and J.-S. Pischke. 2009. Mostly Harmless Econometrics: An Empiricist's Companion. Princeton, NJ: Princeton University Press.
Becketti, S. 2020. Introduction to Time Series Using Stata. Rev. ed. College Station, TX: Stata Press.
Bell, R. M., and D. F. McCaffrey. 2002. Bias reduction in standard errors for linear regression with multi-stage samples. Survey Methodology 28: 169-181.

Cameron, A. C., J. B. Gelbach, and D. L. Miller. 2008. Bootstrap-based improvements for inference with clustered errors. Review of Economics and Statistics 90: 414-427. https://doi.org/10.1162/rest.90.3.414.
Cameron, A. C., and P. K. Trivedi. 2022. Microeconometrics Using Stata. 2nd ed. College Station, TX: Stata Press.
Chatterjee, S., and A. S. Hadi. 2012. Regression Analysis by Example. 5th ed. New York: Wiley.
Davidson, R., and J. G. MacKinnon. 1993. Estimation and Inference in Econometrics. New York: Oxford University Press.
——. 2004. Econometric Theory and Methods. New York: Oxford University Press.
Deb, P., E. C. Norton, and W. G. Manning. 2017. Health Econometrics Using Stata. College Station, TX: Stata Press.
Dohoo, I., W. Martin, and H. Stryhn. 2010. Veterinary Epidemiologic Research. 2nd ed. Charlottetown, Prince Edward Island: VER Inc.
——. 2012. Methods in Epidemiologic Research. Charlottetown, Prince Edward Island: VER Inc.
Dunnington, G. W. 1955. Gauss: Titan of Science. New York: Hafner Publishing.
Duren, P. 2009. Changing faces: The mistaken portrait of Legendre. Notices of the American Mathematical Society 56: 1440-1443.

Filoso, V. 2013. Regression anatomy, revealed. Stata Journal 13: 92-106.
Fuller, W. A., W. J. Kennedy, Jr., D. Schnell, G. Sullivan, and H. J. Park. 1986. PC CARP. Software package. Ames, IA: Statistical Laboratory, Iowa State University.

Gillham, N. W. 2001. A Life of Sir Francis Galton: From African Exploration to the Birth of Eugenics. New York: Oxford University Press.
Gillispie, C. C. 1997. Pierre-Simon Laplace, 1749-1827: A Life in Exact Science. Princeton, NJ: Princeton University Press.

Gould, W. W. 2011a. Understanding matrices intuitively, part 1. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2011/03/03/understanding-matrices-intuitively-part-1/.
__. 2011b. Use poisson rather than regress; tell a friend. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2011/08/22/use-poisson-rather-than-regress-tell-a-friend/.
Hald, A. 1998. A History of Mathematical Statistics from 1750 to 1930. New York: Wiley.
Hamilton, L. C. 2013. Statistics with Stata: Updated for Version 12. 8th ed. Boston: Brooks/Cole.
Hill, R. C., W. E. Griffiths, and G. C. Lim. 2018. Principles of Econometrics. 5th ed. Hoboken, NJ: Wiley.
Hirukawa, M., D. Liu, and A. Prokhorov. 2021. msreg: A command for consistent estimation of linear regression models using matched data. Stata Journal 21: 123-140.

Imbens, G. W., and M. Kolesár. 2016. Robust standard errors in small samples: Some practical advice. Review of Economics and Statistics 98: 701-712. https://doi.org/10.1162/REST_a_00552.
Kohler, U., and F. Kreuter. 2012. Data Analysis Using Stata. 3rd ed. College Station, TX: Stata Press.
Kolesár, M. 2021. Robust standard errors in small samples. https://cran.r-project.org/web/packages/dfadjust/vignettes/dfadjust.pdf.
Kutner, M. H., C. J. Nachtsheim, J. Neter, and W. Li. 2005. Applied Linear Statistical Models. 5th ed. New York: McGraw-Hill/Irwin.

MacKinnon, J. G., and H. L. White, Jr. 1985. Some heteroskedasticity-consistent covariance matrix estimators with improved finite sample properties. Journal of Econometrics 29: 305-325. https://doi.org/10.1016/0304-4076(85)90158-7.
Mehmetoglu, M., and T. G. Jakobsen. 2022. Applied Statistics Using Stata: A Guide for the Social Sciences. 2nd ed. Thousand Oaks, CA: Sage.

Mitchell, M. N. 2021. Interpreting and Visualizing Regression Models Using Stata. 2nd ed. College Station, TX: Stata Press.

Mooi, E., M. Sarstedt, and I. Mooi-Reci. 2018. Market Research: The Process, Data, and Methods Using Stata. Singapore: Springer.
Over, M., D. Jolliffe, and A. Foster. 1996. sg46: Huber correction for two-stage least squares estimates. Stata Technical Bulletin 29: 24-25. Reprinted in Stata Technical Bulletin Reprints, vol. 5, pp. 140-142. College Station, TX: Stata Press.

Pedace, R. 2013. Econometrics for Dummies. Hoboken, NJ: Wiley.
Peracchi, F. 2001. Econometrics. Chichester, UK: Wiley.
Plackett, R. L. 1972. Studies in the history of probability and statistics: XXIX. The discovery of the method of least squares. Biometrika 59: 239-251. https://doi.org/10.2307/2334569.
Pollock, P. H., III, and B. C. Edwards. 2019. A Stata Companion to Political Analysis. 4th ed. Thousand Oaks, CA: CQ Press.

Schonlau, M. 2005. Boosted regression (boosting): An introductory tutorial and a Stata plugin. Stata Journal 5: 330-354.
Stigler, S. M. 1986. The History of Statistics: The Measurement of Uncertainty before 1900. Cambridge, MA: Belknap Press.

Stock, J. H., and M. W. Watson. 2019. Introduction to Econometrics. 4th ed. New York: Pearson.
Studenmund, A. H. 2017. Using Econometrics: A Practical Guide. 7th ed. Boston: Pearson.
Vach, W. 2013. Regression Models as a Tool in Medical Research. Boca Raton, FL: CRC Press.
Weisberg, S. 2014. Applied Linear Regression. 4th ed. Hoboken, NJ: Wiley.
Wooldridge, J. M. 2010. Econometric Analysis of Cross Section and Panel Data. 2nd ed. Cambridge, MA: MIT Press.
-_. 2020. Introductory Econometrics: A Modern Approach. 7th ed. Boston: Cengage.

## Also see

$[R]$ regress postestimation - Postestimation tools for regress
$[R]$ regress postestimation diagnostic plots - Postestimation plots for regress
$[R]$ regress postestimation time series - Postestimation tools for regress with time series
[R] anova - Analysis of variance and covariance
[R] contrast - Contrasts and linear hypothesis tests after estimation
$[\mathrm{R}]$ hetregress - Heteroskedastic linear regression
[R] wildbootstrap - Wild cluster bootstrap inference
[BAYES] bayes: regress - Bayesian linear regression
[BMA] bmaregress - Bayesian model averaging for linear regression
[CAUSAL] Causal inference commands - Introduction to causal inference commands
[FMM] fmm: regress - Finite mixtures of linear regression models
[LASSO] Lasso intro - Introduction to lasso
[META] meta regress - Meta-analysis regression
[MI] Estimation - Estimation commands for use with mi estimate
[SEM] Example 6 - Linear regression
[SEM] Intro 5 - Tour of models
[SP] spregress - Spatial autoregressive models
[SVY] svy estimation - Estimation commands for survey data
[TS] forecast - Econometric model forecasting
[TS] mswitch - Markov-switching regression models
[U] 20 Estimation and postestimation commands

## Title

regress postestimation - Postestimation tools for regress

| Postestimation commands | Predictions | margins |
| :--- | :--- | :--- |
| DFBETA influence statistics | Tests for violation of assumptions | Variance inflation factors |
| Measures of effect size | Methods and formulas | Acknowledgments |
| References | Also see |  |

## Postestimation commands

The following postestimation commands are of special interest after regress:

| Command | Description |
| :--- | :--- |
| dfbeta | DFBETA influence statistics |
| estat hettest | tests for heteroskedasticity |
| estat imtest | information matrix test |
| estat ovtest | Ramsey regression specification-error test for omitted variables |
| estat szroeter | Szroeter's rank test for heteroskedasticity |
| estat vif | variance inflation factors for the independent variables |
| estat esize | $\eta^{2}, \varepsilon^{2}$, and $\omega^{2}$ effect sizes |
| estat moran | Moran's test of residual correlation with nearby residuals |
| lassogof | calculate goodness-of-fit predictions |

These commands are not appropriate with svy estimation results.

The following standard postestimation commands are also available:

| Command | Description |
| :---: | :---: |
| contrast | contrasts and ANOVA-style joint tests of estimates |
| estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estat (svy) | postestimation statistics for survey data |
| estimates | cataloging estimation results |
| etable | table of estimation results |
| *forecast | dynamic forecasts and simulations |
| *hausman | Hausman's specification test |
| lincom | point estimates, standard errors, testing, and inference for linear combinations of coefficients |
| linktest | link test for model specification |
| * lrtest | likelihood-ratio test |
| margins | marginal means, predictive margins, marginal effects, and average marginal effects |
| marginsplot | graph the results from margins (profile plots, interaction plots, etc.) |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients |
| predict | predictions and their SEs, leverage statistics, distance statistics, etc. |
| predictnl | point estimates, standard errors, testing, and inference for generalized predictions |
| pwcompare | pairwise comparisons of estimates |
| suest | seemingly unrelated estimation |
| test | Wald tests of simple and composite linear hypotheses |
| testnl | Wald tests of nonlinear hypotheses |

[^52]
## Predictions

## Description for predict

predict creates a new variable containing predictions such as linear predictions, residuals, standardized residuals, Studentized residuals, Cook's distance, leverage, probabilities, expected values, DFBETAs for varname, standard errors, COVRATIOs, DFITS, and Welsch distances.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

```
predict [type] newvar [if] [in] [, statistic]
```

statistic
Description

Main
xb
residuals
score
rstandard
rstudent
cooksd
leverage| hat
$\operatorname{pr}(a, b)$
$\mathrm{e}(a, b)$
ystar $(a, b)$

* dfbeta (varname)
stdp
stdf
stdr
* covratio
* dfits
* welsch
linear prediction; the default residuals
score; equivalent to residuals
standardized residuals
Studentized (jackknifed) residuals
Cook's distance
leverage (diagonal elements of hat matrix)
$\operatorname{Pr}\left(y_{j} \mid a<y_{j}<b\right)$
$E\left(y_{j} \mid a<y_{j}<b\right)$
$E\left(y_{j}^{*}\right), y_{j}^{*}=\max \left\{a, \min \left(y_{j}, b\right)\right\}$
DFBETA for varname
standard error of the linear prediction
standard error of the forecast
standard error of the residual
COVRATIO
DFITS
Welsch distance

Unstarred statistics are available both in and out of sample; type predict . . . if e (sample) ... if wanted only for the estimation sample. Starred statistics are calculated only for the estimation sample, even when if e(sample) is not specified.
rstandard, rstudent, cooksd, leverage, dfbeta(), stdf, stdr, covratio, dfits, and welsch are not available if any vce() other than vce (ols) was specified with regress.
xb , residuals, score, and stdp are the only options allowed with svy estimation results.
where $a$ and $b$ may be numbers or variables; $a$ missing ( $a \geq$.) means $-\infty$, and $b$ missing ( $b \geq$.) means $+\infty$; see [U] 12.2.1 Missing values.

## Options for predict

$\qquad$ Main
xb , the default, calculates the linear prediction.
residuals calculates the residuals.
score is equivalent to residuals in linear regression.
rstandard calculates the standardized residuals.
rstudent calculates the Studentized (jackknifed) residuals.
cooksd calculates the Cook's $D$ influence statistic (Cook 1977).
leverage or hat calculates the diagonal elements of the projection ("hat") matrix.
$\operatorname{pr}(a, b)$ calculates $\operatorname{Pr}\left(a<\mathbf{x}_{j} \mathbf{b}+u_{j}<b\right)$, the probability that $y_{j} \mid \mathbf{x}_{j}$ would be observed in the interval $(a, b)$.
$a$ and $b$ may be specified as numbers or variable names; $l b$ and $u b$ are variable names;
$\operatorname{pr}(20,30)$ calculates $\operatorname{Pr}\left(20<\mathbf{x}_{j} \mathbf{b}+u_{j}<30\right)$;
$\operatorname{pr}(l b, u b)$ calculates $\operatorname{Pr}\left(l b<\mathbf{x}_{j} \mathbf{b}+u_{j}<u b\right)$; and
$\operatorname{pr}(20, u b)$ calculates $\operatorname{Pr}\left(20<\mathbf{x}_{j} \mathbf{b}+u_{j}<u b\right)$.
$a$ missing $(a \geq$.$) means -\infty$; $\operatorname{pr}(., 30)$ calculates $\operatorname{Pr}\left(-\infty<\mathbf{x}_{j} \mathbf{b}+u_{j}<30\right)$;
$\operatorname{pr}(l b, 30)$ calculates $\operatorname{Pr}\left(-\infty<\mathbf{x}_{j} \mathbf{b}+u_{j}<30\right)$ in observations for which $l b \geq$.
and calculates $\operatorname{Pr}\left(l b<\mathbf{x}_{j} \mathbf{b}+u_{j}<30\right)$ elsewhere.
$b$ missing ( $b \geq$.) means $+\infty$; $\operatorname{pr}(20,$.$) calculates \operatorname{Pr}\left(+\infty>\mathbf{x}_{j} \mathbf{b}+u_{j}>20\right)$; $\operatorname{pr}(20, u b)$ calculates $\operatorname{Pr}\left(+\infty>\mathbf{x}_{j} \mathbf{b}+u_{j}>20\right)$ in observations for which $u b \geq$.
and calculates $\operatorname{Pr}\left(20<\mathbf{x}_{j} \mathbf{b}+u_{j}<u b\right)$ elsewhere.
$\mathrm{e}(a, b)$ calculates $E\left(\mathbf{x}_{j} \mathbf{b}+u_{j} \mid a<\mathbf{x}_{j} \mathbf{b}+u_{j}<b\right)$, the expected value of $y_{j} \mid \mathbf{x}_{j}$ conditional on $y_{j} \mid \mathbf{x}_{j}$ being in the interval $(a, b)$, meaning that $y_{j} \mid \mathbf{x}_{j}$ is truncated. $a$ and $b$ are specified as they are for pr() .
ystar $(a, b)$ calculates $E\left(y_{j}^{*}\right)$, where $y_{j}^{*}=a$ if $\mathbf{x}_{j} \mathbf{b}+u_{j} \leq a, y_{j}^{*}=b$ if $\mathbf{x}_{j} \mathbf{b}+u_{j} \geq b$, and $y_{j}^{*}=\mathbf{x}_{j} \mathbf{b}+u_{j}$ otherwise, meaning that $y_{j}^{*}$ is censored. $a$ and $b$ are specified as they are for pr() .
dfbeta (varname) calculates the DFBETA for varname, the difference between the regression coefficient when the $j$ th observation is included and excluded, said difference being scaled by the estimated standard error of the coefficient. varname must have been included among the regressors in the previously fitted model. The calculation is automatically restricted to the estimation subsample.
stdp calculates the standard error of the prediction, which can be thought of as the standard error of the predicted expected value or mean for the observation's covariate pattern. The standard error of the prediction is also referred to as the standard error of the fitted value.
stdf calculates the standard error of the forecast, which is the standard error of the point prediction for 1 observation. It is commonly referred to as the standard error of the future or forecast value. By construction, the standard errors produced by stdf are always larger than those produced by stdp; see Methods and formulas.
stdr calculates the standard error of the residuals.
covratio calculates COVRATIO (Belsley, Kuh, and Welsch 1980), a measure of the influence of the $j$ th observation based on considering the effect on the variance-covariance matrix of the estimates. The calculation is automatically restricted to the estimation subsample.
dfits calculates DFITS (Welsch and Kuh 1977) and attempts to summarize the information in the leverage versus residual-squared plot into one statistic. The calculation is automatically restricted to the estimation subsample.
welsch calculates Welsch distance (Welsch 1982) and is a variation on dfits. The calculation is automatically restricted to the estimation subsample.

## Remarks and examples for predict

Remarks are presented under the following headings:

Terminology<br>Fitted values and residuals<br>Prediction standard errors<br>Prediction with weighted data<br>Leverage statistics<br>Standardized and Studentized residuals<br>DFITS, Cook's Distance, and Welsch Distance<br>COVRATIO

## Terminology

Many of these commands concern identifying influential data in linear regression. This is, unfortunately, a field that is dominated by jargon, codified and partially begun by Belsley, Kuh, and Welsch (1980). In the words of Chatterjee and Hadi (1986, 416), "Belsley, Kuh, and Welsch's book, Regression Diagnostics, was a very valuable contribution to the statistical literature, but it unleashed on an unsuspecting statistical community a computer speak (à la Orwell), the likes of which we have never seen." Things have only gotten worse since then. Chatterjee and Hadi's (1986, 1988) own attempts to clean up the jargon did not improve matters (see Hoaglin and Kempthorne [1986], Velleman [1986], and Welsch [1986]). We apologize for the jargon, and for our contribution to the jargon in the form of inelegant command names, we apologize most of all.

Model sensitivity refers to how estimates are affected by subsets of our data. Imagine data on $y$ and $x$, and assume that the data are to be fit by the regression $y_{i}=\alpha+\beta x_{i}+\epsilon_{i}$. The regression estimates of $\alpha$ and $\beta$ are $a$ and $b$, respectively. Now imagine that the estimated $a$ and $b$ would be different if a small portion of the dataset, perhaps even one observation, were deleted. As a data analyst, you would like to think that you are summarizing tendencies that apply to all the data, but you have just been told that the model you fit is unduly influenced by one point or just a few points and that, as a matter of fact, there is another model that applies to the rest of the data-a model that you have ignored. The search for subsets of the data that, if deleted, would change the results markedly is a predominant theme of this entry.

There are three key issues in identifying model sensitivity to individual observations, which go by the names residuals, leverage, and influence. In our $y_{i}=a+b x_{i}+e_{i}$ regression, the residuals are, of course, $e_{i}$-they reveal how much our fitted value $\widehat{y}_{i}=a+b x_{i}$ differs from the observed $y_{i}$. A point $\left(x_{i}, y_{i}\right)$ with a corresponding large residual is called an outlier. Say that you are interested in outliers because you somehow think that such points will exert undue influence on your estimates. Your feelings are generally right, but there are exceptions. A point might have a huge residual and yet not affect the estimated $b$ at all. Nevertheless, studying observations with large residuals almost always pays off.
$\left(x_{i}, y_{i}\right)$ can be an outlier in another way-just as $y_{i}$ can be far from $\widehat{y}_{i}, x_{i}$ can be far from the center of mass of the other $x$ 's. Such an "outlier" should interest you just as much as the more traditional outliers. Picture a scatterplot of $y$ against $x$ with thousands of points in some sort of mass
at the lower left of the graph and one point at the upper right of the graph. Now, run a regression line through the points - the regression line will come close to the point at the upper right of the graph and may in fact, go through it. That is, this isolated point will not appear as an outlier as measured by residuals because its residual will be small. Yet this point might have a dramatic effect on our resulting estimates in the sense that, were you to delete the point, the estimates would change markedly. Such a point is said to have high leverage. Just as with traditional outliers, a high leverage point does not necessarily have an undue effect on regression estimates, but if it does not, it is more the exception than the rule.

Now, all of this is a most unsatisfactory state of affairs. Points with large residuals may, but need not, have a large effect on our results, and points with small residuals may still have a large effect. Points with high leverage may, but need not, have a large effect on our results, and points with low leverage may still have a large effect. Can you not identify the influential points and simply have the computer list them for you? You can, but you will have to define what you mean by "influential".
"Influential" is defined with respect to some statistic. For instance, you might ask which points in your data have a large effect on your estimated $a$, which points have a large effect on your estimated $b$, which points have a large effect on your estimated standard error of $b$, and so on, but do not be surprised when the answers to these questions are different. In any case, obtaining such measures is not difficult - all you have to do is fit the regression excluding each observation one at a time and record the statistic of interest which, in the day of the modern computer, is not too onerous. Moreover, you can save considerable computer time by doing algebra ahead of time and working out formulas that will calculate the same answers as if you ran each of the regressions. (Ignore the question of pairs of observations that, together, exert undue influence, and triples, and so on, which remains largely unsolved and for which the brute force fit-every-possible-regression procedure is not a viable alternative.)

## Fitted values and residuals

Typing predict newvar with no options creates newvar containing the fitted values. Typing predict newvar, resid creates newvar containing the residuals.

## > Example 1

Continuing with example 1 from $[\mathrm{R}]$ regress, we wish to fit the following model:

$$
\mathrm{mpg}=\beta_{0}+\beta_{1} \text { weight }+\beta_{2} \text { foreign }+\epsilon
$$



That done, we can now obtain the predicted values from the regression. We will store them in a new variable called pmpg by typing predict pmpg. Because predict produces no output, we will follow that by summarizing our predicted and observed values.

```
. predict pmpg
(option xb assumed; fitted values)
```

. summarize pmpg mpg

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| pmpg | 74 | 21.2973 | 4.709779 | 9.794333 | 29.82151 |
| $m p g$ | 74 | 21.2973 | 5.785503 | 12 | 41 |

## > Example 2: Out-of-sample predictions

We can just as easily obtain predicted values from the model by using a wholly different dataset from the one on which the model was fit. The only requirement is that the data have the necessary variables, which here are weight and foreign.

Using the data on two new cars (the Pontiac Sunbird and the Volvo 260) from newautos.dta, we can obtain out-of-sample predictions (or forecasts) by typing

```
. use https://www.stata-press.com/data/r18/newautos, clear
(New automobile models)
. predict pmpg
(option xb assumed; fitted values)
. list, divider
```

| make | weight | foreign | pmpg |
| ---: | ---: | ---: | ---: |
| Pont. Sunbird | 2690 | Domestic | 23.95829 |
| Volvo 260 | 3170 | Foreign | 19.14607 |

The Pontiac Sunbird has a predicted mileage rating of 23.96 mpg , whereas the Volvo 260 has a predicted rating of 19.15 mpg . In comparison, the actual mileage ratings are 24 for the Pontiac and 17 for the Volvo.

## Prediction standard errors

predict can calculate the standard error of the forecast (stdf option), the standard error of the prediction (stdp option), and the standard error of the residual (stdr option). It is easy to confuse stdf and stdp because both are often called the prediction error. Consider the prediction $\widehat{y}_{j}=\mathbf{x}_{j} \mathbf{b}$, where $\mathbf{b}$ is the estimated coefficient (column) vector and $\mathbf{x}_{j}$ is a (row) vector of independent variables for which you want the prediction. First, $\widehat{y}_{j}$ has a variance due to the variance of the estimated coefficient vector $\mathbf{b}$,

$$
\operatorname{Var}\left(\widehat{y}_{j}\right)=\operatorname{Var}\left(\mathbf{x}_{j} \mathbf{b}\right)=s^{2} h_{j}
$$

where $h_{j}=\mathbf{x}_{j}\left(\mathbf{X}^{\prime} \mathbf{X}\right)^{-1} \mathbf{x}_{j}^{\prime}$ and $s^{2}$ is the mean squared error of the regression. Do not panic over the algebra-just remember that $\operatorname{Var}\left(\widehat{y}_{j}\right)=s^{2} h_{j}$, whatever $s^{2}$ and $h_{j}$ are. stdp calculates this quantity. This is the error in the prediction due to the uncertainty about $\mathbf{b}$.

If you are about to hand this number out as your forecast, however, there is another error. According to your model, the true value of $y_{j}$ is given by

$$
y_{j}=\mathbf{x}_{j} \mathbf{b}+\epsilon_{j}=\widehat{y}_{j}+\epsilon_{j}
$$

and thus the $\operatorname{Var}\left(y_{j}\right)=\operatorname{Var}\left(\widehat{y}_{j}\right)+\operatorname{Var}\left(\epsilon_{j}\right)=s^{2} h_{j}+s^{2}$, which is the square of stdf. stdf, then, is the sum of the error in the prediction plus the residual error.
stdr has to do with an analysis-of-variance decomposition of $s^{2}$, the estimated variance of $y$. The standard error of the prediction is $s^{2} h_{j}$, and therefore $s^{2} h_{j}+s^{2}\left(1-h_{j}\right)=s^{2}$ decomposes $s^{2}$ into the prediction and residual variances.

## > Example 3: Standard error of the forecast

Returning to our model of mpg on weight and foreign, we previously predicted the mileage rating for the Pontiac Sunbird and Volvo 260 as 23.96 and 19.15 mpg , respectively. We now want to put a standard error around our forecast. Remember, the data for these two cars were in newautos.dta:

```
. use https://www.stata-press.com/data/r18/newautos, clear
(New automobile models)
. predict pmpg
(option xb assumed; fitted values)
. predict se_pmpg, stdf
. list, divider
```

| make | weight | foreign | pmpg | se_pmpg |
| ---: | ---: | ---: | ---: | ---: |
| Pont. Sunbird <br> Volvo 260 | 2690 | Domestic | 23.95829 | 3.462791 |

Thus, an approximate $95 \%$ confidence interval for the mileage rating of the Volvo 260 is $19.15 \pm 2 \cdot 3.53=$ [12.09, 26.21].

## Prediction with weighted data

predict can be used after frequency-weighted (fweight) estimation, just as it is used after unweighted estimation. The technical note below concerns the use of predict after analytically weighted (aweight) estimation.

- Technical note

After analytically weighted estimation, predict is willing to calculate only the prediction (no options), residual (residual option), standard error of the prediction (stdp option), and diagonal elements of the projection matrix (hat option). For analytically weighted estimation, the standard error of the forecast and residuals, the standardized and Studentized residuals, and Cook's $D$ are not statistically well-defined concepts.

## Leverage statistics

In addition to providing fitted values and the associated standard errors, the predict command can also be used to generate various statistics used to detect the influence of individual observations. This section provides a brief introduction to leverage (hat) statistics, and some of the following subsections discuss other influence statistics produced by predict.

## > Example 4: Diagonal elements of projection matrix

The diagonal elements of the projection matrix, obtained by the hat option, are a measure of distance in explanatory variable space. leverage is a synonym for hat.

```
. use https://www.stata-press.com/data/r18/auto, clear
(1978 automobile data)
. regress mpg weight foreign
(output omitted)
. predict xdist, hat
. summarize xdist, detail
```

                                    Leverage
    |  | Percentiles | Smallest |  |  |
| ---: | :---: | :---: | :--- | ---: |
| $1 \%$ | .0192325 | .0192325 |  | 74 |
| $5 \%$ | .0192686 | .0192366 |  | 74 |
| $10 \%$ | .0193448 | .019241 | Obs |  |
| $25 \%$ | .0220291 | .0192686 | Sum of wgt. | .0405405 |
| $50 \%$ | .0383797 |  | Mean | .0207624 |
|  |  | Largest | Std. dev. |  |
| $75 \%$ | .0494002 | .0880814 |  | .0004311 |
| $90 \%$ | .0693432 | .099715 | Variance | 1.159745 |
| $95 \%$ | .0880814 | .099715 | Skewness | 4.083313 |

Some $5 \%$ of our sample has an xdist measure in excess of 0.08 . Let's force them to reveal their identities:
. list foreign make mpg if xdist>.08, divider

|  | foreign | make | mpg |
| :--- | :---: | :--- | :---: |
| 24. | Domestic | Ford Fiesta | 28 |
| 26. | Domestic | Linc. Continental | 12 |
| 27. | Domestic | Linc. Mark V | 12 |
| 43. | Domestic | Plym. Champ | 34 |
| 64. | Foreign | Peugeot 604 | 14 |

To understand why these cars are on this list, we must remember that the explanatory variables in our model are weight and foreign and that xdist measures distance in this metric. The Ford Fiesta and the Plymouth Champ are the two lightest domestic cars in our data. The Lincolns are the two heaviest domestic cars, and the Peugeot is the heaviest foreign car.

See lvr2plot in [R] regress postestimation diagnostic plots for information on a leverage-versus-squared-residual plot.

## Standardized and Studentized residuals

The terms standardized and Studentized residuals have meant different things to different authors. In Stata, predict defines the standardized residual as $\widehat{e}_{s_{i}}=e_{i} /\left(s \sqrt{1-h_{i}}\right)$ and the Studentized residual as $r_{i}=e_{i} /\left(s_{(i)} \sqrt{1-h_{i}}\right)$, where $s_{(i)}$ is the root mean squared error of a regression with the $i$ th observation removed. Stata's definition of the Studentized residual is the same as the one given in Bollen and Jackman $(1990,264)$ and is what Chatterjee and Hadi $(1988,74)$ call the "externally Studentized" residual. Stata's "standardized" residual is the same as what Chatterjee and Hadi (1988, 74) call the "internally Studentized" residual.

Standardized and Studentized residuals are attempts to adjust residuals for their standard errors. Although the $\epsilon_{i}$ theoretical residuals are homoskedastic by assumption (that is, they all have the same variance), the calculated $e_{i}$ are not. In fact,

$$
\operatorname{Var}\left(e_{i}\right)=\sigma^{2}\left(1-h_{i}\right)
$$

where $h_{i}$ are the leverage measures obtained from the diagonal elements of hat matrix. Thus, observations with the greatest leverage have corresponding residuals with the smallest variance.

Standardized residuals use the root mean squared error of the regression for $\sigma$. Studentized residuals use the root mean squared error of a regression omitting the observation in question for $\sigma$. In general, Studentized residuals are preferable to standardized residuals for purposes of outlier identification. Studentized residuals can be interpreted as the $t$ statistic for testing the significance of a dummy variable equal to 1 in the observation in question and 0 elsewhere (Belsley, Kuh, and Welsch 1980). Such a dummy variable would effectively absorb the observation and so remove its influence in determining the other coefficients in the model. Caution must be exercised here, however, because of the simultaneous testing problem. You cannot simply list the residuals that would be individually significant at the $5 \%$ level-their joint significance would be far less (their joint significance level would be far greater).

## > Example 5: Standardized and Studentized residuals

In the Terminology section of Remarks and examples for predict, we distinguished residuals from leverage and speculated on the impact of an observation with a small residual but large leverage. If we adjust the residuals for their standard errors, however, the adjusted residual would be (relatively) larger and perhaps large enough so that we could simply examine the adjusted residuals. Taking our price on weight and foreign\#\#c.mpg model from example 1 of [R] regress postestimation diagnostic plots, we can obtain the in-sample standardized and Studentized residuals by typing

```
. use https://www.stata-press.com/data/r18/auto, clear
(1978 automobile data)
. regress price weight foreign##c.mpg
(output omitted)
. predict esta if e(sample), rstandard
. predict estu if e(sample), rstudent
```

In the lvr2plot section of [R] regress postestimation diagnostic plots, we discovered that the VW Diesel has the highest leverage in our data, but a corresponding small residual. The standardized and Studentized residuals for the VW Diesel are
. list make price esta estu if make=="VW Diesel"
71.

| make | price | esta | estu |
| :--- | ---: | ---: | ---: |
| VW Diesel | 5,397 | .6142691 | .6114758 |

The Studentized residual of 0.611 can be interpreted as the $t$ statistic for including a dummy variable for VW Diesel in our regression. Such a variable would not be significant.

## DFITS, Cook's Distance, and Welsch Distance

DFITS (Welsch and Kuh 1977), Cook's Distance (Cook 1977), and Welsch Distance (Welsch 1982) are three attempts to summarize the information in the leverage versus residual-squared plot into one statistic. That is, the goal is to create an index that is affected by the size of the residuals - outliers - and the size of $h_{i}$-leverage. Viewed mechanically, one way to write DFITS (Bollen and Jackman 1990, 265) is

$$
\operatorname{DFITS}_{i}=r_{i} \sqrt{\frac{h_{i}}{1-h_{i}}}
$$

where $r_{i}$ are the Studentized residuals. Thus, large residuals increase the value of DFITS, as do large values of $h_{i}$. Viewed more traditionally, DFITS is a scaled difference between predicted values for the $i$ th case when the regression is fit with and without the $i$ th observation, hence the name.

The mechanical relationship between DFITS and Cook's Distance, $D_{i}$ (Bollen and Jackman 1990, 266), is

$$
D_{i}=\frac{1}{k} \frac{s_{(i)}^{2}}{s^{2}} \text { DFITS }_{i}^{2}
$$

where $k$ is the number of variables (including the constant) in the regression, $s$ is the root mean squared error of the regression, and $s_{(i)}$ is the root mean squared error when the $i$ th observation is omitted. Viewed more traditionally, $D_{i}$ is a scaled measure of the distance between the coefficient vectors when the $i$ th observation is omitted.

The mechanical relationship between DFITS and Welsch's Distance, $W_{i}$ (Chatterjee and Hadi 1988, 123), is

$$
W_{i}=\operatorname{DFITs}_{i} \sqrt{\frac{n-1}{1-h_{i}}}
$$

The interpretation of $W_{i}$ is more difficult because it is based on the empirical influence curve. Although DFITS and Cook's distance are similar, the Welsch distance measure includes another normalization by leverage.

Belsley, Kuh, and Welsch $(1980,28)$ suggest that DFITS values greater than $2 \sqrt{k / n}$ deserve more investigation, and so values of Cook's distance greater than $4 / n$ should also be examined (Bollen and Jackman 1990, 265-266). Through similar logic, the cutoff for Welsch distance is approximately $3 \sqrt{k}$ (Chatterjee and Hadi 1988, 124).

## > Example 6: DFITS influence measure

Continuing with our model of price on weight and foreign\#\#c.mpg, we can obtain the DFITS influence measure:

```
. predict e if e(sample), resid
. predict dfits, dfits
```

We did not specify if e(sample) in computing the DFITS statistic. DFITS is available only over the estimation sample, so specifying if e (sample) would have been redundant. It would have done no harm, but it would not have changed the results.

Our model has $k=5$ independent variables ( $k$ includes the constant) and $n=74$ observations; following the $2 \sqrt{k / n}$ cutoff advice, we type
. list make price e dfits if abs(dfits) > $2 *$ sqrt(5/74), divider

|  | make | price | e | dfits |
| :--- | :--- | ---: | ---: | ---: |
| 12. | Cad. Eldorado | 14,500 | 7271.96 | .9564455 |
| 13. | Cad. Seville | 15,906 | 5036.348 | 1.356619 |
| 24. | Ford Fiesta | 4,389 | 3164.872 | .5724172 |
| 27. | Linc. Mark V | 13,594 | 3109.193 | .5200413 |
| 28. | Linc. Versailles | 13,466 | 6560.912 | .8760136 |
| 42. | Plym. Arrow | 4,647 | -3312.968 | -.9384231 |

We calculate Cook's distance and list the observations greater than the suggested $4 / n$ cutoff:
. predict cooksd if e(sample), cooksd
. list make price e cooksd if cooksd > 4/74, divider

|  | make | price | e | cooksd |
| :--- | :--- | ---: | ---: | ---: |
| 12. | Cad. Eldorado | 14,500 | 7271.96 | .1492676 |
| 13. | Cad. Seville | 15,906 | 5036.348 | .3328515 |
| 24. | Ford Fiesta | 4,389 | 3164.872 | .0638815 |
| 28. | Linc. Versailles | 13,466 | 6560.912 | .1308004 |
| 42. | Plym. Arrow | 4,647 | -3312.968 | .1700736 |

Here we used if e(sample) because Cook's distance is not restricted to the estimation sample by default. It is worth comparing this list with the preceding one.

Finally, we use Welsch distance and the suggested $3 \sqrt{k}$ cutoff:

```
. predict wd, welsch
. list make price e wd if abs(wd) > 3*sqrt(5), divider
```

|  | make | price | e | wd |
| :--- | :--- | ---: | ---: | ---: |
|  | 12. | Cad. Eldorado | 14,500 | 7271.96 |
| 13. | Cad. Seville | 15,906 | 5036.348 | 12.894372 |
| 28. | Linc. Versailles | 13,466 | 6560.912 | 7.703005 |
| 42. | Plym. Arrow | 4,647 | -3312.968 | -8.981481 |

Here we did not need to specify if e(sample) because welsch automatically restricts the prediction to the estimation sample.

## COVRATIO

COVRATIO (Belsley, Kuh, and Welsch 1980) measures the influence of the $i$ th observation by considering the effect on the variance-covariance matrix of the estimates. The measure is the ratio of the determinants of the covariances matrix, with and without the $i$ th observation. The resulting formula is

$$
\operatorname{COVRATIO}_{i}=\frac{1}{1-h_{i}}\left(\frac{n-k-\widehat{e}_{s_{i}}^{2}}{n-k-1}\right)^{k}
$$

where $\widehat{e}_{s_{i}}$ is the standardized residual.

For noninfluential observations, the value of COVRATIO is approximately 1. Large values of the residuals or large values of leverage will cause deviations from 1, although if both are large, COVRATIO may tend back toward 1 and therefore not identify such observations (Chatterjee and Hadi 1988, 139).

Belsley, Kuh, and Welsch (1980) suggest that observations for which

$$
\mid \text { COVRATIO }_{i}-1 \left\lvert\, \geq \frac{3 k}{n}\right.
$$

are worthy of further examination.
$>$ Example 7: COVRATIO influence measure
Using our model of price on weight and foreign\#\#c.mpg, we can obtain the COVRATIO measure and list the observations outside the suggested cutoff by typing

```
. predict covr, covratio
. list make price e covr if abs(covr-1) >= 3*5/74, divider
```

12. 
13. 
14. 
15. 
16. 
17. 
18. 
19. 
20. 
21. 

| make | price | e | covr |
| :--- | ---: | ---: | ---: |
| Cad. Eldorado | 14,500 | 7271.96 | .3814242 |
| Cad. Seville | 15,906 | 5036.348 | .7386969 |
| Linc. Versailles | 13,466 | 6560.912 | .4761695 |
| Plym. Champ | 4,425 | 1621.747 | 1.27782 |
| Audi 5000 | 9,690 | 591.2883 | 1.206842 |
| Datsun 210 | 4,589 | 19.81829 | 1.284801 |
| Peugeot 604 | 12,990 | 1037.184 | 1.348219 |
| Subaru | 3,798 | -909.5894 | 1.264677 |
| VW Diesel | 5,397 | 999.7209 | 1.630653 |
| Volvo 260 | 11,995 | 1327.668 | 1.211888 |

The covratio option automatically restricts the prediction to the estimation sample.

## margins

## Description for margins

margins estimates margins of response for linear predictions.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [options]
```

statistic
xb
$\operatorname{pr}(a, b)$
e $(a, b)$
ystar ( $a, b$ )
residuals
score
rstandard
rstudent
cooksd
leverage hat
dfbeta(varname)
stdp
stdf
stdr
covratio
dfits
welsch

Description
linear prediction; the default
not allowed with margins
not allowed with margins
not allowed with margins
not allowed with margins
not allowed with margins
not allowed with margins not allowed with margins not allowed with margins not allowed with margins not allowed with margins not allowed with margins not allowed with margins not allowed with margins not allowed with margins not allowed with margins not allowed with margins not allowed with margins

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

## DFBETA influence statistics

## Description for dfbeta

dfbeta will calculate one, more than one, or all the DFBETAs after regress. Although predict will also calculate DFBETAs, predict can do this for only one variable at a time. dfbeta is a convenience tool for those who want to calculate DFBETAs for multiple variables. The names for the new variables created are chosen automatically and begin with the letters _dfbeta_.

## Menu for dfbeta

Statistics $>$ Linear models and related $>$ Regression diagnostics $>$ DFBETAs

## Syntax for dfbeta

dfbeta [indepvar $[$ indepvar $[\ldots]]][$, stub (name) $]$

## Option for dfbeta

stub (name) specifies the leading characters dfbeta uses to name the new variables to be generated.
The default is stub(_dfbeta_).

## Remarks and examples for dfbeta

DFBETAs are perhaps the most direct influence measure of interest to model builders. DFBETAS focus on one coefficient and measure the difference between the regression coefficient when the $i$ th observation is included and excluded, the difference being scaled by the estimated standard error of the coefficient. Belsley, Kuh, and Welsch $(1980,28)$ suggest observations with $\mid$ DFBETA $_{i} \mid>2 / \sqrt{n}$ as deserving special attention, but it is also common practice to use 1 (Bollen and Jackman 1990, 267), meaning that the observation shifted the estimate at least one standard error.
> Example 8: DFBETAs influence measure; the dfbeta() option
Using our model of price on weight and foreign\#\#c.mpg, let's first ask which observations have the greatest impact on the determination of the coefficient on $1 . f o r e i g n$. We will use the suggested $2 / \sqrt{n}$ cutoff:

```
. use https://www.stata-press.com/data/r18/auto, clear
(1978 automobile data)
. regress price weight foreign##c.mpg
    (output omitted)
```

```
. sort foreign make
. predict dfor, dfbeta(1.foreign)
. list make price foreign dfor if abs(dfor) > 2/sqrt(74), divider
```

12. 
13. 
14. 
15. 
16. 
17. 
18. 

| make | price | foreign | dfor |
| :--- | ---: | ---: | ---: |
| Cad. Eldorado | 14,500 | Domestic | -.5290519 |
| Cad. Seville | 15,906 | Domestic | .8243419 |
| Linc. Versailles | 13,466 | Domestic | -.5283729 |
| Plym. Arrow | 4,647 | Domestic | -.6622424 |
| Plym. Champ | 4,425 | Domestic | .2371104 |
| Peugeot 604 | 12,990 | Foreign | .2552032 |
| Toyota Corona | 5,719 | Foreign | -.256431 |

The Cadillac Seville shifted the coefficient on 1.foreign 0.82 standard deviations!
Now let us ask which observations have the greatest effect on the mpg coefficient:

```
. predict dmpg, dfbeta(mpg)
. list make price mpg dmpg if abs(dmpg) > 2/sqrt(74), divider
```

12. 
13. 
14. 
15. 
16. 

| make | price | mpg | dmpg |
| :--- | ---: | ---: | ---: |
| Cad. Eldorado | 14,500 | 14 | -.5970351 |
| Cad. Seville | 15,906 | 21 | 1.134269 |
| Linc. Versailles | 13,466 | 14 | -.6069287 |
| Plym. Arrow | 4,647 | 28 | -.8925859 |
| Plym. Champ | 4,425 | 34 | .3186909 |

Once again, we see the Cadillac Seville heading the list, indicating that our regression results may be dominated by this one car.

## > Example 9: DFBETAs influence measure; the dfbeta command

We can use predict, dfbeta() or the dfbeta command to generate the DFBETAs. dfbeta makes up names for the new variables automatically and, without arguments, generates the DFBETAS for all the variables in the regression:

```
. dfbeta
Generating DFBETA variables ...
    _dfbeta_1: DFBETA weight
    _dfbeta_2: DFBETA 1.foreign
    _dfbeta_3: DFBETA mpg
    _dfbeta_4: DFBETA 1.foreign#c.mpg
```

dfbeta created four new variables in our dataset: _dfbeta_1, containing the DFBETAs for weight; _dfbeta_2, containing the DFBETAs for mpg; and so on. Had we wanted only the DFBETAs for mpg and weight, we might have typed
. dfbeta mpg weight
Generating DFBETA variables ...

```
_dfbeta_5: DFBETA weight
_dfbeta_6: DFBETA mpg
```

In the example above, we typed dfbeta mpg weight instead of dfbeta; if we had typed dfbeta followed by dfbeta mpg weight, here is what would have happened:

```
. dfbeta
Generating DFBETA variables ...
    _dfbeta_7: DFBETA weight
    _dfbeta_8: DFBETA 1.foreign
    _dfbeta_9: DFBETA mpg
    _dfbeta_10: DFBETA 1.foreign#c.mpg
. dfbeta mpg weight
Generating DFBETA variables ...
    _dfbeta_11: DFBETA weight
    _dfbeta_12: DFBETA mpg
```

dfbeta would have made up different names for the new variables. dfbeta never replaces existing variables-it instead makes up a different name, so we need to pay attention to dfbeta's output.

## Tests for violation of assumptions

## Description for estat hettest

estat hettest performs three versions of the Breusch-Pagan (1979) and Cook-Weisberg (1983) test for heteroskedasticity. All three versions of this test present evidence against the null hypothesis that $\mathbf{t}=\mathbf{0}$ in $\operatorname{Var}(e)=\sigma^{2} \exp (\mathbf{z t})$. In the normal version, performed by default, the null hypothesis also includes the assumption that the regression disturbances are independent-normal draws with variance $\sigma^{2}$. The normality assumption is dropped from the null hypothesis in the iid and fstat versions, which respectively produce the score and $F$ tests discussed in Methods and formulas. If varlist is not specified, the fitted values are used for $\mathbf{z}$. If varlist or the rhs option is specified, the variables specified are used for $\mathbf{z}$.

## Menu for estat

Statistics > Postestimation

## Syntax for estat hettest

```
estat hettest [varlist] [, rhs [\underline{normal |iid |fstat] mtest[(spec)]]}
```

collect is allowed with estat hettest; see [U] 11.1.10 Prefix commands.

## Options for estat hettest

rhs specifies that tests for heteroskedasticity be performed for the right-hand-side (explanatory) variables of the fitted regression model. The rhs option may be combined with a varlist.
normal, the default, causes estat hettest to compute the original Breusch-Pagan/Cook-Weisberg test, which assumes that the regression disturbances are normally distributed.
iid causes estat hettest to compute the $N * R^{2}$ version of the score test that drops the normality assumption.
fstat causes estat hettest to compute the $F$-statistic version that drops the normality assumption.
mtest $[$ (spec) $]$ specifies that multiple testing be performed. The argument specifies how $p$-values are adjusted. The following specifications, spec, are supported:

| bonferroni | Bonferroni's multiple testing adjustment |
| :---: | :--- |
| holm | Holm's multiple testing adjustment |
| sidak | Šidák's multiple testing adjustment |
| noadjust | no adjustment is made for multiple testing |

mtest may be specified without an argument. This is equivalent to specifying mtest (noadjust); that is, tests for the individual variables should be performed with unadjusted $p$-values. By default, estat hettest does not perform multiple testing. mtest may not be specified with iid or fstat.

## Description for estat imtest

estat imtest performs an information matrix test for the regression model and an orthogonal decomposition into tests for heteroskedasticity, skewness, and kurtosis due to Cameron and Trivedi (1990); White's test for homoskedasticity against unrestricted forms of heteroskedasticity (1980) is available as an option. White's test is usually similar to the first term of the Cameron-Trivedi decomposition.

## Menu for estat

Statistics $>$ Postestimation

## Syntax for estat imtest

estat imtest [, preserve white]
collect is allowed with estat imtest; see [U] 11.1.10 Prefix commands.

## Options for estat imtest

preserve specifies that the data in memory be preserved, all variables and cases that are not needed in the calculations be dropped, and at the conclusion the original data be restored. This option is costly for large datasets. However, because estat imtest has to perform an auxiliary regression on $k(k+1) / 2$ temporary variables, where $k$ is the number of regressors, it may not be able to perform the test otherwise. white specifies that White's original heteroskedasticity test also be performed.

## Description for estat ovtest

estat ovtest performs two versions of the Ramsey (1969) regression specification-error test (RESET) for omitted variables. This test amounts to fitting $y=\mathbf{x b}+\mathbf{z t}+u$ and then testing $\mathbf{t}=\mathbf{0}$. If the rhs option is not specified, powers of the fitted values are used for $\mathbf{z}$. If $\mathbf{r h s}$ is specified, powers of the individual elements of $\mathbf{x}$ are used.

## Menu for estat

Statistics $>$ Postestimation

## Syntax for estat ovtest

estat ovtest [, rhs]
collect is allowed with estat ovtest; see [U] 11.1.10 Prefix commands.

## Option for estat ovtest

rhs specifies that powers of the right-hand-side (explanatory) variables be used in the test rather than powers of the fitted values.

## Description for estat szroeter

estat szroeter performs Szroeter's rank test for heteroskedasticity for each of the variables in varlist or for the explanatory variables of the regression if rhs is specified.

## Menu for estat

Statistics > Postestimation

## Syntax for estat szroeter

estat szroeter [varlist] [, ́ㅡㄴ mtest (spec)]
Either varlist or rhs must be specified.

## Options for estat szroeter

rhs specifies that tests for heteroskedasticity be performed for the right-hand-side (explanatory) variables of the fitted regression model. The rhs option may be combined with a varlist.
mtest (spec) specifies that multiple testing be performed. The argument specifies how $p$-values are adjusted. The following specifications, spec, are supported:

| bonferroni | Bonferroni's multiple testing adjustment |
| :---: | :--- |
| holm | Holm's multiple testing adjustment |
| $\underline{\text { sidak }}$ | Šidák's multiple testing adjustment |
| noadjust | no adjustment is made for multiple testing |

estat szroeter always performs multiple testing. By default, it does not adjust the $p$-values.

## Remarks and examples for estat hettest, estat imtest, estat ovtest, and estat szroeter

We introduce some regression diagnostic commands that are designed to test for certain violations that rvfplot (see $[R]$ regress postestimation diagnostic plots) less formally attempts to detect. estat ovtest provides Ramsey's test for omitted variables-a pattern in the residuals. estat hettest provides a test for heteroskedasticity - the increasing or decreasing variation in the residuals with fitted values, with respect to the explanatory variables, or with respect to yet other variables. The score test implemented in estat hettest (Breusch and Pagan 1979; Cook and Weisberg 1983) performs a score test of the null hypothesis that $b=0$ against the alternative hypothesis of multiplicative heteroskedasticity. estat szroeter provides a rank test for heteroskedasticity, which is an alternative to the score test computed by estat hettest. Finally, estat imtest computes an information matrix test, including an orthogonal decomposition into tests for heteroskedasticity, skewness, and kurtosis (Cameron and Trivedi 1990). The heteroskedasticity test computed by estat imtest is similar to the general test for heteroskedasticity that was proposed by White (1980). Cameron and Trivedi (2022, chap. 3) discuss most of these tests and provides more examples.
> Example 10: estat ovtest, estat hettest, estat szroeter, and estat imtest
We use our model of price on weight and foreign\#\#c.mpg.

```
. use https://www.stata-press.com/data/r18/auto, clear
(1978 automobile data)
. regress price weight foreign##c.mpg
    (output omitted)
. estat ovtest
Ramsey RESET test for omitted variables
Omitted: Powers of fitted values of price
HO: Model has no omitted variables
F(3, 66) = 7.77
Prob > F = 0.0002
. estat hettest
Breusch-Pagan/Cook-Weisberg test for heteroskedasticity
Assumption: Normal error terms
Variable: Fitted values of price
HO: Constant variance
    chi2(1) = 6.50
Prob > chi2 = 0.0108
```

Testing for heteroskedasticity in the right-hand-side variables is requested by specifying the rhs option. By specifying the mtest (bonferroni) option, we request that tests be conducted for each of the variables, with a Bonferroni adjustment for the $p$-values to accommodate our testing multiple hypotheses.

```
. estat hettest, rhs mtest(bonf)
Breusch-Pagan/Cook-Weisberg test for heteroskedasticity
Assumption: Normal error terms
HO: Constant variance
```

| Variable | chi2 | df | p |
| :---: | :---: | :---: | ---: |
| weight <br> foreign <br> Foreign | 15.24 | 1 | $0.0004 *$ |
| mpg <br> foreign\# <br> c.mpg <br> Foreign | 6.15 | 1 | $0.0525 *$ |
| Simultaneous | 15.04 | 1 | $0.0106 *$ |

* Bonferroni-adjusted p-values
. estat szroeter, rhs mtest(holm)
Szroeter's test for homoskedasticity
HO: Variance constant
Ha: Variance monotonic in variables

| Variable | chi2 | df | p |
| :---: | ---: | ---: | ---: |
| weight <br> foreign <br> Foreign | 17.07 | 1 | $0.0001 *$ |
| mpg <br> foreign\# <br> c.mpg <br> Foreign | 6.15 | 1 | $0.0131 *$ |

* Holm-adjusted $p$-values

Finally, we request the information matrix test, which is a conditional moments test with second-, third-, and fourth-order moment conditions.
. estat imtest
Cameron \& Trivedi's decomposition of IM-test

| Source | chi2 | df | p |
| ---: | ---: | ---: | ---: |
| Heteroskedasticity | 18.86 | 10 | 0.0420 |
| Skewness | 11.69 | 4 | 0.0198 |
| Kurtosis | 2.33 | 1 | 0.1273 |
| Total | 32.87 | 15 | 0.0049 |

We find evidence for omitted variables, heteroskedasticity, and nonnormal skewness.
So, why bother with the various graphical commands when the tests seem so much easier to interpret? In part, it is a matter of taste: both are designed to uncover the same problem, and both are, in fact, going about it in similar ways. One is based on a formal calculation, whereas the other is based on personal judgment in evaluating a graph. On the other hand, the tests are seeking evidence of specific problems, whereas judgment is more general. The careful analyst will use both.

We performed the omitted-variable test first. Omitted variables are a more serious problem than heteroskedasticity or the violations of higher moment conditions tested by estat imtest. If this
were not a manual, having found evidence of omitted variables, we would never have run the estat hettest, estat szroeter, and estat imtest commands, at least not until we solved the omitted-variable problem.

## - Technical note

estat ovtest and estat hettest both perform two flavors of their respective tests. By default, estat ovtest looks for evidence of omitted variables by fitting the original model augmented by $\widehat{y}^{2}, \widehat{y}^{3}$, and $\widehat{y}^{4}$, which are the fitted values from the original model. Under the assumption of no misspecification, the coefficients on the powers of the fitted values will be zero. With the rhs option, estat ovtest instead augments the original model with powers (second through fourth) of the explanatory variables (except for dummy variables).
estat hettest, by default, looks for heteroskedasticity by modeling the variance as a function of the fitted values. If, however, we specify a variable or variables, the variance will be modeled as a function of the specified variables. In our example, if we had, a priori, some reason to suspect heteroskedasticity and that the heteroskedasticity is a function of a car's weight, then using a test that focuses on weight would be more powerful than the more general tests such as White's test or the first term in the Cameron-Trivedi decomposition test.
estat hettest, by default, computes the original Breusch-Pagan/Cook-Weisberg test, which includes the assumption of normally distributed errors. Koenker (1981) derived an $N * R^{2}$ version of this test that drops the normality assumption. Wooldridge $(2020,270)$ gives an $F$-statistic version that does not require the normality assumption.

## Stored results for estat hettest, estat imtest, and estat ovtest

estat hettest stores the following results for the (multivariate) score test in r() :
Scalars
$r($ chi2 $) \quad \chi^{2}$ test statistic
$r(\mathrm{df}) \quad$ \#df for the asymptotic $\chi^{2}$ distribution under $H_{0}$
$r(p)$ $p$-value
estat hettest, fstat stores results for the (multivariate) score test in $r()$ :
Scalars
$r(F) \quad$ test statistic
$\mathrm{r}\left(\mathrm{df} \_\mathrm{m}\right) \quad$ \#df of the test for the $F$ distribution under $H_{0}$
$r\left(d f \_r\right) \quad$ \#df of the residuals for the $F$ distribution under $H_{0}$
$r(p) \quad p$-value
estat hettest (if mtest is specified) and estat szroeter store the following in $r()$ :
Matrices
$r$ (mtest) a matrix of test results, with rows corresponding to the univariate tests
mtest[.,1] $\chi^{2}$ test statistic
mtest[.,2] \#df
mtest [., 3] unadjusted $p$-value
mtest [.,4] adjusted $p$-value (if an mtest() adjustment method is specified)
Macros
r(mtmethod) adjustment method for $p$-value
estat imtest stores the following in r() :
Scalars

$$
\begin{aligned}
& r\left(\text { chi2_t) } \quad \text { IM-test statistic }\left(=r\left(c h i 2 \_h\right)+r\left(c h i 2 \_s\right)+r\left(c h i 2 \_k\right)\right)\right. \\
& r\left(d f \_t\right) \quad d f \text { for limiting } \chi^{2} \text { distribution under } H_{0}\left(=r\left(d f \_h\right)+r\left(d f \_s\right)+r\left(d f \_k\right)\right) \\
& r(\text { chi2 }-\mathrm{h}) \quad \text { heteroskedasticity test statistic } \\
& r\left(d f \_h\right) \quad d f \text { for limiting } \chi^{2} \text { distribution under } H_{0} \\
& r \text { (chi2_s) skewness test statistic } \\
& r\left(d f \_s\right) \quad d f \text { for limiting } \chi^{2} \text { distribution under } H_{0} \\
& r\left(\text { chi2 } \_k\right) \quad \text { kurtosis test statistic } \\
& r\left(\mathrm{df} \_\mathrm{k}\right) \quad \mathrm{df} \text { for limiting } \chi^{2} \text { distribution under } H_{0} \\
& r \text { (chi2_w) White's heteroskedasticity test (if white specified) } \\
& r\left(d f \_w\right) \quad d f \text { for limiting } \chi^{2} \text { distribution under } H_{0}
\end{aligned}
$$

estat ovtest stores the following in $r()$ :
Scalars

| $r(p)$ | two-sided $p$-value |
| :--- | :--- |
| $r(F)$ | $F$ statistic |
| $r(d f)$ | degrees of freedom |
| $r\left(d f \_r\right)$ | residual degrees of freedom |

## Variance inflation factors

## Description for estat vif

estat vif calculates the centered or uncentered variance inflation factors (VIFs) for the independent variables specified in a linear regression model.

## Menu for estat

Statistics > Postestimation

## Syntax for estat vif

```
estat vif [, uncentered]
```


## Option for estat vif

uncentered requests the computation of the uncentered variance inflation factors. The uncentered VIFs are often used to detect the collinearity of the regressors with the constant. uncentered must be specified if the regression model did not include a constant term because centered VIFs are not appropriate for these models.

## Remarks and examples for estat vif

Problems arise in regression when the predictors are highly correlated. In this situation, there may be a significant change in the regression coefficients if you add or delete an independent variable. The estimated standard errors of the fitted coefficients are inflated, or the estimated coefficients may not be statistically significant even though a statistical relation exists between the dependent and independent variables.

Data analysts rely on these facts to check informally for the presence of multicollinearity. estat vif, another command for use after regress, calculates the variance inflation factors and tolerances for each of the independent variables.

The output shows the variance inflation factors together with their reciprocals. Some analysts compare the reciprocals with a predetermined tolerance. In the comparison, if the reciprocal of the VIF is smaller than the tolerance, the associated predictor variable is removed from the regression model. However, most analysts rely on informal rules of thumb applied to the VIF; see Chatterjee and Hadi (2012). According to these rules, there is evidence of multicollinearity if

1. The largest VIF is greater than 10 (some choose a more conservative threshold value of 30 ).
2. The mean of all the VIFs is considerably larger than 1.

## > Example 11: estat vif

We examine a regression model fit using the ubiquitous automobile dataset:

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
```

. regress price mpg rep78 trunk headroom length turn displ gear_ratio

| Source | SS | df | MS | Number of obs | = | 69 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | $\mathrm{F}(8,60)$ | = | 6.33 |
| Model | 264102049 | 8 | 33012756.2 | Prob > F | = | 0.0000 |
| Residual | 312694909 | 60 | 5211581.82 | R -squared | = | 0.4579 |
|  |  |  |  | Adj R-squared | = | 0.3856 |
| Total | 576796959 | 68 | 8482308.22 | Root MSE | = | 2282.9 |


| price | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| mpg | -144.84 | 82.12751 | -1.76 | 0.083 | -309.1195 | 19.43948 |
| rep78 | 727.5783 | 337.6107 | 2.16 | 0.035 | 52.25638 | 1402.9 |
| trunk | 44.02061 | 108.141 | 0.41 | 0.685 | -172.2935 | 260.3347 |
| headroom | -807.0996 | 435.5802 | -1.85 | 0.069 | -1678.39 | 64.19062 |
| length | -8.688914 | 34.89848 | -0.25 | 0.804 | -78.49626 | 61.11843 |
| turn | -177.9064 | 137.3455 | -1.30 | 0.200 | -452.6383 | 96.82551 |
| displacement | 30.73146 | 7.576952 | 4.06 | 0.000 | 15.5753 | 45.88762 |
| gear_ratio | 1500.119 | 1110.959 | 1.35 | 0.182 | -722.1303 | 3722.368 |
| _cons | 6691.976 | 7457.906 | 0.90 | 0.373 | -8226.058 | 21610.01 |


| . estat vif |  |  |
| ---: | ---: | ---: |
| Variable | VIF | $1 / \mathrm{VIF}$ |
| length | 8.22 | 0.121614 |
| displacement | 6.50 | 0.153860 |
| turn | 4.85 | 0.205997 |
| gear_ratio | 3.45 | 0.290068 |
| mpg | 3.03 | 0.330171 |
| trunk | 2.88 | 0.347444 |
| headroom | 1.80 | 0.554917 |
| rep78 | 1.46 | 0.686147 |
| Mean VIF | 4.02 |  |

The results are mixed. Although we have no VIFs greater than 10, the mean VIF is greater than 1 , though not considerably so. We could continue the investigation of collinearity, but given that other authors advise that collinearity is a problem only when VIFs exist that are greater than 30 (contradicting our rule above), we will not do so here.

## Example 12: estat vif, with strong evidence of multicollinearity

This example comes from a dataset described in Kutner, Nachtsheim, and Neter $(2004,257)$ that examines body fat as modeled by caliper measurements on the triceps, midarm, and thigh.

| Source | SS | df | MS | Number of obs | = | 20 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | F (3, 16) | = | 21.52 |
| Model | 396.984607 | 3 | 132.328202 | Prob > F | = | 0.0000 |
| Residual | 98.4049068 | 16 | 6.15030667 | R -squared | = | 0.8014 |
|  |  |  |  | Adj R-squared | = | 0.7641 |
| Total | 495.389513 | 19 | 26.0731323 | Root MSE | $=$ | 2.48 |


| bodyfat | Coefficient | Std. err. | t | $\mathrm{P}>\mid \mathrm{tI}$ | [95\% conf. interval] |  |
| ---: | ---: | :--- | :---: | :---: | ---: | ---: |
| triceps | 4.334085 | 3.015511 | 1.44 | 0.170 | -2.058512 | 10.72668 |
| thigh | -2.856842 | 2.582015 | -1.11 | 0.285 | -8.330468 | 2.616785 |
| midarm | -2.186056 | 1.595499 | -1.37 | 0.190 | -5.568362 | 1.19625 |
| _cons | 117.0844 | 99.78238 | 1.17 | 0.258 | -94.44474 | 328.6136 |

. estat vif

| Variable | VIF | $1 /$ VIF |
| ---: | ---: | ---: |
| triceps | 708.84 | 0.001411 |
| thigh | 564.34 | 0.001772 |
| midarm | 104.61 | 0.009560 |
| Mean VIF | 459.26 |  |

Here we see strong evidence of multicollinearity in our model. More investigation reveals that the measurements on the thigh and the triceps are highly correlated:

```
. correlate triceps thigh midarm
(obs=20)
```

|  | triceps | thigh | midarm |
| ---: | ---: | ---: | ---: |
| triceps | 1.0000 |  |  |
| thigh | 0.9238 | 1.0000 |  |
| midarm | 0.4578 | 0.0847 | 1.0000 |

If we remove the predictor tricep from the model (because it had the highest VIF), we get . regress bodyfat thigh midarm

| Source | SS | df | MS |
| ---: | :---: | ---: | :---: |
| Model | 384.279748 | 2 | 192.139874 |
| Residual | 111.109765 | 17 | 6.53586854 |
| Total | 495.389513 | 19 | 26.0731323 |


| Number of obs | $=$ | 20 |
| :--- | :--- | ---: |
| $\mathrm{~F}(2,17)$ | $=$ | 29.40 |
| Prob > F | $=$ | 0.0000 |
| R-squared | $=$ | 0.7757 |
| Adj R-squared | $=$ | 0.7493 |
| Root MSE | $=$ | 2.5565 |


| bodyfat | Coefficient | Std. err. | t | $\mathrm{P}>\mid \mathrm{tl}$ | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | ---: | ---: |
| thigh | .8508818 | .1124482 | 7.57 | 0.000 | .6136367 | 1.088127 |
| midarm | .0960295 | .1613927 | 0.60 | 0.560 | -.2444792 | .4365383 |
| _cons | -25.99696 | 6.99732 | -3.72 | 0.002 | -40.76001 | -11.2339 |


| estat vif <br> Variable | VIF | $1 /$ VIF |
| ---: | ---: | ---: |
| midarm | 1.01 | 0.992831 |
| thigh | 1.01 | 0.992831 |
| Mean VIF | 1.01 |  |

Note how the coefficients change and how the estimated standard errors for each of the regression coefficients become much smaller. The calculated value of $R^{2}$ for the overall regression for the subset model does not appreciably decline when we remove the correlated predictor. Removing an independent variable from the model is one way to deal with multicollinearity. Other methods include ridge regression, weighted least squares, and restricting the use of the fitted model to data that follow the same pattern of multicollinearity. In economic studies, it is sometimes possible to estimate the regression coefficients from different subsets of the data by using cross-section and time series.

All examples above demonstrated the use of centered VIFs. As pointed out by Belsley (1991), the centered VIFs may fail to discover collinearity involving the constant term. One solution is to use the uncentered VIFs instead. According to the definition of the uncentered VIFs, the constant is viewed as a legitimate explanatory variable in a regression model, which allows one to obtain the VIF value for the constant term.

## $>$ Example 13: estat vif, with strong evidence of collinearity with the constant term

Consider the extreme example in which one of the regressors is highly correlated with the constant. We simulate the data and examine both centered and uncentered VIF diagnostics after fitted regression model as follows.
. use https://www.stata-press.com/data/r18/extreme_collin
. regress y one x z

| Source | SS | df | MS | Number of obs | = | 100 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | F (3, 96) | = | 2710.27 |
| Model | 223801.985 | 3 | 74600.6617 | Prob > F | = | 0.0000 |
| Residual | 2642.42124 | 96 | 27.5252213 | R -squared | = | 0.9883 |
|  |  |  |  | Adj R-squared | = | 0.9880 |
| Total | 226444.406 | 99 | 2287.31723 | Root MSE | = | 5.2464 |


| y | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| one | -3.278582 | 10.5621 | -0.31 | 0.757 | -24.24419 | 17.68702 |
| x | 2.038696 | .0242673 | 84.01 | 0.000 | 1.990526 | 2.086866 |
| z | 4.863137 | .2681036 | 18.14 | 0.000 | 4.330956 | 5.395319 |
| _cons | 9.760075 | 10.50935 | 0.93 | 0.355 | -11.10082 | 30.62097 |


| estat vif <br> Variable | VIF | $1 / \mathrm{VIF}$ |
| ---: | ---: | ---: |
| z | 1.03 | 0.968488 |
| x | 1.03 | 0.971307 |
| one | 1.00 | 0.995425 |
| Mean VIF | 1.02 |  |


| estat vif, uncentered |  |  |
| ---: | ---: | ---: |
| Variable | VIF | $1 /$ VIF |
| one | 402.94 | 0.002482 |
| _cons | 401.26 | 0.002492 |
| z | 2.93 | 0.341609 |
| x | 1.13 | 0.888705 |
| Mean VIF | 202.06 |  |

According to the values of the centered VIFs (1.03, 1.03, 1.00), no harmful collinearity is detected in the model. However, by the construction of these simulated data, we know that one is highly collinear with the constant term. As such, the large values of uncentered VIFs for one (402.94) and _cons (401.26) reveal high collinearity of the variable one with the constant term.

## Measures of effect size

## Description for estat esize

estat esize calculates effect sizes for linear models after regress or anova. By default, estat esize reports $\eta^{2}$ estimates (Kerlinger and Lee 2000), which are equivalent to $R^{2}$ estimates. If the option epsilon is specified, estat esize reports $\varepsilon^{2}$ estimates (Grissom and Kim 2012). If the option omega is specified, estat esize reports $\omega^{2}$ estimates (Grissom and Kim 2012). Both $\varepsilon^{2}$ and $\omega^{2}$ are adjusted $R^{2}$ estimates. Confidence intervals for $\eta^{2}$ estimates are estimated by using the noncentral $F$ distribution (Smithson 2001). See Kline (2013) or Thompson (2006) for further information.

## Menu for estat

Statistics $>$ Postestimation

## Syntax for estat esize

```
    estat esize [, epsilon omega level(#)]
collect is allowed with estat esize; see [U] 11.1.10 Prefix commands.
```


## Options for estat esize

epsilon specifies that the $\epsilon^{2}$ estimates of effect size be reported. The default is $\eta^{2}$ estimates. omega specifies that the $\omega^{2}$ estimates of effect size be reported. The default is $\eta^{2}$ estimates.
level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals.

## Remarks and examples for estat esize

Whereas $p$-values are used to assess the statistical significance of a result, measures of effect size are used to assess the practical significance of a result. Effect sizes can be broadly categorized as "measures of group differences" (the $d$ family) and "measures of association" (the $r$ family); see Ellis (2010, table 1.1). The $d$ family includes estimators such as Cohen's $D$, Hedges's $G$, and Glass's $\Delta$ (also see [R] esize). The $r$ family includes estimators such as the point-biserial correlation coefficient, $\eta^{2}, \epsilon^{2}$, and $\omega^{2}$. For an introduction to the concepts and calculation of effect sizes, see Kline (2013) or Thompson (2006). For a more detailed discussion, see Kirk (1996), Ellis (2010), Cumming (2012), Grissom and Kim (2012), and Kelley and Preacher (2012).
> Example 14: Calculating effect sizes for a linear regression model
Suppose we fit a linear regression model for low-birthweight infants.

```
. use https://www.stata-press.com/data/r18/lbw
(Hosmer & Lemeshow data)
```

| Source | SS | df | MS | Number of obs | = | 189 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | F (3, 185) | = | 8.69 |
| Model | 12346897.6 | 3 | 4115632.54 | Prob > F | = | 0.0000 |
| Residual | 87568400.9 | 185 | 473342.708 | R -squared | = | 0.1236 |
|  |  |  |  | Adj R-squared | = | 0.1094 |
| Total | 99915298.6 | 188 | 531464.354 | Root MSE | = | 688 |


| bwt | Coefficient | Std. err. | t | $\mathrm{P}>\mid \mathrm{tl}$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| smoke | -428.0254 | 109.0033 | -3.93 | 0.000 | -643.0746 | -212.9761 |
| race |  |  |  |  |  |  |
| Black <br> Other | -454.1813 | 116.436 | -3.90 | 0.000 | -683.8944 | -224.4683 |
| _cons | 3334.858 | 91.74301 | 36.35 | 0.000 | 3153.86 | 3515.855 |

We can use the estat esize command to calculate $\eta^{2}$ for the entire model and a partial $\eta^{2}$ for each term in the model.

```
. estat esize
Effect sizes for linear models
```

| Source | Eta-squared | df | [95\% conf. interval] |  |
| ---: | :---: | :---: | :---: | :---: |
| Model | .1235736 | 3 | .0399862 | .2041365 |
|  |  |  |  |  |
| smoke | .0769345 | 1 | .0193577 | .1579213 |
| race | .0908394 | 2 | .0233037 | .1700334 |

Note: Eta-squared values for individual model terms are partial.
The overall model effect size is 0.124 . This means that roughly $12.4 \%$ of the variation in bwt is explained by the model. The partial effect size for smoke is 0.077 . This means that roughly $7.7 \%$ of the variation in bwt is explained by smoke after you remove the variation explained by all other terms.

The omega option causes estat esize to report $\omega^{2}$ and partial $\omega^{2}$.

```
. estat esize, omega
Effect sizes for linear models
```

| Source | Omega-squared | df |
| ---: | :---: | ---: |
| Model | .1088457 | 3 |
|  |  |  |
| smoke | .0715877 | 1 |
| race | .0806144 | 2 |

Note: Omega-squared values for individual model terms are partial.

## > Example 15: Calculating effect size for an ANOVA model

We can use estat esize after ANOVA models as well.


| Source | Eta-squared | df | [95\% conf. interval] |  |
| ---: | :---: | :---: | :---: | :---: |
| Model | .1235736 | 3 | .0399862 | .2041365 |
|  |  |  |  |  |
| smoke | .0769345 | 1 | .0193577 | .1579213 |
| race | .0908394 | 2 | .0233037 | .1700334 |

Note: Eta-squared values for individual model terms are partial.

## Technical note

$\eta^{2}$ was developed in the context of analysis of variance. Thus, the published research on the calculation of confidence intervals focuses on cases where the numerator degrees of freedom are relatively small (for example, $\mathrm{df}<20$ ).

Some combinations of the $F$ statistic, numerator degrees of freedom, and denominator degrees of freedom yield confidence limits that do not contain the corresponding estimated value for an $\eta^{2}$. This problem is most commonly observed for larger numerator degrees of freedom.

Nothing in the literature suggests alternative methods for constructing confidence intervals in such cases; therefore, we recommend cautious interpretation of confidence intervals for $\eta^{2}$ when the numerator degrees of freedom are greater than 20.

## Stored results for estat esize

estat esize stores the following results in $r()$ :
Scalars
r(level) confidence level
Matrices
r(esize) a matrix of effect sizes, confidence intervals, degrees of freedom, and $F$ statistics with rows corresponding to each term in the model

## Methods and formulas

See Hamilton (2013, chap. 7), Kohler and Kreuter (2012, sec. 9.3), or Baum (2006, chap. 5) for an overview of using Stata to perform regression diagnostics. See Peracchi (2001, chap. 8) for a mathematically rigorous discussion of diagnostics.

Methods and formulas are presented under the following headings:
predict
Special-interest postestimation commands

## predict

Assume that you have already fit the regression model

$$
\mathbf{y}=\mathbf{X b}+\mathbf{e}
$$

where $\mathbf{X}$ is $n \times k$.
Denote the previously estimated coefficient vector by $\mathbf{b}$ and its estimated variance matrix by $\mathbf{V}$. predict works by recalling various aspects of the model, such as $\mathbf{b}$, and combining that information with the data currently in memory. Let $\mathbf{x}_{j}$ be the $j$ th observation currently in memory, and let $s^{2}$ be the mean squared error of the regression.

If the user specified weights in regress, then $\mathbf{X}^{\prime} \mathbf{X}$ in the following formulas is replaced by $\mathbf{X}^{\prime} \mathbf{D X}$, where $\mathbf{D}$ is defined in Weighted regression under Methods and formulas in $[\mathrm{R}]$ regress.

Let $\mathbf{V}=s^{2}\left(\mathbf{X}^{\prime} \mathbf{X}\right)^{-1}$. Let $k$ be the number of independent variables including the intercept, if any, and let $y_{j}$ be the observed value of the dependent variable.

The predicted value ( xb option) is defined as $\widehat{y}_{j}=\mathbf{x}_{j} \mathbf{b}$.
Let $\ell_{j}$ represent a lower bound for an observation $j$ and $u_{j}$ represent an upper bound. The probability that $y_{j} \mid \mathbf{x}_{j}$ would be observed in the interval $\left(\ell_{j}, u_{j}\right)$-the $\operatorname{pr}(\ell, u)$ option-is

$$
P\left(\ell_{j}, u_{j}\right)=\operatorname{Pr}\left(\ell_{j}<\mathbf{x}_{j} \mathbf{b}+e_{j}<u_{j}\right)=\Phi\left(\frac{u_{j}-\widehat{y}_{j}}{s}\right)-\Phi\left(\frac{\ell_{j}-\widehat{y}_{j}}{s}\right)
$$

where for the $\operatorname{pr}(\ell, u), \mathrm{e}(\ell, u)$, and $\operatorname{ystar}(\ell, u)$ options, $\ell_{j}$ and $u_{j}$ can be anywhere in the range $(-\infty,+\infty)$.

The option $\mathrm{e}(\ell, u)$ computes the expected value of $y_{j} \mid \mathbf{x}_{j}$ conditional on $y_{j} \mid \mathbf{x}_{j}$ being in the interval $\left(\ell_{j}, u_{j}\right)$, that is, when $y_{j} \mid \mathbf{x}_{j}$ is truncated. It can be expressed as

$$
E\left(\ell_{j}, u_{j}\right)=E\left(\mathbf{x}_{j} \mathbf{b}+e_{j} \mid \ell_{j}<\mathbf{x}_{j} \mathbf{b}+e_{j}<u_{j}\right)=\widehat{y}_{j}-s \frac{\phi\left(\frac{u_{j}-\widehat{y}_{j}}{s}\right)-\phi\left(\frac{\ell_{j}-\widehat{y}_{j}}{s}\right)}{\Phi\left(\frac{u_{j}-\widehat{y}_{j}}{s}\right)-\Phi\left(\frac{\ell_{j}-\widehat{y}_{j}}{s}\right)}
$$

where $\phi$ is the normal density and $\Phi$ is the cumulative normal.
You can also compute ystar $(\ell, u)$-the expected value of $y_{j} \mid \mathbf{x}_{j}$, where $y_{j}$ is assumed censored at $\ell_{j}$ and $u_{j}$ :

$$
y_{j}^{*}= \begin{cases}\ell_{j} & \text { if } \mathbf{x}_{j} \mathbf{b}+e_{j} \leq \ell_{j} \\ \mathbf{x}_{j} \mathbf{b}+e_{j} & \text { if } \ell_{j}<\mathbf{x}_{j} \mathbf{b}+e_{j}<u_{j} \\ u_{j} & \text { if } \mathbf{x}_{j} \mathbf{b}+e_{j} \geq u_{j}\end{cases}
$$

This computation can be expressed in several ways, but the most intuitive formulation involves a combination of the two statistics just defined:

$$
y_{j}^{*}=P\left(-\infty, \ell_{j}\right) \ell_{j}+P\left(\ell_{j}, u_{j}\right) E\left(\ell_{j}, u_{j}\right)+P\left(u_{j},+\infty\right) u_{j}
$$

A diagonal element of the projection matrix (hat) or (leverage) is given by

$$
h_{j}=\mathbf{x}_{j}\left(\mathbf{X}^{\prime} \mathbf{X}\right)^{-1} \mathbf{x}_{j}^{\prime}
$$

The standard error of the prediction (the stdp option) is defined as $s_{p_{j}}=\sqrt{\mathbf{x}_{j} \mathbf{V \mathbf { x } _ { j } ^ { \prime }}}$ and can also be written as $s_{p_{j}}=s \sqrt{h_{j}}$.

The standard error of the forecast (stdf) is defined as $s_{f_{j}}=s \sqrt{1+h_{j}}$.
The standard error of the residual (stdr) is defined as $s_{r_{j}}=s \sqrt{1-h_{j}}$.
The residuals (residuals) are defined as $\widehat{e}_{j}=y_{j}-\widehat{y}_{j}$.
The standardized residuals (rstandard) are defined as $\widehat{e}_{s_{j}}=\hat{e}_{j} / s_{r_{j}}$.
The Studentized residuals (rstudent) are defined as

$$
r_{j}=\frac{\widehat{e}_{j}}{s_{(j)} \sqrt{1-h_{j}}}
$$

where $s_{(j)}$ represents the root mean squared error with the $j$ th observation removed, which is given by

$$
s_{(j)}^{2}=\frac{s^{2}(n-k)}{n-k-1}-\frac{\widehat{e}_{j}^{2}}{(n-k-1)\left(1-h_{j}\right)}
$$

where $n$ is the number of observations and $k$ is the number of right-hand-side variables (including the constant).

Cook's $D$ (cooksd) is given by

$$
D_{j}=\frac{\widehat{e}_{s_{j}}^{2}\left(s_{p_{j}} / s_{r_{j}}\right)^{2}}{k}=\frac{h_{j} \widehat{e}_{j}^{2}}{k s^{2}\left(1-h_{j}\right)^{2}}
$$

DFITS (dfits) is given by

$$
\operatorname{DFITS}_{j}=r_{j} \sqrt{\frac{h_{j}}{1-h_{j}}}
$$

Welsch distance (welsch) is given by

$$
W_{j}=\frac{r_{j} \sqrt{h_{j}(n-1)}}{1-h_{j}}
$$

COVRATIO (covratio) is given by

$$
\operatorname{COVRATIO}_{j}=\frac{1}{1-h_{j}}\left(\frac{n-k-\widehat{e}_{s_{j}}^{2}}{n-k-1}\right)^{k}
$$

The DFBETAS (dfbeta) for a particular regressor $x_{i}$ are given by

$$
\operatorname{DFBETA}_{j}=\frac{r_{j} u_{j}}{\sqrt{U^{2}\left(1-h_{j}\right)}}
$$

where $u_{j}$ are the residuals obtained from a regression of $x_{i}$ on the remaining $x$ 's and $U^{2}=\sum_{j} u_{j}^{2}$.

## Special-interest postestimation commands

The omitted-variable test (Ramsey 1969) reported by estat ovtest fits the regression $y_{i}=$ $\mathbf{x}_{i} \mathbf{b}+\mathbf{z}_{i} \mathbf{t}+u_{i}$ and then performs a standard $F$ test of $\mathbf{t}=\mathbf{0}$. The default test uses $\mathbf{z}_{i}=\left(\widehat{y}_{i}^{2}, \widehat{y}_{i}^{3}, \widehat{y}_{i}^{4}\right)$. If $\mathbf{r h s}$ is specified, $\mathbf{z}_{i}=\left(x_{1 i}^{2}, x_{1 i}^{3}, x_{1 i}^{4}, x_{2 i}^{2}, \ldots, x_{m i}^{4}\right)$. In either case, the variables are normalized to have minimum 0 and maximum 1 before powers are calculated.

The test for heteroskedasticity (Breusch and Pagan 1979; Cook and Weisberg 1983) models $\operatorname{Var}\left(e_{i}\right)=\sigma^{2} \exp (\mathbf{z t})$, where $\mathbf{z}$ is a variable list specified by the user, the list of right-hand-side variables, or the fitted values $\mathbf{x} \widehat{\boldsymbol{\beta}}$. The test is of $\mathbf{t}=\mathbf{0}$. Mechanically, estat hettest fits the augmented regression $\widehat{e}_{i}^{2} / \widehat{\sigma}^{2}=a+\mathbf{z}_{i} \mathbf{t}+v_{i}$.

The original Breusch-Pagan/Cook-Weisberg version of the test assumes that the $e_{i}$ are normally distributed under the null hypothesis which implies that the score test statistic $S$ is equal to the model sum of squares from the augmented regression divided by 2 . Under the null hypothesis, $S$ has the $\chi^{2}$ distribution with $m$ degrees of freedom, where $m$ is the number of columns of $\mathbf{z}$.

Koenker (1981) derived a score test of the null hypothesis that $\mathbf{t}=\mathbf{0}$ under the assumption that the $e_{i}$ are independent and identically distributed (i.i.d.). Koenker showed that $S=N * R^{2}$ has a large-sample $\chi^{2}$ distribution with $m$ degrees of freedom, where $N$ is the number of observations and $R^{2}$ is from the augmented regression and $m$ is the number of columns of $\mathbf{z}$. estat hettest, iid produces this version of the test.

Wooldridge $(2020,270)$ showed that an $F$ test of $\mathbf{t}=\mathbf{0}$ in the augmented regression can also be used under the assumption that the $e_{i}$ are i.i.d. estat hettest, fstat produces this version of the test.

Szroeter's class of tests for homoskedasticity against the alternative that the residual variance increases in some variable $x$ is defined in terms of

$$
H=\frac{\sum_{i=1}^{n} h\left(x_{i}\right) e_{i}^{2}}{\sum_{i=1}^{n} e_{i}^{2}}
$$

where $h(x)$ is some weight function that increases in $x$ (Szroeter 1978). $H$ is a weighted average of the $h(x)$, with the squared residuals serving as weights. Under homoskedasticity, $H$ should be approximately equal to the unweighted average of $h(x)$. Large values of $H$ suggest that $e_{i}^{2}$ tends to be large where $h(x)$ is large; that is, the variance indeed increases in $x$, whereas small values of $H$ suggest that the variance actually decreases in $x$. estat szroeter uses $h\left(x_{i}\right)=\operatorname{rank}\left(x_{i}\right.$ in $\left.x_{1} \ldots x_{n}\right)$; see Judge et al. [1985, 452] for details. estat szroeter displays a normalized version of $H$,

$$
Q=\sqrt{\frac{6 n}{n^{2}-1}} H
$$

which is approximately $N(0,1)$ distributed under the null (homoskedasticity).
estat hettest and estat szroeter provide adjustments of $p$-values for multiple testing. The supported methods are described in [R] test.
estat imtest performs the information matrix test for the regression model, as well as an orthogonal decomposition into tests for heteroskedasticity $\delta_{1}$, nonnormal skewness $\delta_{2}$, and nonnormal kurtosis $\delta_{3}$ (Cameron and Trivedi 1990; Long and Trivedi 1993). The decomposition is obtained via three auxiliary regressions. Let $e$ be the regression residuals, $\widehat{\sigma}^{2}$ be the maximum likelihood estimate of $\sigma^{2}$ in the regression, $n$ be the number of observations, $X$ be the set of $k$ variables specified with estat imtest, and $R_{\text {un }}^{2}$ be the uncentered $R^{2}$ from a regression. $\delta_{1}$ is obtained as $n R_{\text {un }}^{2}$ from a regression of $e^{2}-\widehat{\sigma}^{2}$ on the cross products of the variables in $X . \delta_{2}$ is computed as $n R_{\text {un }}^{2}$ from a regression of $e^{3}-3 \widehat{\sigma}^{2} e$ on $X$. Finally, $\delta_{3}$ is obtained as $n R_{\mathrm{un}}^{2}$ from a regression of $e^{4}-6 \widehat{\sigma}^{2} e^{2}-3 \widehat{\sigma}^{4}$ on $X . \delta_{1}, \delta_{2}$, and $\delta_{3}$ are asymptotically $\chi^{2}$ distributed with $1 / 2 k(k+1), K$, and 1 degree of freedom. The information test statistic $\delta=\delta_{1}+\delta_{2}+\delta_{3}$ is asymptotically $\chi^{2}$ distributed with $1 / 2 k(k+3)$ degrees of freedom. White's test for heteroskedasticity is computed as $n R^{2}$ from a regression of $\widehat{u}^{2}$ on $X$ and the cross products of the variables in $X$. This test statistic is usually close to $\delta_{1}$.
estat vif calculates the centered variance inflation factor $\left(\mathrm{VIF}_{c}\right)$ (Chatterjee and Hadi 2012, 248-251) for $x_{j}$, given by

$$
\operatorname{VIF}_{c}\left(x_{j}\right)=\frac{1}{1-\widehat{R}_{j}^{2}}
$$

where $\widehat{R}_{j}^{2}$ is the square of the centered multiple correlation coefficient that results when $x_{j}$ is regressed against all other explanatory variables, including the constant.

The uncentered variance inflation factor ( $\mathrm{VIF}_{u c}$ ) (Belsley 1991, 28-29) for $x_{j}$ is given by

$$
\operatorname{VIF}_{u c}\left(x_{j}\right)=\frac{1}{1-\widetilde{R}_{j}^{2}}
$$

where $\widetilde{R}_{j}^{2}$ is the square of the uncentered multiple correlation coefficient that results when $x_{j}$ is regressed against all other explanatory variables and a constant of 1 . If the original regression model was fit without a constant, the constant would also be omitted from the regression of $x_{j}$.

The methods and formulas for estat esize are described in Methods and formulas of [R] esize.

## Acknowledgments

estat ovtest and estat hettest are based on programs originally written by Richard Goldstein. estat imtest, estat szroeter, and the current version of estat hettest were written by Jeroen Weesie of the Department of Sociology at Utrecht University, The Netherlands. estat imtest is based in part on code written by J. Scott Long of the Department of Sociology at Indiana University, coauthor of the Stata Press book Regression Models for Categorical and Limited Dependent Variables, and author of the Stata Press book The Workflow of Data Analysis Using Stata.

## References

Adkins, L. C., and R. C. Hill. 2011. Using Stata for Principles of Econometrics. 4th ed. Hoboken, NJ: Wiley.
Baum, C. F. 2006. An Introduction to Modern Econometrics Using Stata. College Station, TX: Stata Press.
Belsley, D. A. 1991. Conditional Diagnostics: Collinearity and Weak Data in Regression. New York: Wiley.
Belsley, D. A., E. Kuh, and R. E. Welsch. 1980. Regression Diagnostics: Identifying Influential Data and Sources of Collinearity. New York: Wiley.

Bollen, K. A., and R. W. Jackman. 1990. Regression diagnostics: An expository treatment of outliers and influential cases. In Modern Methods of Data Analysis, ed. J. Fox and J. S. Long, 257-291. Newbury Park, CA: Sage.

Breusch, T. S., and A. R. Pagan. 1979. A simple test for heteroscedasticity and random coefficient variation. Econometrica 47: 1287-1294. https://doi.org/10.2307/1911963.

Cameron, A. C., and P. K. Trivedi. 1990. The information matrix test and its applied alternative hypotheses. Working paper 372, University of California-Davis, Institute of Governmental Affairs.
——. 2022. Microeconometrics Using Stata. 2nd ed. College Station, TX: Stata Press.
Chatterjee, S., and A. S. Hadi. 1986. Influential observations, high leverage points, and outliers in linear regression. Statistical Science 1: 379-393. https://doi.org/10.1214/ss/1177013622.
——. 1988. Sensitivity Analysis in Linear Regression. New York: Wiley.
——. 2012. Regression Analysis by Example. 5th ed. New York: Wiley.
Cook, R. D. 1977. Detection of influential observation in linear regression. Technometrics 19: 15-18. https://doi.org/10.1080/00401706.1977.10489493.

Cook, R. D., and S. Weisberg. 1982. Residuals and Influence in Regression. New York: Chapman and Hall/CRC.
—_. 1983. Diagnostics for heteroscedasticity in regression. Biometrika 70: 1-10. https://doi.org/10.2307/2335938.
Cox, N. J. 2004. Speaking Stata: Graphing model diagnostics. Stata Journal 4: 449-475.
Cumming, G. 2012. Understanding the New Statistics: Effect Sizes, Confidence Intervals, and Meta-Analysis. New York: Routledge.

DeMaris, A. 2004. Regression with Social Data: Modeling Continuous and Limited Response Variables. Hoboken, NJ: Wiley.
Ellis, P. D. 2010. The Essential Guide to Effect Sizes: Statistical Power, Meta-Analysis, and the Interpretation of Research Results. Cambridge: Cambridge University Press.

Grissom, R. J., and J. J. Kim. 2012. Effect Sizes for Research: Univariate and Multivariate Applications. 2nd ed. New York: Routledge.
Hamilton, L. C. 2013. Statistics with Stata: Updated for Version 12. 8th ed. Boston: Brooks/Cole.
Hill, R. C., W. E. Griffiths, and G. C. Lim. 2018. Principles of Econometrics. 5th ed. Hoboken, NJ: Wiley.
Hoaglin, D. C., and P. J. Kempthorne. 1986. Comment [on Chatterjee and Hadi 1986]. Statistical Science 1: 408-412. https://doi.org/10.1214/ss/1177013627.

Hoaglin, D. C., and R. E. Welsch. 1978. The hat matrix in regression and ANOVA. American Statistician 32: 17-22. https://doi.org/10.1080/00031305.1978.10479237.

Huber, C. 2013. Measures of effect size in Stata 13. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2013/09/05/measures-of-effect-size-in-stata-13/.

Judge, G. G., W. E. Griffiths, R. C. Hill, H. Lütkepohl, and T.-C. Lee. 1985. The Theory and Practice of Econometrics. 2nd ed. New York: Wiley.
Kelley, K., and K. J. Preacher. 2012. On effect size. Psychological Methods 17: 137-152. https://doi.org/10.1037/a0028086.

Kerlinger, F. N., and H. B. Lee. 2000. Foundations of Behavioral Research. 4th ed. Belmont, CA: Wadsworth.
Kirk, R. E. 1996. Practical significance: A concept whose time has come. Educational and Psychological Measurement 56: 746-759. https://doi.org/10.1177/0013164496056005002.

Kline, R. B. 2013. Beyond Significance Testing: Statistics Reform in the Behavioral Sciences. 2nd ed. Washington, DC: American Psychological Association.

Koenker, R. 1981. A note on studentizing a test for heteroskedasticity. Journal of Econometrics 17: 107-112. https://doi.org/10.1016/0304-4076(81)90062-2.

Kohler, U., and F. Kreuter. 2012. Data Analysis Using Stata. 3rd ed. College Station, TX: Stata Press.
Kutner, M. H., C. J. Nachtsheim, and J. Neter. 2004. Applied Linear Regression Models. 4th ed. New York: McGraw-Hill.

Lindsey, C., and S. J. Sheather. 2010a. Optimal power transformation via inverse response plots. Stata Journal 10: 200-214.
__. 2010b. Model fit assessment via marginal model plots. Stata Journal 10: 215-225.
Long, J. S., and P. K. Trivedi. 1993. Some specification tests for the linear regression model. Sociological Methods and Research 21: 161-204. Reprinted in Testing Structural Equation Models, ed. K. A. Bollen and J. S. Long, pp. 66-110. Newbury Park, CA: Sage.
Peracchi, F. 2001. Econometrics. Chichester, UK: Wiley.
Ramsey, J. B. 1969. Tests for specification errors in classical linear least-squares regression analysis. Journal of the Royal Statistical Society, Series B 31: 350-371. https://doi.org/10.1111/j.2517-6161.1969.tb00796.x.

Ramsey, J. B., and P. Schmidt. 1976. Some further results on the use of OLS and BLUS residuals in specification error tests. Journal of the American Statistical Association 71: 389-390. https://doi.org/10.1080/01621459.1976.10480355.
Rousseeuw, P. J., and A. M. Leroy. 1987. Robust Regression and Outlier Detection. New York: Wiley.
Smithson, M. 2001. Correct confidence intervals for various regression effect sizes and parameters: The importance of noncentral distributions in computing intervals. Educational and Psychological Measurement 61: 605-632. https://doi.org/10.1177/00131640121971392.
Studenmund, A. H. 2017. Using Econometrics: A Practical Guide. 7th ed. Boston: Pearson.
Szroeter, J. 1978. A class of parametric tests for heteroscedasticity in linear econometric models. Econometrica 46: 1311-1327. https://doi.org/10.2307/1913831.
Thompson, B. 2006. Foundations of Behavioral Statistics: An Insight-Based Approach. New York: Guilford Press.
Velleman, P. F. 1986. Comment [on Chatterjee and Hadi 1986]. Statistical Science 1: 412-413. https://doi.org/10.1214/ss/1177013628.
Velleman, P. F., and R. E. Welsch. 1981. Efficient computing of regression diagnostics. American Statistician 35: 234-242. https://doi.org/10.2307/2683296.

Weisberg, S. 2014. Applied Linear Regression. 4th ed. Hoboken, NJ: Wiley.
Welsch, R. E. 1982. Influence functions and regression diagnostics. In Modern Data Analysis, ed. R. L. Launer and A. F. Siegel, 149-169. New York: Academic Press.
__. 1986. Comment [on Chatterjee and Hadi 1986]. Statistical Science 1: 403-405. https://doi.org/10.1214/ss/1177013625.

Welsch, R. E., and E. Kuh. 1977. Linear Regression Diagnostics. Technical Report 923-977, Massachusetts Institute of Technology, Cambridge, MA.
White, H. L., Jr. 1980. A heteroskedasticity-consistent covariance matrix estimator and a direct test for heteroskedasticity. Econometrica 48: 817-838. https://doi.org/10.2307/1912934.

Wooldridge, J. M. 2020. Introductory Econometrics: A Modern Approach. 7th ed. Boston: Cengage.

## Also see

$[R]$ regress - Linear regression
[R] regress postestimation diagnostic plots - Postestimation plots for regress
$[R]$ regress postestimation time series - Postestimation tools for regress with time series
[SP] estat moran - Moran's test of residual correlation with nearby residuals
[LASSO] lassogof - Goodness of fit after lasso for prediction
[U] 20 Estimation and postestimation commands

## Title

regress postestimation diagnostic plots - Postestimation plots for regress

| Description | rvfplot | avplot | avplots | cprplot | acprplot |
| :--- | :--- | :--- | :--- | :--- | :--- |
| rvpplot | lvr2plot | Methods and formulas | References | Also see |  |

## Description

The following postestimation commands are of special interest after regress:
Command Description

| rvfplot | residual-versus-fitted plot |
| :--- | :--- |
| avplot | added-variable plot |
| avplots | all added-variables plots in one image |
| cprplot | component-plus-residual plot |
| acprplot | augmented component-plus-residual plot |
| rvpplot | residual-versus-predictor plot |
| lvr2plot | leverage-versus-squared-residual plot |

These commands are not appropriate with svy estimation results.
For a discussion of the terminology used in this entry, see the Terminology section of Remarks and examples for predict in [R] regress postestimation.

## rvfplot

## Description for rvfplot

rvfplot graphs a residual-versus-fitted plot, a graph of the residuals against the fitted values.

## Menu for rvfplot

```
Statistics > Linear models and related > Regression diagnostics > Residual-versus-fitted plot
```


## Syntax for rvfplot

```
rvfplot [, rvfplot_options]
```

rvfplot_options Description

Plot
marker_options
marker_label_options
Add plots
addplot (plot) add plots to the generated graph
Y axis, X axis, Titles, Legend, Overall
twoway_options
any options other than by() documented in [G-3] twoway_options

## Options for rvfplot

$\qquad$ Plot
marker_options affect the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G-3] marker_options.
marker_label_options specify if and how the markers are to be labeled; see [G-3] marker_label_options.
$\qquad$ Add plots
addplot (plot) provides a way to add plots to the generated graph. See [G-3] addplot_option.
$\qquad$ Y axis, X axis, Titles, Legend, Overall
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

## Remarks and examples for rvfplot

rvfplot graphs the residuals against the fitted values.

## $>$ Example 1

Using auto.dta described in [U] 1.2.2 Example datasets, we will use regress to fit a model of price on weight, mpg, foreign, and the interaction of foreign with mpg. We specify foreign\#\#c.mpg to obtain the interaction of foreign with mpg; see [U] 11.4.3 Factor variables.

| . use https://www.stata-press.com/data (1978 automobile data) <br> . regress price weight foreign\#\#c.mpg |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Source | SS | df | MS | Number of obs$F(4,69)$ |  | 74 |
| Model | 350319665 | 4 | 87579916.3 | Prob > F |  | 21.22 0.0000 |
| Residual | 284745731 | 69 | 4126749.72 | R-squared <br> Adj R-squared |  | 0.5516 |
|  |  |  |  |  |  | 0.5256 |
| Total | 635065396 | 73 | 8699525.97 | Root MSE |  | 2031.4 |
| price | Coefficient | Std. err | . | $P>\|t\|$ | [95\% conf. | interval] |
| weight | 4.613589 | . 7254961 | 6.36 | 0.000 | 3.166263 | 6.060914 |
| foreign |  |  |  |  |  |  |
| Foreign | 11240.33 | 2751.681 | 4.08 | 0.000 | 5750.878 | 16729.78 |
| mpg | 263.1875 | 110.7961 | 2.38 | 0.020 | 42.15527 | 484.2197 |
| foreign\#c.mpg |  |  |  |  |  |  |
| Foreign | -307.2166 | 108.5307 | -2.83 | 0.006 | -523.7294 | -90.70368 |
| _cons | -14449.58 | 4425.72 | -3.26 | 0.002 | -23278.65 | -5620.51 |

Once we have fit a model, we may use any of the regression diagnostics commands. rvfplot (read residual-versus-fitted plot) graphs the residuals against the fitted values:

```
. rvfplot, yline(0)
```



All the diagnostic plot commands allow the graph twoway and graph twoway scatter options; we specified a yline (0) to draw a line across the graph at $y=0$; see [G-2] graph twoway scatter.

In a well-fitted model, there should be no pattern to the residuals plotted against the fitted values-something not true of our model. Ignoring the two outliers at the top center of the graph, we see curvature in the pattern of the residuals, suggesting a violation of the assumption that price is linear in our independent variables. We might also have seen increasing or decreasing variation in the residuals-heteroskedasticity. Any pattern whatsoever indicates a violation of the least-squares assumptions.

## avplot

## Description for avplot

avplot graphs an added-variable plot (a.k.a. partial-regression leverage plot, partial regression plot, or adjusted partial residual plot) after regress. indepvar may be an independent variable (a.k.a. predictor, carrier, or covariate) that is currently in the model or not.

## Menu for avplot

Statistics $>$ Linear models and related $>$ Regression diagnostics $>$ Added-variable plot

## Syntax for avplot

$$
\text { avplot indepvar }[\text {, avplot_options }]
$$

avplot_options Description
Plot
marker_options change look of markers (color, size, etc.)
marker_label_options add marker labels; change look or position
Reference line
rlopts (cline_options) affect rendition of the reference line
Add plots
addplot (plot) add other plots to the generated graph
Y axis, X axis, Titles, Legend, Overall
twoway_options any options other than by () documented in [G-3] twoway_options

## Options for avplot

$\qquad$ $\sqrt{\text { Plot }}$
marker_options affects the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G-3] marker_options.
marker_label_options specify if and how the markers are to be labeled; see [G-3] marker_label_options.

> Reference line
> rlopts (cline_options) affects the rendition of the reference line. See [G-3] cline_options.

$\xlongequal{\text { Add plots }}$
addplot (plot) provides a way to add plots to the generated graph. See [G-3] addplot_option.
$\qquad$ Y axis, X axis, Titles, Legend, Overall
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

## Remarks and examples for avplot

avplot graphs an added-variable plot, also known as the partial-regression leverage plot.
One of the wonderful features of one-regressor regressions (regressions of $y$ on one $x$ ) is that we can graph the data and the regression line. There is no easier way to understand the regression than to examine such a graph. Unfortunately, we cannot do this when we have more than one regressor. With two regressors, it is still theoretically possible - the graph must be drawn in three dimensions, but with three or more regressors no graph is possible.

The added-variable plot is an attempt to project multidimensional data back to the two-dimensional world for each of the original regressors. This is, of course, impossible without making some concessions. Call the coordinates on an added-variable plot $y$ and $x$. The added-variable plot has the following properties:

- There is a one-to-one correspondence between $\left(x_{i}, y_{i}\right)$ and the $i$ th observation used in the original regression.
- A regression of $y$ on $x$ has the same coefficient and standard error (up to a degree-of-freedom adjustment) as the estimated coefficient and standard error for the regressor in the original regression.
- The "outlierness" of each observation in determining the slope is in some sense preserved.

It is equally important to note the properties that are not listed. The $y$ and $x$ coordinates of the added-variable plot cannot be used to identify functional form, or, at least, not well (see Mallows [1986]). In the construction of the added-variable plot, the relationship between $y$ and $x$ is forced to be linear.

## > Example 2

Let's use the same model as we used in example 1.

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. regress price weight foreign##c.mpg
(output omitted)
```

We can now examine the added-variable plot for mpg. . avplot mpg


This graph suggests a problem in determining the coefficient on mpg. Were this a one-regressor regression, the two points at the top-left corner and the one at the top right would cause us concern, and so it does in our more complicated multiple-regressor case. To identify the problem points, we retyped our command, modifying it to read avplot mpg, mlabel(make), and discovered that the two cars at the top left are the Cadillac Eldorado and the Lincoln Versailles; the point at the top right is the Cadillac Seville. These three cars account for $100 \%$ of the luxury cars in our data, suggesting that our model is misspecified. By the way, the point at the lower right of the graph, also cause for concern, is the Plymouth Arrow, our data entry error.

## - Technical note

Stata's avplot command can be used with regressors already in the model, as we just did, or with potential regressors not yet in the model. In either case, avplot will produce the correct graph. The name "added-variable plot" is unfortunate in the case when the variable is already among the list of regressors but is, we think, still preferable to the name "partial-regression leverage plot" assigned by Belsley, Kuh, and Welsch $(1980,30)$ and more in the spirit of the original use of such plots by Mosteller and Tukey (1977, 271-279). Welsch (1986, 403), however, disagrees: "I am sorry to see that Chatterjee and Hadi [1986] endorse the term 'added-variable plot' when $X_{j}$ is part of the original model" and goes on to suggest the name "adjusted partial residual plot".

## avplots

## Description for avplots

avplots graphs all the added-variable plots in one image.

## Menu for avplots

Statistics $>$ Linear models and related $>$ Regression diagnostics $>$ Added-variable plot

## Syntax for avplots

> avplots [, avplots_options]
avplots_options
Description

Plot
marker_options
marker_label_options
combine_options
Reference line
rlopts (cline_options) affect rendition of the reference line
Y axis, X axis, Titles, Legend, Overall
twoway_options
change look of markers (color, size, etc.)
add marker labels; change look or position
any of the options documented in [G-2] graph combine
any options other than by() documented in [G-3] twoway_options

## Options for avplots

$\qquad$ Plot
marker_options affects the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G-3] marker_options.
marker_label_options specify if and how the markers are to be labeled; see [G-3] marker_label_options. combine_options are any of the options documented in [G-2] graph combine. These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).
$\qquad$ Reference line
rlopts (cline_options) affects the rendition of the reference line. See [G-3] cline_options.
$\sqrt{Y}$ axis, X axis, Titles, Legend, Overall
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

## Remarks and examples for avplots

## > Example 3

In example 2, we used avplot to examine the added-variable plot for mpg in our regression of price on weight and foreign\#\#c.mpg. Now, let's use avplots to graph an added-variable plot for every regressor in the data.


## cprplot

## Description for cprplot

cprplot graphs a component-plus-residual plot (a.k.a. partial residual plot) after regress. indepvar must be an independent variable that is currently in the model.

## Menu for cprplot

Statistics $>$ Linear models and related $>$ Regression diagnostics $>$ Component-plus-residual plot

## Syntax for cprplot

> cprplot indepvar $[$, cprplot_options $]$
> cprplot_options

Plot
marker_options
marker_label_options
Reference line
rlopts (cline_options)
Options
lowess
lsopts(lowess_options)
mspline
msopts (mspline_options)
Add plots
addplot (plot) add other plots to the generated graph
Y axis, X axis, Titles, Legend, Overall
twoway_options
change look of markers (color, size, etc.)
add marker labels; change look or position
affect rendition of the reference line
add a lowess smooth of the plotted points
affect rendition of the lowess smooth
add median spline of the plotted points
affect rendition of the spline

## Options for cprplot

$\qquad$ Plot
marker_options affects the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G-3] marker_options.
marker_label_options specify if and how the markers are to be labeled; see [G-3] marker_label_options.
$\qquad$ Reference line
rlopts (cline_options) affects the rendition of the reference line. See [G-3] cline_options.

Options
lowess adds a lowess smooth of the plotted points to assist in detecting nonlinearities.
lsopts(lowess_options) affects the rendition of the lowess smooth. For an explanation of these options, especially the bwidth() option, see [R] lowess. Specifying lsopts() implies the lowess option.
mspline adds a median spline of the plotted points to assist in detecting nonlinearities.
msopts (mspline_options) affects the rendition of the spline. For an explanation of these options, especially the bands() option, see [G-2] graph twoway mspline. Specifying msopts() implies the mspline option.

Add plots
addplot (plot) provides a way to add other plots to the generated graph. See [G-3] addplot_option.

Y axis, X axis, Titles, Legend, Overall
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

## Remarks and examples for cprplot

Added-variable plots are successful at identifying outliers, but they cannot be used to identify functional form. The component-plus-residual plot (Ezekiel 1924; Larsen and McCleary 1972) is another attempt at projecting multidimensional data into a two-dimensional form, but with different properties. Although the added-variable plot can identify outliers, the component-plus-residual plot cannot. It can, however, be used to examine the functional form assumptions of the model. Both plots have the property that a regression line through the coordinates has a slope equal to the estimated coefficient in the regression model.
> Example 4
We illustrate component-plus-residual plots using a variation of auto.dta.

```
. use https://www.stata-press.com/data/r18/auto1
(Automobile models)
```

. regress price mpg weight

| Source | SS | df | MS |
| ---: | ---: | ---: | ---: |
| Model | 187716578 | 2 | 93858289 |
| Residual | 447348818 | 71 | 6300687.58 |
| Total | 635065396 | 73 | 8699525.97 |


| Number of obs | $=$ | 74 |
| :--- | :--- | ---: |
| $\mathrm{~F}(2,71)$ | $=$ | 14.90 |
| Prob $>$ F | $=$ | 0.0000 |
| R-squared | $=$ | 0.2956 |
| Adj R-squared | $=$ | 0.2757 |
| Root MSE | $=$ | 2510.1 |


| price | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :--- | :---: | :---: | ---: | ---: |
| mpg | -55.9393 | 75.24136 | -0.74 | 0.460 | -205.9663 | 94.08771 |
| weight | 1.710992 | .5861682 | 2.92 | 0.005 | .5422063 | 2.879779 |
| _cons | 2197.9 | 3190.768 | 0.69 | 0.493 | -4164.311 | 8560.11 |

In fact, we know that the effects of mpg in this model are nonlinear-if we added mpg squared to the model, its coefficient would have a $t$ statistic of 2.38 , the $t$ statistic on mpg would become -2.48 , and weight's effect would become about one-third of its current value and become statistically insignificant. Pretend that we do not know this.

The component-plus-residual plot for mpg is

```
. cprplot mpg, mspline msopts(bands(13))
```



We are supposed to examine the above graph for nonlinearities or, equivalently, ask if the regression line, which has slope equal to the estimated effect of mpg in the original model, fits the data adequately. To assist our eyes, we added a median spline. Perhaps some people may detect nonlinearity from this graph, but we assert that if we had not previously revealed the nonlinearity of mpg and if we had not added the median spline, the graph would not overly bother us.

## acprplot

## Description for acprplot

acprplot graphs an augmented component-plus-residual plot (a.k.a. augmented partial residual plot) as described by Mallows (1986). This seems to work better than the component-plus-residual plot for identifying nonlinearities in the data.

## Menu for acprplot

Statistics $>$ Linear models and related $>$ Regression diagnostics $>$ Augmented component-plus-residual plot

## Syntax for acprplot

$$
\text { acprplot indepvar }[\text {, acprplot_options }]
$$

acprplot_options
Description

Plot
marker_options
marker_label_options
Reference line
rlopts (cline_options)
Options
lowess
lsopts (lowess_options)
mspline msopts (mspline_options)

Add plots addplot (plot) add other plots to the generated graph
change look of markers (color, size, etc.)
add marker labels; change look or position
affect rendition of the reference line
add a lowess smooth of the plotted points affect rendition of the lowess smooth add median spline of the plotted points affect rendition of the spline

Y axis, X axis, Titles, Legend, Overall
twoway_options

## Options for acprplot

$\qquad$ Plot
marker_options affects the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G-3] marker_options.
marker_label_options specify if and how the markers are to be labeled; see [G-3] marker_label_options.


Options
lowess adds a lowess smooth of the plotted points to assist in detecting nonlinearities.
lsopts(lowess_options) affects the rendition of the lowess smooth. For an explanation of these options, especially the bwidth () option, see [R] lowess. Specifying lsopts() implies the lowess option.
mspline adds a median spline of the plotted points to assist in detecting nonlinearities.
msopts (mspline_options) affects the rendition of the spline. For an explanation of these options, especially the bands() option, see [G-2] graph twoway mspline. Specifying msopts() implies the mspline option.

Add plots
addplot (plot) provides a way to add other plots to the generated graph. See [G-3] addplot_option.

Y axis, X axis, Titles, Legend, Overall
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

## Remarks and examples for acprplot

In the cprplot section above, we discussed the component-plus-residual plot. Mallows (1986) proposed an augmented component-plus-residual plot that is often more sensitive to detecting nonlinearity.

## > Example 5

Let's compare the augmented component-plus-residual plot with the component-plus-residual plot of example 4.

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. regress price weight foreign##c.mpg
    (output omitted)
. acprplot mpg, mspline msopts(bands(13))
```



It does do somewhat better.

## rvpplot

## Description for rvpplot

rvpplot graphs a residual-versus-predictor plot (a.k.a. independent variable plot or carrier plot), a graph of the residuals against the specified predictor.

## Menu for rvpplot

Statistics $>$ Linear models and related $>$ Regression diagnostics $>$ Residual-versus-predictor plot

## Syntax for rvpplot

rvpplot indepvar [, rvpplot_options]

## rvpplot_options Description

Plot
marker_options
marker_label_options
Add plots
addplot (plot) add plots to the generated graph
Y axis, X axis, Titles, Legend, Overall
twoway_options any options other than by () documented in [G-3] twoway_options

## Options for rvpplot

$\qquad$ Plot
marker_options affects the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G-3] marker_options.
marker_label_options specify if and how the markers are to be labeled; see [G-3] marker_label_options.

[^53]$\qquad$ Y axis, X axis, Titles, Legend, Overall
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

## Remarks and examples for rvpplot

The residual-versus-predictor plot is a simple way to look for violations of the regression assumptions. If the assumptions are correct, there should be no pattern on the graph.
> Example 6
Let's use our model of price on mpg and weight.

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. regress price weight foreign##c.mpg
    (output omitted)
. rvpplot mpg, yline(0)
```



Remember, any pattern counts as a problem, and in this graph, we see that the variation in the residuals decreases as mpg increases.

## Ivr2plot

## Description for Ivr2plot

lvr2plot graphs a leverage-versus-squared-residual plot (a.k.a. L-R plot).

## Menu for Ivr2plot

Statistics $>$ Linear models and related $>$ Regression diagnostics $>$ Leverage-versus-squared-residual plot

## Syntax for Ivr2plot

lvr2plot [, lvr2plot_options]
lvr2plot_options Description
Plot
marker_options change look of markers (color, size, etc.)
marker_label_options
add marker labels; change look or position
Add plots
addplot (plot) add other plots to the generated graph
Y axis, X axis, Titles, Legend, Overall
twoway_options
any options other than by () documented in [G-3] twoway_options

## Options for Ivr2plot

$\qquad$ Plot
marker_options affects the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G-3] marker_options.
marker_label_options specify if and how the markers are to be labeled; see [G-3] marker_label_options.

Add plots
addplot (plot) provides a way to add other plots to the generated graph. See [G-3] addplot_option.

Y axis, X axis, Titles, Legend, Overall
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

## Remarks and examples for Ivr2plot

One of the most useful diagnostic graphs is provided by lvr2plot (leverage-versus-residual-squared plot), a graph of leverage against the (normalized) residuals squared.

## > Example 7

We illustrate 1 vr 2 plot using our model in example 1.

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. regress price weight foreign##c.mpg
    (output omitted)
. lvr2plot
```



The lines on the chart show the average values of leverage and the (normalized) residuals squared. Points above the horizontal line have higher-than-average leverage; points to the right of the vertical line have larger-than-average residuals.

One point immediately catches our eye, and four more make us pause. The point at the top of the graph has high leverage and a smaller-than-average residual. The other points that bother us all have higher-than-average leverage, two with smaller-than-average residuals and two with larger-than-average residuals.

A less pretty but more useful version of the above graph specifies that make be used as the symbol (see [G-3] marker_label_options):

```
. lvr2plot, mlabel(make) mlabp(0) m(none) mlabsize(small)
```



The Vw Diesel, Plymouth Champ, Plymouth Arrow, and Peugeot 604 are the points that cause us the most concern. When we further examine our data, we discover that the vw Diesel is the only diesel in our data and that the data for the Plymouth Arrow were entered incorrectly into the computer. No such simple explanations were found for the Plymouth Champ and Peugeot 604.

## Methods and formulas

See Hamilton (2013, 209-214) and Kohler and Kreuter (2012, sec. 9.3) for a discussion of these diagnostic graphs.

The lvr2plot command plots leverage against the squares of the normalized residuals. The normalized residuals are defined as $\widehat{e}_{n_{j}}=\widehat{e}_{j} /\left(\sum_{i} \widehat{e}_{i}^{2}\right)^{1 / 2}$.

## References

Belsley, D. A., E. Kuh, and R. E. Welsch. 1980. Regression Diagnostics: Identifying Influential Data and Sources of Collinearity. New York: Wiley.

Chatterjee, S., and A. S. Hadi. 1986. Influential observations, high leverage points, and outliers in linear regression. Statistical Science 1: 379-393. https://doi.org/10.1214/ss/1177013622.
Cox, N. J. 2004. Speaking Stata: Graphing model diagnostics. Stata Journal 4: 449-475.
Ezekiel, M. 1924. A method of handling curvilinear correlation for any number of variables. Journal of the American Statistical Association 19: 431-453. https://doi.org/10.2307/2281561.
Gallup, J. L. 2019. Added-variable plots with confidence intervals. Stata Journal 19: 598-614.
Hamilton, L. C. 2013. Statistics with Stata: Updated for Version 12. 8th ed. Boston: Brooks/Cole.
Hoaglin, D. C., and R. E. Welsch. 1978. The hat matrix in regression and ANOVA. American Statistician 32: 17-22. https://doi.org/10.1080/00031305.1978.10479237.
Kohler, U., and F. Kreuter. 2012. Data Analysis Using Stata. 3rd ed. College Station, TX: Stata Press.
Larsen, W. A., and S. J. McCleary. 1972. The use of partial residual plots in regression analysis. Technometrics 14: 781-790. https://doi.org/10.1080/00401706.1972.10488966.
Lindsey, C., and S. J. Sheather. 2010a. Optimal power transformation via inverse response plots. Stata Journal 10: 200-214.

2010b. Model fit assessment via marginal model plots. Stata Journal 10: 215-225.
Mallows, C. L. 1986. Augmented partial residuals. Technometrics 28: 313-319. https://doi.org/10.2307/1268980.
Mosteller, C. F., and J. W. Tukey. 1977. Data Analysis and Regression: A Second Course in Statistics. Reading, MA: Addison-Wesley.
Pedace, R. 2013. Econometrics for Dummies. Hoboken, NJ: Wiley.
Welsch, R. E. 1986. Comment [on Chatterjee and Hadi 1986]. Statistical Science 1: 403-405. https://doi.org/10.1214/ss/1177013625.

## Also see

[ R ] regress - Linear regression
[R] regress postestimation - Postestimation tools for regress
$[\mathrm{R}]$ regress postestimation time series - Postestimation tools for regress with time series
[U] 20 Estimation and postestimation commands

## Title

regress postestimation time series - Postestimation tools for regress with time series

Postestimation commands estat durbinalt
Stored results
References

estat archlm<br>estat dwatson<br>Methods and formulas<br>Also see

estat bgodfrey
Remarks and examples
Acknowledgment

## Postestimation commands

The following postestimation commands for time series are available for regress:
Command Description
estat archlm test for ARCH effects in the residuals
estat bgodfrey
estat durbinalt
estat dwatson
estat sbcusum
estat sbknown
estat sbsingle

Breusch-Godfrey test for higher-order serial correlation
Durbin's alternative test for serial correlation
Durbin-Watson $d$ statistic to test for first-order serial correlation perform cumulative sum test for parameter stability perform tests for a structural break with a known break date perform tests for a structural break with an unknown break date

These commands provide regression diagnostic tools specific to time series. You must tsset your data before using these commands; see [TS] tsset.

## estat archlm

## Description for estat archlm

estat archlm performs Engle's Lagrange multiplier (LM) test for the presence of autoregressive conditional heteroskedasticity.

## Menu for estat

Statistics > Postestimation

## Syntax for estat archlm

estat archlm [, archlm_options]
archlm_options Description

| lags (numlist) | test numlist lag orders |
| :--- | :--- |
| force | allow test after regress, vce(robust) |

[^54]
## Options for estat archlm

lags(numlist) specifies a list of numbers, indicating the lag orders to be tested. The test will be performed separately for each order. The default is order one.
force allows the test to be run after regress, vce(robust). The command will not work if the vce(cluster clustvar) option is specified with regress; see [ $R$ ] regress.

## estat bgodfrey

## Description for estat bgodfrey

estat bgodfrey performs the Breusch-Godfrey test for higher-order serial correlation in the disturbance. This test does not require that all the regressors be strictly exogenous.

## Menu for estat

Statistics $>$ Postestimation

## Syntax for estat bgodfrey

```
estat bgodfrey [, bgodfrey_options]
```

bgodfrey_options Description

| lags(numlist) | test numlist lag orders |
| :--- | :--- |
| nomiss0 | do not use Davidson and MacKinnon's approach |
| small | obtain $p$-values using the $F$ or $t$ distribution |

collect is allowed; see [U] 11.1.10 Prefix commands.

## Options for estat bgodfrey

lags (numlist) specifies a list of numbers, indicating the lag orders to be tested. The test will be performed separately for each order. The default is order one.
nomiss0 specifies that Davidson and MacKinnon's approach (1993, 358), which replaces the missing values in the initial observations on the lagged residuals in the auxiliary regression with zeros, not be used.
small specifies that the $p$-values of the test statistics be obtained using the $F$ or $t$ distribution instead of the default $\chi^{2}$ or normal distribution.

## estat durbinalt

## Description for estat durbinalt

estat durbinalt performs Durbin's alternative test for serial correlation in the disturbance. This test does not require that all the regressors be strictly exogenous.

## Menu for estat

Statistics $>$ Postestimation

## Syntax for estat durbinalt

estat durbinalt [, durbinalt_options]
durbinalt_options
lags (numlist) test numlist lag orders
nomiss0
robust
small
force
Description
do not use Davidson and MacKinnon's approach
compute standard errors using the robust/sandwich estimator
obtain $p$-values using the $F$ or $t$ distribution
allow test after regress, vce(robust) or after newey
collect is allowed; see [U] 11.1.10 Prefix commands.

## Options for estat durbinalt

lags (numlist) specifies a list of numbers, indicating the lag orders to be tested. The test will be performed separately for each order. The default is order one.
nomiss0 specifies that Davidson and MacKinnon's approach (1993, 358), which replaces the missing values in the initial observations on the lagged residuals in the auxiliary regression with zeros, not be used.
robust specifies that the Huber/White/sandwich robust estimator for the variance-covariance matrix be used in Durbin's alternative test.
small specifies that the $p$-values of the test statistics be obtained using the $F$ or $t$ distribution instead of the default $\chi^{2}$ or normal distribution. This option may not be specified with robust, which always uses an $F$ or a $t$ distribution.
force allows the test to be run after regress, vce (robust) and after newey (see [R] regress and [TS] newey). The command will not work if the vce(cluster clustvar) option is specified with regress.

## estat dwatson

## Description for estat dwatson

estat dwatson computes the Durbin-Watson $d$ statistic (Durbin and Watson 1950) to test for first-order serial correlation in the disturbance when all the regressors are strictly exogenous.

## Menu for estat

Statistics $>$ Postestimation

## Syntax for estat dwatson

estat dwatson

collect is allowed; see [U] 11.1.10 Prefix commands.

## Remarks and examples

The Durbin-Watson test is used to determine whether the error term in a linear regression model follows an $\operatorname{AR}(1)$ process. For the linear model

$$
\begin{gathered}
y_{t}=\mathbf{x}_{t} \boldsymbol{\beta}+u_{t} \\
u_{t}=\rho u_{t-1}+\epsilon_{t}
\end{gathered}
$$

the $\mathrm{AR}(1)$ process can be written as

In general, an $\operatorname{AR}(1)$ process requires only that $\epsilon_{t}$ be independent and identically distributed (i.i.d.). The Durbin-Watson test, however, requires $\epsilon_{t}$ to be distributed $N\left(0, \sigma^{2}\right)$ for the statistic to have an exact distribution. Also, the Durbin-Watson test can be applied only when the regressors are strictly exogenous. A regressor $x$ is strictly exogenous if $\operatorname{Corr}\left(x_{s}, u_{t}\right)=0$ for all $s$ and $t$, which precludes the use of the Durbin-Watson statistic with models where lagged values of the dependent variable are included as regressors.

The null hypothesis of the test is that there is no first-order autocorrelation. The Durbin-Watson $d$ statistic can take on values between 0 and 4 and under the null $d$ is equal to 2 . Values of $d$ less than 2 suggest positive autocorrelation $(\rho>0)$, whereas values of $d$ greater than 2 suggest negative autocorrelation $(\rho<0)$. Calculating the exact distribution of the $d$ statistic is difficult, but empirical upper and lower bounds have been established based on the sample size and the number of regressors. Extended tables for the $d$ statistic have been published by Savin and White (1977). For example, suppose you have a model with 30 observations and three regressors (including the constant term). For a test of the null hypothesis of no autocorrelation versus the alternative of positive autocorrelation, the lower bound of the $d$ statistic is 1.284 , and the upper bound is 1.567 at the $5 \%$ significance level. You would reject the null if $d<1.284$, and you would fail to reject if $d>1.567$. A value falling within the range $(1.284,1.567)$ leads to no conclusion about whether or not to reject the null hypothesis.

When lagged dependent variables are included among the regressors, the past values of the error term are correlated with those lagged variables at time $t$, implying that they are not strictly exogenous regressors. The inclusion of covariates that are not strictly exogenous causes the $d$ statistic to be biased toward the acceptance of the null hypothesis. Durbin (1970) suggested an alternative test for models with lagged dependent variables and extended that test to the more general $\operatorname{AR}(p)$ serial correlation process

$$
u_{t}=\rho_{1} u_{t-1}+\cdots+\rho_{p} u_{t-p}+\epsilon_{t}
$$

where $\epsilon_{t}$ is i.i.d. with variance $\sigma^{2}$ but is not assumed or required to be normal for the test.

The null hypothesis of Durbin's alternative test is

$$
H_{0}: \rho_{1}=0, \ldots, \rho_{p}=0
$$

and the alternative is that at least one of the $\rho$ 's is nonzero. Although the null hypothesis was originally derived for an $\operatorname{AR}(p)$ process, this test turns out to have power against $\operatorname{MA}(p)$ processes as well. Hence, the actual null of this test is that there is no serial correlation up to order $p$ because the MA $(p)$ and the $\operatorname{AR}(p)$ models are locally equivalent alternatives under the null. See Godfrey (1988, 113-115) for a discussion of this result.

Durbin's alternative test is in fact a LM test, but it is most easily computed with a Wald test on the coefficients of the lagged residuals in an auxiliary OLS regression of the residuals on their lags and all the covariates in the original regression. Consider the linear regression model

$$
\begin{equation*}
y_{t}=\beta_{1} x_{1 t}+\cdots+\beta_{k} x_{k t}+u_{t} \tag{1}
\end{equation*}
$$

in which the covariates $x_{1}$ through $x_{k}$ are not assumed to be strictly exogenous and $u_{t}$ is assumed to be i.i.d. and to have finite variance. The process is also assumed to be stationary. (See Wooldridge [2020, sec. 11.1] for a discussion of stationarity.) Estimating the parameters in (1) by OLS obtains the residuals $\widehat{u}_{t}$. Next another OLS regression is performed of $\widehat{u}_{t}$ on $\widehat{u}_{t-1}, \ldots, \widehat{u}_{t-p}$ and the other regressors,

$$
\begin{equation*}
\widehat{u}_{t}=\gamma_{1} \widehat{u}_{t-1}+\cdots+\gamma_{p} \widehat{u}_{t-p}+\beta_{1} x_{1 t}+\cdots+\beta_{k} x_{k t}+\epsilon_{t} \tag{2}
\end{equation*}
$$

where $\epsilon_{t}$ stands for the random-error term in this auxiliary OLS regression. Durbin's alternative test is then obtained by performing a Wald test that $\gamma_{1}, \ldots, \gamma_{p}$ are jointly zero. The test can be made robust to an unknown form of heteroskedasticity by using a robust VCE estimator when estimating the regression in (2). When there are only strictly exogenous regressors and $p=1$, this test is asymptotically equivalent to the Durbin-Watson test.

The Breusch-Godfrey test is also an LM test of the null hypothesis of no autocorrelation versus the alternative that $u_{t}$ follows an $\operatorname{AR}(p)$ or $\operatorname{MA}(p)$ process. Like Durbin's alternative test, it is based on the auxiliary regression (2), and it is computed as $N R^{2}$, where $N$ is the number of observations and $R^{2}$ is the simple $R^{2}$ from the regression. This test and Durbin's alternative test are asymptotically equivalent. The test statistic $N R^{2}$ has an asymptotic $\chi^{2}$ distribution with $p$ degrees of freedom. It is valid with or without the strict exogeneity assumption but is not robust to conditional heteroskedasticity, even if a robust VCE is used when fitting (2).

In fitting (2), the values of the lagged residuals will be missing in the initial periods. As noted by Davidson and MacKinnon (1993), the residuals will not be orthogonal to the other covariates in the model in this restricted sample, which implies that the $R^{2}$ from the auxiliary regression will not be zero when the lagged residuals are left out. Hence, Breusch and Godfrey's $N R^{2}$ version of the test may overreject in small samples. To correct this problem, Davidson and MacKinnon (1993) recommend setting the missing values of the lagged residuals to zero and running the auxiliary regression in (2) over the full sample used in (1). This small-sample correction has become conventional for both the Breusch-Godfrey and Durbin's alternative test, and it is the default for both commands. Specifying the nomiss0 option overrides this default behavior and treats the initial missing values generated by regressing on the lagged residuals as missing. Hence, nomiss0 causes these initial observations to be dropped from the sample of the auxiliary regression.

Durbin's alternative test and the Breusch-Godfrey test were originally derived for the case covered by regress without the vce(robust) option. However, after regress, vce(robust) and newey, Durbin's alternative test is still valid and can be invoked if the robust and force options are specified.

## > Example 1: tests for serial correlation

Using data from Klein (1950), we first fit an OLS regression of consumption on the government wage bill:


If we assume that wagegov is a strictly exogenous variable, we can use the Durbin-Watson test to check for first-order serial correlation in the errors.

```
. estat dwatson
Durbin-Watson d-statistic( 2, 22) = . 3217998
```

The Durbin-Watson $d$ statistic, 0.32 , is far from the center of its distribution $(d=2.0)$. Given 22 observations and two regressors (including the constant term) in the model, the lower $5 \%$ bound is about 0.997 , much greater than the computed $d$ statistic. Assuming that wagegov is strictly exogenous, we can reject the null of no first-order serial correlation. Rejecting the null hypothesis does not necessarily mean an AR process; other forms of misspecification may also lead to a significant test statistic. If we are willing to assume that the errors follow an $\operatorname{AR}(1)$ process and that wagegov is strictly exogenous, we could refit the model using arima or prais and model the error process explicitly; see [TS] arima and [TS] prais.

If we are not willing to assume that wagegov is strictly exogenous, we could instead use Durbin's alternative test or the Breusch-Godfrey to test for first-order serial correlation. Because we have only 22 observations, we will use the small option.

| - estat durbinalt, small |
| :--- |
| Durbin's alternative test for autocorrelation |
| lags $(p)$ |

HO: no serial correlation
. estat bgodfrey, small
Breusch-Godfrey LM test for autocorrelation

| $\operatorname{lags}(p)$ | F | df | Prob $>\mathrm{F}$ |
| :---: | :---: | :---: | :---: |
| 1 | 14.264 | $(1, \quad 19)$ | 0.0013 |

HO: no serial correlation
Both tests strongly reject the null of no first-order serial correlation, so we decide to refit the model with two lags of consump included as regressors and then rerun estat durbinalt and estat bgodfrey. Because the revised model includes lagged values of the dependent variable, the Durbin-Watson test is not applicable.

| Source | SS | df | MS |  | Number of obs | = | 20 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Model | 702.660311 | 3 | 234.220104 |  | Prob > F |  | 0.0000 |
| Residual | 85.1596011 | 16 | 5.32247507 |  | R -squared | = | 0.8919 |
|  |  |  |  |  | Adj R-squared |  | 0.8716 |
| Total | 787.819912 | 19 | 41.4642059 |  | Root MSE | = | 2.307 |
| consump | Coefficient | Std. err. | t | $P>\|t\|$ | tl [95\% con |  | interval] |
| wagegovt | . 6904282 | . 3295485 | 2.10 | 0.052 | $52-.0081835$ |  | 1.38904 |
| consump |  |  |  |  |  |  |  |
| L1. | 1.420536 | . 197024 | 7.21 | 0.000 | 0001.002864 |  | 1.838208 |
| L2. | -. 650888 | . 1933351 | -3.37 | 0.004 | $04-1.06074$ |  | -. 241036 |
| _cons | 9.209073 | 5.006701 | 1.84 | 0.084 | $84-1.404659$ |  | 19.82281 |

. estat durbinalt, small lags(1/2)
Durbin's alternative test for autocorrelation

| $\operatorname{lags}(p)$ | F | df | Prob $>\mathrm{F}$ |
| :---: | :---: | :---: | :---: |
| 1 | 0.080 | $\left(\begin{array}{cc}1, & 15\end{array}\right)$ | 0.7805 |
| 2 | 0.260 | $(2, \quad 14)$ | 0.7750 |

HO: no serial correlation
. estat bgodfrey, small lags(1/2)
Breusch-Godfrey LM test for autocorrelation

| lags $(p)$ | F | df | Prob $>\mathrm{F}$ |
| :---: | :---: | :---: | :---: |
| 1 | 0.107 | $\left(\begin{array}{cc}1, & 15 \\ 2 & 0.358\end{array}\right.$ | (14) |

HO: no serial correlation
Although wagegov and the constant term are no longer statistically different from zero at the 5\% level, the output from estat durbinalt and estat bgodfrey indicates that including the two lags of consump has removed any serial correlation from the errors.

Engle (1982) suggests an LM test for checking for autoregressive conditional heteroskedasticity (ARCH) in the errors. The $p$ th-order ARCH model can be written as

$$
\begin{aligned}
\sigma_{t}^{2} & =E\left(u_{t}^{2} \mid u_{t-1}, \ldots, u_{t-p}\right) \\
& =\gamma_{0}+\gamma_{1} u_{t-1}^{2}+\cdots+\gamma_{p} u_{t-p}^{2}
\end{aligned}
$$

To test the null hypothesis of no autoregressive conditional heteroskedasticity (that is, $\gamma_{1}=\cdots=$ $\gamma_{p}=0$ ), we first fit the OLS model (1), obtain the residuals $\widehat{u}_{t}$, and run another OLS regression on the lagged residuals:

$$
\begin{equation*}
\widehat{u}_{t}^{2}=\gamma_{0}+\gamma_{1} \widehat{u}_{t-1}^{2}+\cdots+\gamma_{p} \widehat{u}_{t-p}^{2}+\epsilon \tag{3}
\end{equation*}
$$

The test statistic is $N R^{2}$, where $N$ is the number of observations in the sample and $R^{2}$ is the $R^{2}$ from the regression in (3). Under the null hypothesis, the test statistic follows a $\chi_{p}^{2}$ distribution.

## Example 2: estat archlm

We refit the original model that does not include the two lags of consump and then use estat archlm to see if there is any evidence that the errors are autoregressive conditional heteroskedastic.

| Source | SS | df | MS |  | Number of obs |  | 22 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Model | 532.567711 | 1 | 532.567711 |  | $\mathrm{F}(1,20)$ Prob > F |  | 17.72 0.0004 |
| Residual | 601.207167 | 20 | 30.0603584 |  | R-squared |  | 0.4697 |
|  |  |  |  |  | Adj R-squared |  | 0.4432 |
| Total | 1133.77488 | 21 | 53.9892799 |  | Root MSE |  | 5.4827 |
| consump | Coefficient | Std. err. | t | $\mathrm{P}>\mid \mathrm{t}$ | t\| [95\% conf. interval] |  |  |
| wagegovt | 2.50744 | . 5957173 | 4.21 | 0.000 | 01.264796 |  | 3.750085 |
| _cons | 40.84699 | 3.192183 | 12.80 | 0.000 | - 34.18821 |  | 47.50577 |

```
. estat archlm, lags(1 2 3)
LM test for autoregressive conditional heteroskedasticity (ARCH)
```

| lags $(p)$ | chi2 | df | Prob >chi2 |
| :---: | :---: | :---: | :---: |
| 1 | 5.543 | 1 | 0.0186 |
| 2 | 9.431 | 2 | 0.0090 |
| 3 | 9.039 | 3 | 0.0288 |

H0: no ARCH effects vs. H1: $\operatorname{ARCH}(p)$ disturbance
estat archlm shows the results for tests of ARCH(1), $\operatorname{ARCH}(2)$, and $\operatorname{ARCH}(3)$ effects, respectively. At the $5 \%$ significance level, all three tests reject the null hypothesis that the errors are not autoregressive conditional heteroskedastic. See [TS] arch for information on fitting ARCH models.

## Stored results

estat archlm stores the following in $r()$ :
Scalars

$$
\begin{array}{ll}
r(N) & \text { number of observations } \\
r(\mathrm{k}) & \text { number of regressors } \\
r\left(N_{\text {_gaps }}\right) & \text { number of gaps }
\end{array}
$$

Macros
r(lags) lag order
Matrices
$r(a r c h) \quad$ test statistic for each lag order
$r(d f) \quad$ degrees of freedom
$r(p) \quad$ two-sided $p$-values
estat bgodfrey stores the following in $r()$ :
Scalars

$$
\begin{array}{ll}
r(N) & \text { number of observations } \\
r(k) & \text { number of regressors } \\
r\left(N_{-} \text {gaps }\right) & \text { number of gaps }
\end{array}
$$

Macros
r(lags) lag order
Matrices

| $r($ chi2 $)$ | $\chi^{2}$ statistic for each lag order |
| :--- | :--- |
| $r(F)$ | $F$ statistic for each lag order (small only) |
| $r\left(d f \_r\right)$ | residual degrees of freedom (small only) |
| $r(f)$ | degrees of freedom |
| $r(p)$ | two-sided $p$-values |

estat durbinalt stores the following in r() :
Scalars
$r(N) \quad$ number of observations
$r(k) \quad$ number of regressors
$r$ (N_gaps) number of gaps
Macros
r(lags) lag order
Matrices $r$ (chi2) $\quad \chi^{2}$ statistic for each lag order
$r(F) \quad F$ statistic for each lag order (small only)
$r\left(d f \_r\right) \quad$ residual degrees of freedom (small only)
$r(d f) \quad$ degrees of freedom
$r(p) \quad$ two-sided $p$-values
estat dwatson stores the following in r() :
Scalars

| $r(N)$ | number of observations |
| :--- | :--- |
| $r(k)$ | number of regressors |
| $r\left(N \_g a p s\right)$ | number of gaps |
| $r(d w)$ | Durbin-Watson statistic |

## Methods and formulas

Consider the regression

$$
\begin{equation*}
y_{t}=\beta_{1} x_{1 t}+\cdots+\beta_{k} x_{k t}+u_{t} \tag{4}
\end{equation*}
$$

in which some of the covariates are not strictly exogenous. In particular, some of the $x_{i t}$ may be lags of the dependent variable. We are interested in whether the $u_{t}$ are serially correlated.

The Durbin-Watson $d$ statistic reported by estat dwatson is

$$
d=\frac{\sum_{t=1}^{n-1}\left(\widehat{u}_{t+1}-\widehat{u}_{t}\right)^{2}}{\sum_{t=1}^{n} \widehat{u}_{t}^{2}}
$$

where $\widehat{u}_{t}$ represents the residual of the $t$ th observation.
To compute Durbin's alternative test and the Breusch-Godfrey test against the null hypothesis that there is no $p$ th order serial correlation, we fit the regression in (4), compute the residuals, and then fit the following auxiliary regression of the residuals $\widehat{u}_{t}$ on $p$ lags of $\widehat{u}_{t}$ and on all the covariates in the original regression in (4):

$$
\begin{equation*}
\widehat{u}_{t}=\gamma_{1} \widehat{u}_{t-1}+\cdots+\gamma_{p} \widehat{u}_{t-p}+\beta_{1} x_{1 t}+\cdots+\beta_{k} x_{k t}+\epsilon \tag{5}
\end{equation*}
$$

Durbin's alternative test is computed by performing a Wald test to determine whether the coefficients of $\widehat{u}_{t-1}, \ldots, \widehat{u}_{t-p}$ are jointly different from zero. By default, the statistic is assumed to be distributed $\chi^{2}(\mathrm{p})$. When small is specified, the statistic is assumed to follow an $F(p, N-p-k)$ distribution. The reported $p$-value is a two-sided $p$-value. When robust is specified, the Wald test is performed using the Huber/White/sandwich estimator of the variance-covariance matrix, and the test is robust to an unspecified form of heteroskedasticity.

The Breusch-Godfrey test is computed as $N R^{2}$, where $N$ is the number of observations in the auxiliary regression (5) and $R^{2}$ is the $R^{2}$ from the same regression (5). Like Durbin's alternative test, the Breusch-Godfrey test is asymptotically distributed $\chi^{2}(p)$, but specifying small causes the $p$-value to be computed using an $F(p, N-p-k)$.

By default, the initial missing values of the lagged residuals are replaced with zeros, and the auxiliary regression is run over the full sample used in the original regression of (4). Specifying the nomiss0 option causes these missing values to be treated as missing values, and the observations are dropped from the sample.

Engle's LM test for $\operatorname{ARCH}(p)$ effects fits an OLS regression of $\widehat{u}_{t}^{2}$ on $\widehat{u}_{t-1}^{2}, \ldots, \widehat{u}_{t-p}^{2}$ :

$$
\widehat{u}_{t}^{2}=\gamma_{0}+\gamma_{1} \widehat{u}_{t-1}^{2}+\cdots+\gamma_{p} \widehat{u}_{t-p}^{2}+\epsilon
$$

The test statistic is $n R^{2}$ and is asymptotically distributed $\chi^{2}(p)$.

## Acknowledgment

The original versions of estat archlm, estat bgodfrey, and estat durbinalt were written by Christopher F. Baum of the Department of Economics at Boston College and author of the Stata Press books An Introduction to Modern Econometrics Using Stata and An Introduction to Stata Programming and coauthor of the Stata Press book Environmental Econometrics Using Stata.

## References

Baum, C. F. 2006. An Introduction to Modern Econometrics Using Stata. College Station, TX: Stata Press.
Beran, R. J., and N. I. Fisher. 1998. A conversation with Geoff Watson. Statistical Science 13: 75-93. https://doi.org/10.1214/ss/1028905975.
Breusch, T. S. 1978. Testing for autocorrelation in dynamic linear models. Australian Economic Papers 17: 334-355. https://doi.org/10.1111/j.1467-8454.1978.tb00635.x.

Davidson, R., and J. G. MacKinnon. 1993. Estimation and Inference in Econometrics. New York: Oxford University Press.
Durbin, J. 1970. Testing for serial correlation in least-squares regressions when some of the regressors are lagged dependent variables. Econometrica 38: 410-421. https://doi.org/10.2307/1909547.

Durbin, J., and S. J. Koopman. 2012. Time Series Analysis by State Space Methods. 2nd ed. Oxford: Oxford University Press.
Durbin, J., and G. S. Watson. 1950. Testing for serial correlation in least squares regression. I. Biometrika 37: 409-428. https://doi.org/10.2307/2332391.
_-. 1951. Testing for serial correlation in least squares regression. II. Biometrika 38: 159-177. https://doi.org/10.2307/2332325.
Engle, R. F. 1982. Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation. Econometrica 50: 987-1007. https://doi.org/10.2307/1912773.

Fisher, N. I., and P. Hall. 1998. Geoffrey Stuart Watson: Tributes and obituary (3 December 1921-3 January 1998). Australian and New Zealand Journal of Statistics 40: 257-267. https://doi.org/10.1111/1467-842X.00030.
Godfrey, L. G. 1978. Testing against general autoregressive and moving average error models when the regressors include lagged dependent variables. Econometrica 46: 1293-1301. https://doi.org/10.2307/1913829.
——. 1988. Misspecification Tests in Econometrics: The Lagrange Multiplier Principle and Other Approaches. Econometric Society Monographs, No. 16. Cambridge: Cambridge University Press.
Klein, L. R. 1950. Economic Fluctuations in the United States 1921-1941. New York: Wiley.
Koopman, S. J. 2012. James Durbin, FBA, 1923-2012. Journal of the Royal Statistical Society, Series A 175: 1060-1064. https://doi.org/10.1111/j.1467-985X.2012.01068.x.

Phillips, P. C. B. 1988. The ET Interview: Professor James Durbin. Econometric Theory 4: 125-157. https://doi.org/10.1017/S0266466600011907.

Savin, N. E., and K. J. White. 1977. The Durbin-Watson test for serial correlation with extreme sample sizes or many regressors. Econometrica 45: 1989-1996. https://doi.org/10.2307/1914122.

Wooldridge, J. M. 2020. Introductory Econometrics: A Modern Approach. 7th ed. Boston: Cengage.

James Durbin (1923-2012) was a British statistician who was born in Wigan, near Manchester. He studied mathematics at Cambridge and after military service and various research posts joined the London School of Economics in 1950. Later in life, he was also affiliated with University College London. His many contributions to statistics centered on serial correlation, time series (including major contributions to structural or unobserved components models), sample survey methodology, goodness-of-fit tests, and sample distribution functions, with emphasis on applications in the social sciences. He served terms as president of the Royal Statistical Society and the International Statistical Institute.

Geoffrey Stuart Watson (1921-1998) was born in Victoria, Australia, and earned degrees at Melbourne University and North Carolina State University. After a visit to the University of Cambridge, he returned to Australia, working at Melbourne and then the Australian National University. Following periods at Toronto and Johns Hopkins, he settled at Princeton. Throughout his wide-ranging career, he made many notable accomplishments and important contributions, including the Durbin-Watson test for serial correlation, the Nadaraya-Watson estimator in nonparametric regression, and methods for analyzing directional data.

Leslie G. Godfrey (1946- ) was born in London and earned degrees at the Universities of Exeter and London. He is now a professor of econometrics at the University of York. His interests center on implementation and interpretation of tests of econometric models, including nonnested models.

Trevor Stanley Breusch (1953- ) was born in Queensland and earned degrees at the University of Queensland and Australian National University (ANU). After a post at the University of Southampton, he returned to work at ANU. His background is in econometric methods and his recent interests include political values and social attitudes, earnings and income, and measurement of underground economic activity.

[^55]
## Title

reri - Relative excess risk due to interaction

Description
Options
Acknowledgment

Quick start
Remarks and examples
References

Menu
Stored results Also see

Syntax
Methods and formulas

## Description

reri reports three statistics that assess two-way interactions in an additive model of relative risk. The statistics are of interest for determining how two risk factors interact to produce a positive result for the outcome of interest. The estimates of the interactions themselves are reported as excess relative risks (ERRS). Statistics reported for the interaction are the relative excess risk due to interaction (RERI), attributable proportion (AP), and synergy index (SI); reri supports binomial generalized linear, Poisson, negative binomial, logistic, Cox, parametric survival, and interval-censored parametric and semiparametric survival models.

## Quick start

Report ERRs, RERI, AP, and SI for the interaction of exposures exp1 and $\exp 2$ from a binomial generalized linear model for $y$ fit by binreg
reri binreg y exp1 exp2
Same as above, but use poisson to fit the model
reri poisson y exp1 exp2
Same as above, but report incidence-rate ratios for the interactions rather than ERRS
reri poisson y exp1 exp2, irr
Same as above, but include the covariate x , and specify noisily to view the full Poisson model reri poisson y exp1 exp2 x, irr noisily

Same as above, but report ERRs for the interactions as well as incidence-rate ratios for the full model reri poisson y exp1 exp2 x, irr err noisily

Fit a Cox model using stset data stratified by svar, and report ERRS, RERI, AP, and SI for the interactions of exp1 and exp2

```
reri stcox exp1 exp2, strata(svar)
```

Same as above, but fit a Weibull model
reri streg exp1 exp2, strata(svar) distribution(weibull)
Fit a Weibull model with interval-censored survival data, with variables t 1 and t 2 giving lower and upper endpoints for the censoring interval, and report ERRS, RERI, AP, and SI for the interactions reri stintreg exp1 exp2, distribution(weibull) interval(t1 t2)

## Menu

Statistics $>$ Epidemiology and related $>$ Other $>$ Relative excess risk due to interaction (RERI)

## Syntax

reri cmd depvar varl var2 [control_vars] [if] [in] [weight]
[, options cmd_options]
cmd defines the estimation command to be executed and may be one of the following:
binreg, logistic, nbreg, poisson, stcox, stintcox, stintreg, and streg
depvar is required for all except stcox, stintcox, stintreg, and streg, which do not allow depvar.
varl and var2 are binary variables (integer valued and nonnegative) whose interactions are to be tested for additivity.
options
Description

## Model

baselevel1(\#) specify base level for varl
baselevel2(\#) specify base level for var2
Reporting
noisily
err
coef_type
level(\#)
nofvlabel
nolegend
display output from cmd
report interactions as ERR, the default
report interactions as coef_type rather than ERR
set confidence level; default is level(95)
display numeric values rather than value labels
do not display the legend for the interactions
var1 and var2 cannot be factor variables, but control_vars may contain factor variables; see [U] 11.4.3 Factor variables. cmd_options are any options allowed by cmd, except options or, hr, rd, and vce(jackknife1) are not allowed when $c m d$ is binreg.
When cmd is poisson, vce(robust) is used by default when depvar is 0/1.
bootstrap, by, collect, jackknife, and statsby are allowed; see [U] 11.1.10 Prefix commands.
Weights are allowed if cmd allows them; see [U] 11.1.6 weight.
All postestimation commands behave as they would after cmd ; see the postestimation manual entry for cmd .

## Options

## Model

baselevel1(\#) and baselevel2(\#) specify base levels for varl and var2, respectively, the two variables that give the interactions of interest.

If varl and var 2 are $0 / 1$ variables, then by default, the RERI statistic is a test of whether the ERR of interaction 01 ( 01 meaning varl $=0$ and var $2=1$ ) plus the ERR of interaction 10 equals the ERR of interaction 11. In shorthand notation, we say we are testing $01+10=11$ on the ERR scale.
Specifying baselevel1(\#) or baselevel2(\#) allows the base levels of the interactions to be changed. For example, with $0 / 1$ variables, baselevel1(1) sets the base level for varl to 1 , and RERI tests whether $11+00=01$.

If baselevel1(\#) is not specified, the base level of varl is the default base level of the factor variable i.varl. The default base level is the smallest of the two values of varl, unless the
default has been changed by fvset. This is similarly true for baselevel2(\#) and var2. See [U] 11.4.3.2 Base levels and [R] fvset. Note that base levels cannot be set using factor-variable notation when specifying varl and var2.

## Reporting

noisily displays all output from cmd .
err specifies that the coefficients of the interactions be displayed as ERRs. This is the default. ERR is equal to relative risk minus one. See Remarks and examples below.
coef_type specifies an alternative scale for the coefficients of the interactions instead of ERRs. Allowed coef_type depends on cmd and can be rr, or, irr, or hr , whichever is allowed by cmd. In addition, coef_type can be coef, which displays the interaction terms as unscaled coefficients.
For binreg, coef_type rr displays the interactions as relative-risk ratios. The other models fit by binreg using options or, hr, and rd are not available with reri. For logistic, coef_type or displays the interactions as odds ratios. For poisson and nbreg, coef_type irr displays the interactions as incidence-rate ratios. For stcox, stintcox, stintreg, and streg, coef_type hr displays the interactions as hazard ratios. For the survival-time commands, nohr may be used as a synonym for coef.
When noisily is specified, err and a coef_type may be specified simultaneously. In this case, the table of interactions displayed by reri reports ERRs, and the coefficient table produced by cmd displays coefficients on the coef_type scale.
level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [R] level.
nofvlabel specifies that numeric values rather than value labels be displayed in the legend and $c m d$ output.
nolegend suppresses the display of the legend for the interactions.

## Remarks and examples

Remarks are presented under the following headings:

## Introduction

Additive versus multiplicative interactions
Incidence-rate ratios, hazard ratios, and odds ratios

## Introduction

reri reports three statistics that assess additive interactions of relative risk: RERI, AP, and SI (Lash et al. 2021). These measures are typically of interest when working with epidemiologic data. When one assesses the effect of multiple exposures on an outcome, determining whether one exposure modifies the effect of another can be an important piece of the puzzle.

When there are two exposures that affect an outcome, we often want to model how the exposures interact. That is, for binary exposures $A$ and $B$, we want to model the risk of a positive outcome for subjects having both $A$ and $B$ compared with subjects having only $A$ and those having only $B$.

Typically, a multiplicative model for the interaction is fit to the data: the risk for those with $A$ and $B$ is hypothesized to equal the risk for only $A$ times the risk for only $B$. If the risk for $A$ and $B$ equals the multiplicative risk, then we say there is no interaction of $A$ and $B$. If the risk is greater than the multiplicative risk, we say there is a positive interaction of $A$ and $B$. Multiplicative models
are computationally convenient because they are simple to specify, and it is easy to test whether the interaction term is greater (or less) than multiplicative.

Inherent to this model is the hypothesis that the risk is multiplicative. For many biological processes, however, an additive model of risk may model the process better than a multiplicative one (see, for example, Andersson et al. [2005]). That is, the risk for subjects with $A$ and $B$ is hypothesized to equal the risk for only $A$ plus the risk for only $B$. For this hypothesis, we want to look at how much the risk for $A$ and $B$ is greater (or less) than the additive risk.

The RERI statistic is this risk difference formulated using relative risks. A RERI statistic that is 0 means the risk is additive. One that is positive means the risk is superadditive, and one that is negative means the risk is subadditive. The test of statistical significance of the RERI statistic is straightforward, but it is not as simple as the test for the multiplicative model (see Methods and formulas).

Besides the RERI statistic, there are two other related statistics for additive models: AP and SI. The AP is the proportion of risk of $A$ and $B$ due to the superadditivity of exposures $A$ and $B$. The SI recasts the RERI statistic as a ratio, rather than a difference.

Let's formally define these statistics. Let $p_{A+B+}$ be the probability of having a positive outcome for subjects positive for binary exposure $A$ and positive for binary exposure $B$. Define $p_{A-B+}$, $p_{A+B-}$, and $p_{A-B-}$ similarly. An additive model for the probabilities is

$$
p_{A+B+}=p_{A-B-}+\left(p_{A-B+}-p_{A-B-}\right)+\left(p_{A+B-}-p_{A-B-}\right)
$$

Said in words, the probability of a positive outcome for $A+B+$ equals the probability for the base category $A-B$ - plus the amounts that probabilities for $A-B+$ and $A+B-$ exceed the probability for the base category.

If we divide this equation by $p_{A-B-}$ and note that $p_{A+B+} / p_{A-B-}=\mathrm{RR}_{A+B+}$, the relative risk for $A+B+$ (and similarly for $\mathrm{RR}_{A-B+}$ and $\mathrm{RR}_{A+B-}$ ), we get

$$
\mathrm{RR}_{A+B+}=\mathrm{RR}_{A-B+}+\mathrm{RR}_{A+B-}-1
$$

$E R R$ is defined as $E R R=R R-1$. In terms of ERR, the additive model becomes

$$
\operatorname{ERR}_{A+B+}=\mathrm{ERR}_{A-B+}+\mathrm{ERR}_{A+B-}
$$

The RERI statistic is defined as

$$
\mathrm{RERI}=\mathrm{ERR}_{A+B+}-\mathrm{ERR}_{A-B+}-\mathrm{ERR}_{A+B-}
$$

Hence, RERI is 0 for a perfectly additive model. If it is negative, the effect of the $A+B+$ interaction is less than additive (subadditive). If positive, the effect is more than additive (superadditive).

AP is simply RERI scaled by the relative risk (not ERR) of $A+B+$. That is,

$$
\mathrm{AP}=\frac{\mathrm{RERI}}{\mathrm{RR}_{A+B+}}
$$

When AP is positive, it represents the proportion of risk of $A+B+$ due to the superadditivity of exposures $A$ and $B$ (that is, the amount greater than additivity). Note that it is a proportion of risk, not relative risk, as one can see by multiplying both numerator and denominator by the reference probability $p_{A-B-}$.

The SI is a ratio measure:

$$
\mathrm{SI}=\frac{\mathrm{ERR}_{A+B+}}{\mathrm{ERR}_{A-B+}+\mathrm{ERR}_{A+B-}}
$$

The SI is 1 , rather than 0 , for a perfectly additive model. It is $<1$ for a subadditive model and $>1$ for a superadditive model. Note that because ERRs can be negative (they are always $\geq-1$ ), the SI can be negative, making interpretation difficult. The SI is intended as a measure for cases in which all ERRs of the model are positive, but this will not always be the case. ERRs will be determined by the data and the specification of the model. See Knol et al. (2011) and example 4 for discussions about negative ERRS.

## Additive versus multiplicative interactions

Typically, interactions are fit and tested by specifying main effects and interactions of the main effects (Rothman, Greenland, and Walker 1980). Suppose $x_{A+}$ is a $0 / 1$ variable that is 1 when exposure $A$ is positive and 0 when it is negative. The variable $x_{B+}$ is defined similarly for exposure $B$. Let $x_{A+B+}$ be the $0 / 1$ variable that is 1 when both $A$ and $B$ are positive and 0 otherwise.

When we specify a model and include terms

$$
b_{1} x_{A+}+b_{2} x_{B+}+b_{3} x_{A+B+}
$$

we call $b_{1} x_{A+}$ and $b_{2} x_{B+}$ main effects and $b_{3} x_{A+B+}$ their interaction. The test of the interaction is the test of $b_{3}=0$.

It is a model of interaction that is additive in these terms. However, when the statistics of interest are risk ratios, odds ratios, or hazard ratios, the model is generally called multiplicative because it is multiplicative in these statistics. The risk ratios (or odds ratios or hazard ratios) are given by the exponentiated coefficients; that is, $\mathrm{RR}_{A+}=e^{b_{1}}, \mathrm{RR}_{B+}=e^{b_{2}}$, etc. The risk ratio for $A+B+$ is $\mathrm{RR}_{A+B+}=e^{b_{1}} e^{b_{2}} e^{b_{3}}$. If the interaction is multiplicative, $e^{b_{3}}=1$ and

$$
\mathrm{RR}_{A+B+}=\mathrm{RR}_{A+} \times \mathrm{RR}_{B+}
$$

As mentioned earlier, many biological processes more closely follow an additive model of risk than a multiplicative one. The additive model of ERR is

$$
\operatorname{ERR}_{A+B+}=\operatorname{ERR}_{A-B+}+\mathrm{ERR}_{A+B-}
$$

Note that unlike the multiplicative model with main effects, the three states in the additive model, $A-B+, A+B-$, and $A+B+$, are disjoint.
reri is called with the syntax
reri cmd depvar varl var2...
reri uses varl and var2 to create the three terms for the additive model using factor-variable notation. Suppose varl is xa, a $0 / 1$ variable, indicating whether a subject has exposure $A$. Suppose var 2 is xb for exposure $B$. Then, the terms needed to test for additive ERR are, in factor-variable notation,

```
0.xa#1.xb 1.xa#0.xb 1.xa#1.xb
```

reri creates these terms and calls $c m d$ to fit the model. Then, it uses nlcom to calculate the RERI, AP, and SI statistics and their standard errors. See Methods and formulas.

## Incidence-rate ratios, hazard ratios, and odds ratios

Incidence rates and hazards are measures of risk. An incidence rate is a probability adjusted for time at risk. A hazard is a conditional probability when the distribution is discrete and an incremental conditional probability when the distribution is continuous (Kalbfleisch and Prentice 2002). Hence, incidence-rate ratios and hazard ratios are measures of relative risk. So using RERI, AP, and SI statistics with models that estimate incidence-rate ratios (poisson and nbreg) or hazard ratios (stcox, stintcox, streg, stintreg) presents no difficulties.

Odds ratios, however, are different. As is well known, the odds ratio approximates the risk ratio in the population when the prevalence of the outcome in the population is low (see, for example, Greenland and Thomas [1982]). So if your data are from a case-control study and the population prevalence of a positive outcome is low, then odds ratios from a logistic regression will approximate the risk ratios in the population, and the RERI, AP, and SI statistics are appropriate. On the other hand, if the population prevalence is high, the odds ratio is typically a poor approximation to the risk ratio (unless it is close to 1), and interpretation of the RERI, AP, and SI statistics may be problematic (see, for example, Skrondal [2003]).

If the study you are analyzing is a cohort study, observational study, or a randomized controlled trial with a binary outcome, you can estimate risk directly rather than fitting a logistic regression. A binomial model of risk or a Poisson model can be fit (using an appropriate variance estimator; see Cummings [2009] and example 2).

If time to outcome is measured, then a survival model may be fit, and there are no issues interpreting the RERI and SI statistics. Interpreting the AP statistic is a little trickier because it relates to the hazard at a point in time rather than risk in a population, so the proportion is the proportion of the hazard at a given time.

## > Example 1: Binomial model of risk

We have simulated data on birth defects that mimic the results from Brender et al. (2013), an observational study of birth defects. Three binary outcomes are included in these data and are tube, palate, and heart, representing neural tube defects, palate defects, and heart defects, respectively. Predictors of birth defects are two measures of nitrate intake: drug, with values $0 / 1$, indicating nitrosatable drug exposure; and nitrate, also with values $0 / 1$, representing either low or high daily consumption of nitrates from drinking water.

We load the dataset and describe it.


Sorted by: id
We first examine the outcome tube. We want to see whether the interactions of the two predictors drug and nitrate follow an additive model of risk. Because this is an observational study, a binomial model of risk is appropriate, and we can fit it using binreg.

The syntax of reri is reri cmd depvar varl var2, where varl and var2 are the binary predictors of interest. Note that varl and var2 are specified without using factor-variable notation. reri will create the interactions automatically. Here's the result:

| Fitting binreg ... <br> Interaction of drug and nitrate on a <br> Model: Binomial generalized linear drug\#nitrate <br> - + No\#High <br> + - Yes\#Low <br> + Yes\#High |  |  | Number of observations $=1,284$ |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | ERR | $\begin{gathered} \text { EIM } \\ \text { std. err. } \end{gathered}$ | z | $P>\|z\|$ | [95\% con | interval] |
| drug\#nitrate |  |  |  |  |  |  |
| - + | . 1161419 | . 1571024 | 0.78 | 0.435 | -. 1529507 | . 4707206 |
| + - | . 2385481 | . 2392316 | 1.11 | 0.268 | -. 1517982 | . 8085336 |
| + + | . 7769424 | . 407042 | 2.51 | 0.012 | . 1342 | 1.783922 |
| RERI | . 4222523 | . 453286 | 0.93 | 0.352 | -. 466172 | 1.310677 |
| Attr. prop. | . 2376286 | . 2187062 | 1.09 | 0.277 | -. 1910276 | . 6662848 |
| Synergy ind. | 2.190483 | 2.034328 | 0.84 | 0.398 | . 3548285 | 13.52263 |

Note: $\mathrm{P}>|z|$ for synergy index (SI) is for test $\mathrm{HO}: ~ \mathrm{SI}=1$.
Note that the interactions created by reri represent disjoint groups in the data. The -+ interaction is for drug $=$ No and nitrate $=$ High. The +- and ++ interactions are defined similarly as shown in the legend. The interaction -- is, of course, the reference category.

By default, reri displays the model coefficients for the -+ , +-, and ++ interactions as ERR, which is relative risk minus one. So ERR greater than 0 means that relative risk is greater than 1 . If the model is additive, we would have

$$
\mathrm{ERR}_{++}=\mathrm{ERR}_{-+}+\mathrm{ERR}_{+-}
$$

The RERI statistic is the difference between the two sides of this equation:

$$
\text { RERI }=\text { ERR }_{++}-\text {ERR }_{-+}-\text {ERR }_{+-}
$$

The estimates are ERR ${ }_{-+}=0.116$ and $E R R_{+-}=0.239$, so ERR ${ }_{-+}+\mathrm{ERR}_{+-}=0.355$. The estimate of $E R R_{++}$is 0.777 , which is greater than $E R R_{-+}+E R R_{+-}$. So the model is superadditive. The RERI statistic is 0.422 , the difference between 0.777 and 0.355 .

The $p$-value for RERI is a test of the null hypothesis RERI $=0$. That is, a small $p$-value provides evidence to reject the additive model for the interactions. In this case, $p=0.352$, so we do not have enough evidence to reject the additive model, despite the observed superadditivity.

AP is

$$
\mathrm{AP}=\frac{\mathrm{RERI}}{\mathrm{RR}_{++}}
$$

It is the proportion of risk for the ++ interaction that is due to the risk that is above additive. In this case, AP is 0.238 or $23.8 \%$. Its $p$-value is a test of the null hypothesis AP $=0$. This null hypothesis is equivalent to the null hypothesis RERI $=0$, so $p$-values will be similar.

The SI is

$$
\mathrm{SI}=\frac{\mathrm{ERR}_{++}}{\mathrm{ERR}_{-+}+\mathrm{ERR}_{+-}}
$$

It is how many times larger (or smaller) is $E R R_{++}=0.777$ than $E R R_{-+}+\mathrm{ERR}_{+-}=0.355$. In this example, it is about twice as large, or more precisely, 2.19 times as large. The null hypothesis is SI $=1$, rather than 0 . The SI is a rather odd statistic because it is a test of an additive model using a multiplicative scale!
reri can optionally display the interaction coefficients from binreg as relative risk (that is, risk ratios) rather than ERR. This is done by specifying the rr option. We do so and also specify the noisily option to see the output from binreg.

| Fitting binreg: |  |  |  |
| :---: | :---: | :---: | :---: |
| Iteration 1: Deviance $=1176.736$ |  |  |  |
| Iteration 2: Deviance $=1176.736$ |  |  |  |
| Generalized linear models | Number of obs | = | 1,284 |
| Optimization : MQL Fisher scoring | Residual df | = | 1,280 |
| (IRLS EIM) | Scale parameter | = | 1 |
| Deviance $=1176.735717$ | (1/df) Deviance | $=$ | . 9193248 |
| Pearson $\quad=1283.999016$ | (1/df) Pearson | $=$ | 1.003124 |
| Variance function: $\mathrm{V}(\mathrm{u})=\mathrm{u} *(1-\mathrm{u} / 1)$ | [Binomial] |  |  |
| Link function : $\mathrm{g}(\mathrm{u})=\ln (\mathrm{u})$ | [Log] |  |  |

BIC $=-7985.166$

|  | EIM |  |  |  |  |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| tube | Risk ratio | std. err. | $z$ | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| drug\#nitrate |  |  |  |  |  |  |
| No\#High | 1.116142 | .1571024 | 0.78 | 0.435 | .8470493 | 1.470721 |
| Yes\#Low | 1.238548 | .2392316 | 1.11 | 0.268 | .8482018 | 1.808534 |
| Yes\#High | 1.776942 | .407042 | 2.51 | 0.012 | 1.1342 | 2.783922 |
| _cons | .157969 | .013697 | -21.28 | 0.000 | .1332805 | .1872307 |

[^56]Interaction of drug and nitrate on an additive scale
drug\#nitrate

| -+ | No\#High |
| :--- | :--- |
| +- | Yes\#Low |
| ++ | Yes\#High |


|  | EIM |  |  |  |  |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
|  | Risk ratio | std. err. | $\mathbf{z}$ | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| drug\#nitrate |  |  |  |  |  |  |
| -+ | 1.116142 | .1571024 | 0.78 | 0.435 | .8470493 | 1.470721 |
| +- | 1.238548 | .2392316 | 1.11 | 0.268 | .8482018 | 1.808534 |
| ++ | 1.776942 | .407042 | 2.51 | 0.012 | 1.1342 | 2.783922 |
| RERI | .4222523 | .453286 | 0.93 | 0.352 | -.466172 | 1.310677 |
| Attr. prop. | .2376286 | .2187062 | 1.09 | 0.277 | -.1910276 | .6662848 |
| Synergy ind. | 2.190483 | 2.034328 | 0.84 | 0.398 | .3548285 | 13.52263 |

Note: $\mathrm{P}>|z|$ for synergy index (SI) is for test $\mathrm{HO}: \mathrm{SI}=1$.
We see the output of binreg followed by the output of reri. In this case, reri displays the interaction coefficients as risk ratios, matching the output of binreg. Risk ratios are just ERRs plus one, so point estimates and confidence intervals for the interactions are just shifted by one from the previous results, and standard errors and $p$-values are unchanged. The RERI, AP, and SI statistics are all exactly the same as they were in the previous results.

For comparison, we can fit a binreg model specifying the interaction in the standard manner: main effects for drug and nitrate and a single interaction term for drug $=$ Yes and nitrate $=$ High.


| Number of obs | $=$ | 1,284 |
| :--- | ---: | ---: |
| Residual df | $=$ | 1,280 |
| Scale parameter | $=$ | 1 |
| $(1 / \mathrm{df})$ Deviance | $=$ | .9193248 |
| (1/df) Pearson | $=$ | 1.003124 |
| [Binomial] |  |  |
| [Log] |  |  |
| BIC | $=$ | -7985.166 |


| tube | Risk ratio | EIMEtd err. | $\mathbf{z}$ | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| drug <br> Yes <br> nitrate <br> High | 1.238548 | .2392315 | 1.11 | 0.268 | .8482019 | 1.808533 |
| drug\#nitrate <br> Yes\#High | 1.116142 | .1571023 | 0.78 | 0.435 | .8470494 | 1.47072 |
| _cons | .157969 | .013697 | -21.28 | 0.000 | .1332805 | .1872307 |

Note: _cons estimates baseline risk.

This model is not as easy to interpret as the previous ones and shows no evidence that drug, nitrate, or their interaction has any effect on neural tube defects. However, the previous model fit with disjoint interactions provided strong evidence that the relative risk of neural tube defects was greater for the ++ group when compared with the reference group; in that model, the risk ratio for the ++ group was greater than 1 , and the $p$-value was 0.012 .

Note that when running binreg directly, we must specify the rr model option. When reri calls binreg, it automatically fits an rr model. Specifying rr with reri is merely a display option; it reports relative risk rather than ERR.

## Technical note

As mentioned earlier, when reri is called with, say,

```
reri cmd y x1 x2 ...
```

it creates disjoint interactions and runs

```
cmd y 0.x1#1.x2 1.x1#0.x2 1.x1#1.x2 ...
```

(assuming that x 1 and x 2 are $0 / 1$ and that 0 is taken as the base category for both).
If reri gives an error message and you cannot figure out the reason why, run
cmd y 0.x1\#1.x2 1.x1\#0.x2 1.x1\#1.x2 ...
and see whether this gives any problems such as dropped terms or dropped observations.

## > Example 2: Poisson model

We continue with our previous example. We again fit a model for the outcome tube, representing neural tube birth defects. Our predictors, as in the previous example, are drug (nitrosatable drug exposure) and nitrate (drinking water nitrate intake). The only difference is that we specify poisson rather than binreg.


Note: $\mathrm{P}>|z|$ for synergy index (SI) is for test $\mathrm{HO}: ~ \mathrm{SI}=1$.

We see that the reported point estimates for the interactions, RERI, AP, and SI are identical to the ones estimated when we used binreg. The standard errors are almost the same, differing only in the fourth significant digit.

When the outcome is $0 / 1$, reri poisson by default reports standard errors calculated using vce(robust). This is because when the outcome is $0 / 1$, the distribution is binomial (Bernoulli), not Poisson, and vce (robust) gives standard errors that are valid in this case. The other vcetypes allowed with poisson are available and can be selected by specifying the vce() option with reri. See [R] poisson and [R] vce_option.
binreg by default fits models using iterated, reweighted least-squares optimization and uses the expected information matrix for the variance estimator. poisson uses maximum likelihood optimization. Point estimates will theoretically be the same but will have numerical differences because of the different optimizers used. Different variance estimators are used, so slight differences, as we observed here, are to be expected.

One advantage of using poisson with reri rather than binreg is that maximum likelihood optimization is more robust than iterated, reweighted least-squares optimization. There can be convergence problems with iterated, reweighted least squares when maximum likelihood will converge without any difficulty.
binreg does have an ml option for maximum likelihood optimization. We can also specify vce(robust) to reproduce the results given by poisson.

| Interaction of drug and nitrate on <br> Model: Binomial generalized linear drug\#nitrate <br> - + No\#High <br> + - Yes\#Low <br> + Yes\#High |  |  | Number of observations $=1,284$ |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | ERR | Robust std. err. | z | $P>\|z\|$ | [95\% con | interval] |
| drug\#nitrate |  |  |  |  |  |  |
| - + | . 1161419 | . 1571636 | 0.78 | 0.435 | -. 1530418 | . 4708787 |
| + - | . 2385481 | . 2393248 | 1.11 | 0.268 | -. 1519233 | . 8088004 |
| + + | . 7769424 | . 4072075 | 2.51 | 0.012 | . 133993 | 1.78443 |
| RERI | . 4222523 | . 4534688 | 0.93 | 0.352 | -. 4665302 | 1.311035 |
| Attr. prop. | . 2376286 | . 2187937 | 1.09 | 0.277 | -. 1911993 | . 6664564 |
| Synergy ind. | 2.190483 | 2.035131 | 0.84 | 0.399 | . 3545735 | 13.53235 |

Note: P>|z| for synergy index (SI) is for test HO: SI = 1.
reri binreg with ml and vce(robust) gives the same results as reri poisson. But if we want maximum likelihood with the robust variance estimator, we might as well just use reri poisson.

## > Example 3: Logistic model

We continue with the previous example. These data were simulated to mimic an observational study with full data from several sites over a set period of time. So risk ratios can be estimated directly. But let's pretend it was a case-control study. The outcomes, birth defects, have low prevalences, so odds ratios from a logistic regression should be good approximations to the risk ratios.

We specify logistic with reri:

|  |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | ERR | Std. err | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% con | interval] |
| drug\#nitrate |  |  |  |  |  |  |
| - + | . 141003 | . 1933785 | 0.78 | 0.436 | -. 1814898 | . 5905577 |
| + | . 2965734 | . 3085999 | 1.09 | 0.275 | -. 1867932 | 1.067251 |
| + + | 1.080139 | . 6495028 | 2.35 | 0.019 | . 1280107 | 2.835939 |
| RERI | . 642563 | . 691611 | 0.93 | 0.353 | -. 7129695 | 1.998096 |
| Attr. prop. | . 3089038 | . 2594074 | 1.19 | 0.234 | -. 1995252 | . 8173329 |
| Synergy ind. | 2.468459 | 2.456832 | 0.91 | 0.364 | . 3509409 | 17.36273 |

Note: P>|z| for synergy index (SI) is for test HO: SI = 1 .
The ERRs reported are the odds ratios minus one. Calculating ERR using odds ratios assumes that the odds ratios are approximations to the risk ratios.

Here we see that RERI $=0.643$, $\mathrm{AP}=0.309$, and $\mathrm{SI}=2.47$. When we used poisson and binreg, we got RERI $=0.422, \mathrm{AP}=0.238$, and $\mathrm{SI}=2.19$. Results are similar but not that close numerically.

If we have data from an observational study, a cohort, or a randomized controlled trial, we can estimate risk ratios directly, and we should do so using poisson or binreg. If it is a case-control study, we have no choice but to use logistic.

## Example 4: Negative ERRs

We present data from example 2 in [ST] stcox. The data represent 48 participants in a cancer drug trial. Some receive treatment (drug $=1$ ) and some receive a placebo (drug $=0$ ). The data contain the participants' age and time until death. It has already been stset and is ready to be analyzed using a survival model.

We load the dataset, describe it, and run st to see the survival-time settings.


Sorted by:
. st
-> stset studytime, failure(died)
Survival-time data settings
Failure event: died!=0 \& died<.
Observed time interval: ( 0, studytime]
Exit on or before: failure
The RERI statistic requires two $0 / 1$ predictors. These data have two predictors, drug and age, but age is not $0 / 1$. Let's create a $0 / 1$ variable from age. First, we summarize age:

```
. summarize age, detail
```

Patient's age at start of exp.

|  | Percentiles | Smallest |  |  |
| ---: | ---: | ---: | :--- | ---: |
| $1 \%$ | 47 | 47 |  | 48 |
| $5 \%$ | 48 | 48 |  | 48 |
| $10 \%$ | 49 | 48 | Obs | 55.875 |
| $25 \%$ | 50.5 | 49 | Sum of wgt. |  |
| $50 \%$ | 56 |  | Mean | 5.659205 |
|  |  | Largest | Std. dev. |  |
| $75 \%$ | 60 | 65 |  | 32.0266 |
| $90 \%$ | 65 | 67 | Variance | .3161066 |
| $95 \%$ | 67 | 67 | Skewness | 2.125197 |

The age of the participants ranges from 47 to 67 years. Median age is 56 . Second, we create the variable older with older $=1$ representing those with ages $>56$ and older $=0$ those with ages $\leq 56$. We give it a value label as well.
. generate older $=$ cond (age $>56,1,0)$
. label define older_lbl 0 "age<=56" 1 "age>56"
. label values older older_lbl

Now, we run reri stcox:


```
Notes: P>|z| for synergy index (SI) is for test HO: SI = 1.
    Some estimates of excess relative risk are not positive.
```

The ERR is negative for the $+-($ drug $=1$ and older $=0)$ and $++(\operatorname{drug}=1$ and older $=1)$ interactions. This means the relative risk is less than one.

The RERI, AP, and SI statistics were designed for models in which the -+, +-, and ++ interactions each have positive ERRS. That is, values of 1 for the predictors are hypothesized to be associated with increased risk. Clearly, this is not true here. drug $=1$ is hypothesized to be associated with decreased risk. We incorrectly specified the model for reri. drug $=0$ should be the category hypothesized for a positive ERR.

We can correct this by specifying the option baselevel1(1), which tells reri to use drug $=1$ as the base level for the first predictor.

```
. reri stcox drug older, baselevel1(1)
```

Fitting stcox ...
Interaction of drug and older on an additive scale
Model: Cox proportional hazards Number of observations $=48$

| drug\#older |  |  |
| ---: | :--- | :--- |
| -+ | 1\#age>56 |  |
| +- | $0 \#$ age $<=56$ |  |
| + | + | $0 \#$ age $>56$ |


|  | ERR | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| drug\#older |  |  |  |  |  |  |
| -+ | 2.192339 | 1.962714 | 1.89 | 0.059 | -.0433052 | 9.652331 |
| +- | 10.885 | 7.527446 | 3.91 | 0.000 | 2.434672 | 40.12572 |
| ++ | 18.33507 | 12.94706 | 4.42 | 0.000 | 4.20437 | 70.83291 |
| RERI | 5.257728 | 8.484262 | 0.62 | 0.535 | -11.37112 | 21.88658 |
| Attr. prop. | .271927 | .3309285 | 0.82 | 0.411 | -.376681 | .920535 |
| Synergy ind. | 1.402049 | .677738 | 0.70 | 0.484 | .5436253 | 3.615984 |

Note: $\mathrm{P}>|z|$ for synergy index (SI) is for test $\mathrm{HO}: ~ \mathrm{SI}=1$.

The ERRs are now all positive, and the model is easy to interpret. The RERI statistic is 5.26 , so the effects are superadditive. But the RERI statistic has a $p$-value of 0.535 , so we cannot reject the additive model.

The message of this example is clear. If you have negative ERRs, first check that you correctly specified the hypothesized direction of effects. If the model is correctly specified and there are negative ERRs (not close to 0), then the RERI, AP, and SI statistics may not be useful descriptors for the observed effects.

## Stored results

reri stores the following in $r()$ :

| Scalars |  |
| :--- | :--- |
| $\quad r(N)$ | number of observations |
| $r(r e r i)$ | RERI |
| $r(a p)$ | AP |
| $r(s i)$ | SI |
| $r(l e v e l)$ | confidence level |
| Matrices |  |
| $r\left(r e r i \_t a b l e\right)$ | table of results |

The e() and r() stored results from $c m d$ are returned as well.

## Methods and formulas

The RERI statistic is

$$
\operatorname{RERI}=\operatorname{ERR}_{A+B+}-\operatorname{ERR}_{A-B+}-\operatorname{ERR}_{A+B-}
$$

where $\operatorname{ERR}=\mathrm{RR}-1$ is the ERR and RR denotes relative risk. Here $A+B+$ refers to subjects positive for exposure $A$ and positive for exposure $B$, with $A-B+$ and $A+B$ - defined similarly.

The AP is

$$
\mathrm{AP}=\frac{\mathrm{RERI}}{\mathrm{RR}_{A+B+}}
$$

The SI is

$$
\mathrm{SI}=\frac{\mathrm{ERR}_{A+B+}}{\mathrm{ERR}_{A-B+}+\mathrm{ERR}_{A+B-}}
$$

When incidence-rate ratios, odds ratios, or hazard ratios are estimated rather than risk ratios, they are used in the calculation in place of relative risk.

Standard errors are calculated by nlcom using the "delta method". See Methods and formulas in $[\mathrm{R}]$ nlcom for details. The standard error for the SI is calculated by first calculating the standard error for the logarithm of SI and then transforming the result back to the SI scale.

## Acknowledgment

reri is based on the ic package written by Niels Henrik Bruun of the Aalborg University Hospital, Morten Fenger-Grøn of Jyske Bank, and Anders Prior of Aarhus University (2017).

## References

Andersson, T., L. Alfredsson, H. Källberg, S. Zdravkovic, and A. Ahlbom. 2005. Calculating measures of biological interaction. European Journal of Epidemiology 20: 575-579. https://doi.org/10.1007/s10654-005-7835-x.

Brender, J. D., P. J. Weyer, P. A. Romitti, B. P. Mohanty, M. U. Shinde, A. M. Vuong, J. R. Sharkey, D. Dwivedi, S. A. Horel, J. Kantamneni, J. C. Huber, Jr., Q. Zheng, M. M. Werler, K. E. Kelley, J. S. Griesenbeck, F. B. Zhan, P. H. Langlois, L. Suarez, M. A. Canfield, and the National Birth Defects Prevention Study. 2013. Prenatal nitrate intake from drinking water and selected birth defects in offspring of participants in the National Birth Defects Prevention Study. Environmental Health Perspectives 121: 1083-1089. https://doi.org/10.1289/ehp.1206249.

Bruun, N. H., M. Fenger-Gron, and A. Prior. 2017. ic: Module to compute measures of interaction contrast (biological interaction). Boston College Department of Economics, Statistical Software Components S457975. https://ideas.repec.org/c/boc/bocode/s457975.html.
Cummings, P. 2009. Methods for estimating adjusted risk ratios. Stata Journal 9: 175-196.
Greenland, S., and D. C. Thomas. 1982. On the need for the rare disease assumption in case-control studies. American Journal of Epidemiology 116: 547-553. https://doi.org/10.1093/oxfordjournals.aje.a113439.
Kalbfleisch, J. D., and R. L. Prentice. 2002. The Statistical Analysis of Failure Time Data. 2nd ed. New York: Wiley.
Knol, M. J., T. J. VanderWeele, R. H. H. Groenwold, O. H. Klungel, M. M. Rovers, and D. E. Grobbee. 2011. Estimating measures of interaction on an additive scale for preventive exposures. European Journal of Epidemiology 26: 433-438. https://doi.org/10.1007/s10654-011-9554-9.
Lash, T. L., T. J. VanderWeele, S. Haneuse, and K. J. Rothman. 2021. Modern Epidemiology. 4th ed. Philadelphia: Wolters Kluwer.

Rothman, K. J., S. Greenland, and A. M. Walker. 1980. Concepts of interaction. American Journal of Epidemiology 112: 467-470. https://doi.org/10.1093/oxfordjournals.aje.a113015.
Skrondal, A. 2003. Interaction as departure from additivity in case-control studies: A cautionary note. American Journal of Epidemiology 158: 251-258. https://doi.org/10.1093/aje/kwg113.

## Also see

[R] binreg - Generalized linear models: Extensions to the binomial family
[R] Epitab - Tables for epidemiologists
[R] logistic - Logistic regression, reporting odds ratios
[R] nbreg - Negative binomial regression
[R] poisson - Poisson regression
[ST] stcox - Cox proportional hazards model
[ST] stintcox - Cox proportional hazards model for interval-censored survival-time data
[ST] stintreg - Parametric models for interval-censored survival-time data
[ST] streg - Parametric survival models

## Title

\#review - Review previous commands

## Description Syntax Remarks and examples

## Description

The \#review command displays the last few lines typed at the terminal.

## Syntax

\#review $\left[\#_{1}\left[\#_{2}\right]\right]$

## Remarks and examples

\#review (pronounced pound-review) is a Stata preprocessor command. \#commands do not generate a return code or generate ordinary Stata errors. The only error message associated with \#commands is "unrecognized \#command".

The \#review command displays the last few lines typed at the terminal. If no arguments follow \#review, the last 20 lines typed at the terminal are displayed. The first argument specifies the number of lines to be reviewed, so \#review 10 displays the last 10 lines typed. The second argument specifies the number of lines to be displayed, so \#review 105 displays five lines, starting at the 10th previous line.

Stata reserves a buffer for \#review lines and stores as many previous lines in the buffer as will fit, rolling out the oldest line to make room for the newest. Requests to \#review lines no longer stored will be ignored. Only lines typed at the terminal are placed in the \#review buffer. See [U] 10.5 Editing previous lines in Stata.
> Example 1
Typing \#review by itself will show the last 20 lines you typed at the terminal:

```
. #review
20 webuse auto
19 describe
1 8 \text { notes}
17 * comments go into the #review buffer, too
16 tabulate rep78
15 codebook foreign
1 4 \text { regress mpg weight i.foreign}
13 generate gp100m = 100/mpg
12 label variable gp100m "Gallons per 100 miles"
11 tabulate rep78, summarize(gp100m)
10 ttest gp100m, by(foreign)
9 regress gp100m weight i.foreign
8 regress, beta
7 margins foreign
6 generate gpmw = ((100/mpg)/weight)*1000
5 summarize gpmw
4 \text { twoway scatter price gpmw}
3 regress gpmw i.foreign
2 regress gpmw i.foreign, vce(robust)
1 #review
```

Typing \#review 152 shows the 15th and 14th previous lines:

```
. #review 15 2
15 regress mpg weight i.foreign
14 generate gp100m = 100/mpg
```

- _


## Title

roc - Receiver operating characteristic (ROC) analysis

## Description References

## Description

ROC analysis quantifies the accuracy of diagnostic tests or other evaluation modalities used to discriminate between two states or conditions, which are here referred to as normal and abnormal or control and case. The discriminatory accuracy of a diagnostic test is measured by its ability to correctly classify known normal and abnormal subjects. For this reason, we often refer to the diagnostic test as a classifier. The analysis uses the ROC curve, a graph of the sensitivity versus 1 - specificity of the diagnostic test. The sensitivity is the fraction of positive cases that are correctly classified by the diagnostic test, whereas the specificity is the fraction of negative cases that are correctly classified. Thus the sensitivity is the true-positive rate, and the specificity is the true-negative rate.

There are six ROC commands:

| Command | Entry | Description |
| :--- | :--- | :--- |
| roccomp | $[R]$ roccomp | Tests of equality of ROC areas |
| rocgold | $[R]$ roccomp | Tests of equality of ROC areas against a standard ROC curve |
| rocfit | $[R]$ rocfit | Parametric ROC models |
| rocreg | $[R]$ rocreg | Nonparametric and parametric ROC regression models |
| rocregplot | $[R]$ rocregplot | Plot marginal and covariate-specific ROC curves |
| roctab | $[R]$ roctab | Nonparametric ROC analysis |

Postestimation commands are available after rocfit and rocreg; see [R] rocfit postestimation and [R] rocreg postestimation.

Both nonparametric and parametric (semiparametric) methods have been suggested for generating the ROC curve. The roctab command performs nonparametric ROC analysis for a single classifier. roccomp extends the nonparametric ROC analysis function of roctab to situations where we have multiple diagnostic tests of interest to be compared and tested. The rocgold command also provides ROC analysis for multiple classifiers. rocgold compares each classifier's ROC curve to a "gold standard" ROC curve and makes adjustments for multiple comparisons in the analysis. Both rocgold and roccomp also allow parametric estimation of the ROC curve through a binormal fit. In a binormal fit, both the control and the case populations are normal.

The rocfit command also estimates the ROC curve of a classifier through a binormal fit. Unlike roctab, roccomp, and rocgold, rocfit is an estimation command. In postestimation, graphs of the ROC curve and confidence bands can be produced. Additional tests on the parameters can also be conducted.

ROC analysis can be interpreted as a two-stage process. First, the control distribution of the classifier is estimated, assuming a normal model or using a distribution-free estimation technique. The classifier is standardized using the control distribution to $1-$ percentile value, the false-positive rate. Second, the ROC curve is estimated as the case distribution of the standardized classifier values.

Covariates may affect both stages of ROC analysis. The first stage may be affected, yielding a covariate-adjusted ROC curve. The second stage may also be affected, producing multiple covariatespecific ROC curves.

The rocreg command performs ROC analysis under both types of covariate effects. Both parametric (semiparametric) and nonparametric methods may be used by rocreg. Like rocfit, rocreg is an estimation command and provides many postestimation capabilities.

The global performance of a diagnostic test is commonly summarized by the area under the ROC curve (AUC). This area can be interpreted as the probability that the result of a diagnostic test of a randomly selected abnormal subject will be greater than the result of the same diagnostic test from a randomly selected normal subject. The greater the AUC, the better the global performance of the diagnostic test. Each of the ROC commands provides computation of the AUC.

Citing a lack of clinical relevance for the AUC, other ROC summary measures have been suggested. These include the partial area under the ROC curve for a given false-positive rate $t$ [pAUC $(t)$ ]. This is the area under the ROC curve from the false-positive rate of 0 to $t$. The ROC value at a particular false-positive rate and the false-positive rate for a particular ROC value are also useful summary measures for the ROC curve. These three measures are directly estimated by rocreg during the model fit or postestimation stages. Point estimates of ROC value are computed by the other ROC commands, but no standard errors are reported.

See Pepe (2003) for a discussion of ROC analysis. Pepe has posted Stata datasets and programs used to reproduce results presented in the book (https://www.stata.com/bookstore/pepe.html).

## References

Cook, J. A., and A. Rajbhandari. 2018. heckroccurve: ROC curves for selected samples. Stata Journal 18: 174-183.
Cook, J. A., and V. Ramadas. 2020. When to consult precision-recall curves. Stata Journal 20: 131-148.
Luque-Fernandez, M. A., D. Redondo-Sánchez, and C. Maringe. 2019. cvauroc: Command to compute cross-validated area under the curve for ROC analysis after predictive modeling for binary outcomes. Stata Journal 19: 615-625.
Pepe, M. S. 2003. The Statistical Evaluation of Medical Tests for Classification and Prediction. New York: Oxford University Press.

## Title

roccomp - Tests of equality of ROC areas

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

roccomp and rocgold are used to perform receiver operating characteristic (ROC) analyses with rating and discrete classification data.

The two variables refvar and classvar must be numeric. The reference variable indicates the true state of the observation, such as diseased and nondiseased or normal and abnormal, and must be coded as 0 and 1. The rating or outcome of the diagnostic test or test modality is recorded in classvar, which must be at least ordinal, with higher values indicating higher risk.
roccomp tests the equality of two or more ROC areas obtained from applying two or more test modalities to the same sample or to independent samples. roccomp expects the data to be in wide form when comparing areas estimated from the same sample and in long form for areas estimated from independent samples.
rocgold independently tests the equality of the ROC area of each of several test modalities, specified by classvar, against a "gold standard" ROC curve, goldvar. For each comparison, rocgold reports the raw and the Bonferroni-adjusted $p$-value. Optionally, Šidák's adjustment for multiple comparisons can be obtained.

See $[R]$ rocfit and $[R]$ rocreg for commands that fit maximum-likelihood ROC models.

## Quick start

Equality of AUCs for rating v1 of true state true between samples defined by catvar roccomp true v1, by (catvar)

Equality of AUCs for ratings v1 and v2 for the same sample
roccomp true v1 v2
Same as above, but plot ROC curves without reporting summary statistics and test of equality
roccomp true v1 v2, graph
Same as above, but plot v1 with a dashed line and v2 with a solid line

```
roccomp true v1 v2, graph plot1opts(lpattern(dash)) ///
    plot2opts(lpattern(solid))
```

Use contrast matrix mymat to compare ROC areas for v1, v2, v3, and v4

```
matrix mymat = (1,0,-1,0 \ 0,1,0,-1)
roccomp true v1 v2 v3 v4, test(mymat)
```

Test equality of ROC area for v1 against a "gold standard" gold
rocgold true gold v1

## Menu

## roccomp

Statistics $>$ Epidemiology and related $>$ ROC analysis $>$ Test equality of two or more ROC areas rocgold

Statistics $>$ Epidemiology and related $>$ ROC analysis $>$ Test equality of ROC area against gold standard

## Syntax

Test equality of ROC areas
roccomp refvar classvar [classvars] [if] [in] [weight] [, roccomp_options]

Test equality of ROC area against a standard ROC curve
rocgold refvar goldvar classvar [classvars] [if] [in] [weight] [, rocgold_options]
roccomp_options
Main
by (varname)
test (matname)
graph
norefline
separate
summary
binormal
line\#opts(cline_options)
level(\#)
Plot
plot\#opts(plot_options)
Reference line
rlopts (cline_options)
Y axis, X axis, Titles, Legend, Overall twoway_options

Description
split into groups by variable
use contrast matrix for comparing ROC areas graph the ROC curve
suppress plotting the 45 -degree reference line
place each ROC curve on its own graph
report the area under the ROC curve
estimate areas by using binormal distribution assumption
affect rendition of the \#th binormal fit line
set confidence level; default is level (95)
affect rendition of the \#th ROC curve
affect rendition of the reference line
any options other than by () documented in [G-3] twoway_options
rocgold_options

Main
sidak
test (matname)
graph
norefline
separate
summary
binormal
line\#opts (cline_options)
level(\#)
Plot
plot\#opts (plot_options)
Reference line
rlopts(cline_options)
Y axis, X axis, Titles, Legend, Overall twoway_options

Description
adjust the $p$-value by using Šidák's method
use contrast matrix for comparing ROC areas
graph the ROC curve
suppress plotting the 45-degree reference line
place each ROC curve on its own graph
report the area under the ROC curve
estimate areas by using binormal distribution assumption
affect rendition of the \#th binormal fit line
set confidence level; default is level (95)
affect rendition of the \#th ROC curve; plot 1 is the "gold standard"
affect rendition of the reference line
any options other than by() documented in [G-3] twoway_options
plot_options
marker_options
marker_label_options
cline_options

Description
change look of markers (color, size, etc.)
add marker labels; change look or position
change look of the line
collect is allowed with roccomp and rocgold; see [U] 11.1.10 Prefix commands.
fweights are allowed; see [U] 11.1.6 weight.

## Options

by (varname) (roccomp only) is required when comparing independent ROC areas. The by () variable identifies the groups to be compared.
sidak (rocgold only) requests that the $p$-value be adjusted for the effect of multiple comparisons by using Sidák's method. Bonferroni's adjustment is reported by default.
test (matname) specifies the contrast matrix to be used when comparing ROC areas. By default, the null hypothesis that all areas are equal is tested.
graph produces graphical output of the ROC curve.
norefline suppresses plotting the 45 -degree reference line from the graphical output of the ROC curve.
separate is meaningful only with roccomp and specifies that each ROC curve be placed on its own graph rather than one curve on top of the other.
summary reports the area under the ROC curve, its standard error, and its confidence interval. This option is needed only when also specifying graph.
binormal specifies that the areas under the ROC curves to be compared should be estimated using the binormal distribution assumption. By default, areas to be compared are computed using the trapezoidal rule.
line\#opts(cline_options) affect the rendition of the line representing the \#th ROC curve drawn using the binormal distribution assumption; see [G-3] cline_options. These lines are drawn only if the binormal option is specified.
level(\#) specifies the confidence level, as a percentage, for the confidence intervals. The default is level (95) or as set by set level; see [R] level.

Plot
plot\#opts(plot_options) affect the rendition of the \#th ROC curve-the curve's plotted points connected by lines. The plot_options can affect the size and color of markers, whether and how the markers are labeled, and whether and how the points are connected; see [G-3] marker_options, [G-3] marker_label_options, and [G-3] cline_options.
For rocgold, plot1opts() are applied to the ROC for the gold standard.
$\qquad$ Reference line
rlopts (cline_options) affects the rendition of the reference line; see [G-3] cline_options.
$\qquad$ $\sqrt{Y}$ axis, X axis, Titles, Legend, Overall
twoway_options are any of the options documented in [G-3] twoway_options. These include options for titling the graph (see [G-3] title_options), options for saving the graph to disk (see [G-3] saving_option), and the by () option (see [G-3] by_option).

## Remarks and examples

Remarks are presented under the following headings:

```
Introduction
Comparing areas under the ROC curve
Correlated data
Independent data
Comparing areas with a gold standard
```


## Introduction

roccomp provides comparison of the ROC curves of multiple classifiers. rocgold compares the ROC curves of multiple classifiers with a single "gold standard" classifier. Adjustment of inference for multiple comparisons is also provided by rocgold.

See Pepe (2003) for a discussion of ROC analysis. Pepe has posted Stata datasets and programs used to reproduce results presented in the book (https://www.stata.com/bookstore/pepe.html).

## Comparing areas under the ROC curve

The area under multiple ROC curves can be compared by using roccomp. The command syntax is slightly different if the ROC curves are correlated (that is, different diagnostic tests are applied to the same sample) or independent (that is, diagnostic tests are applied to different samples).

## Correlated data

## $>$ Example 1

Hanley and McNeil (1983) presented data from an evaluation of two computer algorithms designed to reconstruct CT images from phantoms. We will call these two algorithms' modalities 1 and 2. A sample of 112 phantoms was selected; 58 phantoms were considered normal, and the remaining 54 were abnormal. Each of the two modalities was applied to each phantom, and the resulting images were rated by a reviewer using a six-point scale: $1=$ definitely normal, $2=$ probably normal, $3=$ possibly normal, $4=$ possibly abnormal, $5=$ probably abnormal, and $6=$ definitely abnormal. Because each modality was applied to the same sample of phantoms, the two sets of outcomes are correlated.

We list the first 7 observations:

```
. use https://www.stata-press.com/data/r18/ct
(Reconstruction of CT images)
. list in 1/7, sep(0)
```

|  | $\bmod 1$ | $\bmod 2$ | status |
| :--- | ---: | ---: | ---: |
| 1. | 2 | 1 | 0 |
| 2. | 5 | 5 | 1 |
| 3. | 2 | 1 | 0 |
| 4. | 2 | 3 | 0 |
| 5. | 5 | 6 | 1 |
| 6. | 2 | 2 | 0 |
| 7. | 3 | 2 | 0 |

The data are in wide form, which is required when dealing with correlated data. Each observation corresponds to one phantom. The variable mod1 identifies the rating assigned for the first modality, and mod2 identifies the rating assigned for the second modality. The true status of the phantoms is given by status=0 if they are normal and status=1 if they are abnormal. The observations with at least one missing rating were dropped from the analysis.

We plot the two ROC curves and compare their areas.

| . roccomp status mod1 mod2, graph summary |  |  |  |  |  |
| :--- | ---: | ---: | ---: | ---: | ---: |
|  | Obs | ROC <br> area | Std. err. | Asymptotic normal <br> [95\% conf. interval] |  |
| mod1 | 112 | 0.8828 | 0.0317 | 0.82067 | 0.94498 |
| $\bmod 2$ | 112 | 0.9302 | 0.0256 | 0.88005 | 0.98042 |

```
H0: area(mod1) = area(mod2)
    chi2(1) = 2.31 Prob>chi2 = 0.1282
```



By default, roccomp, with the graph option specified, plots the ROC curves on the same graph. Optionally, the curves can be plotted side by side, each on its own graph, by also specifying separate.

For each curve, roccomp reports summary statistics and provides a test for the equality of the area under the curves, using an algorithm suggested by DeLong, DeLong, and Clarke-Pearson (1988).

Although the area under the ROC curve for modality 2 is larger than that of modality 1 , the $\chi^{2}$ test yielded a $p$-value of 0.1282 , suggesting that there is no significant difference between these two areas.

The roccomp command can also be used to compare more than two ROC areas. To illustrate this, we modified the previous dataset by including a fictitious third modality.

```
. use https://www.stata-press.com/data/r18/ct2
(Reconstruction of CT images)
. roccomp status mod1 mod2 mod3, graph summary
```

|  | Obs | $\begin{array}{r} \text { ROC } \\ \text { area } \end{array}$ | Std. err. | Asymptotic normal [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: | :---: |
| mod1 | 112 | 0.8828 | 0.0317 | 0.82067 | 0.94498 |
| mod2 | 112 | 0.9302 | 0.0256 | 0.88005 | 0.98042 |
| mod3 | 112 | 0.9240 | 0.0241 | 0.87670 | 0.97132 |

```
H0: area(mod1) = area(mod2) = area(mod3)
    chi2(2) = 6.54 Prob>chi2 = 0.0381
```



By default, roccomp tests whether the areas under the ROC curves are all equal. Other comparisons can be tested by creating a contrast matrix and specifying test (matname), where matname is the name of the contrast matrix.

For example, assume that we are interested in testing whether the area under the ROC for mod1 is equal to that of mod3. To do this, we can first create an appropriate contrast matrix and then specify its name with the test () option.

Of course, this is a trivial example because we could have just specified

```
. roccomp status mod1 mod3
```

without including mod2 to obtain the same test results. However, for illustration, we will continue with this example.

The contrast matrix must have its number of columns equal to the number of classvars (that is, the total number of ROC curves) and a number of rows less than or equal to the number of classvars, and the elements of each row must add to zero.

```
. matrix C=(1,0,-1)
. roccomp status mod1 mod2 mod3, test(C)
```

|  | ROC |  |  | Asymptotic normal <br> [95\% conf. interval] |  |
| :--- | :---: | :---: | :---: | :---: | ---: |
|  | Obs | area | Std. err. | 0.94498 |  |
| mod1 | 112 | 0.8828 | 0.0317 | 0.82067 | 0.98042 |
| mod2 | 112 | 0.9302 | 0.0256 | 0.88005 | 0.87670 |

```
HO: Comparison as defined by contrast matrix: C
    chi2(1) = 5.25 Prob>chi2 = 0.0220
```

Although all three areas are reported, the comparison is made using the specified contrast matrix.
Perhaps more interesting would be a comparison of the area from mod1 and the average area of $\bmod 2$ and mod3.

```
. matrix C=(1,-.5,-.5)
```

. roccomp status mod1 mod2 mod3, test(C)

|  | Obs | $\begin{array}{r} \text { ROC } \\ \text { area } \end{array}$ | Std. err. | Asymptotic normal [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: | :---: |
| mod1 | 112 | 0.8828 | 0.0317 | 0.82067 | 0.94498 |
| mod2 | 112 | 0.9302 | 0.0256 | 0.88005 | 0.98042 |
| mod3 | 112 | 0.9240 | 0.0241 | 0.87670 | 0.97132 |

H0: Comparison as defined by contrast matrix: C
$\operatorname{chi2}(1)=3.43 \quad$ Prob>chi2 $=0.0642$
Other contrasts could be made. For example, we could test if mod3 is different from at least one of the other two by first creating the following contrast matrix:

```
. matrix C=(-1,0,1 \ 0,-1,1)
. mat list C
C[2,3]
c1 c2 c3
r1 -1 0
r2 0
```


## Independent data

## > Example 2

In example 1, we noted that because each test modality was applied to the same sample of phantoms, the classification outcomes were correlated. Now, assume that we have collected the same data presented by Hanley and McNeil (1983), except that we applied the first test modality to one sample of phantoms and the second test modality to a different sample of phantoms. The resulting measurements are now considered independent.

Here are a few of the observations.

```
. use https://www.stata-press.com/data/r18/ct3
(Reconstruction of CT images)
. list in 1/7, sep(0)
```

| pop | status | rating | mod |
| ---: | ---: | ---: | ---: |
| 12 | 0 | 1 | 1 |
| 31 | 0 | 1 | 2 |
| 1 | 1 | 1 | 1 |
| 3 | 1 | 1 | 2 |
| 28 | 0 | 2 | 1 |
| 19 | 0 | 2 | 2 |
| 3 | 1 | 2 | 1 |

The data are in long form, which is required when dealing with independent data. The data consist of 24 observations: 6 observations corresponding to abnormal phantoms and 6 to normal phantoms evaluated using the first modality, and similarly 6 observations corresponding to abnormal phantoms and 6 to normal phantoms evaluated using the second modality. The number of phantoms corresponding to each observation is given by the pop variable. Once again, we have frequency-weighted data. The variable mod identifies the modality, and rating is the assigned classification.

We can better view our data by using the table command.
. table (mod status) (rating) [fw=pop], totals(mod mod\#status mod\#rating)

|  | Rating |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | 1 | 2 | 3 | 4 | 5 | 6 | Total |
| Modality <br> 1 |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| Status |  |  |  |  |  |  |  |
| 0 | 12 | 28 | 8 | 6 | 4 |  | 58 |
| 1 | 1 | 3 | 6 | 13 | 22 | 9 | 54 |
| Total | 13 | 31 | 14 | 19 | 26 | 9 | 112 |
| 2 |  |  |  |  |  |  |  |
| Status |  |  |  |  |  |  |  |
| 0 | 31 | 19 | 5 | 3 |  |  | 58 |
| 1 | 3 | 2 | 5 | 19 | 15 | 10 | 54 |
| Total | 34 | 21 | 10 | 22 | 15 | 10 | 112 |

The status variable indicates the true status of the phantoms: status $=0$ if they are normal and status $=1$ if they are abnormal.

We now compare the areas under the two ROC curves.

| . roccomp status rating [fw=pop], by (mod) graph summary |  |  |  |  |  |
| :--- | ---: | ---: | ---: | ---: | ---: |
| mod | Obs | ROC <br> area | Std. err. | Asymptotic normal <br> [95\% conf. interval] |  |
| 1 | 112 | 0.8828 | 0.0317 | 0.82067 | 0.94498 |
| 2 | 112 | 0.9302 | 0.0256 | 0.88005 | 0.98042 |

Н०: $\begin{array}{rlr}\operatorname{area}(1) & =\operatorname{area}(2) \\ \operatorname{chi2}(1) & = & 1.35\end{array}$
Prob $>$ chi2 $=0.2447$


## Comparing areas with a gold standard

The area under multiple ROC curves can be compared with a gold standard using rocgold. The command syntax is similar to that of roccomp. The tests are corrected for the effect of multiple comparisons.

## > Example 3

We will use the same data (presented by Hanley and McNeil [1983]) as in the roccomp examples. Let's assume that the first modality is considered to be the standard against which both the second and third modalities are compared.

We want to plot and compare both the areas of the ROC curves of mod2 and mod3 with mod1. Because we consider mod1 to be the gold standard, it is listed first after the reference variable in the rocgold command line.

```
. use https://www.stata-press.com/data/r18/ct2
(Reconstruction of CT images)
. rocgold status mod1 mod2 mod3, graph summary
```

|  | ROC <br> area | Std. err. | chi2 | df | Pr>chi2 | Bonferroni <br> Pr>chi2 |
| :--- | ---: | :---: | :---: | :---: | ---: | ---: |
| mod1 (standard) | 0.8828 | 0.0317 |  |  |  |  |
| mod2 | 0.9302 | 0.0256 | 2.3146 | 1 | 0.1282 | 0.2563 |
| mod3 | 0.9240 | 0.0241 | 5.2480 | 1 | 0.0220 | 0.0439 |



Equivalently, we could have done this in two steps by using the roccomp command.

```
. roccomp status mod1 mod2, graph summary
. roccomp status mod1 mod3, graph summary
```


## Stored results

roccomp stores the following in $r()$ :
Scalars

$$
\begin{array}{llll}
r\left(\mathrm{~N} \_\mathrm{g}\right) & \text { number of groups } & r(\mathrm{df}) & \chi^{2} \text { degrees of freedom } \\
r(\mathrm{p}) & p \text {-value for } \chi^{2} \text { test } & r(\text { chi2 }) & \chi^{2}
\end{array}
$$

Matrices
$r(V)$ variance-covariance matrix
rocgold stores the following in $r()$ :
Scalars
$r\left(N \_g\right)$ number of groups
Matrices
$r(V)$ variance-covariance matrix $\quad r(p) \quad$ vector of $p$-values for $\chi^{2}$ tests
$\begin{array}{ll}\mathrm{r}(\mathrm{chi2}) & \chi^{2} \text { vector } \\ \mathrm{r}(\mathrm{df}) & \chi^{2} \text { degrees-of-freedom vector }\end{array}$

$$
\mathrm{r}(\text { p_adj }) \quad \text { vector of adjusted } p \text {-values }
$$

## Methods and formulas

Assume that we applied a diagnostic test to each of $N_{n}$ normal and $N_{a}$ abnormal subjects. Further assume that the higher the outcome value of the diagnostic test, the higher the risk of the subject being abnormal. Let $\widehat{\theta}$ be the estimated area under the curve, and let $X_{i}, i=1,2, \ldots, N_{a}$ and $Y_{j}, j=1,2, \ldots, N_{n}$ be the values of the diagnostic test for the abnormal and normal subjects, respectively.

Areas under ROC curves are compared using an algorithm suggested by DeLong, DeLong, and Clarke-Pearson (1988). Let $\widehat{\boldsymbol{\theta}}=\left(\widehat{\theta^{1}}, \widehat{\theta^{2}}, \ldots, \widehat{\theta^{k}}\right)$ be a vector representing the areas under $k$ ROC curves. See Methods and formulas in [R] roctab for the definition of these area estimates.

For the $r$ th area, define

$$
V_{10}^{r}\left(X_{i}\right)=\frac{1}{N_{n}} \sum_{j=1}^{N_{n}} \psi\left(X_{i}^{r}, Y_{j}^{r}\right)
$$

and for each normal subject, $j$, define

$$
V_{01}^{r}\left(Y_{j}\right)=\frac{1}{N_{a}} \sum_{i=1}^{N_{a}} \psi\left(X_{i}^{r}, Y_{j}^{r}\right)
$$

where

$$
\psi\left(X^{r}, Y^{r}\right)= \begin{cases}1 & Y^{r}<X^{r} \\ \frac{1}{2} & Y^{r}=X^{r} \\ 0 & Y^{r}>X^{r}\end{cases}
$$

Define the $k \times k$ matrix $\mathbf{S}_{\mathbf{1 0}}$ such that the $(r, s)$ th element is

$$
S_{10}^{r, s}=\frac{1}{N_{a}-1} \sum_{i=1}^{N_{a}}\left\{V_{10}^{r}\left(X_{i}\right)-\widehat{\theta^{r}}\right\}\left\{V_{10}^{s}\left(X_{i}\right)-\widehat{\theta^{s}}\right\}
$$

and $\mathbf{S}_{\mathbf{0 1}}$ such that the $(r, s)$ th element is

$$
S_{01}^{r, s}=\frac{1}{N_{n}-1} \sum_{j=1}^{N_{n}}\left\{V_{01}^{r}\left(Y_{i}\right)-\widehat{\theta^{r}}\right\}\left\{V_{01}^{s}\left(Y_{i}\right)-\widehat{\theta^{s}}\right\}
$$

Then, the covariance matrix is

$$
S=\frac{1}{N_{a}} S_{10}+\frac{1}{N_{n}} S_{01}
$$

Let $\mathbf{L}$ be a contrast matrix defining the comparison, so that

$$
(\widehat{\boldsymbol{\theta}}-\boldsymbol{\theta})^{\prime} \mathbf{L}^{\prime}\left(\mathbf{L} \mathbf{S L}^{\prime}\right)^{-1} \mathbf{L}(\widehat{\boldsymbol{\theta}}-\boldsymbol{\theta})
$$

has a $\chi^{2}$ distribution with degrees of freedom equal to the rank of $\mathbf{L S L}{ }^{\prime}$.

## References

Cleves, M. A. 2002a. Comparative assessment of three common algorithms for estimating the variance of the area under the nonparametric receiver operating characteristic curve. Stata Journal 2: 280-289.
__. 2002b. From the help desk: Comparing areas under receiver operating characteristic curves from two or more probit or logit models. Stata Journal 2: 301-313.
DeLong, E. R., D. M. DeLong, and D. L. Clarke-Pearson. 1988. Comparing the areas under two or more correlated receiver operating characteristic curves: A nonparametric approach. Biometrics 44: 837-845. https://doi.org/10.2307/2531595.

Erdreich, L. S., and E. T. Lee. 1981. Use of relative operating characteristic analysis in epidemiology: A method for dealing with subjective judgment. American Journal of Epidemiology 114: 649-662. https://doi.org/10.1093/oxfordjournals.aje.a113236.
Hanley, J. A., and B. J. McNeil. 1983. A method of comparing the areas under receiver operating characteristic curves derived from the same cases. Radiology 148: 839-843. https://doi.org/10.1148/radiology.148.3.6878708.
Harbord, R. M., and P. Whiting. 2009. metandi: Meta-analysis of diagnostic accuracy using hierarchical logistic regression. Stata Journal 9: 211-229.

Juul, S., and M. Frydenberg. 2021. An Introduction to Stata for Health Researchers. 5th ed. College Station, TX: Stata Press.
Ma, G., and W. J. Hall. 1993. Confidence bands for the receiver operating characteristic curves. Medical Decision Making 13: 191-197. https://doi.org/10.1177/0272989X9301300304.

Pepe, M. S. 2003. The Statistical Evaluation of Medical Tests for Classification and Prediction. New York: Oxford University Press.

Reichenheim, M. E., and A. Ponce de Leon. 2002. Estimation of sensitivity and specificity arising from validity studies with incomplete design. Stata Journal 2: 267-279.

Working, H., and H. Hotelling. 1929. Application of the theory of error to the interpretation of trends. Journal of the American Statistical Association 24 (Suppl.): 73-85. https://doi.org/10.2307/2277011.

[^57]
## Title

rocfit — Parametric ROC models

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

rocfit fits maximum-likelihood ROC models assuming a binormal distribution of the latent variable.
The two variables refvar and classvar must be numeric. The reference variable indicates the true state of the observation, such as diseased and nondiseased or normal and abnormal, and must be coded as 0 and 1 . The rating or outcome of the diagnostic test or test modality is recorded in classvar, which must be at least ordinal, with higher values indicating higher risk.

See $[R]$ roc for other commands designed to perform receiver operating characteristic (ROC) analyses with rating and discrete classification data.

## Quick start

Binary true state, true, as a function of classification variable class
rocfit true class
Same as above, but with frequency weights wvar
rocfit true class [fweight = wvar]
Specify that class is continuous and generate v1 containing classification groups

```
rocfit true class, continuous(3) generate(v1)
```


## Menu

Statistics $>$ Epidemiology and related $>$ ROC analysis $>$ Parametric ROC analysis without covariates

## Syntax

rocfit refvar classvar $[$ if $][$ in $]$ [weight $]$ [, rocfit_options]
rocfit_options Description

Model
continuous(\#)
generate(newvar)
SE
vce (vcetype) vcetype may be oim or opg
Reporting
level(\#)
Maximization
maximize_options
divide classvar into \# groups of approximately equal length create newvar containing classification groups
set confidence level; default is level(95)
control the maximization process; seldom used
collect and fp are allowed; see [U] 11.1.10 Prefix commands.
fweights are allowed; see [U] 11.1.6 weight.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

## Model

continuous(\#) specifies that the continuous classvar be divided into \# groups of approximately equal length. This option is required when classvar takes on more than 20 distinct values.
continuous(.) may be specified to indicate that classvar be used as it is, even though it could have more than 20 distinct values.
generate (newvar) specifies the new variable that is to contain values indicating the groups produced by continuous(\#). generate() may be specified only with continuous().

## SE

vce(vcetype) specifies the type of standard error reported. vcetype may be either oim or opg; see [R] vce_option.

Reporting
level(\#); see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate (\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#),
nrtolerance(\#), nonrtolerance, and from(init_specs); see [R] Maximize. These options are seldom used.

Setting the optimization type to technique (bhhh) resets the default vcetype to vce (opg).

## Remarks and examples

Dorfman and Alf (1969) developed a generalized approach for obtaining maximum likelihood estimates of the parameters for a smooth fitting ROC curve. The most commonly used method for ordinal data, and the one implemented here, is based upon the binormal model; see Pepe (2003), Pepe, Longton, and Janes (2009), and Janes, Longton, and Pepe (2009) for methods of ROC analysis for continuous data, including methods for adjusting for covariates.

The model assumes the existence of an unobserved, continuous, latent variable that is normally distributed (perhaps after a monotonic transformation) in both the normal and abnormal populations with means $\mu_{n}$ and $\mu_{a}$ and variances $\sigma_{n}^{2}$ and $\sigma_{a}^{2}$, respectively. The model further assumes that the $K$ categories of the rating variable result from partitioning the unobserved latent variable by $K-1$ fixed boundaries. The method fits a straight line to the empirical ROC points plotted using normal probability scales on both axes. Maximum likelihood estimates of the line's slope and intercept and the $K-1$ boundaries are obtained simultaneously. See Methods and formulas for details.

The intercept from the fitted line is a measurement of $\left(\mu_{a}-\mu_{n}\right) / \sigma_{a}$, and the slope measures $\sigma_{n} / \sigma_{a}$.

Thus, the intercept is the standardized difference between the two latent population means, and the slope is the ratio of the two standard deviations. The null hypothesis that there is no difference between the two population means is evaluated by testing that the intercept $=0$, and the null hypothesis that the variances in the two populations are equal is evaluated by testing that the slope $=1$.
$>$ Example 1
We use Hanley and McNeil's (1982) dataset, described in example 1 of [R] roctab, to fit a smooth ROC curve assuming a binormal model.

```
. use https://www.stata-press.com/data/r18/hanley
(Tomographic images)
. rocfit disease rating
Fitting binormal model:
Iteration 0: Log likelihood = -123.68069
Iteration 1: Log likelihood = -123.64867
Iteration 2: Log likelihood = -123.64855
Iteration 3: Log likelihood = -123.64855
Binormal model of disease on rating Number of obs = 109
Goodness-of-fit chi2(2) = 0.21
Prob > chi2 = 0.9006
Log likelihood = -123.64855
```

|  | Coefficient | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | $[95 \%$ conf. interval] |  |
| ---: | ---: | :--- | ---: | :--- | ---: | ---: |
| intercept | 1.656782 | 0.310456 | 5.34 | 0.000 | 1.048300 | 2.265265 |
| slope (*) | 0.713002 | 0.215882 | -1.33 | 0.184 | 0.289881 | 1.136123 |
| /cut1 | 0.169768 | 0.165307 | 1.03 | 0.304 | -0.154227 | 0.493764 |
| /cut2 | 0.463215 | 0.167235 | 2.77 | 0.006 | 0.135441 | 0.790990 |
| /cut3 | 0.766860 | 0.174808 | 4.39 | 0.000 | 0.424243 | 1.109477 |
| /cut4 | 1.797938 | 0.299581 | 6.00 | 0.000 | 1.210770 | 2.385106 |


|  |  | Indices from binormal fit |  |  |
| ---: | ---: | :--- | ---: | ---: |
| Index | Estimate | Std. err. | [95\% conf. interval] |  |
| ROC area | 0.911331 | 0.029506 | 0.853501 | 0.969161 |
| delta(m) | 2.323671 | 0.502370 | 1.339044 | 3.308298 |
| d(e) | 1.934361 | 0.257187 | 1.430284 | 2.438438 |
| d(a) | 1.907771 | 0.259822 | 1.398530 | 2.417012 |

(*) z test for slope==1
rocfit outputs the MLE for the intercept and slope of the fitted regression line along with, here, four boundaries (because there are five ratings) labeled /cut1 through /cut4. Also rocfit computes and reports four indices based on the fitted ROC curve: the area under the curve (labeled ROC area), $\delta(m)$ (labeled delta $(\mathrm{m})$ ), $d_{e}$ (labeled d(e)), and $d_{a}$ (labeled d(a)). More information about these indices can be found in Methods and formulas and in Erdreich and Lee (1981).

## Stored results

rocfit stores the following in e():
Scalars
e(N)
e(k)
e(k_eq)
e(k_eq_model)
e(k_dv)
e(df_m)
e(11)
e(chi2_gf)
e(df_gf)
e(p_gf)
e(area)
e(se_area)
e(deltam)
e(se_delm)
e(de)
e(se_de)
$e(d a)$
e(se_da)
$e$ (rank)
e(ic)
e(rc)
e(converged)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(wtype)
e(wexp)
e(title)
e(chi2type)
e(vce)
e(vcetype)
e(opt)
$e$ (which)
e(ml_method)
e(user)
e(technique)
e(properties)
number of observations
number of parameters
number of equations in $e(b)$
number of equations in overall model test
number of dependent variables
model degrees of freedom
log likelihood
goodness-of-fit $\chi^{2}$
goodness-of-fit degrees of freedom
$p$-value for goodness-of-fit test
area under the ROC curve
standard error for the area under the ROC curve
delta(m)
standard area for delta (m)
d(e) index
standard error for d(e) index
$d$ (a) index
standard error for $d(a)$ index
rank of e(V)
number of iterations
return code
1 if converged, 0 otherwise

## rocfit

command as typed
refvar and classvar
weight type
weight expression
title in estimation output
GOF; type of model $\chi^{2}$ test
vcetype specified in vce()
title used to label Std. err.
type of optimization
max or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program
maximization technique
b V

Matrices
e(b)
coefficient vector
e(ilog) iteration $\log$ (up to 20 iterations)
e(gradient)
gradient vector
e(V)
variance-covariance matrix of the estimators
Functions
e(sample) marks estimation sample
In addition to the above, the following is stored in r() :

```
Matrices
```

    \(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values,
    and confidence intervals
    Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

Dorfman and Alf (1969) developed a general procedure for obtaining maximum likelihood estimates of the parameters of a smooth-fitting ROC curve. The most common method, and the one implemented in Stata, is based upon the binormal model.

The model assumes that there is an unobserved continuous latent variable that is normally distributed in both the normal and abnormal populations. The idea is better explained with the following illustration:


The latent variable is assumed to be normally distributed for both the normal and abnormal subjects, perhaps after a monotonic transformation, with means $\mu_{n}$ and $\mu_{a}$ and variances $\sigma_{n}^{2}$ and $\sigma_{a}^{2}$, respectively.

This latent variable is assumed to be partitioned into the $k$ categories of the rating variable by $k-1$ fixed boundaries. In the above figure, the $k=5$ categories of the rating variable identified on the bottom result from the partition of the four boundaries $Z_{1}$ through $Z_{4}$.

Let $R_{j}$ for $j=1,2, \ldots, k$ indicate the categories of the rating variable, let $i=1$ if the subject belongs to the normal group, and let $i=2$ if the subject belongs to the abnormal group.

Then,

$$
p\left(R_{j} \mid i=1\right)=F\left(Z_{j}\right)-F\left(Z_{j-1}\right)
$$

where $Z_{k}=\left(x_{k}-\mu_{n}\right) / \sigma_{n}, F$ is the cumulative normal distribution, $F\left(Z_{0}\right)=0$, and $F\left(Z_{k}\right)=1$. Also,

$$
p\left(R_{j} \mid i=2\right)=F\left(b Z_{j}-a\right)-F\left(b Z_{j-1}-a\right)
$$

where $b=\sigma_{n} / \sigma_{a}$ and $a=\left(\mu_{a}-\mu_{n}\right) / \sigma_{a}$.
The parameters $a, b$ and the $k-1$ fixed boundaries $Z_{j}$ are simultaneously estimated by maximizing the log-likelihood function

$$
\log L=\sum_{i=1}^{2} \sum_{j=1}^{k} r_{i j} \log \left\{p\left(R_{j} \mid i\right)\right\}
$$

where $r_{i j}$ is the number of $R_{j} \mathrm{~s}$ in group $i$.
The area under the fitted ROC curve is computed as

$$
\Phi\left(\frac{a}{\sqrt{1+b^{2}}}\right)
$$

where $\Phi$ is the standard normal cumulative distribution function.
Point estimates for the ROC curve indices are as follows:

$$
\delta(m)=\frac{a}{b} \quad d_{e}=\frac{2 a}{b+1} \quad d_{a}=\frac{a \sqrt{2}}{\sqrt{1+b^{2}}}
$$

Variances for these indices are computed using the delta method.
The $\delta(m)$ estimates $\left(\mu_{a}-\mu_{n}\right) / \sigma_{n}, d_{e}$ estimates $2\left(\mu_{a}-\mu_{n}\right) /\left(\sigma_{a}-\sigma_{n}\right)$, and $d_{a}$ estimates $\sqrt{2}\left(\mu_{a}-\mu_{n}\right) /\left(\sigma_{a}^{2}-\sigma_{n}^{2}\right)^{2}$.

Simultaneous confidence bands for the entire curve are obtained, as suggested by Ma and Hall (1993), by first obtaining Working-Hotelling (1929) confidence bands for the fitted straight line in normal probability coordinates and then transforming them back to ROC coordinates.

## References

Bamber, D. 1975. The area above the ordinal dominance graph and the area below the receiver operating characteristic graph. Journal of Mathematical Psychology 12: 387-415. https://doi.org/10.1016/0022-2496(75)90001-2.
Choi, B. C. K. 1998. Slopes of a receiver operating characteristic curve and likelihood ratios for a diagnostic test. American Journal of Epidemiology 148: 1127-1132. https://doi.org/10.1093/oxfordjournals.aje.a009592.

Dorfman, D. D., and E. Alf, Jr. 1969. Maximum-likelihood estimation of parameters of signal-detection theory and determination of confidence intervals-rating-method data. Journal of Mathematical Psychology 6: 487-496. https://doi.org/10.1016/0022-2496(69)90019-4.

Erdreich, L. S., and E. T. Lee. 1981. Use of relative operating characteristic analysis in epidemiology: A method for dealing with subjective judgment. American Journal of Epidemiology 114: 649-662. https://doi.org/10.1093/oxfordjournals.aje.a113236.
Hanley, J. A., and B. J. McNeil. 1982. The meaning and use of the area under a receiver operating characteristic (ROC) curve. Radiology 143: 29-36. https://doi.org/10.1148/radiology.143.1.7063747.

Janes, H., G. M. Longton, and M. S. Pepe. 2009. Accommodating covariates in receiver operating characteristic analysis. Stata Journal 9: 17-39.
Ma, G., and W. J. Hall. 1993. Confidence bands for the receiver operating characteristic curves. Medical Decision Making 13: 191-197. https://doi.org/10.1177/0272989X9301300304.

Pepe, M. S. 2003. The Statistical Evaluation of Medical Tests for Classification and Prediction. New York: Oxford University Press.

Pepe, M. S., G. M. Longton, and H. Janes. 2009. Estimation and comparison of receiver operating characteristic curves. Stata Journal 9: 1-16.

Working, H., and H. Hotelling. 1929. Application of the theory of error to the interpretation of trends. Journal of the American Statistical Association 24 (Suppl.): 73-85. https://doi.org/10.2307/2277011.

## Also see

[R] rocfit postestimation - Postestimation tools for rocfit
[R] roc - Receiver operating characteristic (ROC) analysis
$[R]$ rocreg - Receiver operating characteristic (ROC) regression
[U] 20 Estimation and postestimation commands

## Title

rocfit postestimation - Postestimation tools for rocfit
Postestimation commands rocplot Remarks and examples Also see

## Postestimation commands

The following command is of special interest after rocfit:

| Command | Description |
| :--- | :--- |
| rocplot | plot the fitted ROC curve and simultaneous confidence bands |

The following standard postestimation commands are also available:

| Command | Description |
| :--- | :--- |
| estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian in- <br> formation criteria (AIC, CAIC, AICc, and BIC) |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estimates | cataloging estimation results <br> etable |
| table of estimation results |  |

[^58]
## rocplot

## Description for rocplot

rocplot plots the fitted ROC curve and simultaneous confidence bands.

## Menu for rocplot

Statistics $>$ Epidemiology and related $>$ ROC analysis $>$ ROC curves after rocfit

## Syntax for rocplot

```
rocplot [, rocplot_options]
```

rocplot_options Description

Main
confband
norefline
level(\#)
Plot
plotopts(plot_options)
Fit line
lineopts(cline_options)
CI plot
ciopts(area_options)
Reference line
rlopts(cline_options)
Add plots
addplot (plot)
Y axis, X axis, Titles, Legend, Overall twoway_options
display confidence bands
suppress plotting the reference line
set confidence level; default is level(95)
affect rendition of the ROC points
affect rendition of the fitted ROC line
affect rendition of the confidence bands
affect rendition of the reference line
add other plots to the generated graph
any options other than by () documented in [G-3] twoway_options
plot_options
marker_options
marker_label_options
cline_options

Description
change look of markers (color, size, etc.)
add marker labels; change look or position change look of the line

## Options for rocplot

$\qquad$ Main
confband specifies that simultaneous confidence bands be plotted around the ROC curve.
norefline suppresses plotting the 45 -degree reference line from the graphical output of the ROC curve.
level(\#) specifies the confidence level, as a percentage, for the confidence bands. The default is level (95) or as set by set level; see [R] level.
$\qquad$ Plot
plotopts (plot_options) affects the rendition of the plotted ROC points, including the size and color of markers, whether and how the markers are labeled, and whether and how the points are connected. For the full list of available plot_options, see [G-3] marker_options, [G-3] marker_label_options, and [G-3] cline_options.
$\qquad$ Fit line
lineopts(cline_options) affects the rendition of the fitted ROC line; see [G-3] cline_options.

## $\bumpeq \widehat{\mathrm{Cl} \text { plot }}$

ciopts (area_options) affects the rendition of the confidence bands; see [G-3] area_options.
$\qquad$ $\sqrt{\text { Reference line }}$
rlopts (cline_options) affects the rendition of the reference line; see [G-3] cline_options.
$\qquad$ Add plots
addplot (plot) provides a way to add other plots to the generated graph. See [G-3] addplot_option.
$\qquad$ Y axis, X axis, Titles, Legend, Overall
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

## Remarks and examples

Remarks are presented under the following headings:
Using lincom and test
Using rocplot

## Using lincom and test

intercept, slope, and /cut\#, shown in example 1 of [R] rocfit, are equation names and not variable names, so they need to be referenced as described in Special syntaxes after multiple-equation estimation of $[\mathrm{R}]$ test. For example, instead of typing

```
. test intercept
intercept not found
r(111);
```

you should type

```
. test [intercept]_cons
( 1) [intercept]_cons = 0
    chi2( 1) = 28.48
    Prob > chi2 = 0.0000
```


## Using rocplot

## > Example 1

In example 1 of [R] rocfit, we fit a ROC curve by typing rocfit disease rating.
In the output table for our model, we are testing whether the variances of the two latent populations are equal by testing that the slope $=1$.

We plot the fitted ROC curve.
. rocplot, confband


Area under curve $=0.9113 \mathrm{se}($ area $)=0.0295$

## Also see

[R] rocfit - Parametric ROC models
[U] 20 Estimation and postestimation commands

## Title

rocreg - Receiver operating characteristic (ROC) regression

Description
Options
Acknowledgments

Quick start
Remarks and examples References

## Menu <br> Stored results

 Also seeSyntax
Methods and formulas

## Description

The rocreg command is used to perform receiver operating characteristic (ROC) analyses with rating and discrete classification data under the presence of covariates.

The two variables refvar and classvar must be numeric. The reference variable indicates the true state of the observation-such as diseased and nondiseased or normal and abnormal-and must be coded as 0 and 1. The refvar coded as 0 can also be called the control population, while the refvar coded as 1 comprises the case population. The rating or outcome of the diagnostic test or test modality is recorded in classvar, which must be ordinal, with higher values indicating higher risk.
rocreg can fit three models: a nonparametric model, a parametric probit model that uses the bootstrap for inference, and a parametric probit model fit using maximum likelihood.

## Quick start

Nonparametric estimation with bootstrap resampling
Area under the ROC curve for test classifier v1 and true state true using seed 20547
rocreg true v1, bseed(20547)
Add v2 as an additional classifier rocreg true v1 v2, bseed(20547)

Same as above, but estimate ROC value for a false-positive rate of 0.7
rocreg true v 1 v 2 , bseed(20547) roc(.7)
Covariate stratification of controls by categorical variable a using seed 121819
rocreg true v1 v2, bseed(121819) ctrlcov(a)
Linear control covariate adjustment with binary variable b and continuous variable x

```
rocreg true v1 v2, bseed(121819) ctrlcov(b x) ctrlmodel(linear)
```


## Parametric estimation

Area under the ROC curve for test classifier v1 and true state true by estimating equations using seed 200512
rocreg true v1, probit bseed(200512)
And save results to myfile.dta for use by rocregplot

```
rocreg true v1, probit bseed(200512) bsave(myfile)
```

Add v2 as a classifier and x as a control covariate in a linear control covariate-adjustment model rocreg true v1 v2, probit bseed(200512) ctrlcov(x) ctrlmodel(linear)

Also treat x as a ROC covariate
rocreg true v1 v2, probit bseed(200512) ctrlcov(x) ///
ctrlmodel(linear) roccov(x)
Estimate AUC by maximum likelihood instead of bootstrap resampling rocreg true v1, probit ml

## Menu

Statistics $>$ Epidemiology and related $>$ ROC analysis $>$ ROC regression models

## Syntax

Perform nonparametric analysis of ROC curve under covariates, using bootstrap
rocreg refvar classvar [classvars] [if] [in] [, np_options]

Perform parametric analysis of ROC curve under covariates, using bootstrap rocreg refvar classvar [classvars] [if] [in], probit [probit_options]

Perform parametric analysis of ROC curve under covariates, using maximum likelihood rocreg refvar classvar [classvars] [if] [in] [weight], probit ml [probit_ml_options]
np_options
Description
Model
auc
roc(numlist)
invroc (numlist)
pauc(numlist)
cluster (varname)
ctrlcov(varlist)
ctrlmodel(strata|linear)
pvc(empirical|normal)
tiecorrected
estimate total area under the ROC curve; the default estimate ROC for given false-positive rates
estimate false-positive rates for given ROC values
estimate partial area under the ROC curve (pAUC) up to each
false-positive rate
variable identifying resampling clusters
adjust control distribution for covariates in varlist stratify or regress on covariates; default is ctrlmodel (strata) use empirical or normal distribution percentile value estimates; default is pvc(empirical)
adjust for tied observations; not allowed with pvc (normal)
Bootstrap
nobootstrap
bseed (\#)
breps(\#)
bootcc
nobstrata
nodots
do not perform bootstrap, just output point estimates
random-number seed for bootstrap
number of bootstrap replications; default is breps (1000)
perform case-control (stratified on refvar) sampling rather than cohort sampling in bootstrap
ignore covariate stratification in bootstrap sampling
suppress bootstrap replication dots
Reporting
level(\#)
set confidence level; default is level (95)
probit_options
Description
Model

* probit
roccov(varlist)
fprpts(\#)
ctrlfprall
cluster (varname)
ctrlcov(varlist)
ctrlmodel (strata|linear)
pvc(empirical|normal)
tiecorrected

Bootstrap
nobootstrap
bseed (\#)
breps (\#)
bootcc
nobstrata
nodots
bsave (filename, . . . )
bfile(filename)
Reporting
level(\#)
fit the probit model
covariates affecting ROC curve
number of false-positive rate points to use in fitting ROC curve; default is fprpts(10)
fit ROC curve at each false-positive rate in control population variable identifying resampling clusters
adjust control distribution for covariates in varlist stratify or regress on covariates; default is ctrlmodel (strata)
use empirical or normal distribution percentile value estimates; default is pvc(empirical)
adjust for tied observations; not allowed with pvc(normal)
do not perform bootstrap, just output point estimates random-number seed for bootstrap
number of bootstrap replications; default is breps (1000)
perform case-control (stratified on refvar) sampling rather than cohort sampling in bootstrap
ignore covariate stratification in bootstrap sampling
suppress bootstrap replication dots
save bootstrap replicates from parametric estimation
use bootstrap replicates dataset for estimation replay
set confidence level; default is level (95)

[^59]probit_ml_options
Description

## Model

* probit
* ml roccov(varlist) cluster (varname) ctrlcov(varlist)

Reporting level(\#) display_options

Maximization
maximize_options
fit the probit model
fit the probit model by maximum likelihood estimation
covariates affecting ROC curve
variable identifying clusters
adjust control distribution for covariates in varlist
set confidence level; default is level (95)
control column formats, line width, and display of omitted variables
control the maximization process; seldom used
*probit and ml are required.
fweights, iweights, and pweights are allowed with maximum likelihood estimation; see [U] 11.1.6 weight.
collect is allowed; see [U] 11.1.10 Prefix commands.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

Options are presented under the following headings:
Options for nonparametric ROC estimation, using bootstrap
Options for parametric ROC estimation, using bootstrap
Options for parametric ROC estimation, using maximum likelihood

## Options for nonparametric ROC estimation, using bootstrap

$\qquad$ Model
auc estimates the total area under the ROC curve. This is the default summary statistic.
roc (numlist) estimates the ROC corresponding to each of the false-positive rates in numlist. The values of numlist must be in the range $(0,1)$.
invroc (numlist) estimates the false-positive rates corresponding to each of the ROC values in numlist. The values of numlist must be in the range $(0,1)$.
pauc (numlist) estimates the partial area under the ROC curve up to each false-positive rate in numlist.
The values of numlist must in the range ( 0,1 ].
cluster (varname) specifies the variable identifying resampling clusters.
ctrlcov(varlist) specifies the covariates to be used to adjust the control population.
ctrlmodel (strata|linear) specifies how to model the control population of classifiers on ctrlcov(). When ctrlmodel(linear) is specified, linear regression is used. The default is ctrlmodel(strata); that is, the control population of classifiers is stratified on the control variables.
pvc(empirical $\mid$ normal) determines how the percentile values of the control population will be calculated. When pvc (normal) is specified, the standard normal cumulative distribution function (CDF) is used for calculation. Specifying pvc(empirical) will use the empirical CDFs of the control population classifiers for calculation. The default is pvc(empirical).
tiecorrected adjusts the percentile values for ties. For each value of the classifier, one half the probability that the classifier equals that value under the control population is added to the percentile value. tiecorrected is not allowed with pvc(normal).

> Bootstrap
> nobootstrap specifies that bootstrap standard errors not be calculated.
> bseed (\#) specifies the random-number seed to be used in the bootstrap.
> breps(\#) sets the number of bootstrap replications. The default is breps(1000).
> bootcc performs case-control (stratified on refvar) sampling rather than cohort bootstrap sampling. nobstrata ignores covariate stratification in bootstrap sampling.
> nodots suppresses bootstrap replication dots.

Reporting
level(\#); see [R] Estimation options.

## Options for parametric ROC estimation, using bootstrap

$\qquad$
probit fits the probit model. This option is required and implies parametric estimation.
roccov(varlist) specifies the covariates that will affect the ROC curve.
fprpts(\#) sets the number of false-positive rate points to use in modeling the ROC curve. These points form an equispaced grid on $(0,1)$. The default is fprpts (10).
ctrlfprall models the ROC curve at each false-positive rate in the control population.
cluster (varname) specifies the variable identifying resampling clusters.
ctrlcov (varlist) specifies the covariates to be used to adjust the control population.
ctrlmodel (strata|linear) specifies how to model the control population of classifiers on ctrlcov(). When ctrlmodel(linear) is specified, linear regression is used. The default is ctrlmodel(strata); that is, the control population of classifiers is stratified on the control variables.
pvc(empirical $\mid$ normal) determines how the percentile values of the control population will be calculated. When pvc (normal) is specified, the standard normal CDF is used for calculation. Specifying pvc (empirical) will use the empirical CDFs of the control population classifiers for calculation. The default is pvc(empirical).
tiecorrected adjusts the percentile values for ties. For each value of the classifier, one half the probability that the classifier equals that value under the control population is added to the percentile value. tiecorrected is not allowed with pvc(normal).

[^60]bseed(\#) specifies the random-number seed to be used in the bootstrap.
breps(\#) sets the number of bootstrap replications. The default is breps (1000).
bootcc performs case-control (stratified on refvar) sampling rather than cohort bootstrap sampling.
nobstrata ignores covariate stratification in bootstrap sampling.
nodots suppresses bootstrap replication dots.
bsave (filename, ...) saves bootstrap replicates from parametric estimation in the given filename with specified options (that is, replace). bsave() is only allowed with parametric analysis using bootstrap.
bfile(filename) specifies to use the bootstrap replicates dataset for estimation replay. bfile() is only allowed with parametric analysis using bootstrap.

Reporting
level(\#); see [R] Estimation options.

## Options for parametric ROC estimation, using maximum likelihood

$\qquad$
probit fits the probit model. This option is required and implies parametric estimation.
ml fits the probit model by maximum likelihood estimation. This option is required and must be specified with probit.
roccov(varlist) specifies the covariates that will affect the ROC curve.
cluster (varname) specifies the variable used for clustering.
ctrlcov(varlist) specifies the covariates to be used to adjust the control population.
$\qquad$ Reporting
level (\#); see [R] Estimation options.
display_options: noomitted, cformat (\%fmt), pformat (\%fmt), sformat (\%fmt), and nolstretch; see [R] Estimation options.
$\qquad$ Maximization
maximize_options: difficult, technique (algorithm_spec), iterate(\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from(init_specs); see [R] Maximize. These options are seldom used. The technique(bhhh) option is not allowed.

## Remarks and examples

Remarks are presented under the following headings:

```
Introduction
ROC statistics
Covariate-adjusted ROC curves
Parametric ROC curves: Estimating equations
Parametric ROC curves: Maximum likelihood
```


## Introduction

Receiver operating characteristic (ROC) analysis provides a quantitative measure of the accuracy of diagnostic tests to discriminate between two states or conditions. These conditions may be referred to as normal and abnormal, nondiseased and diseased, or control and case. We will use these terms interchangeably. The discriminatory accuracy of a diagnostic test is measured by its ability to correctly classify known control and case subjects.

The analysis uses the ROC curve, a graph of the sensitivity versus 1 - specificity of the diagnostic test. The sensitivity is the fraction of positive cases that are correctly classified by the diagnostic test, whereas the specificity is the fraction of negative cases that are correctly classified. Thus, the sensitivity is the true-positive rate, and the specificity is the true-negative rate. We also call 1 - specificity the false-positive rate.

These rates are functions of the possible outcomes of the diagnostic test. At each outcome, a decision will be made by the user of the diagnostic test to classify the tested subject as either normal or abnormal. The true-positive and false-positive rates measure the probability of correct classification or incorrect classification of the subject as abnormal. Given the classification role of the diagnostic test, we will refer to it as the classifier.

Using this basic definition of the ROC curve, Pepe (2000) and Pepe (2003) describe how ROC analysis can be performed as a two-stage process. In the first stage, the control distribution of the classifier is estimated. The specificity is then determined as the percentiles of the classifier values calculated based on the control population. The false-positive rates are calculated as 1 - specificity. In the second stage, the ROC curve is estimated as the cumulative distribution of the case population's "false-positive" rates, also known as the survival function under the case population of the previously calculated percentiles. We use the terms ROC value and true-positive value interchangeably.

This formulation of ROC curve analysis provides simple, nonparametric estimates of several ROC curve summary parameters: area under the ROC curve, partial area under the ROC curve, ROC value for a given false-positive rate, and false-positive rate (also known as invROC) for a given ROC value. In the next section, we will show how to use rocreg to compute these estimates with bootstrap inference. There we will also show how rocreg complements the other nonparametric Stata ROC commands roctab and roccomp.

Other factors beyond condition status and the diagnostic test may affect both stages of ROC analysis. For example, a test center may affect the control distribution of the diagnostic test. Disease severity may affect the distribution of the standardized diagnostic test under the case population. Our analysis of the ROC curve in these situations will be more accurate if we take these covariates into account.

In a nonparametric ROC analysis, covariates may only affect the first stage of estimation; that is, they may be used to adjust the control distribution of the classifier. In a parametric ROC analysis, it is assumed that ROC follows a normal distribution, and thus covariates may enter the model at both stages; they may be used to adjust the control distribution and to model ROC as a function of these covariates and the false-positive rate. In parametric models, both sets of covariates need not be distinct but, in fact, they are often the same.

To model covariate effects on the first stage of ROC analysis, Janes and Pepe (2009) propose a covariate-adjusted ROC curve. We will demonstrate the covariate adjustment capabilities of rocreg in Covariate-adjusted ROC curves.

To account for covariate effects at the second stage, we assume a parametric model. Particularly, the ROC curve is a generalized linear model of the covariates. We will thus have a separate ROC curve for each combination of the relevant covariates. In Parametric ROC curves: Estimating equations, we show how to fit the model with estimating equations and bootstrap inference using rocreg.

This method, documented as the "pdf" approach in Alonzo and Pepe (2002), works well with weak assumptions about the control distribution.

Also in Parametric ROC curves: Estimating equations, we show how to fit a constant-only parametric model (involving no covariates) of the ROC curve with weak assumptions about the control distribution. The constant-only model capabilities of rocreg in this context will be compared with those of rocfit. roccomp has the binormal option, which will allow it to compute area under the ROC curve according to a normal ROC curve, equivalent to that obtained by rocfit. We will compare this functionality with that of rocreg.

In Parametric ROC curves: Maximum likelihood, we demonstrate maximum likelihood estimation of the ROC curve model with rocreg. There we assume a normal linear model for the classifier on the covariates and case-control status. This method is documented in Pepe (2003). We will also demonstrate how to use this method with no covariates, and we will compare rocreg under the constant-only model with rocfit and roccomp.

The rocregplot command is used repeatedly in this entry. This command provides graphical output for rocreg and is documented in [R] rocregplot.

## ROC statistics

roctab computes the ROC curve by calculating the false-positive rate and true-positive rate empirically at every value of the input classifier. It makes no distributional assumptions about the case or control distributions. We can get identical behavior from rocreg by using the default option settings.

## > Example 1: Nonparametric ROC, AUC

Hanley and McNeil (1982) presented data from a study in which a reviewer was asked to classify, using a five-point scale, a random sample of 109 tomographic images from patients with neurological problems. The rating scale was as follows: 1 is definitely normal, 2 is probably normal, 3 is questionable, 4 is probably abnormal, and 5 is definitely abnormal. The true disease status was normal for 58 of the patients and abnormal for the remaining 51 patients.

Here we list 9 of the 109 observations:

```
. use https://www.stata-press.com/data/r18/hanley
(Tomographic images)
. list disease rating in 1/9
```

| disease | rating |
| ---: | ---: |
| 1 | 5 |
| 0 | 1 |
| 1 | 5 |
| 0 | 4 |
| 0 | 1 |
| 0 | 3 |
| 1 | 5 |
| 0 | 5 |
| 0 | 1 |

For each observation, disease identifies the true disease status of the subject ( 0 is normal, 1 is abnormal), and rating contains the classification value assigned by the reviewer.

We run roctab on these data, specifying the graph option so that the ROC curve is rendered. We then calculate the false-positive and true-positive rates of the ROC curve by using rocreg. We graph the rates with rocregplot. Because we focus on rocreg output later, for now we use the quietly prefix to omit the output of rocreg. Both graphs are combined using graph combine (see [G-2] graph combine) for comparison. To ease the comparison, we specify the aspectratio(1) option in roctab; this is the default aspect ratio in rocregplot.

```
. roctab disease rating, graph aspectratio(1) name(a) nodraw title("roctab")
. quietly rocreg disease rating
. rocregplot, name(b) nodraw legend(off) title("rocreg")
. graph combine a b
```




Both roctab and rocreg compute the same false-positive rate and ROC values. The stairstep line connection style of the graph on the right emphasizes the empirical nature of its estimates. The control distribution of the classifier is estimated using the empirical CDF estimate. Similarly, the ROC curve, the distribution of the resulting case observation false-positive rate values, is estimated using the empirical CDF. Note the footnote in the roctab plot. By default, roctab will estimate the area
under the ROC curve (AUC) using a trapezoidal approximation to the estimated false-positive rate and true-positive rate points.

The AUC can be interpreted as the probability that a randomly selected member of the case population will have a larger classifier value than a randomly selected member of the control population. It can also be viewed as the average ROC value, averaged uniformly over the $(0,1)$ false-positive rate domain (Pepe 2003).

The nonparametric estimator of the AUC (DeLong, DeLong, and Clarke-Pearson 1988; Hanley and Hajian-Tilaki 1997) used by rocreg is equivalent to the sample mean of the percentile values of the case observations. Thus to calculate the nonparametric AUC estimate, we only need to calculate the percentile values of the case observations with respect to the control distribution.

This estimate can differ from the trapezoidal approximation estimate. Under discrete classification data, like we have here, there may be ties between classifier values from case to control. The trapezoidal approximation uses linear interpolation between the classifier values to correct for ties. Correcting the nonparametric estimator involves adding a correction term to each observation's percentile value, which measures the probability that the classifier is equal to (instead of less than) the observation's classifier value.

The tie-corrected nonparametric estimate (trapezoidal approximation) is used when we think the true ROC curve is smooth. This means that the classifier we measure is a discretized approximation of a true latent and a continuous classifier.

We now recompute the ROC curve of rating for classifying disease and calculate the AUC. Specifying the tiecorrected option allows tie correction to be used in the rocreg calculation. Under nonparametric estimation, rocreg bootstraps to obtain standard errors and confidence intervals for requested statistics. We use the default 1,000 bootstrap replications to obtain confidence intervals for our parameters. This is a reasonable lower bound to the number of replications (Mooney and Duval 1993) required for estimating percentile confidence intervals. By specifying the summary option in roctab, we will obtain output showing the trapezoidal approximation of the AUC estimate, along with standard error and confidence interval estimates for the trapezoidal approximation suggested by DeLong, DeLong, and Clarke-Pearson (1988).


Nonparametric ROC estimation
Control standardization: empirical, corrected for ties
ROC method : empirical
Area under the ROC curve
Status : disease
Classifier: rating

| AUC | Observed <br> coefficient | Bias | Bootstrap <br> std. err. | [95\% conf. interval] |  |
| :---: | ---: | :---: | :---: | :---: | :---: | :---: |
|  | .8931711 | .0010376 | .0309808 | .8324498 | $.9538923(\mathrm{~N})$ |
|  |  |  |  | .8223829 | $.9475383(\mathrm{P})$ |
|  |  |  |  | .8084577 | .9435818 (BC) |

The estimates of AUC match well. The standard error from roctab is close to the bootstrap standard error calculated by rocreg. The bootstrap standard error generalizes to the more complex models that we consider later, whereas the roctab standard-error calculation does not.

The AUC can be used to compare different classifiers. It is the most popular summary statistic for comparisons (Pepe, Longton, and Janes 2009). roccomp will compute the trapezoidal approximation of the AUC and graph the ROC curves of multiple classifiers. Using the DeLong, DeLong, and ClarkePearson (1988) covariance estimates for the AUC estimate, roccomp performs a Wald test of the null hypothesis that all classifier AUC values are equal. rocreg has similar capabilities.

## > Example 2: Nonparametric ROC, AUC, multiple classifiers

Hanley and McNeil (1983) presented data from an evaluation of two computer algorithms designed to reconstruct CT images from phantoms. We will call these two algorithms modalities 1 and 2. A sample of 112 phantoms was selected; 58 phantoms were considered normal, and the remaining 54
were abnormal. Each of the two modalities was applied to each phantom, and the resulting images were rated by a reviewer using a six-point scale: 1 is definitely normal, 2 is probably normal, 3 is possibly normal, 4 is possibly abnormal, 5 is probably abnormal, and 6 is definitely abnormal. Because each modality was applied to the same sample of phantoms, the two sets of outcomes are correlated.

We list the first seven observations:

```
. use https://www.stata-press.com/data/r18/ct, clear
(Reconstruction of CT images)
. list in 1/7, sep(0)
```

| mod1 | mod2 | status |
| ---: | ---: | ---: |
| 2 | 1 | 0 |
| 5 | 5 | 1 |
| 2 | 1 | 0 |
| 2 | 3 | 0 |
| 5 | 6 | 1 |
| 2 | 2 | 0 |
| 3 | 2 | 0 |

Each observation corresponds to one phantom. The mod1 variable identifies the rating assigned for the first modality, and the mod2 variable identifies the rating assigned for the second modality. The true status of the phantoms is given by status $==0$ if they are normal and status $==1$ if they are abnormal. The observations with at least one missing rating were dropped from the analysis.

A fictitious dataset was created from this true dataset, adding a third test modality. We will use roccomp to compute the AUC statistic for each modality in these data and compare the AUC of the three modalities. We obtain the same behavior from rocreg. As before, the tiecorrected option is specified so that the AUC is calculated with the trapezoidal approximation.

```
. use https://www.stata-press.com/data/r18/ct2
(Reconstruction of CT images)
. roccomp status mod1 mod2 mod3, summary
```

|  | Obs | $\begin{array}{r} \text { ROC } \\ \text { area } \end{array}$ | Std. err. | Asymptot [95\% conf | normal <br> nterval] |
| :---: | :---: | :---: | :---: | :---: | :---: |
| mod1 | 112 | 0.8828 | 0.0317 | 0.82067 | 0.94498 |
| mod2 | 112 | 0.9302 | 0.0256 | 0.88005 | 0.98042 |
| mod3 | 112 | 0.9240 | 0.0241 | 0.87670 | 0.97132 |



HO: All classifiers have equal AUC values
Ha: At least one classifier has a different AUC value
P-value: . 0339546 Test based on bootstrap (N) assumptions.
We see that the AUC estimates are equivalent, and the standard errors are quite close as well. The $p$-value for the tests of equal AUC under rocreg leads to similar inference as the $p$-value from roccomp. The Wald test performed by rocreg uses the joint bootstrap estimate variance matrix of the three AUC estimators rather than the DeLong, DeLong, and Clarke-Pearson (1988) variance estimate used by roccomp.
roccomp is used here on potentially correlated classifiers that are recorded in wide-format data. It can also be used on long-format data to compare independent classifiers. Further details can be found in $[R]$ roccomp.

Citing the AUC's lack of clinical relevance, there is argument against using it as a key summary statistic of the ROC curve (Pepe 2003; Cook 2007). Pepe, Longton, and Janes (2009) suggest using the estimate of the ROC curve itself at a particular point, or the estimate of the false-positive rate at a given ROC value, also known as invROC.

Recall from example 1 how nonparametric rocreg graphs look, with the stairstep pattern in the ROC curve. In an ideal world, the graph would be a smooth one-to-one function, and it would be trivial to map a false-positive rate to its corresponding true-positive rate and vice versa.

However, smooth ROC curves can only be obtained by assuming a parametric model that uses linear interpolation between observed false-positive rates and between observed true-positive rates, and rocreg is certainly capable of that; see example 1 of $[R]$ rocregplot. However, under nonparametric estimation, the mapping between false-positive rates and true-positive rates is not one to one, and estimates tend to be less reliable the further you are from an observed data point. This is somewhat mitigated by using tie-corrected rates (the tiecorrected option).

When we examine continuous data, the difference between the tie-corrected estimates and the standard estimates becomes negligible, and the empirical estimate of the ROC curve becomes close to the smooth ROC curve obtained by linear interpolation. So the nonparametric ROC and invROC estimates work well.

Fixing one rate value of interest can be difficult and subjective (Pepe 2003). A compromise measure is the partial area under the ROC curve (pAUC) (McClish 1989; Thompson and Zucchini 1989). This is the integral of the ROC curve from 0 and above to a given false-positive rate (perhaps the largest clinically acceptable value). Like the AUC estimate, the nonparametric estimate of the pAUC can be written as a sample average of the case observation percentiles, but with an adjustment based on the prescribed maximum false-positive rate (Dodd and Pepe 2003). A tie correction may also be applied so that it reflects the trapezoidal approximation.

We cannot compare rocreg with roctab or roccomp on the estimation of pAUC, because pAUC is not computed by the latter two.

## > Example 3: Nonparametric ROC, other statistics

To see how rocreg estimates ROC, invROC, and pAUC, we will examine a new study. Wieand et al. (1989) examined a pancreatic cancer study with two continuous classifiers, here called y1 (CA 19-9) and y2 (CA 125). This study was also examined in Pepe, Longton, and Janes (2009). The indicator of cancer in a subject is recorded as d. The study was a case-control study, stratifying participants on disease status.

We list the first five observations:

```
. use https://research.fredhutch.org/content/dam/stripe/diagnostic-biomarkers-
> statistical-center/files/wiedat2b.dta, clear
(S. Wieand - Pancreatic cancer diagnostic marker data)
```

. list in $1 / 5$

| $y 1$ | $y 2$ | $d$ |
| ---: | ---: | ---: |
| 28 | 13.3 | no |
| 15.5 | 11.1 | no |
| 8.2 | 16.7 | no |
| 3.4 | 12.6 | no |
| 17.3 | 7.4 | no |

We will estimate the ROC curves at a large value (0.7) and a small value (0.2) of the false-positive rate. These values are specified in $\operatorname{roc}()$. The false-positive rate for ROC or sensitivity value of 0.6 will also be estimated by specifying invroc(). Percentile confidence intervals for these parameters are displayed in the graph obtained by rocregplot after rocreg. The pAUC statistic will be calculated for the false-positive rate of 0.5 , which is specified as an argument to the pauc () option. Following Pepe, Longton, and Janes (2009), we use a stratified bootstrap, sampling separately from the case
and control populations by specifying the bootcc option. This reflects the case-control nature of the study.

All four statistics can be estimated simultaneously by rocreg. For clarity, however, we will estimate each statistic with a separate call to rocreg. rocregplot is used after estimation to graph the ROC and false-positive rate estimates. The display of the individual, observation-specific false-positive rate and ROC values will be omitted in the plot. This is accomplished by specifying msymbol(i) in our plot1opts() and plot2opts() options to rocregplot.
$\begin{array}{ll}\text {. rocreg } \mathrm{d} y 1 \mathrm{y} 2, \operatorname{roc}(.7) \text { bseed(8378923) bootcc nodots } & \\ \text { Bootstrap results } & \\ \text { Number of strata }=2 & \text { Number of obs }=141 \\ & \text { Replications }=1,000\end{array}$
Nonparametric ROC estimation
Control standardization: empirical
ROC method : empirical
ROC curve
Status : d
Classifier: y1

| ROC | Observed <br> coefficient | Bias | Bootstrap <br> std. err. | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | :--- |
|  | .9222222 | .0010222 | .0332527 | .8570482 | $.9873962(\mathrm{~N})$ |
|  |  |  |  | .8555555 | $.9777778(\mathrm{P})$ |
|  |  |  |  | .8555555 | .9777778 (BC) |

Status : d
Classifier: y2

| ROC | Observed <br> coefficient | Bias | Bootstrap <br> std. err. | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
|  | .8888889 | -.0046556 | .0444103 | .8018463 | .9759314 |
|  |  |  |  | .7833333 | .9666666 (N) |
|  |  |  |  | .7666667 | .9555556 (BC) |

HO: All classifiers have equal ROC values
Ha: At least one classifier has a different ROC value
Test based on bootstrap (N) assumptions

| ROC | P-value |
| ---: | ---: |
|  | .7 |
|  | .5537371 |

. rocregplot, plot1opts(msymbol(i)) plot2opts(msymbol(i))


In this study, we see that classifier y1 (CA 19-9) is a uniformly better test than is classifier y2 (CA 125) until high levels of false-positive rate and sensitivity or ROC value are reached. At the high level of false-positive rate, 0.7 , the ROC value does not significantly differ between the two classifiers. This can be seen in the plot by the overlapping confidence intervals.
. rocreg d y1 y2, roc(.2) bseed(8378923) bootcc nodots
Bootstrap results
Number of strata $=2$
Number of obs = 141
Replications $=1,000$
Nonparametric ROC estimation
Control standardization: empirical
ROC method : empirical
ROC curve
Status : d
Classifier: y1

| ROC | Observed <br> coefficient | Bias | Bootstrap <br> std. err. | [95\% conf. interval] |  |  |
| ---: | ---: | ---: | ---: | ---: | ---: | :--- |
|  | .7777778 | .0020778 | .0487666 | .6821969 | .8733586 | $(\mathrm{~N})$ |
|  |  |  |  | .6777778 | $.8722222(\mathrm{P})$ |  |
|  |  |  |  | .6555555 | $.8555555(\mathrm{BC})$ |  |

Status : d
Classifier: y2

| ROC | Observed <br> coefficient | Bias | Bootstrap <br> std. err. | [95\% conf. interval] |  |  |
| ---: | ---: | :---: | :---: | :---: | :---: | :---: |
|  | .4888889 | -.0054 | .1348859 | .2245173 | .7532605 | (N) |
|  |  |  |  | .2222222 | .6944444 | (P) |
|  |  |  |  | .2111111 | .6777778 | (BC) |

HO: All classifiers have equal ROC values
Ha: At least one classifier has a different ROC value
Test based on bootstrap (N) assumptions

| ROC | P-value |
| :--- | ---: |
| .2 | .0461582 |

. rocregplot, plot1opts(msymbol(i)) plot2opts(msymbol(i))


- CA 19-9
- CA 125

The sensitivity for the false-positive rate of 0.2 is found to be higher under y1 than under y2, and this difference is significant at the 0.05 level. In the plot, this is shown by the vertical confidence intervals.

```
. rocreg d y1 y2, invroc(.6) bseed(8378923) bootcc nodots
Bootstrap results
Number of strata = 2 }\quad\mathrm{ Number of obs = 141
```

Nonparametric ROC estimation
Control standardization: empirical
ROC method : empirical
False-positive rate
Status : d
Classifier: y1

| invROC | Observed coefficient | Bias | Bootstrap std. err. | [95\% conf | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: |
| . 6 | 0 | . 0149412 | . 0255885 | -. 0501525 | . 0501525 (N) |
|  |  |  |  | 0 | . 0784314 (P) |
|  |  |  |  | 0 | . 1372549 (BC) |

    Status : d
    Classifier: y2
    | invROC | Observed <br> coefficient | Bias | Bootstrap <br> std. err. | $[95 \%$ conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
|  | .254902 | .0074118 | .0729374 | .1119474 | .3978566 (N) |
|  |  |  |  | .1372549 | .4313726 (P) |
|  |  |  |  | .1176471 | .3921569 (BC) |

HO: All classifiers have equal invROC values
Ha: At least one classifier has a different invROC value
Test based on bootstrap (N) assumptions

| invROC | P-value |
| ---: | ---: |
| .6 | .0010863 |

. rocregplot, plot1opts(msymbol(i)) plot2opts(msymbol(i))


CA 19-9

We find significant evidence that false-positive rates corresponding to a sensitivity of 0.6 are different from y1 to y2. This is visually indicated by the horizontal confidence intervals, which are separated from each other.

| Number of strata $=2$ |  |  |  | $\begin{aligned} & \text { Number of obs }=141 \\ & \text { Replications }=1,000 \end{aligned}$ |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Nonparametric ROC estimation |  |  |  |  |  |  |
| Control standardization: empiricalROC method $:$ empirical |  |  |  |  |  |  |
| Partial area <br> Status <br> Classifier | $\begin{aligned} & \text { nder the ROC } \\ & \text { d } \\ & \text { y1 } \end{aligned}$ | rve |  |  |  |  |
| pAUC | Observed coefficient | Bias | Bootstrap std. err. | [95\% con | interval] |  |
| . 5 | . 3932462 | . 0011971 | . 0219031 | . 3503169 | . 4361755 | (N) |
|  |  |  |  | . 3489107 | . 4338235 | (P) |
|  |  |  |  | . 3453159 | . 4315904 |  |
| Status : d |  |  |  |  |  |  |
| Classifier: y2 |  |  |  |  |  |  |
| pAUC | Observed coefficient | Bias | Bootstrap <br> std. err. | [95\% con | interval] |  |
| . 5 | . 2496732 | . 0033901 | . 0362569 | . 1786109 | . 3207355 | (N) |
|  |  |  |  | . 1837691 | . 3224946 | (P) |
|  |  |  |  | . 1721133 | . 3108932 | (BC) |

HO: All classifiers have equal pAUC values
Ha: At least one classifier has a different pAUC value
Test based on bootstrap (N) assumptions

| pAUC | P-value |
| :--- | ---: |
| .5 | .001023 |

We also find significant evidence supporting the hypothesis that the pAUC for y1 up to a false-positive rate of 0.5 differs from the area of the same region under the ROC curve of y 2 .

## Covariate-adjusted ROC curves

When covariates affect the control distribution of the diagnostic test, thresholds for the test being classified as abnormal may be chosen that vary with the covariate values. These conditional thresholds will be more accurate than the marginal thresholds that would normally be used, because they take into account the specific distribution of the diagnostic test under the given covariate values as opposed to the marginal distribution over all covariate values.

By using these covariate-specific thresholds, we are essentially creating new classifiers for each covariate-value combination, and thus we are creating multiple ROC curves. As explained in Pepe (2003), when the case and control distributions of the covariates are the same, the marginal ROC curve will always be bound above by these covariate-specific ROC curves. So using conditional thresholds will never provide a less powerful test diagnostic in this case.

In the marginal ROC curve calculation, the classifiers are standardized to percentiles according to the control distribution, marginalized over the covariates. Thus, the ROC curve is the CDF of the standardized case observations. The covariate-adjusted ROC curve is the CDF of one minus the conditional control percentiles for the case observations, and the marginal ROC curve is the CDF of one minus the marginal control percentiles for the case observations (Pepe and Cai 2004). Thus, the standardization of classifier to false-positive rate value is conditioned on the specific covariate values under the covariate-adjusted ROC curve.

The covariate-adjusted ROC curve (Janes and Pepe 2009) at a given false-positive rate $t$ is equivalent to the expected value of the covariate-specific ROC at $t$ over all covariate combinations. When the covariates in question do not affect the case distribution of the classifier, the covariate-specific ROC will have the same value at each covariate combination. So here the covariate-adjusted ROC is equivalent to the covariate-specific ROC, regardless of covariate values.

When covariates do affect the case distribution of the classifier, users of the diagnostic test would likely want to model the covariate-specific ROC curves separately. Tools to do this can be found in the parametric modeling discussion in the following two sections. Regardless, the covariate-adjusted ROC curve can serve as a meaningful summary of covariate-adjusted accuracy.

Also note that the ROC summary statistics defined in the previous section have covariate-adjusted analogs. These analogs are estimated in a similar manner as under the marginal ROC curve (Janes, Longton, and Pepe 2009). The options for their calculation in rocreg are identical to those given in the previous section. Further details can be found in Methods and formulas.

## > Example 4: Nonparametric ROC, linear covariate adjustment

Norton et al. (2000) studied data from a neonatal audiology study on three tests to identify hearing impairment in newborns. These data were also studied in Janes, Longton, and Pepe (2009). Here we list 5 of the 5,058 observations.

```
. use https://www.stata-press.com/data/r18/nnhs, clear
(Norton - neonatal audiology data)
. list in 1/5
```

| id | ear | male | currage | d | y1 | y2 | y3 |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| B0157 | R | M | 42.42 | 0 | -3.1 | -9 | -1.5 |
| B0157 | L | M | 42.42 | 0 | -4.5 | -8.7 | -2.71 |
| B0158 | R | M | 40.14 | 1 | -3.2 | -13.2 | -2.64 |
| B0161 | L | F | 38.14 | 0 | -22.1 | -7.8 | -2.59 |
| B0167 | R | F | 37 | 0 | -10.9 | -6.6 | -1.42 |

The classifiers y1 (DPOAE 65 at 2 kHz ), y2 (TEOAE 80 at 2 kHz ), and y3 (ABR) and the hearing impairment indicator $d$ are recorded along with some relevant covariates. The infant's age is recorded in months as currage, and the infant's gender is indicated by male. Over $90 \%$ of the newborns were tested in each ear (ear), so we will cluster on infant ID (id).

Following the strategy of Janes, Longton, and Pepe (2009), we will first perform ROC analysis for the classifiers while adjusting for the covariate effects of the infant's gender and age. This is done by specifying these variables in the ctrlcov() option. We adjust using a linear regression rule, by specifying ctrlmodel (linear). This means that when a user of the diagnostic test chooses a threshold conditional on the age and gender covariates, they assume that the diagnostic test classifier has some linear dependence on age and gender and equal variance as their levels vary. Our cluster adjustment is made by specifying the cluster() option.

We will focus on the first classifier. The percentile, or specificity, values are calculated empirically by default, and thus so are the false-positive rates, ( $1-$ specificity). Also by default, the ROC curve values are empirically defined by the false-positive rates. To draw the ROC curve, we again use rocregplot.

The AUC is calculated by default. For brevity, we specify the nobootstrap option so that bootstrap sampling is not performed. The AUC point estimate will be sufficient for our purposes.


Area under the ROC curve
Status : d
Classifier: y1



Our covariate control adjustment model shows that currage has a negative effect on y1 (DPOAE 65 at 2 kHz ) under the control population. At the 0.001 significance level, we reject that its contribution to y1 is zero, and the point estimate has a negative sign. This result does not directly tell us about the effect of currage on the ROC curve of y1 as a classifier of d. None of the case observations are used in the linear regression, so information on currage for abnormal cases is not used in the model. This result does show us how to calculate false-positive rates for tests that use thresholds conditional on a child's sex and current age. We will see how currage affects the ROC curve when y1 is used as a classifier and conditional thresholds are used based on male and currage in the following section, Parametric ROC curves: Estimating equations.

## Technical note

Under this nonparametric estimation, rocreg saved the false-positive rate for each observation's y1 values in the utility variable $\quad \mathrm{fpr}-\mathrm{y} 1$. The true-positive rates are stored in the utility variable _roc_y1. For other models, say with classifier yname, these variables would be named _fpr_yname and _roc_yname. They will also be overwritten with each call of rocreg. The variables _roc_* and _fpr_* are usually for internal rocreg use only and are overwritten with each call of rocreg. They are only created for nonparametric models or parametric models that do not involve ROC covariates. In these models, covariates may only affect the first stage of estimation, the control distribution, and not the ROC curve itself. In parametric models that allow ROC covariates, different covariate values would lead to different ROC curves.

To see how the covariate-adjusted ROC curve estimate differs from the standard marginal estimate, we will reestimate the ROC curve for classifier y1 without covariate adjustment. We rename these variables before the new estimation and then draw an overlaid twoway line (see [G-2] graph twoway line) plot to compare the two.

```
. rename _fpr_y1 o_fpr_y1
. rename _roc_y1 o_roc_y1
. label variable o_roc_y1 "Covariate adjusted"
. rocreg d y1, cluster(id) nobootstrap
Nonparametric ROC estimation Number of obs = 5,058
Control standardization: empirical
ROC method : empirical
Area under the ROC curve
    Status : d
    Classifier: y1
```



```
. label variable _roc_y1 "Marginal"
. label variable o_fpr_y1 "False-positive rate for y1"
. twoway line _roc_y1 _fpr_y1, sort(_fpr_y1 _roc_y1) connect(J) ||
> line o_roc_y1 o_fpr_y1, sort(o_fpr_y1 o_roc_y1)
> connect(J) lpattern(dash) aspectratio(1) legend(cols(1))
```



Though they are close, particularly in AUC, there are clearly some points of difference between the estimates. So the covariate-adjusted ROC curve may be useful here.

In our examples thus far, we have used the empirical CDF estimator to estimate the control distribution. rocreg allows some flexibility here. The pvc (normal) option may be specified to calculate the percentile values according to a Gaussian distribution of the control.

Covariate adjustment in rocreg may also be performed with stratification instead of linear regression. Under the stratification method, the unique values of the stratified covariates each define separate parameters for the control distribution of the classifier. A user of the diagnostic test chooses a threshold based on the control distribution conditioned on the unique covariate value parameters.

We will demonstrate the use of normal percentile values and covariate stratification in our next example.

## > Example 5: Nonparametric ROC, covariate stratification

The hearing test study of Stover et al. (1996) examined the effectiveness of negative signal-to-noise ratio, nsnr, as a classifier of hearing loss. The test was administered under nine different settings, corresponding to different frequency, xf, and intensity, $x l$, combinations. Here we list 10 of the 1,848 observations.

```
. use https://www.stata-press.com/data/r18/dp, clear
(Stover - DPOAE test data)
. list in 1/10
```

| id | d | nsnr | $x f$ | $x l$ | $x d$ |
| ---: | ---: | ---: | ---: | ---: | ---: |
| 101 | 1 | 18 | 10.01 | 5.5 | 3.5 |
| 101 | 1 | 19 | 20.02 | 5.5 | 3 |
| 101 | 1 | 7.6 | 10.01 | 6 | 3.5 |
| 101 | 1 | 15 | 20.02 | 6 | 3 |
| 101 | 1 | 16 | 10.01 | 6.5 | 3.5 |
| 101 | 1 | 5.8 | 20.02 | 6.5 | 3 |
| 102 | 0 | -2.6 | 10.01 | 5.5 | . |
| 102 | 0 | -3 | 14.16 | 5.5 | . |
| 102 | 1 | 10 | 20.02 | 5.5 | 1 |
| 102 | 0 | -5.8 | 10.01 | 6 | . |

Hearing loss is represented by d. The covariate $x d$ is a measure of the degree of hearing loss. We will use this covariate in later analysis, because it only affects the case distribution of the classifier. Multiple measurements are taken for each individual, id, so we will cluster by individual.

We evaluate the effectiveness of nsnr using xf and xl as stratification covariates with rocreg; the default method of covariate adjustment.

As mentioned before, the default false-positive rate calculation method in rocreg estimates the conditional control distribution of the classifiers empirically. For comparison, we will also estimate a separate ROC curve using false-positive rates assuming the conditional control distribution is normal. This behavior is requested by specifying the pvc (normal) option. Using the rocregplot option name() to store the ROC plots and using the graph combine command, we are able to compare the Gaussian and empirical ROC curves side by side. As before, for brevity we specify the nobootstrap option to suppress bootstrap sampling.


```
. rocreg d nsnr, pvc(normal) ctrlcov(xf xl) cluster(id) nobootstrap
Nonparametric ROC estimation Number of obs = 1,848
Covariate control : stratification
Control variables : xf xl
Control standardization: normal
ROC method : empirical
Area under the ROC curve
    Status : d
    Classifier: nsnr
```


. rocregplot, title(Normal FPR) name(b) nodraw
. graph combine a b, xsize(5)


On cursory visual inspection, we see little difference between the two curves. The AUC values are close as well. So it is sensible to assume that we have Gaussian percentile values for control standardization.

## Parametric ROC curves: Estimating equations

We now assume a parametric model for covariate effects on the second stage of ROC analysis. Particularly, the ROC curve is a probit model of the covariates. We will thus have a separate ROC curve for each combination of the relevant covariates.

Under weak assumptions about the control distribution of the classifier, we can fit this model by using estimating equations as described in Alonzo and Pepe (2002). This method can be also be used without covariate effects in the second stage, assuming a parametric model for the single (constant only) ROC curve. Covariates may still affect the first stage of estimation, so we parametrically model the single covariate-adjusted ROC curve (from the previous section). The marginal ROC curve, involving no covariates in either stage of estimation, can be fit parametrically as well.

In addition to the Alonzo and Pepe (2002) explanation, further details are given in Pepe, Longton, and Janes (2009); Janes, Longton, and Pepe (2009); Pepe (2003); and Janes and Pepe (2009).

The parametric models that we consider assume that the ROC curve is a cumulative distribution function $g$ invoked with input of a linear polynomial in the corresponding quantile function invoked on the false-positive rate $u$. In this context, we assume that $g$ corresponds to a standard normal cumulative distribution function, $\Phi$. So the corresponding quantile function is $\Phi^{-1}$. The constant intercept of the polynomial may depend on covariates, but the slope term $\alpha$ (the quantile coefficient) may not.

$$
\operatorname{ROC}(u)=g\left\{\mathbf{x}^{\prime} \boldsymbol{\beta}+\alpha g^{-1}(u)\right\}
$$

The first step of the algorithm involves the choice of false-positive rates to use in the parametric fit. These are typically a set of equispaced points spanning the interval ( 0,1 ). Alonzo and Pepe (2002) examined the effect of fitting large and small sets of points, finding that relatively small sets could be used with little loss of efficiency. Alternatively, the set can be formed by using the observed false-positive rates in the data (Pepe 2003). Further details on the algorithm are provided in Methods and formulas.

Under parametric estimation, all the summary measures we defined earlier, except the AUC, are not calculated until postestimation. In models with covariates, each covariate combination would yield a different ROC curve and thus different summary parameters, so no summary parameters are initially estimated. In marginal parametric models (where there are no ROC covariates, but there are potentially control covariates), we will calculate the AUC and leave the other measures for postestimation; see $[R]$ rocreg postestimation. As with the other parameters, we bootstrap for standard errors and inference.

We will now demonstrate how rocreg performs the Alonzo and Pepe (2002) algorithm using the previous section's examples and others.

## > Example 6: Parametric ROC, linear covariate adjustment

We return to the neonatal audiology study with gender and age covariates (Norton et al. 2000), which we discussed in example 4. Janes, Longton, and Pepe (2009) suspected the current age of the infant would play a role in the case distribution of the classifier y1 (DPOAE 65 at 2 kHz ). They postulated a probit link between the ROC curve and the covariate-adjusted false-positive rates. We follow their investigation and reach similar results.

In example 4, we saw the results of adjusting for the currage and male variables in the control population for classifier $y 1$. Now, we see how currage affects the ROC curve when y1 is used with thresholds conditioned on male and currage.

We specify the covariates that should affect the ROC curve in the roccov() option. By default, rocreg will choose 10 equally spaced false-positive rates in the $(0,1)$ interval as fitting points. The fprpts() option allows the user to specify more or fewer points. We specify the bsave() option with the nnhs 2 y 1 dataset so that we can use the bootstrap resamples in postestimation.

```
. use https://www.stata-press.com/data/r18/nnhs, clear
(Norton - neonatal audiology data)
. rocreg d y1, probit ctrlcov(currage male) ctrlmodel(linear) roccov(currage)
> cluster(id) bseed(56930) bsave(nnhs2y1) nodots
Bootstrap results Number of obs = 5,056
```

Parametric ROC estimation
Covariate control : linear regression
Control variables : currage male
Control standardization: empirical
ROC method : parametric Link: probit
Status : d
Classifier: y1
Covariate control adjustment model:
Linear regression

| Number of obs | $=$ | 4,907 |
| :--- | :--- | ---: |
| $\mathrm{~F}(2,2685)$ | $=$ | 13.80 |
| Prob $>\mathrm{F}$ | $=$ | 0.0000 |
| R-squared | $=$ | 0.0081 |
| Root MSE | $=$ | 7.7515 |

(Std. err. adjusted for 2,686 clusters in id)

|  | y1 | Coefficient | Robust |  |  |  |
| ---: | ---: | :---: | :---: | :---: | ---: | ---: |
| std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |  |  |
| currage | -.2032456 | .0389032 | -5.22 | 0.000 | -.2795288 | -.1269624 |
| male | .2471744 | .2603598 | 0.95 | 0.343 | -.2633516 | .7577005 |
| _cons | -1.239484 | 1.487855 | -0.83 | 0.405 | -4.156942 | 1.677973 |

    Status : d
    Classifier: y1
    ROC Model :
    (Replications based on 2,741 clusters in id)

| y1 | Observed coefficient | Bias | Bootstrap std. err. | [95\% conf. interval] |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| _cons | -1.272505 | -. 058656 | 1.157249 | -3.540671 | . 995661 | (N) |
|  |  |  |  | -3.703316 | . 8687538 | (P) |
|  |  |  |  | -3.550433 | 1.094785 | (BC) |
| currage | . 0448228 | . 0015634 | . 0300731 | -. 0141194 | . 1037649 | ( N ) |
|  |  |  |  | -. 0107322 | . 108762 | (P) |
|  |  |  |  | -. 0156332 | . 1044122 | (BC) |
| probit |  |  |  |  |  |  |
| _cons | . 9372393 | . 0153781 | . 0739921 | . 7922176 | 1.082261 | (N) |
|  |  |  |  | . 8027433 | 1.108293 | (P) |
|  |  |  |  | . 78655 | 1.077874 | (BC) |

Note how the number of clusters-here infants-changes from the covariate control adjustment model fit to the ROC model. The control fit is limited to control cases and thus fewer infants. The ROC is fit on all the data, so the variance is adjusted for all clustering on all infants.

With a 0.05 level of statistical significance, we cannot reject the null hypothesis that currage has no effect on the ROC curve at a given false-positive rate. This is because each of our $95 \%$ bootstrap confidence intervals contains 0 . This corresponds with the finding in Janes, Longton, and Pepe (2009) where the reported $95 \%$ intervals each contained 0 . We cannot reject that the intercept parameter $\beta_{0}$, reported as _cons in the main table, is 0 at the 0.05 level either. The slope parameter $\alpha$, reported
as _cons in the probit table, is close to 1 and cannot be rejected as being 1 at the 0.05 level. Under the assumption that the ROC coefficients except $\alpha$ are 0 and that $\alpha=1$, the ROC curve at false-positive rate $u$ is equal to $u$. In other words, we cannot reject that the false-positive rate is equal to the true-positive rate, and so the test is noninformative. Further investigation of the results requires postestimation; see $[R]$ rocreg postestimation.

The fitting point set can be formed by using the observed false-positive rates (Pepe 2003). Our next example will illustrate this.

## > Example 7: Parametric ROC, covariate stratification

We return to the hearing test study of Stover et al. (1996), which we discussed in example 5. Pepe (2003) suspected that intensity, xd, would play a role in the case distribution of the negative signal-to-noise ratio (nsnr) classifier. A ROC regression was fit with covariate adjustment for $x f$ and xl with stratification, and for ROC covariates $\mathrm{xf}, \mathrm{xl}$, and xd . There is no prohibition against the same covariate being used in the first and second stages of ROC calculation. The false-positive rate fitting point set was composed of all observed false-positive rates in the control data.

We fit the model with rocreg here. Using observed false-positive rates as the fitting point set can make the dataset very large, so fitting the model is computationally intensive. We demonstrate the fitting algorithm without precise confidence intervals, focusing instead on the coefficient estimates and standard errors. We will thus perform only 50 bootstrap replications, a reasonable number to obtain accurate standard error estimates (Mooney and Duval 1993). The number of replications is specified in the breps () option.

The ROC covariates are specified in roccov(). We specify that all observed false-positive rates in the control observations be used as fitting points with the ctrlfprall option. The nobstrata option specifies that the bootstrap is not stratified. The covariate stratification in the first stage of estimation does not affect the resampling. We will return to this example in postestimation, so we save the bootstrap results in the nsnrf dataset with the bsave() option.


We obtain results similar to those reported in Pepe (2003, 159). We find that the coefficients for xl and xd differ from 0 at the 0.05 level of significance. So over certain covariate combinations, we can have a variety of informative tests using nsnr as a classifier.

As mentioned before, when there are no covariates, rocreg can still fit a parametric model for the ROC curve of a classifier by using the Alonzo and Pepe (2002) method. roccomp and rocfit can fit marginal probit models as well. We will compare the behavior of rocreg with that of roccomp and rocfit for probit models without covariates.

When the binormal option is specified, roccomp calculates the AUC for input classifiers according to the maximum likelihood algorithm of rocfit. The rocfit algorithm expects discrete classifiers but can slice continuous classifiers into discrete partitions. Further, the case and control distributions are both assumed normal. Actually, the observed classification values are taken as discrete indicators of the latent normally distributed classification values. This method is documented in Dorfman and Alf (1969).

Alonzo and Pepe (2002) compared their estimating equations probability density function method (with empirical estimation of the false-positive rates) to the maximum likelihood approach of Dorfman and Alf (1969) and found that they had similar efficiency and mean squared error. So we should expect rocfit and rocreg to give similar results when fitting a simple probit model.

## > Example 8: Parametric ROC, marginal model

We return to the Hanley and McNeil (1982) data. We will fit a probit model to the ROC curve, assuming that the rating variable is a discrete indicator of an underlying latent normal random variable in both the case and control populations of disease. We invoke rocfit with the default options. rocreg is invoked with the probit option. The percentile values are calculated empirically. Because there are fewer categories than 10, there will be fewer than 10 false-positive rates that trigger a different true-positive rate value. So for efficiency, we invoke rocreg with the ctrlfprall option.

| Binormal model of disease on rating |  |  |  | Number of obs |  | 109 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Goodness-of-f | chi2(2) = | 0.21 |  |  |  |  |
| Prob > chi2 | $=$ | 0.9006 |  |  |  |  |
| Log likelihood | $=$ | -123.64855 |  |  |  |  |
|  | Coefficient | Std. err. | z | $P>\|z\|$ | [95\% con | interval] |
| intercept | 1.656782 | 0.310456 | 5.34 | 0.000 | 1.048300 | 2.265265 |
| slope (*) | 0.713002 | 0.215882 | -1.33 | 0.184 | 0.289881 | 1.136123 |
| /cut1 | 0.169768 | 0.165307 | 1.03 | 0.304 | -0.154227 | 0.493764 |
| /cut2 | 0.463215 | 0.167235 | 2.77 | 0.006 | 0.135441 | 0.790990 |
| /cut3 | 0.766860 | 0.174808 | 4.39 | 0.000 | 0.424243 | 1.109477 |
| /cut4 | 1.797938 | 0.299581 | 6.00 | 0.000 | 1.210770 | 2.385106 |
| Index | Estimate | Indi <br> Std. err. | es frc | inorma | fit <br> [95\% conf | interval] |
| ROC area | 0.911331 | 0.029506 |  |  | 0.853501 | 0.969161 |
| delta (m) | 2.323671 | 0.502370 |  |  | 1.339044 | 3.308298 |
| d(e) | 1.934361 | 0.257187 |  |  | 1.430284 | 2.438438 |
| d(a) | 1.907771 | 0.259822 |  |  | 1.398530 | 2.417012 |

(*) z test for slope==1


We see that the intercept and slope parameter estimates are close. The intercept (_cons in the main table) is clearly nonzero. Under rocreg, the slope (_cons in the probit table) and its percentile and bias-corrected confidence intervals are close to those of rocfit. The area under the ROC curve for each of the rocreg and rocfit estimators also matches closely.

Now, we will compare the parametric fit of rocreg under the constant probit model with roccomp.

## > Example 9: Parametric ROC, marginal model, multiple classifiers

We now use the fictitious dataset generated from Hanley and McNeil (1983). To fit a probit model using roccomp, we specify the binormal option. Our specification of rocreg remains the same as before.
rocregplot is used to render the model produced by rocreg. We specify several graph options to both roccomp and rocregplot to ease comparison. When the binormal option is specified along with graph, roccomp will draw the binormal fitted lines in addition to connected line plots of the empirical false-positive and true-positive rates.

In this plot, we overlay scatterplots of the empirical false-positive rates (because percentile value calculation defaulted to pvc(empirical)) and the parametric true-positive rates.

```
. use https://www.stata-press.com/data/r18/ct2, clear
(Reconstruction of CT images)
. roccomp status mod1 mod2 mod3, summary binormal graph aspectratio(1)
> plot1opts(connect(i) msymbol(o))
> plot2opts(connect(i) msymbol(s))
> plot3opts(connect(i) msymbol(t))
> legend(label(1 "Modality 1") label(3 "Modality 2")
            label(5 "Modality 3") label(2 "Modality 1 fit")
            label(4 "Modality 2 fit") label(6 "Modality 3 fit")
            order(1 3 5 2 4 6) pos(6))
    title(roccomp) name(a) nodraw
Fitting binormal model for: mod1
Fitting binormal model for: mod2
Fitting binormal model for: mod3
                                    ROC
```



```
. rocreg status mod1 mod2 mod3, probit ctrlfprall bseed(867340912) nodots
```

Bootstrap results
Number of obs = 112 Replications $=999$

```
Parametric ROC estimation
Control standardization: empirical
ROC method : parametric Link: probit
Status : status
Classifier: mod1
ROC Model :
```

| mod1 | Observed coefficient | Bias | Bootstrap std. err. | [95\% conf. interval] |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| _cons | 1.726034 | . 164964 | . 5823832 | . 5845836 | 2.867484 | (N) |
|  |  |  |  | 1.197595 | 3.410778 |  |
|  |  |  |  | 1.154531 | 3.027969 | (BC) |
| probit |  |  |  |  |  |  |
| _cons | . 9666323 | . 1104948 | . 4635417 | . 0581071 | 1.875157 | (N) |
|  |  |  |  | . 5102274 | 2.319844 | (P) |
|  |  |  |  | . 5193889 | 2.319844 | (BC) |


| AUC | Observed coefficient | Bias | Bootstrap std. err. | [95\% conf. interval] |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | . 8927007 | . 000062 | . 0306285 | . 83267 | . 9527315 | (N) |
|  |  |  |  | . 8297837 | . 946722 | (P) |
|  |  |  |  | . 8262202 | . 9423347 | (BC) |

Status : status
Classifier: mod2
ROC Model :

| mod2 | Observed coefficient | Bias | Bootstrap std. err. | [95\% conf. interval] |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| _cons | 1.696811 | . 0760455 | . 4750493 | . 7657314 | 2.627891 | (N) |
|  |  |  |  | 1.191126 | 2.854689 |  |
|  |  |  |  | 1.205256 | 2.916377 | (BC) |
| probit |  |  |  |  |  |  |
| _cons | . 4553828 | . 0245707 | . 304156 | -. 140752 | 1.051518 | (N) |
|  |  |  |  | . 0857558 | 1.070745 | (P) |
|  |  |  |  | . 1495717 | 1.434937 | (BC) |


| AUC | Observed <br> coefficient | Bias | Bootstrap <br> std. err. | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
|  | .938734 | -.0033942 | .0268351 | .8861382 | .9913297 |
|  |  |  |  | .875 | .9774636 (N) |
|  |  |  |  | .8775983 | .9777322 (BC) |
|  |  |  |  |  |  |

Status : status
Classifier: mod3
ROC Model :


HO: All classifiers have equal AUC values
Ha: At least one classifier has a different AUC value
P-value: . 0599896 Test based on bootstrap (N) assumptions.
. rocregplot, title(rocreg) nodraw name(b)
> plot1opts(msymbol(o)) plot2opts(msymbol(s)) plot3opts(msymbol(t))

```
. graph combine a b, xsize(5)
```



We see differing true-positive rate values in the scattered points, which is expected because roccomp gives the empirical estimate and rocreg gives the parametric estimate. However, the estimated curves and areas under the ROC curve look similar. Using the Wald test based on the bootstrap covariance, rocreg rejects the null hypothesis that each test has the same AUC at the 0.1 significance level. roccomp formulates the asymptotic covariance using the rocfit estimates of AUC. Examination of its output leads to rejection of the null hypothesis that the AUCs are equal across each test at the 0.05 significance level.

## Parametric ROC curves: Maximum likelihood

The Alonzo and Pepe (2002) method of fitting a parametric model to the ROC curve is powerful because it can be generally applied, but that can be a limitation as well. Whenever we invoke the method and want anything other than point estimates of the parameters, we must perform bootstrap resampling.

An alternative is to use maximum likelihood inference to fit the ROC curve. This method can save computational time by avoiding the bootstrap.
rocreg implements maximum likelihood estimation for ROC curve analysis when both the case and control populations are normal. Particularly, the classifier is a normal linear model on certain covariates, and the covariate effect and variance of the classifier may change between the case and control populations. This model is defined in Pepe (2003, 145).

$$
y=\mathbf{z}^{\prime} \boldsymbol{\beta}_{\mathbf{0}}+D \mathbf{x}^{\prime} \boldsymbol{\beta}_{\mathbf{1}}+\sigma(D) \epsilon
$$

Our error term, $\epsilon$, is a standard normal random variable. The variable $D$ is our true status variable, being 1 for the case population observations and 0 for the control population observations. The variance function $\sigma$ is defined as

$$
\sigma(D)=\sigma_{0}(D=0)+\sigma_{1}(D=1)
$$

This provides two variance parameters in the model and does not depend on covariate values.

Suppose a covariate $x_{i}$ is present in $\mathbf{z}$ and $\mathbf{x}$. The coefficient $\beta_{1 i}$ represents the interaction effect of the $x_{i}$ and $D$. It is the extra effect that $x_{i}$ has on classifier $y$ under the case population, $D=1$, beyond the main effect $\beta_{0 i}$. These $\beta_{1}$ coefficients are directly related to the ROC curve of $y$.

Under this model, the ROC curve is derived to be

$$
\operatorname{ROC}(u)=\Phi\left[\frac{1}{\sigma_{1}}\left\{\mathbf{x}^{\prime} \boldsymbol{\beta}_{\mathbf{1}}+\sigma_{0} \Phi^{-1}(u)\right\}\right]
$$

For convenience, we reparameterize the model at this point, creating the parameters $\beta_{i}=\sigma_{1}^{-1} \beta_{1 i}$ and $\alpha=\sigma_{1}^{-1} \sigma_{0}$. We refer to $\beta_{0}$ as the constant intercept, $\mathbf{i}^{\prime}$ cons. The parameter $\alpha$ is referred to as the constant slope, s_cons.

$$
\operatorname{ROC}(u)=\Phi\left\{\mathbf{x}^{\prime} \boldsymbol{\beta}+\alpha \Phi^{-1}(u)\right\}
$$

We may interpret the final coefficients as the standardized linear effect of the ROC covariate on the classifier under the case population. The marginal effect of the covariate on the classifier in the control population is removed, and it is rescaled by the case population standard deviation of the classifier when all ROC covariate effects are removed. An appreciable effect on the classifier by a ROC covariate in this measure leads to an appreciable effect on the classifier's ROC curve by the ROC covariate.

The advantage of estimating the control coefficients $\boldsymbol{\beta}_{0}$ is similar to the gains of estimating the covariate control models in the estimating equations ROC method and nonparametric ROC estimation. This model would similarly apply when evaluating a test that is conditioned on control covariates.

Again, we note that under parametric estimation, all the summary measures we defined earlier except the AUC are not calculated until postestimation. In models with covariates, each covariate combination would yield a different ROC curve and thus different summary parameters, so no summary parameters are estimated initially. In marginal parametric models, we will calculate the AUC and leave the other measures for postestimation. There is a simple closed-form formula for the AUC under the probit model. Using this formula, the delta method can be invoked for inference on the AUC. Details on AUC estimation for probit marginal models are found in Methods and formulas.

We will demonstrate the maximum likelihood method of rocreg by revisiting the models of the previous section.

## > Example 10: Maximum likelihood ROC, single classifier

Returning to the hearing test study of Stover et al. (1996), we use a similar covariate grouping as before. The frequency xf and intensity xl are control covariates ( $\mathbf{z}$ ), while all three covariates $\mathrm{xf}, \mathrm{xl}$, and hearing loss degree xd are case covariates ( x ). In example 7, we fit this model using the Alonzo and Pepe (2002) method. Earlier we stratified on the control covariates and estimated the conditioned control distribution of nsnr empirically. Now, we assume a normal linear model for nsnr on $x f$ and $x l$ under the control population.

We fit the model by specifying the control covariates in the ctrlcov() option and the case covariates in the roccov() option. The ml option tells rocreg to perform maximum likelihood estimation.

```
. use https://www.stata-press.com/data/r18/dp, clear
(Stover - DPOAE test data)
. rocreg d nsnr, ctrlcov(xf xl) roccov(xf xl xd) probit ml cluster(id) nolog
Parametric ROC estimation Number of obs = 112
                                    Replications = 999
Covariate control : linear regression
Control variables : xf xl
Control standardization: normal
ROC method : parametric Link: probit
```

Status : d
Classifiers: nsnr
Classifier : nsnr
Covariate control adjustment model:
(Std. err. adjusted for 208 clusters in id)

|  | Coefficient | Robust std. err. | z | $P>\|z\|$ | [95\% conf | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| casecov |  |  |  |  |  |  |
| xf | . 4690907 | . 1408683 | 3.33 | 0.001 | . 192994 | . 7451874 |
| xl | -3.187785 | . 8976521 | -3.55 | 0.000 | -4.947151 | -1.42842 |
| xd | 3.042998 | . 3569756 | 8.52 | 0.000 | 2.343339 | 3.742657 |
| _cons | 23.48064 | 5.692069 | 4.13 | 0.000 | 12.32439 | 34.63689 |
| casesd |  |  |  |  |  |  |
| _cons | 7.979708 | . 354936 | 22.48 | 0.000 | 7.284047 | 8.67537 |
| ctrlcov |  |  |  |  |  |  |
| xf | -. 1447499 | . 0615286 | -2.35 | 0.019 | -. 2653438 | -. 0241561 |
| xl | -. 8631348 | . 2871976 | -3.01 | 0.003 | -1.426032 | -. 3002378 |
| _cons | 1.109477 | 1.964004 | 0.56 | 0.572 | -2.7399 | 4.958854 |
| ctrlsd |  |  |  |  |  |  |
| _cons | 7.731203 | . 3406654 | 22.69 | 0.000 | 7.063511 | 8.398894 |

    Status : d
    ROC Model :
    (Std. err. adjusted for 208 clusters in id)

|  |  | Robust |  |  |  |  |  |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: | ---: |
|  |  | Coefficient | std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| nsnr |  |  |  |  |  |  |  |
|  | i_cons | 2.942543 | .7569821 | 3.89 | 0.000 | 1.458885 | 4.426201 |
|  | xf | .0587854 | .0175654 | 3.35 | 0.001 | .024358 | .0932129 |
|  | xl | -.3994865 | .1171914 | -3.41 | 0.001 | -.6291775 | -.1697955 |
|  | xd | .381342 | .0449319 | 8.49 | 0.000 | .2932771 | .4694068 |
|  | s_cons | .9688578 | .0623476 | 15.54 | 0.000 | .8466587 | 1.091057 |

We find the results are similar to those of example 7. Frequency (xf) and intensity (xl) have a negative effect on the classifier nsnr in the control population.

The negative control effect is mitigated for xf in the case population, but the effect for xl is even more negative there. Hearing loss severity, $x d$, has a positive effect on $n s n r$ in the case population, and it is undefined in the control population.

The ROC coefficients are shown in the ROC Model table. Each are different from 0 at the 0.05 level. At this level, we also cannot conclude that the variances differ from case to control populations,
because 1 is in the $95 \%$ confidence interval for s_cons, the ratio of the case to control standard deviation parameters.

Both frequency (xf) and hearing loss severity ( xd ) make a positive contribution to the ROC curve and thus make the test more powerful. Intensity ( xl ) has a negative effect on the ROC curve and weakens the test. We previously saw in example 5 that the control distribution appears to be normal, so using maximum likelihood to fit this model is a reasonable approach.

This model was also fit in Pepe (2003, 147). Pepe used separate least-squares estimates for the case and control samples. We obtain similar results for the coefficients, but the maximum likelihood fitting yields slightly different standard deviations by considering both case and control observations concurrently. In addition, a misprint in Pepe (2003, 147) reports a coefficient of -4.91 for xl in the case population instead of -3.19 as reported by Stata.

Inference on multiple classifiers using the Alonzo and Pepe (2002) estimating equation method is performed by fitting each model separately and bootstrapping to determine the dependence of the estimates. Using the maximum likelihood method, we also fit each model separately. We use suest (see [R] suest) to estimate the joint variance-covariance of our parameter estimates.

For our models, we can view the score equation for each model as an estimating equation. The estimate that solves the estimating equation (that makes the score 0 ) is asymptotically normal with a variance matrix that can be estimated using the inverse of the squared scores. By stacking the score equations of the separate models, we can estimate the variance matrix for all the parameter estimates by using this rule. This is an informal explanation; further details can be found in $[\mathrm{R}]$ suest and in the references Rogers (1993); White (1982 and 1996).

Now, we will examine a case with multiple classification variables.
> Example 11: Maximum likelihood ROC, multiple classifiers
We return to the neonatal audiology study with gender and age covariates (Norton et al. 2000). In example 6, we fit a model with male and currage as control covariates, and currage as a ROC covariate for the classifier y1 (DPOAE 65 at 2 kHz ). We will refit this model, extending it to include the classifier y2 (TEOAE 80 at 2 kHz ).
. use https://www.stata-press.com/data/r18/nnhs
(Norton - neonatal audiology data)
. rocreg d y1 y2, probit ml ctrlcov(currage male) roccov(currage) cluster(id)
> nolog
Parametric ROC estimation Number of obs $=1,848$
Covariate control : linear regression
Control variables : currage male
Control standardization: normal
ROC method : parametric Link: probit
Status : d
Classifiers: y1 y2
Classifier : y1
Covariate control adjustment model:

|  | Coefficient | Std. err. | z | $P>\|z\|$ | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| casecov |  |  |  |  |  |  |
| currage | . 494211 | . 2126672 | 2.32 | 0.020 | . 077391 | . 9110311 |
| _cons | -15.00403 | 8.238094 | -1.82 | 0.069 | -31.1504 | 1.142338 |
| casesd |  |  |  |  |  |  |
| _cons | 8.49794 | . 4922792 | 17.26 | 0.000 | 7.533091 | 9.46279 |
| ctrlcov |  |  |  |  |  |  |
| currage | -. 2032048 | . 0323803 | -6.28 | 0.000 | -. 266669 | -. 1397406 |
| male | . 2369359 | . 2201391 | 1.08 | 0.282 | -. 1945288 | . 6684006 |
| _cons | -1.23534 | 1.252775 | -0.99 | 0.324 | -3.690734 | 1.220055 |
| ctrlsd |  |  |  |  |  |  |
| _cons | 7.749156 | . 0782225 | 99.07 | 0.000 | 7.595843 | 7.902469 |

Classifier : y2
Covariate control adjustment model:

|  | Coefficient | Std. err | z | $P>\|z\|$ | [95\% con | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| casecov |  |  |  |  |  |  |
| currage | . 5729861 | . 2422662 | 2.37 | 0.018 | . 0981532 | 1.047819 |
| _cons | -18.2597 | 9.384968 | -1.95 | 0.052 | -36.6539 | . 1344949 |
| casesd |  |  |  |  |  |  |
| _cons | 9.723858 | . 5632985 | 17.26 | 0.000 | 8.619813 | 10.8279 |
| ctrlcov |  |  |  |  |  |  |
| currage | -. 1694575 | . 0291922 | -5.80 | 0.000 | -. 2266732 | -. 1122419 |
| male | . 7122587 | . 1993805 | 3.57 | 0.000 | . 3214802 | 1.103037 |
| _cons | -5.651728 | 1.129452 | -5.00 | 0.000 | -7.865415 | -3.438042 |
| ctrlsd |  |  |  |  |  |  |
| _cons | 6.986167 | . 0705206 | 99.07 | 0.000 | 6.84795 | 7.124385 |

```
Status : d
ROC Model :
```

(Std. err. adjusted for 2,741 clusters in id)

|  |  | Robust |  |  |  |  |  |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: | ---: |
|  |  | Coefficient | std. err. | $\mathbf{z}$ | $\mathrm{P}>\mid \mathrm{zl}$ | [95\% conf. interval] |  |
| y1 |  |  |  |  |  |  |  |
|  | i_cons | -1.765608 | 1.105393 | -1.60 | 0.110 | -3.932138 | .4009225 |
|  | currage | .0581566 | .0290177 | 2.00 | 0.045 | .0012828 | .1150303 |
|  | s_cons | .9118864 | .0586884 | 15.54 | 0.000 | .7968593 | 1.026913 |
| y2 |  |  |  |  |  |  |  |
|  | i_cons | -1.877825 | .905174 | -2.07 | 0.038 | -3.651933 | -.1037167 |
|  | currage | .0589258 | .0235849 | 2.50 | 0.012 | .0127002 | .1051514 |
|  | s_cons | .7184563 | .0565517 | 12.70 | 0.000 | .607617 | .8292957 |

Both classifiers have similar results. The results for y1 show the same direction as the estimating equation results in example 6 . However, we can now reject the null hypothesis that the ROC currage coefficient is 0 at the 0.05 level.

In example 6, we could not reject that the slope parameter s_cons was 1 and that the constant intercept or ROC coefficient for current age was 0 . The resulting ROC curve implied a noninformative test using y1 as a classifier. This is not the case with our current results. As currage increases, we expect a steeper ROC curve and thus a more powerful test, for both classifiers y1 (DPOAE 65 at 2 kHz ) and y2 (TEOAE 80 at 2 kHz ).

In example 10, the clustering of observations within infant id was adjusted in the individual fit of nsnr. In our current example, the adjustment for the clustering of observations within id is performed during concurrent estimation, as opposed to during the individual classifier fits (as in example 10). This adjustment, performed by suest, is still accurate.

Now, we will fit constant probit models and compare rocreg with rocfit and roccomp with the binormal option. Our first applications of rocfit and roccomp are taken directly from examples 8 and 9. The Dorfman and Alf (1969) algorithm that rocfit works with uses discrete classifiers or uses slicing to make a classifier discrete. So we are applying the maximum likelihood method of rocreg on discrete classification data here, where it expects continuous data. We expect to see some discrepancies, but we do not find great divergence in the estimates. After revisiting examples 8 and 9 , we will fit a probit model with a continuous classifier and no covariates using rocreg, and we will compare the results with those from rocfit.

## > Example 12: Maximum likelihood ROC, marginal model

Using the Hanley and McNeil (1982) data, discussed in example 1 and in example 8, we fit a constant probit model of the classifier rating with true status disease. rocreg is invoked with the ml option and compared with rocfit.
. use https://www.stata-press.com/data/r18/hanley, clear
(Tomographic images)
. rocfit disease rating, nolog
Binormal model of disease on rating $\quad$ Number of obs $=109$
Goodness-of-fit chi2(2) = 0.21
Prob > chi2 $=0.9006$
Log likelihood $=-123.64855$

|  | Coefficient | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| intercept | 1.656782 | 0.310456 | 5.34 | 0.000 | 1.048300 | 2.265265 |
| slope (*) | 0.713002 | 0.215882 | -1.33 | 0.184 | 0.289881 | 1.136123 |
| /cut1 | 0.169768 | 0.165307 | 1.03 | 0.304 | -0.154227 | 0.493764 |
| /cut2 | 0.463215 | 0.167235 | 2.77 | 0.006 | 0.135441 | 0.790990 |
| /cut3 | 0.766860 | 0.174808 | 4.39 | 0.000 | 0.424243 | 1.109477 |
| /cut4 | 1.797938 | 0.299581 | 6.00 | 0.000 | 1.210770 | 2.385106 |
| Index | Estimate | Std. err. | es from | oinorm | fit [95\% con | interval] |
| ROC area | 0.911331 | 0.029506 |  |  | 0.853501 | 0.969161 |
| delta(m) | 2.323671 | 0.502370 |  |  | 1.339044 | 3.308298 |
| d(e) | 1.934361 | 0.257187 |  |  | 1.430284 | 2.438438 |
| d(a) | 1.907771 | 0.259822 |  |  | 1.398530 | 2.417012 |

(*) z test for slope==1
. rocreg disease rating, probit ml nolog
Binormal model of disease on rating Number of obs = 109
Log likelihood = -123.64855 $\quad$ Prob $>$ chi2 $=$.
Control standardization: normal
ROC method : parametric Link: probit
Status : disease
Classifiers: rating
Classifier : rating
Covariate control adjustment model:

|  | Coefficient | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| :--- | ---: | :--- | ---: | :--- | ---: | :--- |
| casecov _cons | 2.3357 | .2334285 | 10.01 | 0.000 | 1.878188 | 2.793211 |
| casesd _cons | 1.117131 | .1106124 | 10.10 | 0.000 | .9003344 | 1.333927 |
| ctrlcov _cons | 2.017241 | .1732589 | 11.64 | 0.000 | 1.67766 | 2.356823 |
| ctrlsd _cons | 1.319501 | .1225125 | 10.77 | 0.000 | 1.07938 | 1.559621 |

```
Status : disease
ROC Model :
```

|  | Coefficient | Std. err. | $z$ | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | :--- | :---: | :---: | :---: | :---: |
| rating |  |  |  |  |  |  |
| i_cons | 2.090802 | .2941411 | 7.11 | 0.000 | 1.514297 | 2.667308 |
| s_cons | 1.181151 | .1603263 | 7.37 | 0.000 | .8669177 | 1.495385 |
| auc | .9116494 | .0261658 | 34.84 | 0.000 | .8603654 | .9629333 |

We compare the estimates for these models:

|  | rocfit | rocreg, ml |
| :--- | :--- | :--- |
| slope | 0.7130 | 1.1812 |
| SE of slope | 0.2159 | 0.1603 |
| intercept | 1.6568 | 2.0908 |
| SE of intercept | 0.3105 | 0.2941 |
| AUC | 0.9113 | 0.9116 |
| SE of AUC | 0.0295 | 0.0262 |

We find that both the intercept and the slope are estimated as higher with the maximum likelihood method under rocreg than with rocfit. The AUC (ROC area in rocfit) is close for both commands. We find that the standard errors of each of these estimates is slightly lower under rocreg than rocfit as well.

Both rocfit and rocreg suggest that the slope parameter of the ROC curve (slope in rocfit and s_cons in rocreg) is not significantly different from 1 . Thus, we cannot reject that the classifier has the same variance in both case and control populations. There is, however, significant evidence that the intercepts (i_cons in rocreg and intercept in rocfit) differ from 0 . Because of the positive direction of the intercept estimates, the ROC curve for rating as a classifier of disease suggests that rating provides an informative test. This is also suggested by the high AUC, which is significantly different from 0.5 , that is, a flip of a coin.

## > Example 13: Maximum likelihood ROC, marginal model, multiple classifiers

We use the fictitious dataset generated from Hanley and McNeil (1983), which we previously used in example 2 and in example 9. To fit a probit model using roccomp, we specify the binormal option. We perform parametric, maximum likelihood ROC analysis using rocreg. We use rocregplot to plot the ROC curves created by rocreg.

```
. use https://www.stata-press.com/data/r18/ct2, clear
(Reconstruction of CT images)
. roccomp status mod1 mod2 mod3, summary binormal graph aspectratio(1)
> plot1opts(connect(i) msymbol(o))
> plot2opts(connect(i) msymbol(s))
> plot3opts(connect(i) msymbol(t))
> legend(label(1 "Modality 1") label(3 "Modality 2")
            label(5 "Modality 3") label(2 "Modality 1 fit")
            label(4 "Modality 2 fit") label(6 "Modality 3 fit")
            order(1 3 5 2 4 6) pos(6)) title(roccomp) name(a) nodraw
Fitting binormal model for: mod1
Fitting binormal model for: mod2
Fitting binormal model for: mod3
ROC
\begin{tabular}{lrrrrr} 
& Obs & area & Std. err. & \multicolumn{1}{c}{ [95\% conf. interval] } \\
\hline mod1 & 112 & 0.8945 & 0.0305 & 0.83482 & 0.95422 \\
mod2 & 112 & 0.9382 & 0.0264 & 0.88647 & 0.99001 \\
mod3 & 112 & 0.9376 & 0.0223 & 0.89382 & 0.98139
\end{tabular}
```

H0: area(mod1) = area(mod2) = area(mod3)
chi2 2 ) $=8.27 \quad$ Prob>chi2 $=0.0160$
. rocreg status mod1 mod2 mod3, probit ml nolog
Parametric ROC estimation Number of obs = 109
Control standardization: normal
ROC method : parametric Link: probit
Status : status
Classifiers: mod1 mod2 mod3
Classifier : mod1
Covariate control adjustment model:

|  | Coefficient | Std. err. | $z$ | P>\|z| | [95\% conf. interval] |  |
| :--- | :---: | :---: | :---: | :---: | :---: | :---: |
| casecov _cons | 2.118135 | .2165905 | 9.78 | 0.000 | 1.693626 | 2.542645 |
| casesd _cons | 1.166078 | .1122059 | 10.39 | 0.000 | .9461589 | 1.385998 |
| ctrlcov _cons | 2.344828 | .1474147 | 15.91 | 0.000 | 2.0559 | 2.633755 |
| ctrlsd _cons | 1.122677 | .1042379 | 10.77 | 0.000 | .9183746 | 1.32698 |

Classifier : mod2
Covariate control adjustment model:

|  | Coefficient | Std. err. | $z$ | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| :--- | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| casecov _cons | 2.659642 | .2072731 | 12.83 | 0.000 | 2.253395 | 3.06589 |
| casesd _cons | 1.288468 | .1239829 | 10.39 | 0.000 | 1.045466 | 1.53147 |
| ctrlcov _cons | 1.655172 | .1105379 | 14.97 | 0.000 | 1.438522 | 1.871823 |
| ctrlsd _cons | .8418313 | .0781621 | 10.77 | 0.000 | .6886365 | .9950262 |


|  | Coefficient | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| :--- | :---: | :---: | :---: | :---: | :---: | :---: |
| casecov _cons | 2.353768 | .1973549 | 11.93 | 0.000 | 1.966959 | 2.740576 |
| casesd _cons | 1.143359 | .1100198 | 10.39 | 0.000 | .9277243 | 1.358994 |
| ctrlcov _cons | 2.275862 | .1214094 | 18.75 | 0.000 | 2.037904 | 2.51382 |
| ctrlsd _cons | .9246267 | .0858494 | 10.77 | 0.000 | .7563649 | 1.092888 |


|  |  |  |  | Robust |  |  |  |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: | ---: |
|  |  | Coefficient | std. err. | z | P>\|z| | [95\% conf. interval] |  |
| mod1 |  |  |  |  |  |  |  |
|  | i_cons | 1.81646 | .3144804 | 5.78 | 0.000 | 1.20009 | 2.432831 |
|  | s_cons | .9627801 | .1364084 | 7.06 | 0.000 | .6954245 | 1.230136 |
|  | auc | .904657 | .0343518 | 26.34 | 0.000 | .8373287 | .9719853 |
| mod2 |  |  |  |  |  |  |  |
|  | i_cons | 2.064189 | .3267274 | 6.32 | 0.000 | 1.423815 | 2.704563 |
|  | s_cons | .6533582 | .1015043 | 6.44 | 0.000 | .4544135 | .8523029 |
|  | auc | .9580104 | .0219713 | 43.60 | 0.000 | .9149473 | 1.001073 |
| mod3 |  |  |  |  |  |  |  |
|  | i_cons | 2.058643 | .2890211 | 7.12 | 0.000 | 1.492172 | 2.625113 |
|  | s_cons | .8086932 | .1163628 | 6.95 | 0.000 | .5806262 | 1.03676 |
|  | auc | .9452805 | .0236266 | 40.01 | 0.000 | .8989732 | .9915877 |

HO: All classifiers have equal AUC values
Ha: At least one classifier has a different AUC value
P-value: . 0808808

```
. rocregplot, title(rocreg) nodraw name(b)
> plot1opts(msymbol(o)) plot2opts(msymbol(s)) plot3opts(msymbol(t))
. graph combine a b, xsize(5)
```



- Modality 1
- Modality 2
- Modality 3
- Modality 1 fit
- Modality 2 fit
——Modality 3 fit

- Modality 1
- Modality 2
$\Delta$ Modality 3
- Modality 1 fit
- Modality 2 fit
- Modality 3 fit

We compare the AUC estimates for these models:

|  | roccomp | rocreg, ml |
| :--- | :--- | :--- |
| mod1 | 0.8945 | 0.9047 |
| $\bmod 2$ | 0.9382 | 0.9580 |
| $\bmod 3$ | 0.9376 | 0.9453 |

Each classifier has a higher estimated AUC under rocreg than roccomp. Each curve appears to be raised and smoothed in the rocreg fit as compared with roccomp. They are different, but not drastically different. The inference on whether the curve areas are the same is similar to example 9 . We reject equality at the 0.10 level under rocreg and at the 0.05 level under roccomp.

Each intercept is significantly different from 0 at the 0.05 level and is estimated in a positive direction. Though all but classifier mod2 has 1 in their slope confidence intervals, the high intercepts suggest steep ROC curves and powerful tests.

Also note that the false-positive and true-positive rate points are calculated empirically in the roccomp graph and parametrically in rocreg. In example 9, the false-positive rates calculated by rocreg were calculated empirically, similar to roccomp. But in this example, the rates are calculated based on normal percentiles.

Now, we will generate an example to compare rocfit and rocreg under maximum likelihood estimation of a continuous classifier.

## > Example 14: Maximum likelihood ROC, graphical comparison with rocfit

We generate 500 realizations of a population under threat of disease. One quarter of the population has the disease. A classifier x is measured, which has a control distribution of $N(1,3)$ and a case distribution of $N(1+5,2)$. We will invoke rocreg with the ml option on this generated data. We specify the continuous() option for rocfit and invoke it on the data as well. The continuous() option tells rocfit how many discrete slices to partition the data into before fitting.

For comparison of the two curves, we will use the rocfit postestimation command, rocplot; see [R] rocfit postestimation. This command graphs the empirical false-positive and true-positive rates with an overlaid fit of the binormal curve estimated by rocfit. rocplot also supports an addplot() option. We use the saved variables from rocreg in this option to overlay a line plot of the rocreg fit.

```
. clear
. set seed 8675309
. set obs 500
Number of observations (_N) was 0, now 500.
. generate d = runiform() < . 25
. quietly generate double epsilon = 3*invnormal(runiform()) if d == 0
. quietly replace epsilon = 2*invnormal(runiform()) if d == 1
. quietly generate double x = 1 + d*5 + epsilon
. rocreg d x, probit ml nolog
Parametric ROC estimation Number of obs = 112
Control standardization: normal
ROC method : parametric Link: probit
Status : d
Classifiers: x
Classifier : x
Covariate control adjustment model:
\begin{tabular}{l|cccccc}
\hline & Coefficient & Std. err. & \(z\) & P>|z| & [95\% conf. interval] \\
\hline casecov _cons & 4.823931 & .2305469 & 20.92 & 0.000 & 4.372067 & 5.275795 \\
\hline casesd _cons & 1.926652 & .1204158 & 16.00 & 0.000 & 1.690642 & 2.162663 \\
\hline ctrlcov _cons & 1.14378 & .155409 & 7.36 & 0.000 & .8391841 & 1.448376 \\
\hline ctrlsd _cons & 2.99742 & .1098907 & 27.28 & 0.000 & 2.782038 & 3.212802 \\
\hline
\end{tabular}
```

    Status : d
    ROC Model :
    |  |  | Coefficient | Std. err. | $\mathbf{z}$ | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | ---: | :--- | :---: | :---: | ---: | ---: |
| x |  |  |  |  |  |  |  |
|  | i_cons | 2.503789 | .1969952 | 12.71 | 0.000 | 2.117686 | 2.889893 |
|  | s_cons | 1.555766 | .1127296 | 13.80 | 0.000 | 1.33482 | 1.776712 |
|  | auc | .912102 | .0123921 | 73.60 | 0.000 | .8878139 | .9363902 |

. rocfit $\mathrm{d} x$, continuous(10) nolog
Binormal model of d on $\mathrm{x} \quad$ Number of obs $=500$
Goodness-of-fit chi2(7) = 1.33
Prob > chi2 $=0.9877$
Log likelihood $=-914.15521$

|  | Coefficient | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :--- | ---: | :--- | ---: | ---: |
| intercept | 2.647297 | 0.277012 | 9.56 | 0.000 | 2.104362 | 3.190231 |
| slope (*) | 1.670103 | 0.195433 | 3.43 | 0.001 | 1.287062 | 2.053145 |
| /cut1 | -2.079091 | 0.153221 | -13.57 | 0.000 | -2.379398 | -1.778783 |
| /cut2 | -1.383360 | 0.093448 | -14.80 | 0.000 | -1.566515 | -1.200205 |
| /cut3 | -0.905227 | 0.075606 | -11.97 | 0.000 | -1.053413 | -0.757041 |
| /cut4 | -0.252654 | 0.065679 | -3.85 | 0.000 | -0.381382 | -0.123925 |
| /cut5 | 0.310051 | 0.065913 | 4.70 | 0.000 | 0.180863 | 0.439239 |
| /cut6 | 0.915048 | 0.072958 | 12.54 | 0.000 | 0.772054 | 1.058043 |
| /cut7 | 1.512188 | 0.092153 | 16.41 | 0.000 | 1.331570 | 1.692805 |
| /cut8 | 2.095878 | 0.136662 | 15.34 | 0.000 | 1.828026 | 2.363731 |
| /cut9 | 2.516563 | 0.181939 | 13.83 | 0.000 | 2.159970 | 2.873156 |


|  |  | Indices from binormal fit |  |  |
| ---: | ---: | :--- | ---: | ---: |
| Index | Estimate | Std. err. | [95\% conf. interval] |  |
| ROC area | 0.913079 | 0.012942 | 0.887713 | 0.938445 |
| delta(m) | 1.585110 | 0.107531 | 1.374352 | 1.795868 |
| d(e) | 1.982917 | 0.121777 | 1.744239 | 2.221596 |
| d(a) | 1.923275 | 0.115671 | 1.696565 | 2.149985 |

```
(*) z test for slope==1
. rocplot, plotopts(msymbol(i)) lineopts(lpattern(dash))
> norefline addplot(line _roc_x _fpr_x, sort(_fpr_x _roc_x)
> lpattern(solid)) aspectratio(1) legend(off)
```



We find that the curves are close. As before, the rocfit estimates are lower for the slope and intercept than under rocreg. The AUC estimates are close. Though the slope confidence interval contains 1, a high ROC intercept suggests a steep ROC curve and thus a powerful test.

## Stored results

Nonparametric rocreg stores the following in e():
Scalars

| e(N) | number of observations |
| :--- | :--- |
| e(N_strata) | number of covariate strata |
| e(N_clust) | number of clusters |
| e(level) | confidence level for bootstrap CIs |
| e(rank) | rank of e(V) |

Macros
e(cmd)
e(cmdline)
e(classvars)
e(refvar)
e(ctrlmodel)
e(ctrlcov)
e(pvc)
e(title)
e(tiecorrected)
e(nobootstrap)
e(rngstate)
e(breps)
e(bootcc)
e(nobstrata)
e(clustvar)
e(exp\#)
e(roc)
e(invroc)
e(pauc)
e(auc)
e(vce)
e(properties)
Matrices
e(b)
e(V)
e(b_bs)
e(reps)
e(bias)
e(se)
e(z0)
e(ci_normal)
e(ci_percentile)
e(ci_bc)
rocreg
command as typed
classification variable list
status variable, reference variable
covariate-adjustment specification
covariate-adjustment variables
percentile value calculation method
title in estimation output
tiecorrected, if specified
nobootstrap, if specified
random-number state used in bootstrap, if bootstrap was performed
number of bootstrap resamples, if bootstrap performed
bootcc, if specified
nobstrata, if specified
name of cluster variable
expression for the \#th statistic
false-positive rates where ROC was estimated
ROC values where false-positive rates were estimated
false-positive rates where pAUC was estimated
indicates that AUC was calculated
bootstrap
b V (or b if bootstrap not performed)
tions
e(sample) marks estimation sample
In addition to the above, the following is stored in r() :

```
Matrices
    r(table)
```

matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

Parametric, bootstrap rocreg stores the following in e():
Scalars
e(N)
e(N_strata)
e(N_clust)
e(level)
e(rank)
Macros
e(cmd)
e(cmdline)
e(classvars)
e(refvar)
e(ctrlmodel)
e(ctrlcov)
e(pvc)
e(title)
e(tiecorrected)
e(probit)
e(roccov)
e(fprpts)
e(ctrlfprall)
e(nobootstrap)
e(rngstate)
e(breps)
e(bootcc)
e(nobstrata)
e(clustvar)
e(exp\#)
e(vce)
e(properties)
e(predict)
Matrices
e(b)
e(V)
e(b_bs)
e(reps)
e(bias)
e(se)
e(z0)
e(ci_normal)
e(ci_percentile)
e(ci_bc)
number of observations
number of covariate strata
number of clusters
confidence level for bootstrap CIs
rank of $e(V)$
rocreg
command as typed
classification variable list
status variable, reference variable
covariate-adjustment specification
covariate-adjustment variables
percentile value calculation method
title in estimation output
tiecorrected, if specified
probit, if specified
ROC covariates
number of points used as false-positive rate fit points
indicates whether all observed false-positive rates were used as fit points
nobootstrap, if specified
random-number state used in bootstrap
number of bootstrap resamples
bootcc, if specified
nobstrata, if specified
name of cluster variable
expression for the \#th statistic
bootstrap
b V (or b if nobootstrap is specified)
program used to implement predict
coefficient vector
variance-covariance matrix of the estimators
bootstrap estimates
number of nonmissing results
estimated biases
estimated standard errors
median biases
normal-approximation confidence intervals
percentile confidence intervals
bias-corrected confidence intervals
Functions
e(sample) marks estimation sample
In addition to the above, the following is stored in r() :

```
Matrices
    r(table)
```

matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

Parametric, maximum likelihood rocreg stores the following in e() :
Scalars
$e(N)$ number of observations
$e$ (N_clust) number of clusters
$e$ (rank) rank of e(V)
Macros
e(cmd) rocreg
e(cmdline) command as typed
e(classvars) classification variable list
e(refvar) status variable
e(ctrlmodel) linear
e(ctrlcov)
e(roccov)
e(probit)
e(pvc)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(vce)
e(vcetype)
e(ml)
e(predict)
Matrices
$e(b)$
control population covariates
ROC covariates
probit, if specified
normal
weight type
weight expression
title in estimation output
name of cluster variable
cluster if clustering used
robust if multiple classifiers or clustering used
ml , if specified
program used to implement predict
e(V)
coefficient vector
variance-covariance matrix of the estimators
Functions
e(sample) marks estimation sample
In addition to the above, the following is stored in r() :
Matrices
$r(t a b l e) \quad$ matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

Assume that we applied a diagnostic test to each of $N_{0}$ control and $N_{1}$ case subjects. Further assume that the higher the outcome value of the diagnostic test, the higher the risk of the subject being abnormal. Let $y_{1 i}, i=1,2, \ldots, N_{1}$, and $y_{0 j}, j=1,2, \ldots, N_{0}$, be the values of the diagnostic test for the case and control subjects, respectively. The true status variable $D$ identifies an observation as case $D=1$ or control $D=0$. The CDF of the classifier $Y$ is $F$. Conditional on $D$, we write the CDF as $F_{D}$.

Methods and formulas are presented under the following headings:

ROC statistics<br>Covariate-adjusted ROC curves<br>Parametric ROC curves: Estimating equations<br>Parametric ROC curves: Maximum likelihood

## ROC statistics

We obtain these definitions and their estimates from Pepe (2003) and Pepe, Longton, and Janes (2009). The false-positive and true-positive rates at cutoff $c$ are defined as

$$
\begin{aligned}
& \operatorname{FPR}(y)=P(Y \geq y \mid D=0) \\
& \operatorname{TPR}(y)=P(Y \geq y \mid D=1)
\end{aligned}
$$

The true-positive rate, or ROC value at false-positive rate $u$, is given by

$$
\operatorname{ROC}(u)=P\left(1-F_{0}(Y) \leq u \mid D=1\right)
$$

When $Y$ is continuous, the false-positive rate can be written as

$$
\operatorname{FPR}(y)=1-F_{0}(y)
$$

The empirical CDF for the sample $z_{1}, \ldots, z_{n}$ is given by

$$
\widehat{F}(z)=\sum_{i=1}^{n} \frac{I\left(z<z_{i}\right)}{n}
$$

The empirical estimates $\widehat{\text { PPR }}$ and $\widehat{\text { ROC }}$ both use this empirical CDF estimator.
The area under the ROC curve is defined as

$$
\mathrm{AUC}=\int_{0}^{1} \operatorname{ROC}(u) d_{u}
$$

The partial area under the ROC curve for false-positive rate $a$ is defined as

$$
\operatorname{pAUC}(a)=\int_{0}^{a} \operatorname{ROC}(u) d_{u}
$$

The nonparametric estimate for the AUC is given by

$$
\widehat{\mathrm{AUC}}=\sum_{i=1}^{N_{1}} \frac{1-\widehat{\mathrm{FPR}}\left(y_{1 i}\right)}{N_{1}}
$$

The nonparametric estimate of pAUC is given by

$$
\widehat{\operatorname{pAUC}}(a)=\sum_{i=1}^{N_{1}} \frac{\max \left\{1-\widehat{\mathrm{FPR}}\left(y_{1 i}\right)-(1-a), 0\right\}}{N_{1}}
$$

For discrete classifiers, a correction term is subtracted from the false-positive rate estimate so that the $\widehat{A U C}$ and $\widehat{p A U C}$ estimates correspond with a trapezoidal approximation to the area of the ROC curve.

$$
\operatorname{FPR}^{c}(y)=1-\widehat{F}_{0}(y)-\frac{1}{2} \sum_{j=1}^{N_{0}} \frac{I\left(y=y_{0 j}\right)}{N_{0}}
$$

In the nonparametric estimation of the ROC curve, all inference is performed using the bootstrap command (see [R] bootstrap). rocreg also allows users to calculate the ROC curve and related statistics by assuming a normal control distribution. So these formulas are updated by replacing $F_{0}$ by $\Phi$ (with adjustment of the marginal mean and variance of the control distribution).

## Covariate-adjusted ROC curves

Suppose we observe covariate vector $Z$ in addition to the classifier $Y$. Let $Z_{1 i}, i=1,2, \ldots, N_{1}$, and $Z_{0 j}, j=1,2, \ldots, N_{0}$, be the values of the covariates for the case and control subjects, respectively.

The covariate-adjusted ROC curve is defined by Janes and Pepe (2009) as

$$
\operatorname{AROC}(t)=E\left\{\operatorname{ROC}\left(t \mid Z_{0}\right)\right\}
$$

It is calculated by replacing the marginal control CDF estimate, $\widehat{F}_{0}$, with the conditional control CDF estimate, $\widehat{F}_{0 Z}$. If we used a normal control CDF, then we would replace the marginal control mean and variance with the conditional control mean and variance. The formulas of the previous section can be updated for covariate-adjustment by making this substitution of the conditional CDF for the marginal CDF in the false-positive rate calculation.

Because the calculation of the ROC value is now performed based on the conditionally calculated false-positive rate, no further conditioning is made in its calculation under nonparametric estimation.
rocreg supports covariate adjustment with stratification and linear regression. Under stratification, separate parameters are estimated for the control distribution at each level of the covariates. Under linear regression, the classifier is regressed on the covariates over the control distribution, and the resulting coefficients serve as parameters for $\widehat{F}_{0 Z}$.

## Parametric ROC curves: Estimating equations

Under nonparametric estimation of the ROC curve with covariate adjustment, no further conditioning occurs in the ROC curve calculation beyond the use of covariate-adjusted false-positive rates as inputs.

Under parametric estimation of the ROC curve, we can relax this restriction. We model the ROC curve as a cumulative distribution function $g$ (standard normal $\Phi$ ) invoked with input of a linear polynomial in the corresponding quantile function (here $\Phi^{-1}$ ) invoked on the false-positive rate $u$. The constant intercept of the polynomial may depend on covariates; the slope term $\alpha$ (quantile coefficient) may not.

$$
\operatorname{ROC}(u)=g\left\{\mathbf{x}^{\prime} \boldsymbol{\beta}+\alpha g^{-1}(u)\right\}
$$

Pepe (2003) notes that having a binormal ROC $(g=\Phi)$ is equivalent to specifying that some monotone transformation of the data exists to make the case and control classifiers normally distributed. This specification applies to the marginal case and control.

Under weak assumptions about the control distribution of the classifier, we can fit this model by using estimating equations (Alonzo and Pepe 2002). The method can be used without covariate effects in the second stage, assuming a parametric model for the single ROC curve. Using the Alonzo and Pepe (2002) method, the covariate-adjusted ROC curve may be fit parametrically. The marginal ROC curve, involving no covariates in either stage of estimation, can be fit parametrically as well. In addition to the Alonzo and Pepe (2002) explanation, further details are given in Pepe, Longton, and Janes (2009); Janes, Longton, and Pepe (2009); Pepe (2003); and Janes and Pepe (2009).

The algorithm can be described as follows:

1. Estimate the false-positive rates of the classifier fpr. These may be computed in any fashion outlined so far: covariate-adjusted, empirically, etc.
2. Determine a set of $n_{p}$ false-positive rates to use as fitting points $f_{1}, \ldots, f_{n_{p}}$. These may be an equispaced grid on $(0,1)$ or the set of observed false-positive rates from part 1.
3. Expand the case observation portion of the data to include a subobservation for each fitting point. So there are now $N_{1}\left(n_{p}-1\right)$ additional observations in the data.
4. Generate a new dummy variable u . For subobservation $j, \mathrm{u}=I\left(\mathrm{fpr} \leq f_{j}\right)$.
5. Generate a new variable quant containing the quantiles of the false-positive rate fitting points. For subobservation $j$, quant $=g^{-1}\left(f_{j}\right)$.
6. Perform a binary regression (probit, $g=\Phi$ ) of fpr on the covariates x and quantile variable quant.

The coefficients of part 6 are the coefficients of the ROC model. The coefficients of the covariates coincide naturally with estimates of $\boldsymbol{\beta}$, and the $\alpha$ parameter is estimated by the coefficient on quant. Because the method is so general and makes few distributional assumptions, bootstrapping must be performed for inference. If multiple classifiers are to be fit, the algorithm is performed separately for each in each bootstrap, and the bootstrap is used to estimate covariances.

We mentioned earlier that in parametric estimation, the AUC was the only summary parameter that could be estimated initially. This is true when we fit the marginal probit model because there are no covariates in part 6 of the algorithm.

To calculate the AUC statistic under a marginal probit model, we use the formula

$$
\mathrm{AUC}=\Phi\left(\frac{\beta_{0}}{\sqrt{1+\alpha^{2}}}\right)
$$

Alternatively, the AUC for the probit model can be calculated as pAUC(1) in postestimation. Under both models, bootstrapping is performed for inference on the AUC.

## Parametric ROC curves: Maximum likelihood

rocreg supports another form of parametric ROC estimation: maximum likelihood with a normally distributed classifier. This method assumes that the classifier is a normal linear model on certain covariates, and the covariate effect and variance of the classifier may change between the case and control populations. The model is defined in Pepe (2003, 145).

$$
y=\mathbf{z}^{\prime} \boldsymbol{\beta}_{\mathbf{0}}+D \mathbf{x}^{\prime} \boldsymbol{\beta}_{\mathbf{1}}+\sigma(D) \epsilon
$$

Our error term, $\epsilon$, is a standard normal random variable. The variable $D$ is our true status variable, being 1 for the case population observations and 0 for the control population observations. The variance function $\sigma$ is defined as

$$
\sigma(D)=\sigma_{0}(D=0)+\sigma_{1}(D=1)
$$

This provides two variance parameters in the model and does not depend on covariate values.
Under this model, the ROC curve is easily derived to be

$$
\operatorname{ROC}(u)=\Phi\left[\frac{1}{\sigma_{1}}\left\{\mathbf{x}^{\prime} \boldsymbol{\beta}_{\mathbf{1}}+\sigma_{0} \Phi^{-1}(u)\right\}\right]
$$

We reparameterize the model, creating the parameters $\beta_{i}=\sigma_{1}^{-1} \beta_{1 i}$ and $\alpha=\sigma_{1}^{-1} \sigma_{0}$. We refer to $\beta_{0}$ as the constant intercept, i_cons. The parameter $\alpha$ is referred to as the constant slope, s_cons.

$$
\operatorname{ROC}(u)=\Phi\left\{\mathbf{x}^{\prime} \boldsymbol{\beta}+\alpha \Phi^{-1}(u)\right\}
$$

The original model defining the classifier $y$ leads to the following single observation likelihoods for $D=0$ and $D=1$ :

$$
\begin{gathered}
L\left(\boldsymbol{\beta}_{\mathbf{0}}, \boldsymbol{\beta}_{\mathbf{1}}, \sigma_{1}, \sigma_{0}, \mid D=0, y, \mathbf{z}, \mathbf{x}\right)=\frac{1}{\sqrt{2 \pi} \sigma_{0}} \exp \frac{-\left(y-\mathbf{z}^{\prime} \boldsymbol{\beta}_{\mathbf{0}}\right)^{2}}{2 \sigma_{0}^{2}} \\
L\left(\boldsymbol{\beta}_{\mathbf{0}}, \boldsymbol{\beta}_{\mathbf{1}}, \sigma_{1}, \sigma_{0}, \mid D=1, y, \mathbf{z}, \mathbf{x}\right)=\frac{1}{\sqrt{2 \pi} \sigma_{1}} \exp \frac{-\left(y-\mathbf{z}^{\prime} \boldsymbol{\beta}_{\mathbf{0}}-\mathbf{x}^{\prime} \boldsymbol{\beta}_{\mathbf{1}}\right)^{2}}{2 \sigma_{1}^{2}}
\end{gathered}
$$

These can be combined to yield the observation-level log likelihood:

$$
\begin{aligned}
\ln L\left(\boldsymbol{\beta}_{\mathbf{0}}, \boldsymbol{\beta}_{\mathbf{1}}, \sigma_{1}, \sigma_{0}, \mid D, y, \mathbf{z}, \mathbf{x}\right)= & -\frac{\ln 2 \pi}{2} \\
& -I(D=0)\left\{\ln \sigma_{0}+\frac{\left(y-\mathbf{z}^{\prime} \boldsymbol{\beta}_{\mathbf{0}}\right)^{2}}{2 \sigma_{0}^{2}}\right\} \\
& -I(D=1)\left\{\ln \sigma_{1}+\frac{\left(y-\mathbf{z}^{\prime} \boldsymbol{\beta}_{\mathbf{0}}-\mathbf{x}^{\prime} \boldsymbol{\beta}_{\mathbf{1}}\right)^{2}}{2 \sigma_{1}^{2}}\right\}
\end{aligned}
$$

When there are multiple classifiers, each classifier is fit separately with maximum likelihood. Then, the results are combined by stacking the scores and using the sandwich variance estimator. For more information, see [R] suest and the references White (1982); Rogers (1993); and White (1996).

## Acknowledgments

We thank Margaret S. Pepe, Holly Janes, and Gary Longton (retired) of the Fred Hutchinson Cancer Research Center for providing the inspiration for the rocreg command and for illuminating many useful datasets for its documentation.

## References

Alonzo, T. A., and M. S. Pepe. 2002. Distribution-free ROC analysis using binary regression techniques. Biostatistics 3: 421-432. https://doi.org/10.1093/biostatistics/3.3.421.

Cleves, M. A. 2002a. Comparative assessment of three common algorithms for estimating the variance of the area under the nonparametric receiver operating characteristic curve. Stata Journal 2: 280-289.
__. 2002b. From the help desk: Comparing areas under receiver operating characteristic curves from two or more probit or logit models. Stata Journal 2: 301-313.
Cook, N. R. 2007. Use and misuse of the receiver operating characteristic curve in risk prediction. Circulation 115: 928-935. https://doi.org/10.1161/CIRCULATIONAHA.106.672402.

DeLong, E. R., D. M. DeLong, and D. L. Clarke-Pearson. 1988. Comparing the areas under two or more correlated receiver operating characteristic curves: A nonparametric approach. Biometrics 44: 837-845. https://doi.org/10.2307/2531595.
Dodd, L. E., and M. S. Pepe. 2003. Partial AUC estimation and regression. Biometrics 59: 614-623. https://doi.org/10.1111/1541-0420.00071.
Dorfman, D. D., and E. Alf, Jr. 1969. Maximum-likelihood estimation of parameters of signal-detection theory and determination of confidence intervals-rating-method data. Journal of Mathematical Psychology 6: 487-496. https://doi.org/10.1016/0022-2496(69)90019-4.

Hanley, J. A., and K. O. Hajian-Tilaki. 1997. Sampling variability of nonparametric estimates of the areas under receiver operating characteristic curves: An update. Academic Radiology 4: 49-58. https://doi.org/10.1016/S1076-6332(97)80161-4.
Hanley, J. A., and B. J. McNeil. 1982. The meaning and use of the area under a receiver operating characteristic (ROC) curve. Radiology 143: 29-36. https://doi.org/10.1148/radiology.143.1.7063747.
——. 1983. A method of comparing the areas under receiver operating characteristic curves derived from the same cases. Radiology 148: 839-843. https://doi.org/10.1148/radiology.148.3.6878708.

Janes, H., G. M. Longton, and M. S. Pepe. 2009. Accommodating covariates in receiver operating characteristic analysis. Stata Journal 9: 17-39.

Janes, H., and M. S. Pepe. 2009. Adjusting for covariate effects on classification accuracy using the covariate-adjusted receiver operating characteristic curve. Biometrika 96: 371-382. https://doi.org/10.1093/biomet/asp002.

Lora, D., I. Contador, J. F. Pérez-Regadera, and A. Gómez de la Cámara. 2016. Features of the area under the receiver operating characteristic (ROC) curve. A good practice. Stata Journal 16: 185-196.

McClish, D. K. 1989. Analyzing a portion of the ROC curve. Medical Decision Making 9: 190-195. https://doi.org/10.1177/0272989X8900900307.
Mooney, C. Z., and R. D. Duval. 1993. Bootstrapping: A Nonparametric Approach to Statistical Inference. Newbury Park, CA: Sage.
Norton, S. J., M. P. Gorga, J. E. Widen, R. C. Folsom, Y. Sininger, B. Cone-Wesson, B. R. Vohr, K. Mascher, and K. Fletcher. 2000. Identification of neonatal hearing impairment: Evaluation of transient evoked otoacoustic emission, distortion product otoacoustic emission, and auditory brain stem response test performance. Ear and Hearing 21: 508-528. https://doi.org/10.1097/00003446-200010000-00013.
Pepe, M. S. 1998. Three approaches to regression analysis of receiver operating characteristic curves for continuous test results. Biometrics 54: 124-135. https://doi.org/10.2307/2534001.
. 2000. Receiver operating characteristic methodology. Journal of the American Statistical Association 95: 308-311. https://doi.org/10.2307/2669554.
. 2003. The Statistical Evaluation of Medical Tests for Classification and Prediction. New York: Oxford University Press.

Pepe, M. S., and T. Cai. 2004. The analysis of placement values for evaluating discriminatory measures. Biometrics 60: 528-535. https://doi.org/10.1111/j.0006-341X.2004.00200.x.
Pepe, M. S., G. M. Longton, and H. Janes. 2009. Estimation and comparison of receiver operating characteristic curves. Stata Journal 9: 1-16.

Rogers, W. H. 1993. sg16.4: Comparison of nbreg and glm for negative binomial. Stata Technical Bulletin 16: 7. Reprinted in Stata Technical Bulletin Reprints, vol. 3, pp. 82-84. College Station, TX: Stata Press.
Stover, L., M. P. Gorga, S. T. Neely, and D. Montoya. 1996. Toward optimizing the clinical utility of distortion product otoacoustic emission measurements. Journal of the Acoustical Society of America 100: 956-967. https://doi.org/10.1121/1.416207.

Thompson, M. L., and W. Zucchini. 1989. On the statistical analysis of ROC curves. Statistics in Medicine 8: 1277-1290. https://doi.org/10.1002/sim. 4780081011.

White, H. L., Jr. 1982. Maximum likelihood estimation of misspecified models. Econometrica 50: 1-25. https://doi.org/10.2307/1912526.
——. 1996. Estimation, Inference and Specification Analysis. Cambridge: Cambridge University Press.
Wieand, S., M. H. Gail, B. R. James, and K. L. James. 1989. A family of nonparametric statistics for comparing diagnostic markers with paired or unpaired data. Biometrika 76: 585-592. https://doi.org/10.2307/2336123.

## Also see

[R] rocreg postestimation - Postestimation tools for rocreg
$[R]$ rocregplot - Plot marginal and covariate-specific ROC curves after rocreg
[R] rocfit - Parametric ROC models
$[\mathrm{R}]$ roc - Receiver operating characteristic (ROC) analysis
[U] 20 Estimation and postestimation commands

## Title

rocreg postestimation - Postestimation tools for rocreg

| Postestimation commands | predict | estat |
| :--- | :--- | :--- |
| Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |

## Postestimation commands

The following commands are of special interest after rocreg:
Command Description
estat nproc nonparametric ROC curve estimation, keeping fit information from rocreg rocregplot plot marginal and covariate-specific ROC curves

The following standard postestimation commands are also available:

| Command | Description |
| :--- | :--- |
| estimates | cataloging estimation results <br> table of estimation results |
| etable | point estimates, standard errors, testing, and inference for linear combinations of <br> coefficients |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations <br> of coefficients |
| predict | roc curve predictions |
| test | Wald tests of simple and composite linear hypotheses |
| testnl |  |

## predict

## Description for predict

Use of predict after fitting a parametric model with rocreg allows calculation of all the ROC curve summary statistics for covariate-specific ROC curves. The ROC values for given false-positive rates, false-positive rate for given ROC values, area under the ROC curve (AUC), and partial areas under the ROC curve (pAUC) for a given false-positive rate can all be calculated.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

```
predict [type] newvar [if] [in] [, statistic options]
```


## statistic

Main
at (varname)
auc
roc
invroc
pauc
classvar(varname)

Description
input variable for statistic
total area under the ROC curve; the default
ROC values for given false-positive rates in at ()
false-positive rate for given ROC values in at ()
partial area under the ROC curve up to each false-positive rate in at ()
statistic for given classifier
options
Options
intpts(\#)
se(newvar)
ci (stubname)
level(\#)
*bfile(filename, ...)
*btype ( $n|p| b c$ )

Description
points in numeric integration of pAUC calculation predict standard errors
produce confidence intervals, stored as variables with prefix stubname and suffixes _l and _u
set confidence level; default is level (95)
load dataset containing bootstrap replicates from rocreg
produce normal-based ( n ), percentile ( p ), or bias-corrected (bc) confidence intervals; default is btype( n )

[^61]
## Options for predict

Main
at (varname) records the variable to be used as input for the above predictions. auc predicts the total area under the ROC curve defined by the covariate values in the data. This is the default statistic.
roc predicts the ROC values for false-positive rates stored in varname specified in at (). invroc predicts the false-positive rates for given ROC values stored in varname specified in at (). pauc predicts the partial area under the ROC curve up to each false-positive rate stored in varname specified in at ().
classvar(varname) performs the prediction for the specified classifier.
$\qquad$ Options
intpts(\#) specifies that \# points be used in the pAUC calculation.
se(newvar) specifies that standard errors be produced and stored in newvar.
ci(stubname) requests that confidence intervals be produced and the lower and upper bounds be stored in stubname_l and stubname_u, respectively.
level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals.
bfile (filename, ...) uses bootstrap replicates of parameters from rocreg stored in filename to estimate standard errors and confidence intervals of predictions.
btype ( $n|p| b c$ ) specifies whether to produce normal-based (n), percentile (p), or bias-corrected (bc) confidence intervals. The default is btype( n ).

## estat

## Description for estat

estat nproc allows calculation of all the ROC curve summary statistics for covariate-specific ROC curves, as well as for a nonparametric ROC estimation. Under nonparametric estimation, a single ROC curve is estimated by rocreg. Covariates can affect this estimation, but there are no separate covariate-specific ROC curves. Thus, the input arguments for estat nproc are taken in the command line rather than from the data as variable values.

## Menu for estat

Statistics $>$ Postestimation

## Syntax for estat nproc

estat nproc [, estat_nproc_options]
estat_nproc_options Description

Main
auc estimate total area under the ROC curve
roc(numlist) estimate ROC values for given false-positive rates
invroc (numlist) estimate false-positive rate for given ROC values
pauc (numlist) estimate partial area under the ROC curve up to each false-positive rate

At least one option must be specified.

## Options for estat nproc

$\qquad$ Main
auc estimates the total area under the ROC curve.
roc (numlist) estimates the ROC for each of the false-positive rates in numlist. The values in numlist must be in the range $(0,1)$.
invroc (numlist) estimates the false-positive rate for each of the ROC values in numlist. The values in numlist must be in the range $(0,1)$.
pauc (numlist) estimates the partial area under the ROC curve up to each false-positive rate in numlist. The values in numlist must be in the range $(0,1]$.

## Remarks and examples

Remarks are presented under the following headings:

## Using predict after rocreg

predict, after parametric rocreg, predicts the AUC, the ROC value, the false-positive rate (invROC), or the paUC value. The default is auc.

We begin by estimating the area under the ROC curve for each of the three age-specific ROC curves in example 1 of $[R]$ rocregplot: 30,40 , and 50 months.

## > Example 1: Parametric ROC, AUC

In example 6 of [R] rocreg, a probit ROC model was fit to audiology test data from Norton et al. (2000). The estimating equations method of Alonzo and Pepe (2002) was used to fit the model. Gender and age were covariates that affected the control distribution of the classifier y1 (DPOAE 65 at 2 kHz ). Age was a ROC covariate for the model, so we fit separate ROC curves at each age.

Following Janes, Longton, and Pepe (2009), we drew the ROC curves for ages 30, 40, and 50 months in example 1 of [R] rocregplot. Now, we use predict to estimate the AUC for the ROC curve at each of those ages.

The bootstrap dataset saved by rocreg in example 6 of $[R]$ rocreg, nnhs $2 y 1 . d t a$, is used in the bfile() option.

We will store the AUC prediction in the new variable predAUC. We specify the se() option with the new variable name seAUC to produce an estimate of the prediction's standard error. By specifying the stubname cin in ci(), we tell predict to create normal-based confidence intervals (the default) as new variables cin_l and cin_u.

```
. use https://www.stata-press.com/data/r18/nnhs
(Norton - neonatal audiology data)
. rocreg d y1, probit ctrlcov(currage male) ctrlmodel(linear) roccov(currage)
> cluster(id) bseed(56930) bsave(nnhs2y1)
    (output omitted)
. set obs 5061
Number of observations (_N) was 5,058, now 5,061.
. quietly replace currage = 30 in 5059
. quietly replace currage = 40 in 5060
. quietly replace currage = 50 in 5061
. predict predAUC in 5059/5061, auc se(seAUC) ci(cin) bfile(nnhs2y1)
. list currage predAUC seAUC cin* in 5059/5061
```

|  | currage | predAUC | seAUC | cin_l | cin_u |
| :--- | ---: | ---: | ---: | ---: | ---: |
|  | 30 | .5209999 | .076013 | .3720171 | .6699827 |
| 5060. | 40 | .6479176 | .0284218 | .592212 | .7036232 |
| 5061. | 50 | .7601378 | .0794346 | .6044489 | .9158267 |

As expected, we find the AUC to increase with age.
Essentially, we have a stored bootstrap sample of ROC covariate coefficient estimates in nnhs $2 \mathrm{y} 1 . \mathrm{dta}$. We calculate the AUC using each set of coefficient estimates, resulting in a sample of AUC estimates. Then, the bootstrap standard error and confidence intervals are calculated based on this AUC sample. Further details of the computation of the standard error and percentile confidence intervals can be found in Methods and formulas and in [R] bootstrap.

We can also produce percentile or bias-corrected confidence intervals by specifying btype(p) or btype (bc), which we now demonstrate.

```
. drop *AUC*
. predict predAUC in 5059/5061, auc se(seAUC) ci(cip) bfile(nnhs2y1) btype(p)
. list currage predAUC cip* in 5059/5061
```

|  | currage | predAUC | cip_l | cip_u |
| :--- | ---: | ---: | ---: | ---: |
| 5059. | 30 | .5209999 | .3625608 | .6615624 |
| 5060. | 40 | .6479176 | .5874361 | .702619 |
| 5061. | 50 | .7601378 | .5836248 | .8910214 |

. drop *AUC*
. predict predAUC in 5059/5061, auc se(seAUC) ci(cibc) bfile(nnhs2y1) btype(bc)
. list currage predAUC cibc* in 5059/5061

|  | currage | predAUC | cibc_l | cibc_u |
| :--- | ---: | ---: | ---: | ---: |
| 5059. | 30 | .5209999 | .3733315 | .6669934 |
| 5060. | 40 | .6479176 | .5901812 | .7038136 |
| 5061. | 50 | .7601378 | .5738162 | .8882028 |

predict can also estimate the ROC value and the false-positive rate (invROC).

## > Example 2: Parametric ROC, invROC, and ROC value

In example 7 of [R] rocreg, we fit the ROC curve for status variable hearing loss (d) and classifier negative signal-to-noise ratio nsnr with ROC covariates frequency (xf), intensity ( xl ), and hearing loss severity ( xd ). The data were obtained from Stover et al. (1996). The model fit was probit with bootstrap resampling. We saved 50 bootstrap replications in the dataset nsnrf.dta.

The covariate value combinations $\mathrm{xf}=10.01$, $\mathrm{xl}=5.5$, and $\mathrm{xd}=.5$, and $\mathrm{xf}=10.01$, $\mathrm{xl}=$ 6.5 , and $x d=4$ are of interest. In example 3 of $[R]$ rocregplot, we estimated the ROC values for false-positive rates 0.2 and 0.7 and the false-positive rate for a ROC value of 0.5 by using rocregplot. We will use predict to replicate the estimation.

We begin by appending observations with our desired covariate combinations to the data. We also create two new variables: rocinp, which contains the ROC values for which we wish to predict the corresponding invROC values, and invrocinp, which contains the invROC values corresponding to the ROC values we wish to predict.

```
. clear
. input xf xl xd rocinp invrocinp
                                xf xl xd rocinp invrocinp
    1. 10.01 5.5 .5 .2 .
    2. 10.01 6.5 4 .2 .
    3. 10.01 5.5 .5 .7 .5
    4. 10.01 6.5 4 .7 . 5
    5. end
. save newdata
file newdata.dta saved
. use https://www.stata-press.com/data/r18/dp
(Stover - DPOAE test data)
. quietly rocreg d nsnr, ctrlcov(xf xl) roccov(xf xl xd) probit cluster(id)
> nobstrata ctrlfprall bseed(156385) breps(50) ctrlmodel(strata) bsave(nsnrf)
. append using newdata
. list xf xl xd invrocinp rocinp in 1849/1852
```

|  | xf | xl | xd | invroc~p | rocinp |
| :--- | ---: | ---: | ---: | ---: | ---: |
| 1849. | 10.01 | 5.5 | .5 | . | .2 |
| 1850. | 10.01 | 6.5 | 4 | . | .2 |
| 1851. | 10.01 | 5.5 | .5 | .5 | .7 |
| 1852. | 10.01 | 6.5 | 4 | .5 | .7 |

Now, we will use predict to estimate the ROC value for the false-positive rates stored in rocinp. We specify the roc option, and we specify rocinp in the at () option. The other options, se() and ci(), are used to obtain standard errors and confidence intervals, respectively. The dataset of bootstrap samples, nsnrf.dta, is specified in bfile(). After prediction, we list the point estimates and standard errors.

```
. predict rocit in 1849/1852, roc at(rocinp) se(seroc) ci(cin) bfile(nsnrf)
. list xf xl xd rocinp rocit seroc if !missing(rocit)
```

|  | xf | xl | xd | rocinp | rocit | seroc |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: |
|  | 1849. | 10.01 | 5.5 | .5 | .2 | .7652956 |
| 1850. | 10.01 | 6.5 | 4 | .2 | .9672505 | .016278785 |
| 1851. | 10.01 | 5.5 | .5 | .7 | .9835816 | .0133583 |
| 1852. | 10.01 | 6.5 | 4 | .7 | .999428 | .0007784 |

These results match example 3 of $[R]$ rocregplot. We list the confidence intervals next. These also conform to the rocregplot results from example 3 in $[R]$ rocregplot. We begin with the confidence intervals for ROC under the covariate values $\mathrm{xf}=10.01, \mathrm{xl}=5.5$, and $\mathrm{xd}=.5$.
. list $x f$ xl xd rocinp rocit cin* if inlist(_n, 1849, 1851)

|  | xf | xl | xd | rocinp | rocit | cin_1 | cin_u |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: | ---: |
|  | 1849. | 10.01 | 5.5 | .5 | .2 | .7652956 | .6429572 |
| 1851. | 10.01 | 5.5 | .5 | .7 | .9835816 | .9573998 | 1.0097634 |

Now, we list the ROC confidence intervals under the covariate values $\mathrm{xf}=10.01, \mathrm{xl}=6.5$, and $x d=4$.
. list xf xl xd rocinp rocit cin* if inlist(_n, 1850, 1852)

|  | xf | xl | xd | rocinp | rocit | cin_l | cin_u |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: | ---: |
|  | 1850. | 10.01 | 6.5 | 4 | .2 | .9672505 | .9353452 |
| 1852. | 10.01 | 6.5 | 4 | .7 | .999428 | .9979024 | 1.000954 |

Now, we will predict the false-positive rate for a ROC value by specifying the invroc option. We pass the invrocinp variable as an argument to the at () option. Again, we list the point estimates and standard errors first.

```
. drop ci*
. predict invrocit in 1849/1852, invroc at(invrocinp) se(serocinv) ci(cin)
> bfile(nsnrf)
. list xf xl xd invrocinp invrocit serocinv if !missing(invrocit)
```

|  | xf | xl | xd | invroc~p | invrocit | serocinv |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: |
|  | 10.01 | 5.5 | .5 | .5 | .0615144 | .0209516 |
| 1852. | 10.01 | 6.5 | 4 | .5 | .0043298 | .003835 |

These also match those of example 3 of $[\mathrm{R}]$ rocregplot. Listing the confidence intervals shows identical results as well. First, we list the confidence intervals under the covariate values $\mathrm{xf}=10.01$, $x l=5.5$, and $x d=.5$.
. list xf xl xd invrocinp invrocit cin* in 1851
1851.

| xf | xl | xd | invroc $\sim p$ | invrocit | cin_l | cin_u |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| 10.01 | 5.5 | .5 | .5 | .0615144 | .0204499 | .1025789 |

Now, we list the confidence intervals for false-positive rate under the covariate values $\mathrm{xf}=10.01$, $\mathrm{xl}=6.5$, and $\mathrm{xd}=4$.
. list xf xl xd invrocinp invrocit cin* in 1852
1852.

| xf | xl | xd | invroc $\sim$ | invrocit | cin_l | cin_u |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| 10.01 | 6.5 | 4 | .5 | .0043298 | -.0031867 | .0118463 |

The predict command can also be used after a maximum-likelihood ROC model is fit.

## > Example 3: Maximum likelihood ROC, invROC, and ROC value

In the previous example, we revisited the estimating equations fit of a probit model with ROC covariates frequency (xf), intensity (xl), and hearing loss severity (xd) to the Stover et al. (1996) audiology study data. A maximum likelihood fit of the same model was performed in example 10 of $[R]$ rocreg. In example 2 of $[R]$ rocregplot, we used rocregplot to estimate ROC values and false-positive rates for this model under two covariate configurations. We will use predict to obtain the same estimates. We will also estimate the partial area under the ROC curve.

We append the data as in the previous example. This leads to the following four final observations in the data.

```
. use https://www.stata-press.com/data/r18/dp, clear
(Stover - DPOAE test data)
. rocreg d nsnr, probit ctrlcov(xf xl) roccov(xf xl xd) ml cluster(id)
    (output omitted)
. append using newdata
. list xf xl xd invrocinp rocinp in 1849/1852
```

|  | xf | xl | xd | invroc $\sim \mathrm{p}$ | rocinp |
| :--- | ---: | ---: | ---: | ---: | ---: |
|  | 1849. | 10.01 | 5.5 | .5 | . |
| 1850. | 10.01 | 6.5 | 4 | .2 |  |
| 1851. | 10.01 | 5.5 | .5 | .5 | .2 |
| 1852. | 10.01 | 6.5 | 4 | .5 | .7 |
|  |  |  |  |  |  |

Now, we predict the ROC value for false-positive rates of 0.2 and 0.7. Under maximum likelihood prediction, only Wald-type confidence intervals are produced. We specify a new variable name for the standard error in the se() option and a stubname for the confidence interval variables in the ci() option.

- predict rocit in 1849/1852, roc at(rocinp) se(seroc) ci(ci)
. list xf xl xd rocinp rocit seroc ci_l ci_u if !missing(rocit), noobs

| xf | xl | xd | rocinp | rocit | seroc | ci_l | ci_u |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| 10.01 | 5.5 | .5 | .2 | .7608593 | .0510501 | .660803 | .8609157 |
| 10.01 | 6.5 | 4 | .2 | .9499408 | .0179824 | .914696 | .9851856 |
| 10.01 | 5.5 | .5 | .7 | .978951 | .0097382 | .9598644 | .9980376 |
| 10.01 | 6.5 | 4 | .7 | .9985001 | .0009657 | .9966073 | 1.000393 |

These results match our estimates in example 2 of $[R]$ rocregplot. We also match example 2 of $[R]$ rocregplot when we estimate the false-positive rate for a ROC value of 0.5 .

```
. drop ci*
. predict invrocit in 1851/1852, invroc at(invrocinp) se(serocinv) ci(ci)
. list xf xl xd invrocinp invrocit serocinv ci_l ci_u if !missing(invrocit),
> noobs
```

| xf | xl | xd | invroc $\sim p$ | invrocit | serocinv | ci_l | ci_u |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| 10.01 | 5.5 | .5 | .5 | .0578036 | .0198626 | .0188736 | .0967336 |
| 10.01 | 6.5 | 4 | .5 | .0055624 | .0032645 | -.0008359 | .0119607 |

## > Example 4: Maximum likelihood ROC, pAUC, and ROC value

In example 13 of [R] rocreg, we fit a maximum-likelihood marginal probit model to each classifier of the fictitious dataset generated from Hanley and McNeil (1983). In example 5 of [R] rocregplot, rocregplot was used to draw the ROC for the mod1 and mod3 classifiers. Estimates of the ROC value and false-positive rate were also obtained with Wald-type confidence intervals.

We return to this example, this time using predict to estimate the ROC value and false-positive rate. We will also estimate the pAUC for the false-positive rates of 0.3 and 0.8 .

First, we add the input variables to the data. The variable paucinp will hold the 0.3 and 0.8 false-positive rates that we will input to pAUC. The variable invrocinp holds the ROC value of 0.8 for which we will estimate the false-positive rate. Finally, the variable rocinp holds the false-positive rates of 0.15 and 0.75 for which we will estimate the ROC value.

```
. use https://www.stata-press.com/data/r18/ct2, clear
(Reconstruction of CT images)
. rocreg status mod1 mod2 mod3, probit ml
    (output omitted)
. quietly generate paucinp = . 3 in 111
. quietly replace paucinp = . }8\mathrm{ in }11
. quietly generate invrocinp = . }8\mathrm{ in 112
. quietly generate rocinp = . }15\mathrm{ in 111
. quietly replace rocinp = . }75\mathrm{ in }11
```

Then, we estimate the ROC value for false-positive rates 0.15 and 0.75 under classifier mod1. The point estimate is stored in roc1. Wald confidence intervals and standard errors are also estimated. We find that these results match those of example 5 of $[R]$ rocregplot.

```
. predict roc1 in 111/112, classvar(mod1) roc at(rocinp) se(sr1) ci(cir1)
. list rocinp roc1 sr1 cir1* in 111/112
```

|  | rocinp | roc1 | sr1 | cir1_1 | cir1_u |
| :--- | ---: | ---: | ---: | ---: | ---: |
| 111. | .15 | .7934935 | .0801363 | .6364293 | .9505578 |
| 112. | .75 | .9931655 | .0069689 | .9795067 | 1.006824 |

Now, we perform the same estimation under the classifier mod3.

```
. predict roc3 in 111/112, classvar(mod3) roc at(roci) se(sr3) ci(cir3)
. list rocinp roc3 sr3 cir3* in 111/112
```

|  | rocinp | roc3 | sr3 | cir3_1 | cir3_u |
| :--- | ---: | ---: | ---: | ---: | ---: |
| 111. | .15 | .8888596 | .0520118 | .7869184 | .9908009 |
| 112. | .75 | .9953942 | .0043435 | .9868811 | 1.003907 |

Next, we estimate the false-positive rate for the ROC value of 0.8 . These results also match example 5 of $[R]$ rocregplot.

```
. predict invroc1 in 112, classvar(mod1) invroc at(invrocinp) se(sir1) ci(ciir1)
. list invrocinp invroc1 sir1 ciir1* in 112
```

112. 

| invroc~p | invroc1 | sir1 | ciir1_1 | ciir1_u |
| ---: | ---: | ---: | ---: | ---: |
| .8 | .1556435 | .069699 | .0190361 | .292251 |

. predict invroc3 in 112, classvar(mod3) invroc at(invrocinp) se(sir3) ci(ciir3)
. list invrocinp invroc3 sir3 ciir3* in 112
112.

| invroc~p | invroc3 | sir3 | ciir3_1 | ciir3_u |
| ---: | ---: | ---: | ---: | ---: |
| .8 | .0661719 | .045316 | -.0226458 | .1549896 |

Finally, we estimate the pAUC for false-positive rates of 0.3 and 0.8 . The point estimate is calculated by numeric integration. Wald confidence intervals are obtained with the delta method. Further details are presented in Methods and formulas.

```
. predict pauc1 in 111/112, classvar(mod1) pauc at(paucinp) se(sp1) ci(cip1)
```

. list paucinp pauc1 sp1 cip1* in 111/112

|  | paucinp | pauc1 | sp1 | cip1_1 | cip1_u |
| :--- | ---: | ---: | ---: | ---: | ---: |
| 111. | .3 | .221409 | .0240351 | .174301 | .268517 |
| 112. | .8 | .7033338 | .0334766 | .6377209 | .7689466 |

. predict pauc3 in $111 / 112$, classvar (mod3) pauc at (paucinp) se(sp3) ci(cip3)
. list paucinp pauc3 sp3 cip3* in 111/112

|  | paucinp | pauc3 | sp3 | cip3_1 | cip3_u |
| :--- | ---: | ---: | ---: | ---: | ---: |
|  | 111. | .3 | .2540215 | .0173474 | .2200213 |
| 112. | .8 | .7420408 | .0225192 | .6979041 | .7861776 |

## Using estat nproc

When you initially use rocreg to fit a nonparametric ROC curve, you can obtain bootstrap estimates of a ROC value, false-positive rate, area under the ROC curve, and partial area under the ROC curve. The estat nproc command allows the user to estimate these parameters after rocreg has originally been used.

The seed and resampling settings used by rocreg are used by estat nproc. So the results for these new statistics are identical to what they would be if they had been initially estimated in the rocreg command. These new statistics, together with those previously estimated in rocreg, are returned in $r()$.

We demonstrate with an example.

## > Example 5: Nonparametric ROC, invROC, and pAUC

In example 3 of $[\mathrm{R}]$ rocreg, we examined data from a pancreatic cancer study (Wieand et al. 1989). Two continuous classifiers, y1 (CA 19-9) and y2 (CA 125), were used for the true status variable d. In that example, we estimated various quantities including the false-positive rate for a ROC value of 0.6 and the pAUC for a false-positive rate of 0.5 . Here we replicate that estimation with a call to rocreg to estimate the former and follow that with a call to estat nproc to estimate the latter. For simplicity, we restrict estimation to classifier y1 (CA 19-9).

We start by executing rocreg, estimating the false-positive rate for a ROC value of 0.6 . This value is specified in invroc(). Case-control resampling is used by specifying the bootcc option.
. use https://research.fredhutch.org/content/dam/stripe/diagnostic-biomarkers> statistical-center/files/wiedat2b.dta, clear
(S. Wieand - Pancreatic cancer diagnostic marker data)
. rocreg d y1, invroc(.6) bseed(8378923) bootcc nodots
Bootstrap results
Number of strata $=2 \quad$ Number of obs $=141$
Replications $=1,000$
Nonparametric ROC estimation
Control standardization: empirical
ROC method : empirical
False-positive rate
Status : d
Classifier: y1

| invROC | Observed coefficient | Bias | Bootstrap std. err. | [95\% conf. interval] |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| . 6 | 0 | . 0149412 | . 0255885 | -. 0501525 | . 0501525 | ( N ) |
|  |  |  |  | 0 | . 0784314 | (P) |
|  |  |  |  | 0 | . 1372549 | (BC) |

Now, we will estimate the pAUC for the false-positive rate of 0.5 using estat nproc and the pauc() option.

```
. matrix list e(b)
symmetric e(b)[1,1]
            y1:
    invroc_1
y1 0
. estat nproc, pauc(.5)
Bootstrap results
```

Number of strata $=2 \quad$ Number of obs $=141$
Replications $=1,000$
Nonparametric ROC estimation
Control standardization: empirical
ROC method : empirical
False-positive rate
Status : d
Classifier: y1

| invROC | Observed <br> coefficient | Bias | Bootstrap <br> std. err. | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
|  | 0 | .0149412 | .0255885 | -.0501525 | .0501525 |
|  |  |  |  | 0 | .0784314 |
|  |  |  | 0 | .1372549 | $(\mathrm{P})$ |
|  |  |  |  |  |  |

Partial area under the ROC curve
Status : d
Classifier: y1


```
. matrix list r(b)
r(b)[1,2]
    y1: pauc_1
y1 0 . 39324619
. matrix list e(b)
symmetric e(b)[1,1]
            y1:
        invroc_1
y1 0
. matrix list r(V)
symmetric r(V)[2,2]
\begin{tabular}{rrr} 
& \(y 1:\) & \(y 1:\) \\
y1:invroc_1 & invroc_1 & pauc_1 \\
y1:pauc_1 & -.00065477 & \\
\end{tabular}
. matrix list e(V)
symmetric e(V)[1,1]
    y1:
        invroc_1
y1:invroc_1 .00065477
```

The advantages of using estat nproc are twofold. First, you can estimate additional parameters of interest without having to respecify the bootstrap settings you did with rocreg; instead estat nproc uses the bootstrap settings that were stored by rocreg. Second, parameters estimated with estat nproc are added to those parameters estimated by rocreg and returned in the matrices $r(b)$ (parameter estimates) and $r(V)$ (variance-covariance matrix). Thus, you can also obtain correlations between any quantities you wish to estimate.

## Stored results

estat nproc stores the following in $r()$ :
Matrices

```
    r(b) coefficient vector
    r(V) variance-covariance matrix of the estimators
    r(ci_normal) normal-approximation confidence intervals
    r(ci_percentile) percentile confidence intervals
    r(ci_bc) bias-corrected confidence intervals
```


## Methods and formulas

Details on computation of the nonparametric ROC curve and the estimation of the parametric ROC curve model coefficients can be found in [R] rocreg. Here we describe how to estimate the ROC curve summary statistics for a parametric model. The cumulative distribution function, $g$, can be the standard normal cumulative distribution function, $\Phi$.

Methods and formulas are presented under the following headings:

[^62]
## Parametric model: Summary parameter definition

Conditioning on covariates $\mathbf{x}$, we have the following ROC curve model:

$$
\operatorname{ROC}(u)=g\left\{\mathbf{x}^{\prime} \boldsymbol{\beta}+\alpha g^{-1}(u)\right\}
$$

$\mathbf{x}$ can be constant, and $\boldsymbol{\beta}=\beta_{0}$, the constant intercept.
We can solve this equation to obtain the false-positive rate value $u$ for a ROC value of $r$ :

$$
u=g\left[\left\{g^{-1}(r)-\mathbf{x}^{\prime} \boldsymbol{\beta}\right\} \alpha^{-1}\right]
$$

The partial area under the ROC curve for the false-positive rate $u$ is defined by

$$
\operatorname{pAUC}(u)=\int_{o}^{u} g\left\{\mathbf{x}^{\prime} \boldsymbol{\beta}+\alpha g^{-1}(t)\right\} d t
$$

The area under the ROC curve is defined by

$$
\mathrm{AUC}=\int_{0}^{1} g\left\{\mathbf{x}^{\prime} \boldsymbol{\beta}+\alpha g^{-1}(t)\right\} d t
$$

When $g$ is the standard normal cumulative distribution function $\Phi$, we can express the AUC as

$$
\mathrm{AUC}=\Phi\left(\frac{\mathbf{x}^{\prime} \boldsymbol{\beta}}{\sqrt{1+\alpha^{2}}}\right)
$$

## Maximum likelihood estimation

We allow maximum likelihood estimation under probit parametric models, so $g=\Phi$. The ROC value, false-positive rate, and AUC parameters all have closed-form expressions in terms of the covariate values $\mathbf{x}$, coefficient vector $\boldsymbol{\beta}$, and slope parameter $\alpha$. So to estimate these three types of summary parameters, we use the delta method (Oehlert 1992; Phillips and Park 1988). Particularly, we use the nl com command (see [R] nlcom) to implement the delta method.

To estimate the partial area under the ROC curve for false-positive rate $u$, we use numeric integration. A trapezoidal approximation is used in calculating the integrals. A numeric integral of the $\operatorname{ROC}(t)$ function conditioned on the covariate values $\mathbf{x}$, coefficient vector estimate $\widehat{\boldsymbol{\beta}}$, and slope parameter estimate $\widehat{\alpha}$ is computed over the range $t=[0, u]$. This gives us the point estimate of pAUC $(u)$.

To calculate the standard error and confidence intervals for the point estimate of pAUC $(u)$, we again use the delta method. Details on the delta method algorithm can be found in Methods and formulas of $[\mathrm{R}]$ nlcom and the earlier mentioned references.

Under maximum likelihood estimation, the coefficient estimates $\widehat{\boldsymbol{\beta}}$ and slope estimate $\widehat{\alpha}$ are asymptotically normal with variance matrix $\mathbf{V}$. For convenience, we rename the parameter vector $\left[\boldsymbol{\beta}^{\prime}, \alpha\right]$ to the $k$-parameter vector $\boldsymbol{\theta}=\left[\theta_{1}, \ldots, \theta_{k}\right]$. We will also explicitly refer to the conditioning of the ROC curve by $\boldsymbol{\theta}$ in its mention as $\operatorname{ROC}(t, \boldsymbol{\theta})$.

Under the delta method, the continuous scalar function of the estimate $\widehat{\boldsymbol{\theta}}, f(\widehat{\boldsymbol{\theta}})$ has asymptotic mean $f(\boldsymbol{\theta})$ and asymptotic covariance

$$
\widehat{\operatorname{Var}}\{f(\widehat{\boldsymbol{\theta}})\}=\mathbf{f V f}^{\prime}
$$

where $\mathbf{f}$ is the $1 \times k$ matrix of derivatives for which

$$
\mathbf{f}_{1 j}=\frac{\partial f(\boldsymbol{\theta})}{\partial \theta_{j}} \quad j=1, \ldots, k
$$

The asymptotic covariance of $f(\widehat{\boldsymbol{\theta}})$ is estimated and then used in conjunction with $f(\widehat{\boldsymbol{\theta}})$ for further inference, including Wald confidence intervals, standard errors, and hypothesis testing.

In the case of $\operatorname{pAUC}(u)$ estimation, our $f(\widehat{\boldsymbol{\theta}})$ is the aforementioned numeric integral of the ROC curve. It estimates $f(\boldsymbol{\theta})$, the true integral of the ROC curve on the $[0, u]$ range. The $\mathbf{V}$ variance matrix is estimated using the likelihood information that rocreg calculated, and the estimation is performed by rocreg itself.

The partial derivatives of $f(\boldsymbol{\theta})$ can be determined by using Leibnitz's rule (Weisstein 2011):

$$
\mathbf{f}_{1 j}=\frac{\partial}{\partial \theta_{j}} \int_{0}^{u} \operatorname{ROC}(t, \boldsymbol{\theta}) d t=\int_{0}^{u} \frac{\partial}{\partial \theta_{j}} \operatorname{ROC}(t, \boldsymbol{\theta}) d t \quad j=1, \ldots, k
$$

When $\theta_{j}$ corresponds with the slope parameter $\alpha$, we obtain the following partial derivative:

$$
\frac{\partial}{\partial \alpha} \operatorname{pAUC}(u)=\int_{0}^{u} \phi\left\{\mathbf{x}^{\prime} \boldsymbol{\beta}+\alpha \Phi^{-1}(t)\right\} \Phi^{-1}(t) d t
$$

The partial derivative of $f(\boldsymbol{\theta})[\operatorname{pAUC}(\mathrm{u})]$ for $\beta_{0}$ is the following:

$$
\frac{\partial}{\partial \beta_{0}} \operatorname{pAUC}(u)=\int_{0}^{u} \phi\left\{\mathbf{x}^{\prime} \boldsymbol{\beta}+\alpha \Phi^{-1}(t)\right\} d t
$$

For a nonintercept coefficient, we obtain the following:

$$
\frac{\partial}{\partial \beta_{i}} \operatorname{pAUC}(u)=\int_{0}^{u} x_{i} \phi\left\{\mathbf{x}^{\prime} \boldsymbol{\beta}+\alpha \Phi^{-1}(t)\right\} d t
$$

We can estimate each of these integrals by numeric integration, plugging in the estimates $\widehat{\boldsymbol{\beta}}$ and $\widehat{\alpha}$ for the parameters. This, together with the previously calculated estimate $\widehat{\mathbf{V}}$, provides an estimate of the asymptotic covariance of $f(\widehat{\boldsymbol{\theta}})=\widehat{\mathrm{pAUC}}(u)$, which allows us to perform further statistical inference on $\operatorname{pAUC}(u)$.

## Estimating equations estimation

When we fit a model using the Alonzo and Pepe (2002) estimating equations method, we use the bootstrap to perform inference on the ROC curve summary parameters. Each bootstrap sample provides a sample of the coefficient estimates $\boldsymbol{\beta}$ and the slope estimates $\alpha$. Using the formulas in Parametric model: Summary parameter definition under Methods and formulas, we can obtain an estimate of the ROC, false-positive rate, or AUC for each resample. Using numeric integration (with the trapezoidal approximation), we can also estimate the pAUC of the resample.

By making these calculations, we obtain a bootstrap sample of our summary parameter estimate. We then obtain bootstrap standard errors, normal approximation confidence intervals, percentile confidence intervals, and bias-corrected confidence intervals using this bootstrap sample. Further details can be found in $[R]$ bootstrap.

## References

Alonzo, T. A., and M. S. Pepe. 2002. Distribution-free ROC analysis using binary regression techniques. Biostatistics 3: 421-432. https://doi.org/10.1093/biostatistics/3.3.421.
Choi, B. C. K. 1998. Slopes of a receiver operating characteristic curve and likelihood ratios for a diagnostic test. American Journal of Epidemiology 148: 1127-1132. https://doi.org/10.1093/oxfordjournals.aje.a009592.
Hanley, J. A., and B. J. McNeil. 1983. A method of comparing the areas under receiver operating characteristic curves derived from the same cases. Radiology 148: 839-843. https://doi.org/10.1148/radiology.148.3.6878708.
Janes, H., G. M. Longton, and M. S. Pepe. 2009. Accommodating covariates in receiver operating characteristic analysis. Stata Journal 9: 17-39.
Norton, S. J., M. P. Gorga, J. E. Widen, R. C. Folsom, Y. Sininger, B. Cone-Wesson, B. R. Vohr, K. Mascher, and K. Fletcher. 2000. Identification of neonatal hearing impairment: Evaluation of transient evoked otoacoustic emission, distortion product otoacoustic emission, and auditory brain stem response test performance. Ear and Hearing 21: 508-528. https://doi.org/10.1097/00003446-200010000-00013.
Oehlert, G. W. 1992. A note on the delta method. American Statistician 46: 27-29. https://doi.org/10.2307/2684406.
Phillips, P. C. B., and J. Y. Park. 1988. On the formulation of Wald tests of nonlinear restrictions. Econometrica 56: 1065-1083. https://doi.org/10.2307/1911359.
Stover, L., M. P. Gorga, S. T. Neely, and D. Montoya. 1996. Toward optimizing the clinical utility of distortion product otoacoustic emission measurements. Journal of the Acoustical Society of America 100: 956-967. https://doi.org/10.1121/1.416207.

Weisstein, E. W. 2011. Leibniz integral rule. From Mathworld-A Wolfram Web Resource. https://mathworld.wolfram.com/LeibnizIntegralRule.html.
Wieand, S., M. H. Gail, B. R. James, and K. L. James. 1989. A family of nonparametric statistics for comparing diagnostic markers with paired or unpaired data. Biometrika 76: 585-592. https://doi.org/10.2307/2336123.

## Also see

$[\mathrm{R}]$ rocreg - Receiver operating characteristic (ROC) regression
[R] rocregplot - Plot marginal and covariate-specific ROC curves after rocreg
[U] 20 Estimation and postestimation commands

## Title

rocregplot - Plot marginal and covariate-specific ROC curves after rocreg

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| probit_options | common_options | boot_options | Remarks and examples |
| Methods and formulas | References | Also see |  |

## Description

Under parametric estimation, rocregplot plots the fitted ROC curves for specified covariate values and classifiers. If rocreg, probit or rocreg, probit ml were previously used, the false-positive rates (for specified ROC values) and ROC values (for specified false-positive rates) for each curve may also be plotted, along with confidence intervals.

Under nonparametric estimation, rocregplot will plot the fitted ROC curves using the _fpr_* and _roc_* variables produced by rocreg. Point estimates and confidence intervals for false-positive rates and ROC values that were computed in rocreg may be plotted as well.

## Quick start

Plot ROC curve after any rocreg command
rocregplot
Name graph mygraph and save as myfile.gph
rocregplot, name(mygraph) saving(myfile)
Plot ROC curve only for v2 after fitting a model with v1 and v2 rocregplot, classvars(v2)

Add bias-corrected CI for the ROC value at a false-positive rate of 0.7 from estimation with bootstrap resampling that specified roc(.7)
rocregplot, btype(bc)

Plots following parametric estimation only
Plot curve evaluated at $\mathrm{x}=30$ and $\mathrm{x}=50$
rocregplot, at1 $(x=30)$ at2 $(x=50)$
After bootstrap resampling with resampled observations saved in myfile.dta
rocregplot, at1 $(x=30)$ at2 $(x=50)$ bfile(myfile)
And draw CI for ROC values at false-positive rates $0.1,0.2$, and 0.3
rocregplot, at1 ( $\mathrm{x}=30$ ) at2 $(\mathrm{x}=50) \operatorname{roc}(.1$. 2 .3) bfile(myfile)

## Menu

Statistics $>$ Epidemiology and related $>$ ROC analysis $>$ ROC curves after rocreg

## Syntax

Plot ROC curve after nonparametric analysis
rocregplot [, common_options boot_options]

Plot ROC curve after parametric analysis using bootstrap rocregplot [, probit_options common_options boot_options]

Plot ROC curve after parametric analysis using maximum likelihood rocregplot [, probit_options common_options]
probit_options
Description
Main
at (varname=\# [varname=\#...]) value of specified covariates and mean of unspecified covariate [at1 (varname=\# [varname=\#...]) [at2 $($ varname $=\#[$ varname $=\# . .]$. [....]]]
*roc(numlist) show estimated ROC values for given false-positive rates
*invroc (numlist)
level(\#)
show estimated false-positive rates for given ROC values set confidence level; default is level (95)

## Curve

line\#opts (cline_options)
affect rendition of ROC curve \#

[^63]common_options
Description

Main
classvars (varlist)
norefline
Scatter
plot\#opts (scatter_options)

Reference line
rlopts (cline_options)
Y axis, X axis, Titles, Legend, Overall
twoway_options
restrict plotting of ROC curves to specified classifiers suppress plotting the reference line
affect rendition of the classifier \#s false-positive rate and ROC scatter points; not allowed with at ()
affect rendition of the reference line
any options other than by () documented in [G-3] twoway_options
boot_options
Bootstrap
*bfile(filename)
btype( $n|p| b c$ )

Description
load dataset containing bootstrap replicates from rocreg plot normal-based ( n ), percentile ( p ), or bias-corrected (bc) confidence intervals; default is btype ( n )

[^64]
## probit_options

$\qquad$ Main
at (varname=\# ...) requests that the covariates specified by varname be set to \#. By default, rocregplot evaluates the function by setting each covariate to its mean value. This option causes the ROC curve to be evaluated at the value of the covariates listed in at () and at the mean of all unlisted covariates.
at1 (varname=\# ...), at2 (varname $=\# \ldots$ ) , .., at10 (varname $=\# \ldots$ ) specify that ROC curves (up to 10) be plotted on the same graph. at1(), at2(),..., at10() work like the at () option. They request that the function be evaluated at the value of the covariates specified and at the mean of all unlisted covariates. at1() specifies the values of the covariates for the first curve, at2() specifies the values of the covariates for the second curve, and so on.
roc (numlist) specifies that estimated ROC values for given false-positive rates be graphed.
invroc (numlist) specifies that estimated false-positive rates for given ROC values be graphed.
level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals. level() may be specified with either roc() or invroc().

Curve
line\#opts(cline_options) affects the rendition of ROC curve \#. See [G-3] cline_options.

## common_options

$\qquad$ Main
classvars (varlist) restricts plotting ROC curves to specified classification variables. norefline suppresses plotting the reference line.
$\qquad$ Scatter
plot\#opts(scatter_options) affects the rendition of classifier \#'s false-positive rate and ROC scatter points. This option applies only to non-ROC covariate estimation graphing. See [G-2] graph twoway scatter.
$\qquad$ Reference line
rlopts (cline_options) affects the rendition of the reference line. See [G-3] cline_options.

Y axis, X axis, Titles, Legend, Overall
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and options for saving the graph to disk (see [G-3] saving_option).

## boot_options

$\rightleftharpoons$ Bootstrap
bfile(filename) uses bootstrap replicates of parameters from rocreg stored in filename to estimate standard errors and confidence intervals of predictions. bfile() must be specified with either roc() or invroc() if parametric estimation with bootstrapping was used.
btype ( $n|p| b c$ ) indicates the desired type of confidence interval rendering. $n$ draws normal-based, $p$ draws percentile, and bc draws bias-corrected confidence intervals for specified false-positive rates and ROC values in roc() and invroc(). The default is btype(n).

## Remarks and examples

Remarks are presented under the following headings:
Plotting covariate-specific ROC curves
Plotting marginal ROC curves

## Plotting covariate-specific ROC curves

The rocregplot command is also demonstrated in $[R]$ rocreg. We will further demonstrate its use with several examples. Particularly, we will show how rocregplot can draw the ROC curves of covariate models that have been fit using rocreg.

## > Example 1: Parametric ROC

In example 6 of [R] rocreg, we fit a probit ROC model to audiology test data from Norton et al. (2000). The estimating equation method of Alonzo and Pepe (2002) was used to the fit the model. Gender and age were covariates that affected the control distribution of the classifier y1 (DPOAE 65 at 2 kHz ). Age was a ROC covariate for the model, so we fit separate ROC curves at each age.

Following Janes, Longton, and Pepe (2009), we draw the ROC curves for ages 30, 40, and 50 months. The at1(), at2(), and at3() options are used to specify the age covariates.

```
. use https://www.stata-press.com/data/r18/nnhs
(Norton - neonatal audiology data)
. rocreg d y1, probit ctrlcov(currage male) ctrlmodel(linear) roccov(currage)
> cluster(id) bseed(56930) bsave(nnhs2y1, replace)
    (output omitted)
. rocregplot, at1(currage=30) at2(currage=40) at3(currage=50)
```



Here we use the default entries of the legend, which indicate the "at \#" within the specified at* options and the classifier to which the curve corresponds. ROC curve one corresponds with currage $=30$, two with currage $=40$, and three with currage $=50$. The positive effect of age on the ROC curve is evident. At an age of 30 months (currage=30), the ROC curve of y1 (DPOAE 65 at 2 kHz ) is nearly equivalent to that of a noninformative test that gives equal probability to hearing loss. At age 50 months (currage $=50$ ), corresponding to some of the oldest children in the study, the ROC curve shows that test y1 (DPOAE 65 at 2 kHz ) is considerably more powerful than the noninformative test.

You may create your own legend by specifying the legend() option. The default legend is designed for the possibility of multiple covariates. Here we could change the legend entries to currage values and gain some extra clarity. However, this may not be feasible when there are many covariates present.

We can also use rocregplot after maximum likelihood estimation.
> Example 2: Maximum likelihood ROC
We return to the audiology study with frequency (xf), intensity ( xl ), and hearing loss severity (xd) covariates from Stover et al. (1996) that we examined in example 10 of [R] rocreg. Negative
signal-to-noise ratio is again used as a classifier. Using maximum likelihood, we fit a probit model to these data with the indicated ROC covariates.

After fitting the model, we wish to compare the ROC curves of two covariate combinations. The first has an intensity value of 5.5 (the lowest intensity, corresponding to 55 decibels) and a frequency of 10.01 (the lowest frequency, corresponding to 1001 hertz). We give the first combination a hearing loss severity value of 0.5 (the lowest). The second covariate combination has the same frequency, but the highest intensity value of 6.5 ( 65 decibels). We give this second covariate set a higher severity value of 4 . We will visually compare the two ROC curves resulting from these two covariate value combinations.

We specify false-positive rates of 0.7 first followed by 0.2 in the roc() option to visually compare the size of the ROC curve at large and small false-positive rates. Because maximum likelihood estimation was used to fit the model, a Wald confidence interval is produced for the estimated ROC value and false-positive rate parameters. Further details are found in Methods and formulas.

```
. use https://www.stata-press.com/data/r18/dp
(Stover - DPOAE test data)
. rocreg d nsnr, probit ctrlcov(xf xl) roccov(xf xl xd) ml cluster(id)
    (output omitted)
```

```
. rocregplot, at1(xf=10.01, xl=5.5, xd=.5) at2(xf=10.01, xl=6.5, xd=4) roc(.7)
```

ROC curve
Status : d
Classifier: nsnr
Under covariates:

|  | at1 |
| ---: | ---: |
| $x f$ | 10.01 |
| $x l$ | 5.5 |
| $x d$ | .5 |


| ROC | Coefficient | Std. err. | [95\% conf. interval] |  |
| ---: | ---: | ---: | :--- | :--- |
| .7 | .978951 | .0097382 | .9598645 | .9980376 |

Under covariates:

|  | at2 |
| ---: | ---: |
| $x f$ | 10.01 |
| $x l$ | 6.5 |
| xd | 4 |


| ROC | Coefficient | Std. err. | [95\% conf. interval] |  |
| ---: | ---: | :--- | :--- | :--- |
| .7 | .9985001 | .0009657 | .9966073 | 1.000393 |



At the higher false-positive rate value of 0.7 , we see little difference in the ROC values and note that the confidence intervals nearly overlap. Now, we view the same curves with the lower false-positive rate compared.

```
. rocregplot, at1(xf=10.01, xl=5.5, xd=.5) at2(xf=10.01, xl=6.5, xd=4) roc(.2)
```

ROC curve
Status : d
Classifier: nsnr
Under covariates:

|  | at1 |
| ---: | ---: |
| $x f$ | 10.01 |
| $x l$ | 5.5 |
| xd | .5 |


| ROC | Coefficient | Std. err. | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :--- |
| .2 | .7608593 | .0510501 | .660803 | .8609157 |

Under covariates:

|  | at2 |
| ---: | ---: |
| $x f$ | 10.01 |
| $x l$ | 6.5 |
| $x d$ | 4 |


| ROC | Coefficient | Std. err. | [95\% conf. interval] |  |
| ---: | ---: | :--- | :--- | :--- |
| .2 | .9499408 | .0179824 | .914696 | .9851856 |



The lower false-positive rate of 0.2 shows clearly distinguishable ROC values. Now, we specify option invroc (.5) to view how the false-positive rates vary at a ROC value of 0.5 .

```
. rocregplot, at1(xf=10.01, xl=5.5, xd=.5) at2(xf=10.01, xl=6.5, xd=4) invroc(.5)
False-positive rate
    Status : d
    Classifier: nsnr
Under covariates:
\begin{tabular}{r|r} 
& at1 \\
\hline\(x f\) & 10.01 \\
\(x l\) & 5.5 \\
xd & .5
\end{tabular}
\begin{tabular}{r|rrll}
\hline invROC & Coefficient & Std. err. & [95\% conf. interval] \\
\hline .5 & .0578036 & .0198626 & .0188736 & .0967336 \\
\hline
\end{tabular}
```

Under covariates:

|  | at2 |
| ---: | ---: |
| $x f$ | 10.01 |
| $x l$ | 6.5 |
| xd | 4 |


| invROC | Coefficient | Std. err. | [95\% conf. interval] |  |
| ---: | ---: | :--- | :--- | :--- |
| .5 | .0055624 | .0032645 | -.0008359 | .0119607 |



At a ROC value of 0.5, the false-positive rates for both curves are small and close to one another.

## Technical note

We can use the testnl command to support our visual observations with statistical inference. We use it to perform a Wald test of the null hypothesis that the two ROC curves just rendered are equal at a false-positive rate of 0.7.

```
. testnl normal(_b[i_cons]+10.01*_b[xf]+5.5*_b[xl]
                            +.5*_b[xd]+_b[s_cons]*invnormal(.7)) =
        normal(_b[i_cons]+10.01*_b[xf]+6.5*_b[xl]
            +4*_b[xd]+_b[s_cons]*invnormal (.7))
(1) normal(_b[i_cons]+10.01*_b[xf]+5.5*_b[xl] +.5*_b[xd]+_b[s_cons]*invnormal(.7)) =
    normal(_b[i_cons]+10.01*_b[xf]+6.5*_b[xl] + 4*_b[xd]+_b[s_cons]*invnormal (.7))
            chi2(1) = 
```

The test is significant at the 0.05 level, and thus we find that the two curves are significantly different. Now, we will use testnl again to test equality of the false-positive rates for each curve with a ROC value of 0.5 . The inverse ROC formula used is derived in Methods and formulas.

```
. testnl normal((invnormal(.5)-(_b[i_cons]+10.01*_b[xf]+5.5*_b[xl]+.5*_b[xd]))
    /_b[s_cons]) =
    normal((invnormal(.5)-(_b[i_cons]+10.01*_b[xf]+6.5*_b[xl]+4*_b[xd]))
    /_b[s_cons])
    (1) normal((invnormal(.5)-(_b[i_cons]+10.01*_b[xf]+5.5*_b[xl]+.5*_b[xd]))
> /_b[s_cons]) = normal((invnormal(.5)-(_b[i_cons]+10.01*_b[xf]+6.5*_b[xl]
> +4*_b[xd])) /_b[s_cons])
        chi2(1) = 8.01
        Prob > chi2 = 0.0046
```

We again reject the null hypothesis that the two curves are equal at the 0.05 level.

The model of our last example was also fit using the estimating equations method in example 7 of $[R]$ rocreg. We will demonstrate rocregplot after that model fit as well.

## > Example 3: Parametric ROC, invROC, and ROC value

In example 2, we used rocregplot after a maximum likelihood model fit of the ROC curve for classifier $n s n r$ and covariates frequency ( $x f$ ), intensity ( xl ), and hearing loss severity ( xd ). The data were obtained from the audiology study described in Stover et al. (1996). In example 7 of $[R]$ rocreg, we fit the model using the estimating equations method of Alonzo and Pepe (2002). Under this method, bootstrap resampling is used to make inferences. We saved 50 bootstrap replications in nsnrf.dta, which we re-create below.

We use rocregplot to draw the ROC curves for $n s n r$ under the covariate values $\mathrm{xf}=10.01$, $x l=5.5$, and $\mathrm{xd}=.5$, and $\mathrm{xf}=10.01$, $\mathrm{xl}=6.5$, and $\mathrm{xd}=4$. The at\#() options are used to specify the covariate values. The previous bootstrap results are made available to rocregplot with the bfile() option. As before, we will specify 0.2 and 0.7 as false-positive rates in the roc() option and 0.5 as a ROC value in the invroc () option. We do not specify btype() and thus our graph will contain normal-based bootstrap confidence bands, the default.
. use https://www.stata-press.com/data/r18/dp
(Stover - DPOAE test data)
. rocreg d nsnr, probit ctrlcov(xf xl) roccov(xf xl xd) cluster(id)
> nobstrata ctrlfprall bseed(156385) breps(50) bsave(nsnrf, replace) (output omitted)
. rocregplot, at1 ( $\mathrm{xf}=10.01$, $\mathrm{xl}=5.5, \mathrm{xd}=.5$ ) at2 $(\mathrm{xf}=10.01, \mathrm{xl}=6.5, \mathrm{xd}=4)$
> roc(.7) bfile(nsnrf)
ROC curve
Status : d
Classifier: nsnr
Under covariates:

|  | at1 |
| ---: | ---: |
| $x f$ | 10.01 |
| xl | 5.5 |
| xd | .5 |

(Replications based on 208 clusters in id)

| ROC | Observed <br> coefficient | Bias | Bootstrap <br> std. err. | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
|  | .9835816 | .0026679 | .0133583 | .9573998 | 1.009763 (N) |
|  |  |  |  | .9475966 | $.9983087(P)$ |
|  |  |  |  | .9453403 | .9983087 (BC) |

Under covariates:

|  | at2 |
| ---: | ---: |
| $x f$ | 10.01 |
| $x l$ | 6.5 |
| $x d$ | 4 |

(Replications based on 208 clusters in id)

| ROC | Observed <br> coefficient | Bias | Bootstrap <br> std. err. | $[95 \%$ conf. interval] |  |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
|  | .999428 | .0001748 | .0007784 | .9979024 | 1.000954 | (N) |
|  |  |  |  | .9975154 | .9999899 (P) |  |
|  |  |  |  | .9965358 | .9999899 (BC) |  |



As shown in the graph, we find that the ROC values at a false-positive rate of 0.7 are close together, as they were in the maximum likelihood estimation in example 2 . We now repeat this process for the lower false-positive rate of 0.2 by using the roc (.2) option.

```
. rocregplot, at1(xf=10.01, xl=5.5, xd=.5) at2(xf=10.01, xl=6.5, xd=4)
> roc(.2) bfile(nsnrf)
ROC curve
    Status : d
    Classifier: nsnr
```

Under covariates:

|  | at1 |
| ---: | ---: |
| $x f$ | 10.01 |
| xl | 5.5 |
| xd | .5 |

(Replications based on 208 clusters in id)

| ROC | Observed <br> coefficient | Bias | Bootstrap <br> std. err. | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
|  | .7652956 | -.006687 | .0624187 | .6429572 | $.8876341(\mathrm{~N})$ |
|  |  |  |  | .6654987 | $.8837934(\mathrm{P})$ |
|  |  |  |  | .6444601 | $.8837934(\mathrm{BC})$ |

Under covariates:

|  | at2 |
| ---: | ---: |
| $x f$ | 10.01 |
| $x l$ | 6.5 |
| xd | 4 |

(Replications based on 208 clusters in id)

| ROC | Observed <br> coefficient | Bias | Bootstrap <br> std. err. | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
|  | .9672505 | -.0015124 | .0162785 | .9353452 | .9991558 (N) |
|  |  |  |  | .9372331 | .9976121 |
|  |  |  |  | .9218445 | .9926328 (BC) |



The ROC values are slightly higher at the false-positive rate of 0.2 than they were in the maximum likelihood estimation in example 2. To see if the false-positive rates differ at a ROC value of 0.5 , we specify the invroc (.5) option.

```
. rocregplot, at1(xf=10.01, xl=5.5, xd=.5) at2(xf=10.01, xl=6.5, xd=4)
> invroc(.5) bfile(nsnrf)
False-positive rate
    Status : d
    Classifier: nsnr
```

Under covariates:

|  | at1 |
| ---: | ---: |
| xf | 10.01 |
| xl | 5.5 |
| xd | .5 |

(Replications based on 208 clusters in id)

| invROC | Observed <br> coefficient | Bias | Bootstrap <br> std.err. | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
|  | .0615144 | .0012744 | .0209516 | .0204499 | $.1025789(\mathrm{~N})$ |
|  |  |  |  | .0234184 | $.09672(\mathrm{P})$ |
|  |  |  |  | .0234184 | .09672 (BC) |

Under covariates:

|  | at2 |
| ---: | ---: |
| $x f$ | 10.01 |
| $x l$ | 6.5 |
| $x d$ | 4 |

(Replications based on 208 clusters in id)

| invROC | Observed <br> coefficient | Bias | Bootstrap <br> std. err. | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
|  | .0043298 | -.000139 | .003835 | -.0031867 | .0118463 (N) |
|  |  |  |  | .0002028 | $.0136293(\mathrm{P})$ |
|  |  |  |  | .0012615 | .0192129 (BC) |



The point estimates of the ROC value and false-positive rate are both computed directly using the point estimates of the ROC coefficients. Calculation of the standard errors and confidence intervals is slightly more complicated. Essentially, we have stored a sample of our ROC covariate coefficient estimates in nsnrf.dta. We then calculate the ROC value or false-positive rate estimates using each set of coefficient estimates, resulting in a sample of point estimates. Then, the bootstrap standard error and confidence intervals are calculated based on these bootstrap samples. Details of the computation of the standard error and percentile confidence intervals can be found in Methods and formulas and in [R] bootstrap.

As mentioned in [R] rocreg, 50 resamples is a reasonable lower bound for obtaining bootstrap standard errors (Mooney and Duval 1993). However, it may be too low for obtaining percentile and bias-corrected confidence intervals. Normal-based confidence intervals are valid when the bootstrap distribution exhibits normality. See $[R]$ bootstrap postestimation for more details.

We can assess the normality of the bootstrap distribution by using a normal probability plot. Stata provides this in the pnorm command (see [R] Diagnostic plots). We will use nsnrf.dta to draw a normal probability plot for the ROC estimate corresponding to a false-positive rate of 0.2 . We use the covariate values $\mathrm{xf}=10.01, \mathrm{xl}=6.5$, and $\mathrm{xd}=4$.

```
. use nsnrf
(bootstrap: rocregstat)
. generate double rocp2 = nsnr_b_i_cons + 10.01*nsnr_b_xf + 6.5*nsnr_b_xl +
> 4*nsnr_b_xd+nsnr_b_s_cons*invnormal(.2)
. replace rocp2 = normal(rocp2)
(50 real changes made)
```

- pnorm rocp2


The closeness of the points to the horizontal line on the normal probability plot shows us that the bootstrap distribution is approximately normal. So it is reasonable to use the normal-based confidence intervals for ROC at a false-positive rate of 0.2 under covariate values $\mathrm{xf}=10.01$, $\mathrm{xl}=6.5$, and $x d=4$.

## Plotting marginal ROC curves

The rocregplot command can also be used after fitting models with no covariates. We will demonstrate this with an empirical ROC model fit in $[\mathrm{R}]$ rocreg.
> Example 4: Nonparametric ROC
We run rocregplot after fitting the single-classifier, empirical ROC model shown in example 1 of $[R]$ rocreg. There we empirically predicted the ROC curve of the classifier rating for the true status variable disease from the Hanley and McNeil (1982) data. The rocreg command saves variables _roc_rating and _fpr_rating, which give the ROC values and false-positive rates, respectively, for every value of rating. These variables are used by rocregplot to render the ROC curve.

```
. use https://www.stata-press.com/data/r18/hanley, clear
(Tomographic images)
. rocreg disease rating, noboot
Nonparametric ROC estimation Number of obs = 109
Control standardization: empirical
ROC method : empirical
Area under the ROC curve
    Status : disease
    Classifier: rating
```



```
. rocregplot
```



We end our discussion of rocregplot by showing its use after a marginal probit model.
$>$ Example 5: Maximum likelihood ROC, invROC, and ROC value
In example 13 of [ R$]$ rocreg, we fit a maximum-likelihood probit model to each classifier of the fictitious dataset generated from Hanley and McNeil (1983).

We use rocregplot after the original rocreg command to draw the ROC curves for classifiers mod1 and mod3. This is accomplished by specifying the two variables in the classvars() option. We will use the roc() option to obtain confidence intervals for ROC values at false-positive rates of 0.15 and 0.75 . We will specify the invroc() option to obtain false-positive rate confidence intervals for a ROC value of 0.8 . As mentioned previously, these are Wald confidence intervals.

First, we will view results for a false-positive rate of 0.75 .

```
. use https://www.stata-press.com/data/r18/ct2, clear
(Reconstruction of CT images)
. rocreg status mod1 mod2 mod3, probit ml
    (output omitted)
. rocregplot, classvars(mod1 mod3) roc(.75)
ROC curve
    Status : status
    Classifier: mod1
```

| ROC | Coefficient | Std. err. | [95\% conf. interval] |  |
| ---: | ---: | :---: | ---: | :--- |
| .75 | .9931655 | .0069689 | .9795067 | 1.006824 |

Status : status
Classifier: mod3

| ROC | Coefficient | Std. err. | [95\% conf. interval] |  |
| ---: | ---: | :---: | :--- | :--- |
| .75 | .9953942 | .0043435 | .9868811 | 1.003907 |



We see that the estimates for each of the two ROC curves are close. Because this is a marginal model, the actual false-positive rate and the true-positive rate for each observation are plotted in the graph. The added point estimates of the ROC value at false-positive rate 0.75 are shown as diamond (mod3) and circle (mod1) symbols in the upper-right-hand corner of the graph at FPR $=0.75$. Confidence bands are also plotted at $\operatorname{FPR}=0.75$ but are so narrow that they are barely noticeable. Under both classifiers, the ROC value at 0.75 is very high. Now, we will compare these results to those with a lower false-positive rate of 0.15 .

| ROC curve <br> Status : status <br> Classifier: mod1 |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: |
|  |  |  |  |  |
| ROC | Coefficient | Std. err. | [95\% con | interval] |
| . 15 | . 7934935 | . 0801363 | . 6364292 | . 9505578 |
| Status : status <br> Classifier: mod3 |  |  |  |  |
| ROC | Coefficient | Std. err. | [95\% con | interval] |
| . 15 | . 8888596 | . 0520118 | . 7869184 | . 9908008 |



The ROC value for the false-positive rate of 0.15 is more separated in the two classifiers. Here we see that mod3 has a larger ROC value than mod1 for this false-positive rate, but the confidence intervals of the estimates overlap.

By specifying invroc (.8), we obtain invROC confidence intervals corresponding to a ROC value of 0.8.

| classvars (mod1 mod3) invroc(.8) |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: |
| False-positive rate |  |  |  |  |
| Status : status |  |  |  |  |
| Classifier: mod1 |  |  |  |  |
| invROC | Coefficient | Std. err. | [95\% con | interval] |
| . 8 | . 1556435 | . 069699 | . 019036 | . 2922509 |
| Classifier: mod3 |  |  |  |  |
| invROC | Coefficient | Std. err. | [95\% con | interval] |
| . 8 | .0661719 | . 045316 | -. 0226458 | . 1549896 |



For estimation of the false-positive rate at a ROC value of 0.8 , the confidence intervals overlap. Both classifiers require only a small false-positive rate to achieve a ROC value of 0.8.

## Methods and formulas

Details on computation of the nonparametric ROC curve and the estimation of the parametric ROC curve model coefficients can be found in $[\mathrm{R}]$ rocreg. Here we describe how to estimate the ROC values and false-positive rates of a parametric model. The cumulative distribution function $g$ can be the standard normal cumulative distribution function.

Methods and formulas are presented under the following headings:
Parametric model: Summary parameter definition
Maximum likelihood estimation
Estimating equations estimation

## Parametric model: Summary parameter definition

Conditioning on covariates $\mathbf{x}$, we have the following ROC curve model:

$$
\operatorname{ROC}(u)=g\left\{\mathbf{x}^{\prime} \boldsymbol{\beta}+\alpha g^{-1}(u)\right\}
$$

$\mathbf{x}$ can be constant, and $\boldsymbol{\beta}=\beta_{0}$, the constant intercept.
With simple algebra, we can solve this equation to obtain the false-positive rate value $u$ for a ROC value of $r$ :

$$
u=g\left[\left\{g^{-1}(r)-\mathbf{x}^{\prime} \boldsymbol{\beta}\right\} \alpha^{-1}\right]
$$

## Maximum likelihood estimation

We allow maximum likelihood estimation under probit parametric models, so $g=\Phi$. The ROC value and false-positive rate parameters all have closed-form expressions in terms of the covariate values $\mathbf{x}$, coefficient vector $\boldsymbol{\beta}$, and slope parameter $\alpha$. Thus to estimate these two types of summary parameters, we use the delta method (Oehlert 1992; Phillips and Park 1988). Particularly, we use the nlcom command (see [R] nlcom) to implement the delta method.

Under maximum likelihood estimation, the coefficient estimates $\widehat{\boldsymbol{\beta}}$ and slope estimate $\widehat{\alpha}$ are asymptotically normal with variance matrix V. For convenience, we rename the parameter vector [ $\left.\boldsymbol{\beta}^{\prime}, \alpha\right]$ to the $k$-parameter vector $\boldsymbol{\theta}=\left[\theta_{1}, \ldots, \theta_{k}\right]$. We will also explicitly refer to the conditioning of the ROC curve by $\boldsymbol{\theta}$ in its mention as $\operatorname{ROC}(t, \boldsymbol{\theta})$.

Under the delta method, the continuous scalar function of the estimate $\widehat{\boldsymbol{\theta}}, f(\widehat{\boldsymbol{\theta}})$ has asymptotic mean $f(\boldsymbol{\theta})$ and asymptotic covariance

$$
\widehat{\operatorname{Var}}\{f(\widehat{\boldsymbol{\theta}})\}=\mathbf{f V f}^{\prime}
$$

where $\mathbf{f}$ is the $1 \times k$ matrix of derivatives for which

$$
\mathbf{f}_{1 j}=\frac{\partial f(\boldsymbol{\theta})}{\partial \theta_{j}} \quad j=1, \ldots, k
$$

The asymptotic covariance of $f(\widehat{\boldsymbol{\theta}})$ is estimated and then used in conjunction with $f(\widehat{\boldsymbol{\theta}})$ for further inference, including Wald confidence intervals, standard errors, and hypothesis testing.

## Estimating equations estimation

When we fit a model using the Alonzo and Pepe (2002) estimating equations method, we use the bootstrap to perform inference on the ROC curve summary parameters. Each bootstrap sample provides a sample of the coefficient estimates $\boldsymbol{\beta}$ and the slope estimates $\alpha$. Using the formulas above, we can obtain an estimate of the ROC value or false-positive rate for each resample.

By making these calculations, we obtain a bootstrap sample of our summary parameter estimate. We then obtain bootstrap standard errors, normal approximation confidence intervals, percentile confidence intervals, and bias-corrected confidence intervals using this bootstrap sample. Further details can be found in $[R]$ bootstrap.

## References

Alonzo, T. A., and M. S. Pepe. 2002. Distribution-free ROC analysis using binary regression techniques. Biostatistics 3: 421-432. https://doi.org/10.1093/biostatistics/3.3.421.

Bamber, D. 1975. The area above the ordinal dominance graph and the area below the receiver operating characteristic graph. Journal of Mathematical Psychology 12: 387-415. https://doi.org/10.1016/0022-2496(75)90001-2.
Choi, B. C. K. 1998. Slopes of a receiver operating characteristic curve and likelihood ratios for a diagnostic test. American Journal of Epidemiology 148: 1127-1132. https://doi.org/10.1093/oxfordjournals.aje.a009592.

Hanley, J. A., and B. J. McNeil. 1982. The meaning and use of the area under a receiver operating characteristic (ROC) curve. Radiology 143: 29-36. https://doi.org/10.1148/radiology.143.1.7063747.

[^65]Janes, H., G. M. Longton, and M. S. Pepe. 2009. Accommodating covariates in receiver operating characteristic analysis. Stata Journal 9: 17-39.

Lora, D., I. Contador, J. F. Pérez-Regadera, and A. Gómez de la Cámara. 2016. Features of the area under the receiver operating characteristic (ROC) curve. A good practice. Stata Journal 16: 185-196.
Mooney, C. Z., and R. D. Duval. 1993. Bootstrapping: A Nonparametric Approach to Statistical Inference. Newbury Park, CA: Sage.

Norton, S. J., M. P. Gorga, J. E. Widen, R. C. Folsom, Y. Sininger, B. Cone-Wesson, B. R. Vohr, K. Mascher, and K. Fletcher. 2000. Identification of neonatal hearing impairment: Evaluation of transient evoked otoacoustic emission, distortion product otoacoustic emission, and auditory brain stem response test performance. Ear and Hearing 21: 508-528. https://doi.org/10.1097/00003446-200010000-00013.

Oehlert, G. W. 1992. A note on the delta method. American Statistician 46: 27-29. https://doi.org/10.2307/2684406.
Phillips, P. C. B., and J. Y. Park. 1988. On the formulation of Wald tests of nonlinear restrictions. Econometrica 56: 1065-1083. https://doi.org/10.2307/1911359.
Stover, L., M. P. Gorga, S. T. Neely, and D. Montoya. 1996. Toward optimizing the clinical utility of distortion product otoacoustic emission measurements. Journal of the Acoustical Society of America 100: 956-967. https://doi.org/10.1121/1.416207.

## Also see

$[R]$ rocreg - Receiver operating characteristic (ROC) regression
$[R]$ rocreg postestimation - Postestimation tools for rocreg
[U] 20 Estimation and postestimation commands

## Title

roctab - Nonparametric ROC analysis

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

The above command is used to perform receiver operating characteristic (ROC) analyses with rating and discrete classification data.

The two variables refvar and classvar must be numeric. The reference variable indicates the true state of the observation, such as diseased and nondiseased or normal and abnormal, and must be coded as 0 and 1. The rating or outcome of the diagnostic test or test modality is recorded in classvar, which must be at least ordinal, with higher values indicating higher risk.
roctab performs nonparametric ROC analyses. By default, roctab calculates the area under the ROC curve. Optionally, roctab can plot the ROC curve, display the data in tabular form, and produce Lorenz-like plots.

See [R] rocfit for a command that fits maximum-likelihood ROC models.

## Quick start

AUC, standard error, and CI for ROC curve of binary true state, true, as a function of classification variable v

```
roctab true v
```

Also plot the ROC curve
roctab true v, graph summary
Same as above, but plot sensitivity versus specificity instead of the ROC curve roctab true v, specificity summary

Exact binomial CI instead of asymptotic normal CI for AUC estimate

```
roctab true v, binomial
```

Gini and Pietra indices

```
roctab true v, lorenz
```


## Menu

Statistics $>$ Epidemiology and related $>$ ROC analysis $>$ Nonparametric ROC analysis without covariates

## Syntax

roctab refvar classvar $[$ if ] [in] [weight] [, options]
options
Description

Main
lorenz
binomial
nolabel
detail
table
bamber
hanley
graph
norefline
summary
specificity
level(\#)
Plot
plotopts(plot_options)
Reference line rlopts(cline_options)

Add plots addplot (plot)

Y axis, X axis, Titles, Legend, Overall twoway_options
report Gini and Pietra indices
calculate exact binomial confidence intervals
display numeric codes rather than value labels
show details on sensitivity/specificity for each cutpoint
display the raw data in a $2 \times k$ contingency table
calculate standard errors by using the Bamber method
calculate standard errors by using the Hanley method
graph the ROC curve
suppress plotting the 45 -degree reference line
report the area under the ROC curve
graph sensitivity versus specificity
set confidence level; default is level (95)
affect rendition of the ROC curve
affect rendition of the reference line
add other plots to the generated graph
any options other than by () documented in [G-3] twoway_options
collect is allowed; see [U] 11.1.10 Prefix commands.
fweights are allowed; see [U] 11.1.6 weight.
plot_options
Description
marker_options
marker_label_options
cline_options
change look of markers (color, size, etc.)
add marker labels; change look or position
change look of the line

## Options

$\qquad$ Main
lorenz specifies that the Gini and Pietra indices be reported. Optionally, graph will plot the Lorenz-like curve.
binomial specifies that exact binomial confidence intervals be calculated.
nolabel specifies that numeric codes be displayed rather than value labels.
detail outputs a table displaying the sensitivity, specificity, the percentage of subjects correctly classified, and two likelihood ratios for each possible cutpoint of classvar.
table outputs a $2 \times k$ contingency table displaying the raw data.
bamber specifies that the standard error for the area under the ROC curve be calculated using the method suggested by Bamber (1975). Otherwise, standard errors are obtained as suggested by DeLong, DeLong, and Clarke-Pearson (1988).
hanley specifies that the standard error for the area under the ROC curve be calculated using the method suggested by Hanley and McNeil (1982). Otherwise, standard errors are obtained as suggested by DeLong, DeLong, and Clarke-Pearson (1988).
graph produces graphical output of the ROC curve. If lorenz is specified, graphical output of a Lorenz-like curve will be produced.
norefline suppresses plotting the 45-degree reference line from the graphical output of the ROC curve.
summary reports the area under the ROC curve, its standard error, and its confidence interval. If lorenz is specified, Lorenz indices are reported. This option is needed only when also specifying graph.
specificity produces a graph of sensitivity versus specificity instead of sensitivity versus ( 1 - specificity). specificity implies graph.
level(\#) specifies the confidence level, as a percentage, for the confidence intervals. The default is level(95) or as set by set level; see [R] level.
$\qquad$ Plot
plotopts(plot_options) affects the rendition of the plotted ROC curve-the curve's plotted points connected by lines. The plot_options can affect the size and color of markers, whether and how the markers are labeled, and whether and how the points are connected; see [G-3] marker_options, [G-3] marker_label_options, and [G-3] cline_options.

Reference line
rlopts (cline_options) affects the rendition of the reference line; see [G-3] cline_options.

Add plots
addplot (plot) provides a way to add other plots to the generated graph; see [G-3] addplot_option.

Y axis, X axis, Titles, Legend, Overall
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

## Remarks and examples

Remarks are presented under the following headings:

## Introduction

The roctab command provides nonparametric estimation of the ROC for a given classifier and true-status reference variable. The Lorenz curve functionality of roctab, which provides an alternative to standard ROC analysis, is discussed in Lorenz-like curves.

See Pepe (2003) for a discussion of ROC analysis. Pepe has posted Stata datasets and programs used to reproduce results presented in the book (https://www.stata.com/bookstore/pepe.html).

## Nonparametric ROC curves

The points on the nonparametric ROC curve are generated using each possible outcome of the diagnostic test as a classification cutpoint and computing the corresponding sensitivity and $1-$ specificity. These points are then connected by straight lines, and the area under the resulting ROC curve is computed using the trapezoidal rule.

## > Example 1

Hanley and McNeil (1982) presented data from a study in which a reviewer was asked to classify, using a five-point scale, a random sample of 109 tomographic images from patients with neurological problems. The rating scale was as follows: $1=$ definitely normal, $2=$ probably normal, $3=$ questionable, $4=$ probably abnormal, and $5=$ definitely abnormal. The true disease status was normal for 58 of the patients and abnormal for the remaining 51 patients.

Here we list 9 of the 109 observations:

```
. use https://www.stata-press.com/data/r18/hanley
(Tomographic images)
. list disease rating in 1/9
```

| disease | rating |
| ---: | ---: |
| 1 | 5 |
| 0 | 1 |
| 1 | 5 |
| 0 | 4 |
| 0 | 1 |
| 0 | 3 |
| 1 | 5 |
| 0 | 5 |
| 0 | 1 |

For each observation, disease identifies the true disease status of the subject $(0=$ normal, $1=$ abnormal), and rating contains the classification value assigned by the reviewer.

We can use roctab to calculate and plot the nonparametric ROC curve by specifying both the summary and graph options. By also specifying the table option, we obtain a contingency table summarizing our dataset.
. roctab disease rating, table graph summary


Area under ROC curve $=0.8932$

By default, roctab reports the area under the curve, its standard error, and its confidence interval. The graph option can be used to plot the ROC curve.

The ROC curve is plotted by computing the sensitivity and specificity using each value of the rating variable as a possible cutpoint. A point is plotted on the graph for each of the cutpoints. These plotted points are joined by straight lines to form the ROC curve, and the area under the ROC curve is computed using the trapezoidal rule.

We can tabulate the computed sensitivities and specificities for each of the possible cutpoints by specifying detail.

| Cutpoint | Sensitivity | Specificity | Correctly <br> classified | LR+ | LR- |
| :---: | :---: | :---: | :---: | :---: | :---: |
| ( $>=1$ ) | 100.00\% | 0.00\% | 46.79\% | 1.0000 |  |
| ( >= 2 ) | 94.12\% | 56.90\% | 74.31\% | 2.1835 | 0.1034 |
| ( >= 3 ) | 90.20\% | 67.24\% | 77.98\% | 2.7534 | 0.1458 |
| ( >= 4 ) | 86.27\% | 77.59\% | 81.65\% | 3.8492 | 0.1769 |
| ( >= 5 ) | 64.71\% | 96.55\% | 81.65\% | 18.7647 | 0.3655 |
| ( > 5 ) | 0.00\% | 100.00\% | 53.21\% |  | 1.0000 |


| Obs | ROC <br> area | Std. err. | Asymptotic normal <br> [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | ---: |
| 109 | 0.8932 | 0.0307 | 0.83295 | 0.95339 |

Each cutpoint in the table indicates the ratings used to classify tomographs as being from an abnormal subject. For example, the first cutpoint $(>=1)$ indicates that all tomographs rated as 1 or greater are classified as coming from abnormal subjects. Because all tomographs have a rating of 1 or greater, all are considered abnormal. Consequently, all abnormal cases are correctly classified (sensitivity $=100 \%$ ), but none of the normal patients is classified correctly (specificity $=0 \%$ ). For the second cutpoint ( $>=2$ ), tomographs with ratings of 1 are classified as normal, and those with ratings of 2 or greater are classified as abnormal. The resulting sensitivity and specificity are $94.12 \%$ and $56.90 \%$, respectively. Using this cutpoint, we correctly classified $74.31 \%$ of the 109 tomographs. Similar interpretations can be used on the remaining cutpoints. As mentioned, each cutpoint corresponds to a point on the nonparametric ROC curve. The first cutpoint $(>=1)$ corresponds to the point at $(1,1)$, and the last cutpoint $(>5)$ corresponds to the point at $(0,0)$.
detail also reports two likelihood ratios suggested by Choi (1998): the likelihood ratio for a positive test result ( $\mathrm{LR}+$ ) and the likelihood ratio for a negative test result ( $\mathrm{LR}-$ ). The $\mathrm{LR}+$ is the ratio of the probability of a positive test among the truly positive subjects to the probability of a positive test among the truly negative subjects. The LR- is the ratio of the probability of a negative test among the truly positive subjects to the probability of a negative test among the truly negative subjects. Choi points out that LR+ corresponds to the slope of the line from the origin to the point on the ROC curve determined by the cutpoint. Similarly, LR- corresponds to the slope from the point $(1,1)$ to the point on the ROC curve determined by the cutpoint.

By default, roctab calculates the standard error for the area under the curve by using an algorithm suggested by DeLong, DeLong, and Clarke-Pearson (1988) and asymptotic normal confidence intervals. Optionally, standard errors based on methods suggested by Bamber (1975) or Hanley and McNeil (1982) can be computed by specifying bamber or hanley, respectively, and an exact binomial confidence interval can be obtained by specifying binomial.
. roctab disease rating, bamber

| Obs | $\begin{array}{r} \text { ROC } \\ \text { area } \end{array}$ | $\begin{aligned} & \text { Bamber } \\ & \text { std. err. } \end{aligned}$ | Asymptotic normal [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: |
| 109 | 0.8932 | 0.0306 | 0.83317 | 0.95317 |
| . roctab disease rating, hanley binomial |  |  |  |  |
| Obs | $\begin{array}{r} \text { ROC } \\ \text { area } \end{array}$ | $\begin{aligned} & \text { Hanley } \\ & \text { std. err. } \end{aligned}$ | Binomi [95\% conf | $\begin{aligned} & \text { exact } \\ & \text { interval] } \end{aligned}$ |
| 109 | 0.8932 | 0.0320 | 0.81559 | 0.94180 |

## Lorenz-like curves

For applications where it is known that the risk status increases or decreases monotonically with increasing values of the diagnostic test, the ROC curve and associated indices are useful in assessing the overall performance of a diagnostic test. When the risk status does not vary monotonically with increasing values of the diagnostic test, however, the resulting ROC curve can be nonconvex and its indices can be unreliable. For these situations, Lee (1999) proposed an alternative to the ROC analysis based on Lorenz-like curves and the associated Pietra and Gini indices.

Lee (1999) mentions at least three specific situations where results from Lorenz curves are superior to those obtained from ROC curves: 1) a diagnostic test with similar means but very different standard deviations in the abnormal and normal populations, 2) a diagnostic test with bimodal distributions in either the normal or abnormal population, and 3) a diagnostic test distributed symmetrically in the normal population and skewed in the abnormal.

When the risk status increases or decreases monotonically with increasing values of the diagnostic test, the ROC and Lorenz curves yield interchangeable results.

## > Example 2

To illustrate the use of the lorenz option, we constructed a fictitious dataset that yields results similar to those presented in Table III of Lee (1999). The data assume that a 12 -point rating scale was used to classify 442 diseased and 442 healthy subjects. We list a few of the observations.

```
. use https://www.stata-press.com/data/r18/lorenz, clear
. list in 1/7, noobs sep(0)
```

| disease | class | pop |
| ---: | ---: | ---: |
| 0 | 5 | 66 |
| 1 | 11 | 17 |
| 0 | 6 | 85 |
| 0 | 3 | 19 |
| 0 | 10 | 19 |
| 0 | 2 | 7 |
| 1 | 4 | 16 |

The data consist of 24 observations: 12 observations from diseased individuals and 12 from nondiseased individuals. Each observation corresponds to one of the 12 classification values of the rating-scale variable, class. The number of subjects represented by each observation is given by the pop variable, making this a frequency-weighted dataset. The data were generated assuming a binormal distribution of the latent variable with similar means for the normal and abnormal populations but with the standard deviation for the abnormal population five times greater than that of the normal population.

```
. roctab disease class [fweight=pop], graph summary
                    ROC Asymptotic normal
        Obs area Std. err. [95% conf. interval]
\begin{tabular}{lllll}
884 & 0.5774 & 0.0215 & 0.53517 & 0.61959
\end{tabular}
```



Area under ROC curve $=0.5774$

The resulting ROC curve is nonconvex or, as termed by Lee, "wiggly". Lee argues that for this and similar situations, the Lorenz curve and indices are preferred.

```
. roctab disease class [fweight=pop], lorenz summary graph
Lorenz curve
```

Pietra index $=0.6493$
Gini index $=0.7441$

Lorenz curve


Like ROC curves, a more bowed Lorenz curve suggests a better diagnostic test. This bowedness is quantified by the Pietra index, which is geometrically equivalent to twice the largest triangle that can be inscribed in the area between the curve and the diagonal line, and the Gini index, which is equivalent to twice the area between the Lorenz curve and the diagonal. Lee (1999) provides several additional interpretations for the Pietra and Gini indices.

## Stored results

roctab stores the following in $r()$ :
Scalars

```
    r(N) number of observations
    r(se) standard error for the area under the ROC curve
    r(lb) lower bound of CI for the area under the ROC curve
    r(ub) upper bound of CI for the area under the ROC curve
    r(level) confidence level
    r(area) area under the ROC curve
    r(pietra) Pietra index
    r(gini) Gini index
    r(cutpoints) description of cutpoints (detail only)
    r(detail) matrix with details on sensitivity and specificity for each cutpoint (detail only)
```

Macros
Matrices

## Methods and formulas

Assume that we applied a diagnostic test to each of $N_{n}$ normal and $N_{a}$ abnormal subjects. Further assume that the higher the outcome value of the diagnostic test, the higher the risk of the subject being abnormal. Let $\widehat{\theta}$ be the estimated area under the curve, and let $X_{i}, i=1,2, \ldots, N_{a}$ and $Y_{j}, j=1,2, \ldots, N_{n}$ be the values of the diagnostic test for the abnormal and normal subjects, respectively.

The points on the nonparametric ROC curve are generated using each possible outcome of the diagnostic test as a classification cutpoint and computing the corresponding sensitivity and $1-$ specificity. These points are then connected by straight lines, and the area under the resulting ROC curve is computed using the trapezoidal rule.

The default standard error for the area under the ROC curve is computed using the algorithm described by DeLong, DeLong, and Clarke-Pearson (1988). For each abnormal subject, $i$, define

$$
V_{10}\left(X_{i}\right)=\frac{1}{N_{n}} \sum_{j=1}^{N_{n}} \psi\left(X_{i}, Y_{j}\right)
$$

and for each normal subject, $j$, define

$$
V_{01}\left(Y_{j}\right)=\frac{1}{N_{a}} \sum_{i=1}^{N_{a}} \psi\left(X_{i}, Y_{j}\right)
$$

where

$$
\psi(X, Y)=\left\{\begin{array}{cc}
1 & Y<X \\
\frac{1}{2} & Y=X \\
0 & Y>X
\end{array}\right.
$$

Define

$$
S_{10}=\frac{1}{N_{a}-1} \sum_{i=1}^{N_{a}}\left\{V_{10}\left(X_{i}\right)-\widehat{\theta}\right\}^{2}
$$

and

$$
S_{01}=\frac{1}{N_{n}-1} \sum_{j=1}^{N_{n}}\left\{V_{01}\left(Y_{j}\right)-\widehat{\theta}\right\}^{2}
$$

The variance of the estimated area under the ROC curve is given by

$$
\operatorname{var}(\widehat{\theta})=\frac{1}{N_{a}} S_{10}+\frac{1}{N_{n}} S_{01}
$$

The hanley standard error for the area under the ROC curve is computed using the algorithm described by Hanley and McNeil (1982). It requires the calculation of two quantities: $Q_{1}$ is $\operatorname{Pr}$ (two randomly selected abnormal subjects will both have a higher score than a randomly selected normal subject), and $Q_{2}$ is $\operatorname{Pr}$ (one randomly selected abnormal subject will have a higher score than any two randomly selected normal subjects). The Hanley and McNeil variance of the estimated area under the ROC curve is

$$
\operatorname{var}(\widehat{\theta})=\frac{\widehat{\theta}(1-\widehat{\theta})+\left(N_{a}-1\right)\left(Q_{1}-\widehat{\theta}^{2}\right)+\left(N_{n}-1\right)\left(Q_{2}-\widehat{\theta}^{2}\right)}{N_{a} N_{n}}
$$

The bamber standard error for the area under the ROC curve is computed using the algorithm described by Bamber (1975). For any two $Y$ values, $Y_{j}$ and $Y_{k}$, and any $X_{i}$ value, define

$$
b_{y y x}=p\left(Y_{j}, Y_{k}<X_{i}\right)+p\left(X_{i}<Y_{j}, Y_{k}\right)-2 p\left(Y_{j}<X_{i}<Y_{k}\right)
$$

and similarly, for any two $X$ values, $X_{i}$ and $X_{l}$, and any $Y_{j}$ value, define

$$
b_{x x y}=p\left(X_{i}, X_{l}<Y_{j}\right)+p\left(Y_{j}<X_{i}, X_{l}\right)-2 p\left(X_{i}<Y_{j}<X_{l}\right)
$$

Bamber's unbiased estimate of the variance for the area under the ROC curve is
$\operatorname{var}(\widehat{\theta})=\frac{1}{4}\left(N_{a}-1\right)\left(N_{n}-1\right)\left\{p(X \neq Y)+\left(N_{a}-1\right) b_{x x y}+\left(N_{n}-1\right) b_{y y x}-4\left(N_{a}+N_{n}-1\right)(\widehat{\theta}-0.5)^{2}\right\}$
Asymptotic confidence intervals are constructed and reported by default, assuming a normal distribution for the area under the ROC curve.

Exact binomial confidence intervals are calculated as described in $[\mathrm{R}]$ ci, with $p$ equal to the area under the ROC curve.

## References

Bamber, D. 1975. The area above the ordinal dominance graph and the area below the receiver operating characteristic graph. Journal of Mathematical Psychology 12: 387-415. https://doi.org/10.1016/0022-2496(75)90001-2.
Choi, B. C. K. 1998. Slopes of a receiver operating characteristic curve and likelihood ratios for a diagnostic test. American Journal of Epidemiology 148: 1127-1132. https://doi.org/10.1093/oxfordjournals.aje.a009592.

Cleves, M. A. 2002a. Comparative assessment of three common algorithms for estimating the variance of the area under the nonparametric receiver operating characteristic curve. Stata Journal 2: 280-289.
__. 2002b. From the help desk: Comparing areas under receiver operating characteristic curves from two or more probit or logit models. Stata Journal 2: 301-313.
DeLong, E. R., D. M. DeLong, and D. L. Clarke-Pearson. 1988. Comparing the areas under two or more correlated receiver operating characteristic curves: A nonparametric approach. Biometrics 44: 837-845.
https://doi.org/10.2307/2531595.
Erdreich, L. S., and E. T. Lee. 1981. Use of relative operating characteristic analysis in epidemiology: A method for dealing with subjective judgment. American Journal of Epidemiology 114: 649-662. https://doi.org/10.1093/oxfordjournals.aje.a113236.

Hanley, J. A., and B. J. McNeil. 1982. The meaning and use of the area under a receiver operating characteristic (ROC) curve. Radiology 143: 29-36. https://doi.org/10.1148/radiology.143.1.7063747.
Harbord, R. M., and P. Whiting. 2009. metandi: Meta-analysis of diagnostic accuracy using hierarchical logistic regression. Stata Journal 9: 211-229.

Hong, L., G. Alfani, C. Gigliarano, and M. Bonneti. 2018. giniinc: A Stata package for measuring inequality from incomplete income and survival data. Stata Journal 18: 692-715.
Juul, S., and M. Frydenberg. 2021. An Introduction to Stata for Health Researchers. 5th ed. College Station, TX: Stata Press.

Lee, W. C. 1999. Probabilistic analysis of global performances of diagnostic tests: Interpreting the Lorenz curvebased summary measures. Statistics in Medicine 18: 455-471. https://doi.org/10.1002/(sici)1097-0258(19990228)18: 4〈455::aid-sim44〉3.0.co;2-a.
Ma, G., and W. J. Hall. 1993. Confidence bands for the receiver operating characteristic curves. Medical Decision Making 13: 191-197. https://doi.org/10.1177/0272989X9301300304.
Pacifico, D., and F. Poege. 2017. Estimating measures of multidimensional poverty with Stata. Stata Journal 17: 687-703.
Pepe, M. S. 2003. The Statistical Evaluation of Medical Tests for Classification and Prediction. New York: Oxford University Press.

Reichenheim, M. E., and A. Ponce de Leon. 2002. Estimation of sensitivity and specificity arising from validity studies with incomplete design. Stata Journal 2: 267-279.
Working, H., and H. Hotelling. 1929. Application of the theory of error to the interpretation of trends. Journal of the American Statistical Association 24 (Suppl.): 73-85. https://doi.org/10.2307/2277011.

## Also see

[R] logistic postestimation - Postestimation tools for logistic
$[\mathrm{R}]$ roc - Receiver operating characteristic (ROC) analysis
[ R ] roccomp - Tests of equality of ROC areas
[R] rocfit - Parametric ROC models
$[R]$ rocreg - Receiver operating characteristic (ROC) regression

## Title

rreg - Robust regression

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| Acknowledgment | References | Also see |  |

## Description

rreg performs one version of robust regression of depvar on indepvars.
Also see Robust standard errors in $[R]$ regress for standard regression with robust variance estimates and $[R]$ qreg for quantile (including median) regression.

## Quick start

Robust regression of y on x
rreg y x
Add a categorical covariate a using factor-variable syntax
rreg y x i.a
Same as above, but set the tuning parameter to 8
rreg y x i.a, tune (8)
Generate a new variable wvar containing the final weight for each observation
rreg y x i.a, genwt(wvar)
Set confidence level to $99 \%$
rreg y x i.a, level(99)

## Menu

Statistics $>$ Linear models and related $>$ Other $>$ Robust regression

## Syntax

$$
\text { rreg depvar }[\text { indepvars }][\text { if }][\text { in }][\text {, options }]
$$

options
Model
tune (\#)
Reporting
level(\#)
genwt (newvar)
display_options

Description
use \# as the biweight tuning constant; default is tune (7)
set confidence level; default is level (95)
create newvar containing the weights assigned to each observation
control columns and column formats, row spacing, line width,
display of omitted variables and base and empty cells, and
factor-variable labeling

## Optimization

optimization_options
graph
coeflegend
control the optimization process; seldom used
graph weights during convergence
display legend instead of statistics
indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
depvar and indepvars may contain time-series operators; see [U] 11.4.4 Time-series varlists.
by, collect, mfp, mi estimate, rolling, and statsby are allowed; see [U] 11.1.10 Prefix commands.
coeflegend does not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

$\qquad$ Model
tune(\#) is the biweight tuning constant. The default is tune (7), meaning seven times the median absolute deviation from the median residual; see Methods and formulas. Lower tuning constants downweight outliers rapidly but may lead to unstable estimates (less than 6 is not recommended). Higher tuning constants produce milder downweighting.

Reporting
level (\#); see [R] Estimation options.
genwt (newvar) creates the new variable newvar containing the weights assigned to each observation. display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.
optimization_options: iterate(\#), tolerance(\#), [no]log. iterate() specifies the maximum number of iterations; iterations stop when the maximum change in weights drops below tolerance(); and log/nolog specifies whether to show the iteration log (see set iterlog in [R] set iter). These options are seldom used.
graph allows you to graphically watch the convergence of the iterative technique. The weights obtained from the most recent round of estimation are graphed against the weights obtained from the previous round.

The following option is available with rreg but is not shown in the dialog box:
coeflegend; see [R] Estimation options.

## Remarks and examples

rreg first performs an initial screening based on Cook's distance $>1$ to eliminate gross outliers before calculating starting values and then performs Huber iterations followed by biweight iterations, as suggested by Li (1985).

## > Example 1

We wish to examine the relationship between mileage rating, weight, and location of manufacture for the 74 cars in the auto.dta. As a point of comparison, we begin by fitting an ordinary regression:


We now compare this with the results from rreg:

| Huber iteration 1: Maximum difference in weights $=.80280176$ |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Huber iteration 2: Maximum difference in weights $=.2915438$ |  |  |  |  |  |  |
| Huber iteration 3: Maximum difference in weights $=.08911171$ |  |  |  |  |  |  |
| Huber iteration 4: Maximum difference in weights $=.02697328$ |  |  |  |  |  |  |
| Biweight iteration 5: Maximum difference in weights $=.29186818$ |  |  |  |  |  |  |
| Biweight iteration 6: Maximum difference in weights $=.11988101$ |  |  |  |  |  |  |
| Biweight iteration 7: Maximum difference in weights $=.03315872$ |  |  |  |  |  |  |
| Biweight iteration 8: Maximum difference in weights $=.00721325$ |  |  |  |  |  |  |
| Robust regression |  |  |  | Number of obs $=74$ |  |  |
|  |  |  |  | F( 2 | 71) $=$ | 168.32 |
|  |  |  |  | Prob > | = | 0.0000 |
| mpg | Coefficient | Std. err | t | $P>\|t\|$ | [95\% conf | interval] |
| weight | -. 0063976 | . 0003718 | -17.21 | 0.000 | -. 007139 | -. 0056562 |
| foreign | -3.182639 | . 627964 | -5.07 | 0.000 | -4.434763 | -1.930514 |
| _cons | 40.64022 | 1.263841 | 32.16 | 0.000 | 38.1202 | 43.16025 |

Note the large change in the foreign coefficient.

## - Technical note

It would have been better if we had fit the previous robust regression by typing rreg mpg weight foreign, genwt(w). The new variable, w, would then contain the estimated weights. Let's pretend that we did this:

```
. rreg mpg weight foreign, genwt(w)
    (output omitted)
. summarize w, detail
```

Robust Regression Weight

|  | Percentiles | Smallest |  |  |
| ---: | :---: | ---: | :--- | ---: |
| $1 \%$ | 0 | 0 |  |  |
| $5 \%$ | .0442957 | 0 | 0 | Obs |
| $10 \%$ | .4674935 |  | Sum of wgt. | 74 |
| $25 \%$ | .8894815 | .0442957 | Mean | .8509966 |
| $50 \%$ | .9690193 |  | Std. dev. | .2746451 |
|  |  | Largest |  |  |
| $75 \%$ | .9949395 | .9996715 |  | .0754299 |
| $90 \%$ | .9989245 | .9996953 | Variance | -2.287952 |
| $95 \%$ | .9996715 | .9997343 | Skewness | 6.874605 |

We discover that 3 observations in our data were dropped altogether (they have weight 0 ). We could further explore our data:

```
. sort w
. list make mpg weight w if w <.467, sep(0)
```

| make | mpg | weight | W |
| :--- | ---: | ---: | ---: |
| Datsun 210 | 35 | 2,020 | 0 |
| Subaru | 35 | 2,050 | 0 |
| VW Diesel | 41 | 2,040 | 0 |
| Plym. Arrow | 28 | 3,260 | .04429567 |
| Cad. Seville | 21 | 4,290 | .08241943 |
| Toyota Corolla | 31 | 2,200 | .10443129 |
| Olds 98 | 21 | 4,060 | .28141296 |

Being familiar with the automobile data, we immediately spotted two things: the VW is the only diesel car in our data, and the weight recorded for the Plymouth Arrow is incorrect.

## > Example 2

If we specify no explanatory variables, rreg produces a robust estimate of the mean:


The estimate is given by the coefficient on _cons. The mean is 20.69 with an estimated standard error of 0.6418 . The $95 \%$ confidence interval is $[19.4,22.0]$. By comparison, ci means (see $[R]$ ci) gives us the standard calculation:

| . ci means mpg |  |  |  |  |  |
| ---: | ---: | ---: | ---: | ---: | ---: |
| Variable | Obs | Mean | Std. err. | [95\% conf. interval] |  |
| mpg | 74 | 21.2973 | .6725511 | 19.9569 | 22.63769 |

Stata has an active user community working in the area of robust statistics. You may also be interested in robust estimators and outlier detection tools from Verardi and McCathie (2012), Verardi and Dehon (2010), Jann (2010), and Verardi and Croux (2009), to name a few. See [R] ssc for how to install community-contributed packages.

## Stored results

rreg stores the following in e() :

```
Scalars
        e(N)
        e(mss)
        e(df_m)
        e(rss)
        e(df_r)
        e(r2)
        e(r2_a)
        e(F)
        e(rmse)
        e(rank)
Macros
        e(cmd)
        e(cmdline)
        e(depvar)
        e(genwt)
        e(title)
        e(model)
        e(properties)
        e(predict)
        e(marginsok)
        e(asbalanced)
        e(asobserved)
        number of observations
        model sum of squares
        model degrees of freedom
        residual sum of squares
        residual degrees of freedom
        R
        adjusted R}\mp@subsup{R}{}{2
        F statistic
        root mean squared error
        rank of e(V)
        reg
        command as typed
        name of dependent variable
        variable containing the weights
        title in estimation output
        ols
        b V
        e(marginsok)
        program used to implement predict
        predictions allowed by margins
        factor variables fvset as asbalanced
        factor variables fvset as asobserved
Matrices
        e(b)
        coefficient vector
        e(V)
        variance-covariance matrix of the estimators
Functions
    e(sample) marks estimation sample
```

In addition to the above, the following is stored in r() :

```
Matrices
```

    \(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values,
    and confidence intervals
    Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

See Berk (1990), Goodall (1983), and Rousseeuw and Leroy (1987) for a general description of the issues and methods. Hamilton (1991a, 1992) provides a more detailed description of rreg and some Monte Carlo evaluations.
rreg begins by fitting the regression (see [R] regress), calculating Cook's $D$ (see [R] predict and $[\mathrm{R}]$ regress postestimation), and excluding any observation for which $D>1$.

Thereafter, rreg works iteratively: it performs a regression, calculates case weights from absolute residuals, and regresses again using those weights. Iterations stop when the maximum change in weights drops below tolerance (). Weights derive from one of two weight functions, Huber weights and biweights. Huber weights (Huber 1964) are used until convergence, and then, from that result, biweights are used until convergence. The biweight was proposed by Beaton and Tukey (1974, 151152) after the Princeton robustness study (Andrews et al. 1972) had compared various estimators. Both weighting functions are used because Huber weights have problems dealing with severe outliers, whereas biweights sometimes fail to converge or have multiple solutions. The initial Huber weighting should improve the behavior of the biweight estimator.

In Huber weighting, cases with small residuals receive weights of 1 ; cases with larger residuals receive gradually smaller weights. Let $e_{i}=y_{i}-\mathbf{X}_{i} \mathbf{b}$ represent the $i$ th-case residual. The $i$ th scaled residual $u_{i}=e_{i} / s$ is calculated, where $s=M / 0.6745$ is the residual scale estimate and $M=\operatorname{med}\left(\left|e_{i}-\operatorname{med}\left(e_{i}\right)\right|\right)$ is the median absolute deviation from the median residual. Huber estimation obtains case weights:

$$
w_{i}= \begin{cases}1 & \text { if }\left|u_{i}\right| \leq c_{h} \\ c_{h} /\left|u_{i}\right| & \text { otherwise }\end{cases}
$$

rreg defines $c_{h}=1.345$, so downweighting begins with cases whose absolute residual exceeds (1.345/0.6745) $M \approx 2 M$.

With biweights, all cases with nonzero residuals receive some downweighting, according to the smoothly decreasing biweight function

$$
w_{i}= \begin{cases}\left\{1-\left(u_{i} / c_{b}\right)^{2}\right\}^{2} & \text { if }\left|u_{i}\right| \leq c_{b} \\ 0 & \text { otherwise }\end{cases}
$$

where $c_{b}=4.685 \times$ tune () $/ 7$. Thus when tune ()$=7$, cases with absolute residuals of (4.685/0.6745) $M \approx 7 M$ or more are assigned 0 weight and thus are effectively dropped. Goodall (1983, 377) suggests using a value between 6 and 9 , inclusive, for tune() in the biweight case and states that performance is good between 6 and 12, inclusive.

The tuning constants $c_{h}=1.345$ and $c_{b}=4.685$ (assuming tune() is set at the default 7) give rreg about $95 \%$ of the efficiency of OLS when applied to data with normally distributed errors (Hamilton 1991b). Lower tuning constants downweight outliers more drastically (but give up Gaussian efficiency); higher tuning constants make the estimator more like OLS.

Standard errors are calculated using the pseudovalues approach described in Street, Carroll, and Ruppert (1988).

## Acknowledgment

The current version of rreg is due to the work of Lawrence Hamilton of the Department of Sociology at the University of New Hampshire.

## References

Andrews, D. F., P. J. Bickel, F. R. Hampel, P. J. Huber, W. H. Rogers, and J. W. Tukey. 1972. Robust Estimates of Location: Survey and Advances. Princeton, NJ: Princeton University Press.
Beaton, A. E., and J. W. Tukey. 1974. The fitting of power series, meaning polynomials, illustrated on band-spectroscopic data. Technometrics 16: 147-185. https://doi.org/10.2307/1267936.
Berk, R. 1990. A primer on robust regression. In Modern Methods of Data Analysis, ed. J. Fox and J. S. Long, 292-324. Newbury Park, CA: Sage.

Goodall, C. 1983. M-estimators of location: An outline of the theory. In Understanding Robust and Exploratory Data Analysis, ed. D. C. Hoaglin, C. F. Mosteller, and J. W. Tukey, 339-431. New York: Wiley.
Gould, W. W., and W. H. Rogers. 1994. Quantile regression as an alternative to robust regression. In 1994 Proceedings of the Statistical Computing Section. Alexandria, VA: American Statistical Association.

Hamilton, L. C. 1991a. srd1: How robust is robust regression? Stata Technical Bulletin 2: 21-26. Reprinted in Stata Technical Bulletin Reprints, vol. 1, pp. 169-175. College Station, TX: Stata Press.

[^66]1992. Regression with Graphics: A Second Course in Applied Statistics. Belmont, CA: Duxbury.

Huber, P. J. 1964. Robust estimation of a location parameter. Annals of Mathematical Statistics 35: 73-101. https://doi.org/10.1214/aoms/1177703732.
Jann, B. 2010. robreg: Stata module providing robust regression estimators. Boston College Department of Economics, Statistical Software Components S457114. https://ideas.repec.org/c/boc/bocode/s457114.html.

Li, G. 1985. Robust regression. In Exploring Data Tables, Trends, and Shapes, ed. D. C. Hoaglin, C. F. Mosteller, and J. W. Tukey, 281-340. New York: Wiley.
Mosteller, C. F., and J. W. Tukey. 1977. Data Analysis and Regression: A Second Course in Statistics. Reading, MA: Addison-Wesley.
Relles, D. A., and W. H. Rogers. 1977. Statisticians are fairly robust estimators of location. Journal of the American Statistical Association 72: 107-111. https://doi.org/10.2307/2286917.
Rousseeuw, P. J., and A. M. Leroy. 1987. Robust Regression and Outlier Detection. New York: Wiley.
Street, J. O., R. J. Carroll, and D. Ruppert. 1988. A note on computing robust regression estimates via iteratively reweighted least squares. American Statistician 42: 152-154. https://doi.org/10.2307/2684491.

Verardi, V., and C. Croux. 2009. Robust regression in Stata. Stata Journal 9: 439-453.
Verardi, V., and C. Dehon. 2010. Multivariate outlier detection in Stata. Stata Journal 10: 259-266.
Verardi, V., and A. McCathie. 2012. The S-estimator of multivariate location and scatter in Stata. Stata Journal 12: 299-307.

## Also see

[R] rreg postestimation - Postestimation tools for rreg
[R] qreg - Quantile regression
$[R]$ regress - Linear regression
[MI] Estimation - Estimation commands for use with mi estimate
[U] 20 Estimation and postestimation commands

## Title

rreg postestimation - Postestimation tools for rreg

Postestimation commands predict margins Also see

## Postestimation commands

The following postestimation commands are available after rreg:

| Command | Description |
| :--- | :--- |
| contrast | contrasts and ANOVA-style joint tests of estimates |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estimates | cataloging estimation results |
| etable | table of estimation results |
| * forecast | dynamic forecasts and simulations <br> point estimates, standard errors, testing, and inference for linear combinations of |
| margins | coefficients |
| marginal means, predictive margins, marginal effects, and average marginal effects |  |
| nlcom | graph the results from margins (profile plots, interaction plots, etc.) <br> point estimates, standard errors, testing, and inference for nonlinear combinations |
| predict | of coefficients |
| predictions and their SEs, residuals, etc. |  |
| pwcompare | point estimates, standard errors, testing, and inference for generalized predictions <br> pairwise comparisons of estimates |
| test | Wald tests of simple and composite linear hypotheses |
| testnl | Wald tests of nonlinear hypotheses |

[^67]
## predict

## Description for predict

predict creates a new variable containing predictions such as linear predictions, standard errors, residuals, and diagonal elements of the hat matrix.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

```
predict [type] newvar [if] [in] [, statistic]
```

statistic Description

Main
xb linear prediction; the default
stdp standard error of the linear prediction
residuals residuals
hat diagonal elements of the hat matrix
These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

## Options for predict

$\qquad$ Main
xb , the default, calculates the linear prediction.
stdp calculates the standard error of the linear prediction.
residuals calculates the residuals.
hat calculates the diagonal elements of the hat matrix. You must have run the rreg command with the genwt() option.

## margins

## Description for margins

margins estimates margins of response for linear predictions.

## Menu for margins

Statistics > Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [options]
```

statistic Description
$\mathrm{xb} \quad$ linear prediction; the default
stdp not allowed with margins
residuals not allowed with margins
hat not allowed with margins

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

## Also see

[R] rreg - Robust regression
[U] 20 Estimation and postestimation commands

## Title

runtest - Test for random order

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| Acknowledgment | References |  |  |

## Description

runtest tests whether the observations of varname are serially independent-that is, whether they occur in a random order - by counting how many runs there are above and below a threshold. By default, the median is used as the threshold. A small number of runs indicates positive serial correlation; a large number indicates negative serial correlation.

## Quick start

Run test for the serial independence of the observations of v 1 runtest v1

Use the mean of v 1 , rather than the median, as the threshold for counting runs
runtest v1, mean
Use 10 as the threshold for counting runs
runtest v1, threshold(10)
Same as above, but ignore any values of v1 equal to the threshold when counting runs runtest v1, threshold(10) drop

## Menu

Statistics $>$ Nonparametric analysis $>$ Tests of hypotheses $>$ Test for random order

## Syntax

runtest varname [in] [, options]
options

| continuity | continuity correction <br> ignore values equal to the threshold <br> drop |
| :--- | :--- |
| split | threshold; default is to count as below |
| mean | use mean as threshold; default is median |
| threshold(\#) | assign arbitrary threshold; default is median |

drop
split
mean
threshold(\#)

## Description

## continuity correction

ignore values equal to the threshold
randomly split values equal to the threshold as above or below the threshold; default is to count as below
use mean as threshold; default is median
assign arbitrary threshold; default is median
collect is allowed; see [U] 11.1.10 Prefix commands.

## Options

continuity specifies a continuity correction that may be helpful in small samples. If there are fewer than 10 observations either above or below the threshold, however, the tables in Swed and Eisenhart (1943) provide more reliable critical values. By default, no continuity correction is used.
drop directs runtest to ignore any values of varname that are equal to the threshold value when counting runs and tabulating observations. By default, runtest counts a value as being above the threshold when it is strictly above the threshold and as being below the threshold when it is less than or equal to the threshold.
split directs runtest to randomly split values of varname that are equal to the threshold. In other words, when varname is equal to threshold, a "coin" is flipped. If it comes up heads, the value is counted as above the threshold. If it comes up tails, the value is counted as below the threshold. mean directs runtest to tabulate runs above and below the mean rather than the median.
threshold(\#) specifies an arbitrary threshold to use in counting runs. For example, if varname has already been coded as a $0 / 1$ variable, the median generally will not be a meaningful separating value.

## Remarks and examples

runtest performs a nonparametric test of the hypothesis that the observations of varname occur in a random order by counting how many runs there are above and below a threshold. If varname is positively serially correlated, it will tend to remain above or below its median for several observations in a row; that is, there will be relatively few runs. If, on the other hand, varname is negatively serially correlated, observations above the median will tend to be followed by observations below the median and vice versa; that is, there will be relatively many runs.

By default, runtest uses the median for the threshold, and this is not necessarily the best choice. If mean is specified, the mean is used instead of the median. If threshold(\#) is specified, \# is used. Because runtest divides the data into two states-above and below the threshold-it is appropriate for data that are already binary; for example, win or lose, live or die, rich or poor, etc. Such variables are often coded as 0 for one state and 1 for the other. Here you should specify threshold (0) because, by default, runtest separates the observations into those that are greater than the threshold and those that are less than or equal to the threshold.

As with most nonparametric procedures, the treatment of ties complicates the test. Observations equal to the threshold value are ties and can be treated in one of three ways. By default, they are treated as if they were below the threshold. If drop is specified, they are omitted from the calculation and the total number of observations is adjusted. If split is specified, each is randomly assigned to the above- and below-threshold groups. The random assignment is different each time the procedure is run unless you specify the random-number seed; see $[R]$ set seed.

## > Example 1

We can use runtest to check regression residuals for serial correlation.

```
. use https://www.stata-press.com/data/r18/run1
. scatter resid year, connect(l) yline(0) title(Regression residuals)
```



The graph gives the impression that these residuals are positively correlated. Excursions above or below zero - the natural threshold for regression residuals - tend to last for several observations. runtest can evaluate the statistical significance of this impression.

```
. runtest resid, thresh(0)
N(resid <= 0) = 8
N(resid > 0) = 8
        obs = 16
    N(runs) = 5
        z = -2.07
        Prob>|z| = . 04
```

There are five runs in these 16 observations. Using the normal approximation to the true distribution of the number of runs, the five runs in this series are fewer than would be expected if the residuals were serially independent. The $p$-value is 0.04 , indicating a two-sided significant result at the $5 \%$ level. If the alternative hypothesis is positive serial correlation, rather than any deviation from randomness, then the one-sided $p$-value is $0.04 / 2=0.015$. With so few observations, however, the normal approximation may be inaccurate. (Tables compiled by Swed and Eisenhart list five runs as the $5 \%$ critical value for a one-sided test.)
runtest is a nonparametric test. It ignores the magnitudes of the observations and notes only whether the values are above or below the threshold. We can demonstrate this feature by reducing the information about the regression residuals in this example to a $0 / 1$ variable that indicates only whether a residual is positive or negative.

```
. generate byte sign = resid>0
. runtest sign, thresh(0)
N(sign <= 0) = 8
N(sign > 0) = 8
            obs = 16
        N(runs) = 5
            z = -2.07
        Prob>|z| = . 04
```

As expected, runtest produces the same answer as before.

## - Technical note

The run test can also be used to test the null hypothesis that two samples are drawn from the same underlying distribution. The run test is sensitive to differences in the shapes, as well as the locations, of the empirical distributions.

Suppose, for example, that two different additives are added to the oil in 10 different cars during an oil change. The cars are run until a viscosity test determines that another oil change is needed, and the number of miles traveled between oil changes is recorded. The data are

```
. use https://www.stata-press.com/data/r18/additive, clear
. list
```

|  | additive | miles |
| :---: | :---: | :---: |
| 1. | 1 | 4024 |
| 2. | 1 | 4756 |
| 3. | 1 | 7993 |
| 4. | 1 | 5025 |
| 5. | 1 | 4188 |
| 6. | 2 | 3007 |
| 7. | 2 | 1988 |
| 8. | 2 | 1051 |
| 9. | 2 | 4478 |
| 10. | 2 | 4232 |

To test whether the additives generate different distributions of miles between oil changes, we sort the data by miles and then use runtest to see whether the marker for each additive occurs in random order:

```
. sort miles
. runtest additive, thresh(1)
N(additive <= 1) = 5
N(additive > 1) = 5
            obs = 10
        N(runs) = 4
            z = -1.34
        Prob>|z| = . 18
```

Here the additives do not produce statistically different results.

- Technical note

A test that is related to the run test is the runs up-and-down test. In the latter test, the data are classified not by whether they lie above or below a threshold but by whether they are steadily increasing or decreasing. Thus an unbroken string of increases in the variable of interest is counted as one run, as is an unbroken string of decreases. According to Madansky (1988), the run test is superior to the runs up-and-down test for detecting trends in the data, but the runs up-and-down test is superior for detecting autocorrelation.
runtest can be used to perform a runs up-and-down test. Using the regression residuals from the example above, we can perform a runtest on their first differences:

```
. use https://www.stata-press.com/data/r18/run1
. generate resid_D = resid - resid[_n-1]
(1 missing value generated)
. runtest resid_D, thresh(0)
N(resid_D <= 0) = 7
N(resid_D > 0) = 8
            obs = 15
    N(runs) = 6
            z = -1.33
    Prob>|z| = . 18
```

Edgington (1961) has compiled a table of the small-sample distribution of the runs up-and-down statistic, and this table is reprinted in Madansky (1988). For large samples, the $z$ statistic reported by runtest is incorrect for the runs up-and-down test. Let $N$ be the number of observations ( 15 here), and let $r$ be the number of runs (6). The expected number of runs in the runs up-and-down test is

$$
\mu_{r}=\frac{2 N-1}{3}
$$

the variance is

$$
\sigma_{r}^{2}=\frac{16 N-29}{90}
$$

and the correct $z$ statistic is

$$
\widehat{z}=\frac{r-\mu_{r}}{\sigma_{r}}
$$

## Technical note

runtest will tolerate missing values at the beginning or end of a series, as occurred in the technical note above (generating first differences resulted in a missing value for the first observation). runtest, however, will issue an error message if there are any missing observations in the interior of the series (in the portion covered by the in range qualifier). To perform the test anyway, simply drop the missing observations before using runtest.

## Stored results

runtest stores the following in $r()$ :
Scalars

| $r(N)$ | number of observations |
| :--- | :--- |
| $r\left(N \_\right.$below $)$ | number below the threshold |
| $r\left(N \_a b o v e\right)$ | number above the threshold |
| $r($ mean $)$ | expected number of runs |

$$
\begin{array}{ll}
\mathrm{r}(\mathrm{p}) & p \text {-value of } z \\
\mathrm{r}(\mathrm{z}) & z \text { statistic } \\
\mathrm{r}\left(\mathrm{n} \_ \text {runs }\right) & \text { number of runs } \\
\mathrm{r}(\text { Var }) & \text { variance of the number of runs }
\end{array}
$$

## Methods and formulas

runtest begins by calculating the number of observations below the threshold, $n_{0}$; the number of observations above the threshold, $n_{1}$; the total number of observations, $N=n_{0}+n_{1}$; and the number of runs, $r$. These statistics are always reported, so the exact tables of critical values in Swed and Eisenhart (1943) may be consulted if necessary.

The expected number of runs under the null is

$$
\mu_{r}=\frac{2 n_{0} n_{1}}{N}+1
$$

the variance is

$$
\sigma_{r}^{2}=\frac{2 n_{0} n_{1}\left(2 n_{0} n_{1}-N\right)}{N^{2}(N-1)}
$$

and the normal approximation test statistic is

$$
\widehat{z}=\frac{r-\mu_{r}}{\sigma_{r}}
$$

## Acknowledgment

runtest was written by Sean Becketti, author of the Stata Press book Introduction to Time Series Using Stata, Revised Edition.

## References

Edgington, E. S. 1961. Probability table for number of runs of signs of first differences in ordered series. Journal of the American Statistical Association 56: 156-159. https://doi.org/10.2307/2282342.

Madansky, A. 1988. Prescriptions for Working Statisticians. New York: Springer.
Swed, F. S., and C. Eisenhart. 1943. Tables for testing randomness of grouping in a sequence of alternatives. Annals of Mathematical Statistics 14: 66-87. https://doi.org/10.1214/aoms/1177731494.

## Title

scobit - Skewed logistic regression

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

scobit fits a maximum-likelihood skewed logit model.

## Quick start

Skewed logistic regression of binary variable y on x 1 and x 2
scobit y x1 x2
With robust standard errors
scobit $y$ x1 x2, vce(robust)
Same as above, and display coefficients and std. err. with two digits to the right of the decimal scobit y x1 x2, vce(robust) cformat(\%8.2f)

Same as above, and also display $p$-values with two digits to the right of the decimal
scobit y x1 x2, vce(robust) cformat(\%8.2f) pformat(\%5.2f)

## Menu

Statistics $>$ Binary outcomes $>$ Skewed logistic regression

## Syntax

scobit depvar [indepvars] [if] [in] [weight] [, options]
options
Description

## Model

noconstant
offset (varname)
asis
constraints (constraints)
suppress constant term
include varname in model with coefficient constrained to 1
retain perfect predictor variables
apply specified linear constraints
SE/Robust
vce (vcetype)
vcetype may be oim, robust, cluster clustvar, opg, bootstrap, or jackknife

Reporting
level(\#)
nocnsreport
display_options
set confidence level; default is level (95)
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

Maximization
maximize_options
collinear
coeflegend
control the maximization process
keep collinear variables
display legend instead of statistics
indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
bootstrap, by, collect, fp, jackknife, nestreg, rolling, statsby, stepwise, and svy are allowed; see [U] 11.1.10 Prefix commands.
Weights are not allowed with the bootstrap prefix; see [R] bootstrap.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

$\qquad$ Model
noconstant, offset (varname), constraints (constraints); see [R] Estimation options.
asis forces retention of perfect predictor variables and their associated perfectly predicted observations and may produce instabilities in maximization; see [R] probit.

SE/Robust
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
$\qquad$ Reporting
level(\#); see [R] Estimation options.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fint), pformat (\%fmt), sformat (\% $\% m t$ ), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate(\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance (\#), nonrtolerance, and from(init_specs); see [R] Maximize. These options are seldom used.

Setting the optimization type to technique (bhhh) resets the default vcetype to vce (opg).
The following options are available with scobit but are not shown in the dialog box:
collinear, coeflegend; see [R] Estimation options.

## Remarks and examples

Remarks are presented under the following headings:
Skewed logistic model
Robust standard errors

## Skewed logistic model

scobit fits maximum likelihood models with dichotomous dependent variables coded as $0 / 1$ (or, more precisely, coded as 0 and not 0 ).
$>$ Example 1
We have data on the make, weight, and mileage rating of 22 foreign and 52 domestic automobiles. We wish to fit a model explaining whether a car is foreign based on its mileage. Here is an overview of our data:


Sorted by: foreign
Note: Dataset has changed since last saved.
. inspect foreign
foreign: Car origin
Number of observations


| Negative | - | - | - |
| :--- | ---: | ---: | ---: |
| Zero | 52 | 52 | - |
| Positive | 22 | 22 | - |
|  | 74 | 74 | - |
| Total | - |  |  |
| Missing | - |  |  |

74
(2 unique values)
foreign is labeled and all values are documented in the label.
The variable foreign takes on two unique values, 0 and 1 . The value 0 denotes a domestic car, and 1 denotes a foreign car.

The model that we wish to fit is

$$
\operatorname{Pr}(\text { foreign }=1)=F\left(\beta_{0}+\beta_{1} \mathrm{mpg}\right)
$$

where $F(z)=1-1 /\{1+\exp (z)\}^{\alpha}$.
To fit this model, we type

```
. scobit foreign mpg
Fitting logistic model:
Iteration 0: Log likelihood = -45.03321
Iteration 1: Log likelihood = -39.380959
Iteration 2: Log likelihood = -39.288802
Iteration 3: Log likelihood = -39.28864
Iteration 4: Log likelihood = -39.28864
Fitting full model:
Iteration 0: Log likelihood = -39.28864
Iteration 1: Log likelihood = -39.286393
Iteration 2: Log likelihood = -39.284415
Iteration 3: Log likelihood = -39.284234
Iteration 4: Log likelihood = -39.284197
Iteration 5: Log likelihood = -39.284196
```

| Skewed logistic regression |  |  |  | Number of obs Zero outcomes |  | 74 52 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Log likelihood | $=-39.2842$ |  |  | Nonze | outcomes | 22 |
| foreign | Coefficient | Std. err. | z | $P>\|z\|$ | [95\% con | interval] |
| mpg | . 1813879 | . 2407362 | 0.75 | 0.451 | -. 2904463 | . 6532222 |
| _cons | -4.274883 | 1.399305 | -3.06 | 0.002 | -7.017471 | -1.532295 |
| /lnalpha | -. 4450405 | 3.879885 | -0.11 | 0.909 | -8.049476 | 7.159395 |
| alpha | . 6407983 | 2.486224 |  |  | . 0003193 | 1286.133 |

LR test of alpha=1: chi2(1) = 0.01
Prob > chi2 $=0.9249$
Note: Likelihood-ratio tests are recommended for inference with scobit models.
We find that cars yielding better gas mileage are less likely to be foreign. The likelihood-ratio test at the bottom of the output indicates that the model is not significantly different from a logit model. Therefore, we should use the more parsimonious model.

## - Technical note

Stata interprets a value of 0 as a negative outcome (failure) and treats all other values (except missing) as positive outcomes (successes). Thus if the dependent variable takes on the values 0 and 1 , then 0 is interpreted as failure and 1 as success. If the dependent variable takes on the values 0 , 1 , and 2 , then 0 is still interpreted as failure, but both 1 and 2 are treated as successes.

Formally, when we type scobit $y x$, Stata fits the model

$$
\operatorname{Pr}\left(y_{j} \neq 0 \mid \mathbf{x}_{j}\right)=1-1 /\left\{1+\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}\right)\right\}^{\alpha}
$$

## Robust standard errors

If you specify the vce(robust) option, scobit reports robust standard errors as described in [U] 20.22 Obtaining robust variance estimates. For the model of foreign on mpg, the robust calculation increases the standard error of the coefficient on mpg by around $25 \%$ :
. scobit foreign mpg, vce(robust) nolog

| Skewed logistic regression | Number of obs | $=$ | 74 |
| :--- | :--- | :--- | :--- |
|  |  | Zero outcomes | $=$ |
| 52 |  |  |  |
| Log pseudolikelihood $=-39.2842$ | Nonzero outcomes | $=$ | 22 |


| foreign | Coefficient | Robust |  |  |  |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| std. err. | z | P>\|z| | [95\% conf. interval] |  |  |  |
| mpg | .1813879 | .3028487 | 0.60 | 0.549 | -.4121847 | .7749606 |
| _cons | -4.274883 | 1.335521 | -3.20 | 0.001 | -6.892455 | -1.657311 |
| /lnalpha | -.4450405 | 4.71561 | -0.09 | 0.925 | -9.687466 | 8.797385 |
| alpha | .6407983 | 3.021755 |  |  | .0000621 | 6616.919 |

Without vce(robust), the standard error for the coefficient on mpg was reported to be 0.241 , with a resulting confidence interval of $[-0.29,0.65]$.

Specifying the vce(cluster clustvar) option relaxes the independence assumption required by the skewed logit estimator to being just independence between clusters. To demonstrate this, we will switch to a different dataset.

## $>$ Example 2

We are studying the unionization of women in the United States and have a dataset with 26,200 observations on 4,434 women between 1970 and 1988. For our purposes, we will use the variables age (the women were $14-26$ in 1968 and the data thus span the age range of 16-46), grade (years of schooling completed, ranging from 0 to 18), not_smsa ( $28 \%$ of the person-time was spent living outside an SMSA-standard metropolitan statistical area), south ( $41 \%$ of the person-time was in the South), and year. Each of these variables is included in the regression as a covariate along with the interaction between south and year. This interaction, along with the south and year variables, is specified in the scobit command using factor-variables notation, south\#\#c.year. We also have variable union. Overall, $22 \%$ of the person-time is marked as time under union membership and $44 \%$ of these women have belonged to a union.

We fit the following model, ignoring that women are observed an average of 5.9 times each in these data:

```
. use https://www.stata-press.com/data/r18/union, clear
(NLS Women 14-24 in 1968)
```

. scobit union age grade not_smsa south\#\#c.year, nrtol(1e-3)
(output omitted)

| Skewed logistic regression | Number of obs | $=$ | 26,200 |
| :--- | :--- | :--- | :--- |
|  | Zero outcomes | $=$ | 20,389 |
| Log likelihood $=-13540.61$ | Nonzero outcomes | $=$ | 5,811 |


| union | Coefficient | Std. err. | z | $P>\|z\|$ | [95\% con | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| age | . 0185363 | . 0043615 | 4.25 | 0.000 | . 0099879 | . 0270848 |
| grade | . 0452801 | . 0057124 | 7.93 | 0.000 | . 034084 | . 0564761 |
| not_smsa | -. 1886826 | . 0317801 | -5.94 | 0.000 | -. 2509705 | -. 1263947 |
| 1 . south | -1.422372 | . 3949301 | -3.60 | 0.000 | -2.196421 | -. 6483233 |
| year | -. 0133016 | . 0049575 | -2.68 | 0.007 | -. 0230181 | -. 0035851 |
| south\#c.year $1$ | . 0105663 | . 0049233 | 2.15 | 0.032 | . 0009167 | . 0202158 |
| _cons | -10.3557 | 68.97573 | -0.15 | 0.881 | -145.5456 | 124.8342 |
| /lnalpha | 9.136018 | 68.97398 | 0.13 | 0.895 | -126.0505 | 144.3225 |
| alpha | 9283.72 | 640335.1 |  |  | $1.81 \mathrm{e}-55$ | $4.77 e+62$ |
| LR test of alpha=1: chi2(1) = 3.76 |  |  |  |  | Prob > chi2 $=0.0524$ |  |
| Note: Likelihood-ratio tests are recommended for inference with scobit models. |  |  |  |  |  |  |

The reported standard errors in this model are probably meaningless. Women are observed repeatedly, so the observations are not independent. Looking at the coefficients, we find a large southern effect against unionization and a different time trend for the south. The vce (cluster clustvar) option provides a way to fit this model and obtains correct standard errors:

| Skewed logistic regression |  |  |  | Number of obs $=$ 26,200 <br> Zero outcomes $=$ 20,389 <br> Nonzero outcomes $=$ 5,811 |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| (Std. err. adjusted for 4,434 clusters in idcode) |  |  |  |  |  |  |
| union | Coefficient | Robust std. err. | z | $P>\|z\|$ | [95\% conf | interval] |
| age | . 0185363 | . 0084867 | 2.18 | 0.029 | . 0019027 | . 03517 |
| grade | . 0452801 | . 0125765 | 3.60 | 0.000 | . 0206306 | . 0699295 |
| not_smsa | -. 1886826 | . 0642037 | -2.94 | 0.003 | -. 3145194 | -. 0628457 |
| 1. south | -1.422372 | . 5064933 | -2.81 | 0.005 | -2.415081 | -. 4296635 |
| year | -. 0133016 | . 0090622 | -1.47 | 0.142 | -. 0310632 | . 0044599 |
| south\#c.year <br> 1 | . 0105663 | . 0063172 | 1.67 | 0.094 | -. 0018153 | . 0229478 |
| _cons | -10.3557 | . 9412174 | -11.00 | 0.000 | -12.20046 | -8.510951 |
| /lnalpha | 9.136018 | . 7426654 | 12.30 | 0.000 | 7.68042 | 10.59162 |
| alpha | 9283.72 | 6894.697 |  |  | 2165.53 | 39799.71 |

scobit, vce(cluster clustvar) is robust to assumptions about within-cluster correlation. That is, it inefficiently sums within cluster for the standard error calculation rather than attempting to exploit what might be assumed about the within-cluster correlation (as do the xtgee population-averaged models; see [XT] xtgee).

## Technical note

The scobit model can be difficult to fit because of the functional form. Often, it requires many iterations, or the optimizer prints out warning and informative messages during the optimization. For example, without the nrtol ( $1 \mathrm{e}-3$ ) option, the model using the union dataset will not converge. See $[R]$ Maximize for details about the optimizer.

## - Technical note

The main reason for using scobit rather that logit is that the effects of the regressors on the probability of success are not constrained to be the largest when the probability is 0.5 . Rather, the independent variables might show their largest impact when the probability of success is 0.3 or 0.6 . This added flexibility results because the scobit function, unlike the logit function, can be skewed and is not constrained to be mirror symmetric about the 0.5 probability of success.

As Nagler (1994) pointed out, the point of maximum impact is constrained under the scobit model to fall within the interval $\left(0,1-e^{(-1)}\right)$ or approximately $(0,0.63)$. Achen (2002) notes that if we believe the maximum impact to be outside that range, we can instead fit the "power logit" model by simply reversing the 0 s and 1 s of our outcome variable and fitting a scobit model on failure, rather than success. We would need to reverse the signs of the coefficients if we wanted to interpret them in terms of impact on success, or we could leave them as they are and interpret them in terms of impact on failure. The important thing to remember is that the scobit model, unlike the logit model, is not invariant to the choice of which result is assigned to success.

## Stored results

scobit stores the following in e():
Scalars
$e(N) \quad$ number of observations
$e(k) \quad$ number of parameters
e(k_eq)
e(k_aux)
e(k_dv)
e(ll)
e(ll_c)
e(N_f)
e(N_s)
e(alpha)
e(N_clust)
e(chi2_c)
e(rank)
e(ic)
e(rc)
e(converged)
Macros

$$
e(c m d)
$$

e(cmdline)
e(depvar)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(offset)
e(chi2_ct)
e(vce)
e(vcetype)
e(opt)
e(which)
e(ml_method)
e(user)
$e$ (technique)
e(properties)
e(predict)
e(footnote)
e(asbalanced)
e(asobserved)
Matrices

```
e(b)
e(Cns)
e(ilog)
e(gradient)
e(V)
e(V_modelbased)
```

Functions
e(sample)
number of equations in e(b)
number of auxiliary parameters
number of dependent variables
log likelihood
$\log$ likelihood, comparison model
number of failures (zero outcomes)
number of successes (nonzero outcomes)
alpha
number of clusters
$\chi^{2}$ for comparison test
rank of e(V)
number of iterations
return code
1 if converged, 0 otherwise
scobit
command as typed
name of dependent variable
weight type
weight expression
title in estimation output
name of cluster variable
linear offset variable
Wald or LR; type of model $\chi^{2}$ test corresponding to e(chi2_c)
vcetype specified in vce()
title used to label Std. err.
type of optimization
max or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement predict
program used to implement the footnote display
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
constraints matrix
iteration $\log$ (up to 20 iterations)
gradient vector
variance-covariance matrix of the estimators
model-based variance
marks estimation sample
In addition to the above, the following is stored in r() :

```
Matrices
    r(table)
```

matrix containing the coefficients with their standard errors, test statistics, $p$-values, and confidence intervals

Note that results stored in $r()$ are updated when the command is replayed and will be replaced when any r -class command is run after the estimation command.

## Methods and formulas

Skewed logit analysis is an alternative to logit that relaxes the assumption that individuals with initial probability of 0.5 are most sensitive to changes in independent variables.

The log-likelihood function for skewed logit is

$$
\ln L=\sum_{j \in S} w_{j} \ln F\left(\mathbf{x}_{j} \mathbf{b}\right)+\sum_{j \notin S} w_{j} \ln \left\{1-F\left(\mathbf{x}_{j} \mathbf{b}\right)\right\}
$$

where $S$ is the set of all observations $j$ such that $y_{j} \neq 0, F(z)=1-1 /\{1+\exp (z)\}^{\alpha}$, and $w_{j}$ denotes the optional weights. $\ln L$ is maximized as described in [ R$]$ Maximize.

This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce (robust) and vce (cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.
scobit also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

## References

Achen, C. H. 2002. Toward a new political methodology: Microfoundations and ART. Annual Review of Political Science 5: 423-450. https://doi.org/10.1146/annurev.polisci.5.112801.080943.
Nagler, J. 1994. Scobit: An alternative estimator to logit and probit. American Journal of Political Science 38: 230-255. https://doi.org/10.2307/2111343.

## Also see

[R] scobit postestimation - Postestimation tools for scobit
[R] cloglog - Complementary log-log regression
[ R ] glm - Generalized linear models
[R] logistic - Logistic regression, reporting odds ratios
[SVY] svy estimation - Estimation commands for survey data
[U] 20 Estimation and postestimation commands

## Title

scobit postestimation - Postestimation tools for scobit
Postestimation commands predict margins Remarks and examples

## Postestimation commands

The following postestimation commands are available after scobit:

| Command | Description |
| :---: | :---: |
| contrast | contrasts and ANOVA-style joint tests of estimates |
| estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estat (svy) | postestimation statistics for survey data |
| estimates | cataloging estimation results |
| etable | table of estimation results |
| * forecast | dynamic forecasts and simulations |
| lincom | point estimates, standard errors, testing, and inference for linear combinations of coefficients |
| * lrtest | likelihood-ratio test |
| margins | marginal means, predictive margins, marginal effects, and average marginal effects |
| marginsplot | graph the results from margins (profile plots, interaction plots, etc.) |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients |
| predict | probabilities, linear predictions and their SEs, etc. |
| predictnl | point estimates, standard errors, testing, and inference for generalized predictions |
| pwcompare | pairwise comparisons of estimates |
| suest | seemingly unrelated estimation |
| test | Wald tests of simple and composite linear hypotheses |
| testnl | Wald tests of nonlinear hypotheses |

[^68]
## predict

## Description for predict

predict creates a new variable containing predictions such as probabilities, linear predictions, and standard errors.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

```
predict [type] newvar [if] [in] [, statistic nooffset]
predict [type] stub* [if] [in], scores
```


## statistic Description

Main
$\mathrm{pr} \quad$ probability of a positive outcome; the default
$\mathrm{xb} \quad \mathbf{x}_{j} \mathbf{b}$, linear prediction
stdp standard error of the linear prediction
These statistics are available both in and out of sample; type predict ... if e(sample)... if wanted only for the estimation sample.

## Options for predict

$\qquad$ Main
pr , the default, calculates the probability of a positive outcome.
xb calculates the linear prediction.
stdp calculates the standard error of the linear prediction.
nooffset is relevant only if you specified offset (varname) for scobit. It modifies the calculations made by predict so that they ignore the offset variable; the linear prediction is treated as $\mathbf{x}_{j} \mathbf{b}$ rather than as $\mathbf{x}_{j} \mathbf{b}+$ offset $_{j}$.
scores calculates equation-level score variables.
The first new variable will contain $\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)$.
The second new variable will contain $\partial \ln L / \partial \ln \alpha$.

## margins

## Description for margins

margins estimates margins of response for probabilities and linear predictions.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...)...] [options]
```

statistic
Description
pr
probability of a positive outcome; the default
$\mathrm{xb} \quad \mathbf{x}_{j} \mathbf{b}$, linear prediction
stdp not allowed with margins

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see $[R]$ margins.

## Remarks and examples

Once you have fit a model, you can obtain the predicted probabilities by using the predict command for both the estimation sample and other samples; see [U] 20 Estimation and postestimation commands and $[R]$ predict. Here we will make only a few additional comments.
predict without arguments calculates the predicted probability of a positive outcome. With the xb option, it calculates the linear combination $\mathbf{x}_{j} \mathbf{b}$, where $\mathbf{x}_{j}$ are the independent variables in the $j$ th observation and $\mathbf{b}$ is the estimated parameter vector.

With the stdp option, predict calculates the standard error of the prediction, which is not adjusted for replicated covariate patterns in the data.
> Example 1
In example 1 of [R] scobit, we fit the model scobit foreign mpg. To obtain predicted probabilities, we type

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. keep make mpg weight foreign
. scobit foreign mpg
    (output omitted)
. predict p
(option pr assumed; Pr(foreign))
. summarize foreign p
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline foreign & 74 & .2972973 & .4601885 & 0 & 1 \\
\(p\) & 74 & .2974049 & .182352 & .0714664 & .871624
\end{tabular}
```


## Also see

[R] scobit - Skewed logistic regression
[U] 20 Estimation and postestimation commands

## Title

sdtest - Variance-comparison tests

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

sdtest performs tests on the equality of standard deviations (variances). In the first form, sdtest tests that the standard deviation of varname is \#. In the second form, sdtest performs the same test, using the standard deviations of the two groups defined by groupvar. In the third form, sdtest tests that varname $_{1}$ and varname $2_{2}$ have the same standard deviation.
sdtesti is the immediate form of sdtest; see [U] 19 Immediate commands.
Both the traditional $F$ test for the homogeneity of variances and Bartlett's generalization of this test to $K$ samples are sensitive to the assumption that the data are drawn from an underlying Gaussian distribution. See, for example, the cautionary results discussed by Markowski and Markowski (1990). Levene (1960) proposed a test statistic for equality of variance that was found to be robust under nonnormality. Then Brown and Forsythe (1974) proposed alternative formulations of Levene's test statistic that use more robust estimators of central tendency in place of the mean. These reformulations were demonstrated to be more robust than Levene's test when dealing with skewed populations.
robvar reports Levene's robust test statistic $\left(W_{0}\right)$ for the equality of variances between the groups defined by groupvar and the two statistics proposed by Brown and Forsythe that replace the mean in Levene's formula with alternative location estimators. The first alternative ( $W_{50}$ ) replaces the mean with the median. The second alternative replaces the mean with the $10 \%$ trimmed mean $\left(W_{10}\right)$.

## Quick start

Test that the standard deviation of $v 1$ is equal to 2

```
sdtest v1=2
```

Equality of standard deviations (variances) test for v1 comparing the two groups defined by catvar1 sdtest v1, by(catvar1)

Robust equality of variances test for v1 comparing the groups defined by catvar1 robvar v1, by (catvar1)

Compare the variances of v2 and v3
sdtest v2 == v3
Same as above, but with separate tests for each group defined by catvar2
by catvar2, sort: sdtest v2 == v3
Test $\mathrm{sd}_{1}=\mathrm{sd}_{2}$ for $\mathrm{sd}_{1}=34, \mathrm{sd}_{2}=45, N_{1}=143$, and $N_{2}=184$
sdtesti 143 . 34184 . 45

## Menu

sdtest
Statistics $>$ Summaries, tables, and tests $>$ Classical tests of hypotheses $>$ Variance-comparison test

## sdtesti

Statistics $>$ Summaries, tables, and tests $>$ Classical tests of hypotheses $>$ Variance-comparison test calculator robvar

Statistics $>$ Summaries, tables, and tests $>$ Classical tests of hypotheses $>$ Robust equal-variance test

## Syntax

One-sample variance-comparison test

$$
\text { sdtest varname }==\#[\text { if }][\mathrm{in}][, \underline{l} \mathrm{evel}(\#)]
$$

Two-sample variance-comparison test using groups

$$
\text { sdtest varname }[\text { if }][\text { in }] \text {, by (groupvar) }[\underline{l} \operatorname{level}(\#)]
$$

Two-sample variance-comparison test using variables

$$
{\text { sdtest } \text { varname }_{1}==\text { varname }_{2}[\text { if }][\text { in }][, \text { level(\#) }]}^{2}
$$

Immediate form of one-sample variance-comparison test

```
sdtesti # obs {#mean |.} # sd # val [, level(#)]
```

Immediate form of two-sample variance-comparison test


Robust tests for equality of variances
robvar varname [if] [in], by (groupvar)
by and collect are allowed with sdtest and robvar, and collect is allowed with sdtesti; see [U] 11.1.10 Prefix commands.

## Options

level(\#) specifies the confidence level, as a percentage, for confidence intervals of the means. The default is level (95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals.
by (groupvar) specifies the groupvar that defines the groups to be compared. For sdtest, there should be two groups, but for robvar there may be more than two groups. Do not confuse the by () option with the by prefix; both may be specified.

## Remarks and examples

Remarks are presented under the following headings:

Basic form<br>Immediate form<br>Robust test

## Basic form

sdtest performs two different statistical tests: one testing equality of variances and the other testing that the standard deviation is equal to a known constant. Which test it performs is determined by whether you type a variable name or a number to the right of the equal sign.
> Example 1: One-sample test of variance
We have a sample of 74 automobiles. For each automobile, we know the mileage rating. We wish to test whether the overall standard deviation is 5 mpg :

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. sdtest mpg == 5
One-sample test of variance
```

| Variable | Obs | Mean | Std. err. | Std. dev. | [95\% conf | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| mpg | 74 | 21.2973 | . 6725511 | 5.785503 | 19.9569 | 22.63769 |
| H0. $\begin{aligned} \text { sd } & =s d(m p g) ~\end{aligned}$ |  |  |  | chi2 $=97.7384$ |  |  |
|  |  |  |  | Degree | f freedom | 73 |
| Ha : |  | Ha: sd ! $=5$ |  |  | Ha: sd > 5 |  |
| $\operatorname{Pr}(\mathrm{C}<$ | . 9717 | $2 * \operatorname{Pr}(\mathrm{C}>\mathrm{c})=0.0565$ |  |  | $\operatorname{Pr}(\mathrm{C}>\mathrm{c})=0.0283$ |  |

## > Example 2: Variance ratio test

We are testing the effectiveness of a new fuel additive. We run an experiment on 12 cars, running each without and with the additive. The data can be found in $[R]$ ttest. The results for each car are stored in the variables mpg1 and mpg2:

| Variable | Obs | Mean | Std. err. | Std. dev. | [95\% conf | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| mpg1 | 12 | 21 | . 7881701 | 2.730301 | 19.26525 | 22.73475 |
| mpg2 | 12 | 22.75 | . 9384465 | 3.250874 | 20.68449 | 24.81551 |
| Combined | 24 | 21.875 | . 6264476 | 3.068954 | 20.57909 | 23.17091 |
| $\begin{aligned} & \text { H0: }: \\ & \text { ratio }=s d(m p g 1) / \operatorname{sd}(m p g 2) \end{aligned}$ |  |  |  | $\begin{aligned} f & =0.7054 \\ \text { Degrees of freedom } & =11,11\end{aligned}$ |  |  |
|  |  |  |  |  |  |  |
| Ha: ratio < 1 |  | Ha: ratio != 1 |  |  | Ha: ratio > 1 |  |
|  |  | $2 * \operatorname{Pr}(\mathrm{~F}<\mathrm{f})=0.5725$ |  |  | $\operatorname{Pr}(\mathrm{F}>\mathrm{f})=0.7138$ |  |

We cannot reject the hypothesis that the standard deviations are the same.

In [R] ttest, we draw an important distinction between paired and unpaired data, which, in this example, means whether there are 12 cars in a before-and-after experiment or 24 different cars. For sdtest, on the other hand, there is no distinction. If the data had been unpaired and stored as described in [R] ttest, we could have typed sdtest mpg, by (treated), and the results would have been the same.

## Immediate form

## > Example 3: sdtesti

Immediate commands are used not with data, but with reported summary statistics. For instance, to test whether a variable on which we have 75 observations and a reported standard deviation of 6.5 comes from a population with underlying standard deviation 6 , we would type

| . sdtesti 75.6 .56 |
| :--- |
| One-sample test of variance |

The mean plays no role in the calculation, so it may be omitted.
To test whether the variable comes from a population with the same standard deviation as another for which we have a calculated standard deviation of 7.5 over 65 observations, we would type

| . sdtesti 75.6 .565 .7 .5 |
| :--- |
| Variance ratio test |

## Robust test

> Example 4: robvar
We wish to test whether the standard deviation of the length of stay for patients hospitalized for a given medical procedure differs by gender. Our data consist of observations on the length of hospital stay for 1778 patients: 884 males and 894 females. Length of stay, lengthstay, is highly skewed (skewness coefficient $=4.912591$ ) and thus violates Bartlett's normality assumption. Therefore, we use robvar to compare the variances.


For these data, we cannot reject the null hypothesis that the variances are equal. However, Bartlett's test yields a significance probability of 0.0319 because of the pronounced skewness of the data.

## Technical note

robvar implements both the conventional Levene's test centered at the mean and a median-centered test. In a simulation study, Conover, Johnson, and Johnson (1981) compare the properties of the two tests and recommend using the median test for asymmetric data, although for small sample sizes the test is somewhat conservative. See Carroll and Schneider (1985) for an explanation of why both mean- and median-centered tests have approximately the same level for symmetric distributions, but for asymmetric distributions the median test is closer to the correct level.

## Stored results

sdtest and sdtesti store the following in $r()$ :
Scalars

| $r(N)$ | number of observations |
| :--- | :--- |
| $r\left(p \_1\right)$ | lower one-sided $p$-value |
| $r(p-u)$ | upper one-sided $p$-value |
| $r(p)$ | two-sided $p$-value |
| $r(F)$ | $F$ statistic |
| $r(s d)$ | standard deviation |
| $r(s d-1)$ | standard deviation for first variable |
| $r(s d-2)$ | standard deviation for second variable |
| $r(d f)$ | degrees of freedom |
| $r(d f-1)$ | numerator degrees of freedom |
| $r(d f-2)$ | denominator degrees of freedom |
| $r(c h i 2)$ | $\chi^{2}$ |

robvar stores the following in r() :
Scalars

| $r(N)$ | number of observations |
| :--- | :--- |
| $r(w 50)$ | Brown and Forsythe's $F$ statistic (median) |
| $r(p-w 50)$ | Brown and Forsythe's $p$-value |
| $r(w 0)$ | Levene's $F$ statistic |
| $r(p-w 0)$ | Levene's $p$-value |
| $r(\mathrm{w} 10)$ | Brown and Forsythe's $F$ statistic (trimmed mean) |
| $r(\mathrm{p}-\mathrm{w} 10)$ | Brown and Forsythe's $p$-value (trimmed mean) |
| $r(d f-1)$ | numerator degrees of freedom |
| $r(d f-2)$ | denominator degrees of freedom |

## Methods and formulas

See Armitage et al. (2002, 149-153) or Bland (2015, 144-145) for an introduction and explanation of the calculation of these tests.

The test for $\sigma=\sigma_{0}$ is given by

$$
\chi^{2}=\frac{(n-1) s^{2}}{\sigma_{0}^{2}}
$$

which is distributed as $\chi^{2}$ with $n-1$ degrees of freedom.
The test for $\sigma_{x}^{2}=\sigma_{y}^{2}$ is given by

$$
F=\frac{s_{x}^{2}}{s_{y}^{2}}
$$

which is distributed as $F$ with $n_{x}-1$ and $n_{y}-1$ degrees of freedom.
Let $X_{i j}$ be the $j$ th observation of $X$ for the $i$ th group. Let $Z_{i j}=\left|X_{i j}-\bar{X}_{i}\right|$, where $\bar{X}_{i}$ is the mean of $X$ in the $i$ th group. Levene's test statistic is

$$
W_{0}=\frac{\sum_{i} n_{i}\left(\bar{Z}_{i}-\bar{Z}\right)^{2} /(g-1)}{\sum_{i} \sum_{j}\left(Z_{i j}-\bar{Z}_{i}\right)^{2} / \sum_{i}\left(n_{i}-1\right)}
$$

where $n_{i}$ is the number of observations in group $i$ and $g$ is the number of groups. $W_{50}$ is obtained by replacing $\bar{X}_{i}$ with the $i$ th group median of $X_{i j}$, whereas $W_{10}$ is obtained by replacing $\bar{X}_{i}$ with the $10 \%$ trimmed mean for group $i$.

## References

Armitage, P., G. Berry, and J. N. S. Matthews. 2002. Statistical Methods in Medical Research. 4th ed. Oxford: Blackwell.
Bland, M. 2015. An Introduction to Medical Statistics. 4th ed. Oxford: Oxford University Press.
Brown, M. B., and A. B. Forsythe. 1974. Robust tests for the equality of variances. Journal of the American Statistical Association 69: 364-367. https://doi.org/10.2307/2285659.
Carroll, R. J., and H. Schneider. 1985. A note on Levene's tests for equality of variances. Statistics and Probability Letters 3: 191-194. https://doi.org/10.1016/0167-7152(85)90016-1.

Conover, W. J., M. E. Johnson, and M. M. Johnson. 1981. A comparative study of tests for homogeneity of variances, with applications to the outer continental shelf bidding data. Technometrics 23: 351-361. https://doi.org/10.2307/1268225.
Gastwirth, J. L., Y. R. Gel, and W. Miao. 2009. The impact of Levene's test of equality of variances on statistical theory and practice. Statistical Science 24: 343-360. https://doi.org/10.1214/09-STS301.

Levene, H. 1960. Robust tests for equality of variances. In Contributions to Probability and Statistics: Essays in Honor of Harold Hotelling, ed. I. Olkin, S. G. Ghurye, W. Hoeffding, W. G. Madow, and H. B. Mann, 278-292. Menlo Park, CA: Stanford University Press.
Markowski, C. A., and E. P. Markowski. 1990. Conditions for the effectiveness of a preliminary test of variance. American Statistician 44: 322-326. https://doi.org/10.2307/2684360.

## Also see

$[\mathrm{R}]$ ttest $-t$ tests (mean-comparison tests)

## Title

search - Search Stata documentation and other resources

| Description | Quick start | Menu |
| :--- | :--- | :--- |
| Syntax | Options for search | Option for set searchdefault |
| Remarks and examples | Acknowledgment | Also see |

## Description

search searches a keyword database and the Internet for Stata materials related to your query.
Capitalization of the words following search is irrelevant, as is the inclusion or exclusion of special characters such as commas and hyphens.
set searchdefault affects the default behavior of the search command. all is the default.
search, all is the best way to search for information on a topic across all sources, including the system help, the FAQs at the Stata website, the Stata Journal, and all Stata-related Internet sources including community-contributed additions. From the results, you can click to go to a source or to install additions.

## Quick start

Search local keyword database and materials available via Stata's net command for all words (word1, word2, and word3)
search word1 word2 word3
Same as above, but match any word (word1, word2, or word3)
search word1 word2 word3, or
Search for Stata Journal articles about new commands like regress
search regress, sj
Search the manuals for entries about tabulate and related commands
search tabulate, manual

## Menu

Help > Search...

## Syntax

search word [word ...] [, search_options]
set searchdefault $\{$ all $\mid$ local $\mid$ net $\} \quad[, \underline{p e r m a n e n t l y}]$
search_options Description

| all | search across both the local keyword database and the net material; the default |
| :---: | :---: |
| local | search using Stata's keyword database |
| net | search across materials available via Stata's net command |
| author | search by author's name |
| entry | search by entry ID |
| exact | search across both the local keyword database and the net materials; prevents matching on abbreviations |
| faq | search the FAQs posted to the Stata and other select websites |
| historical | search entries that are of historical interest only |
| or | list an entry if any of the words typed after search are associated with the entry |
| manual | arch the entries in the Stata Documentation |
| sj | search the entries in the Stata Journal |

## Options for search

all, the default (unless changed by set searchdefault), specifies that the search be performed across both the local keyword database and the net materials. The results of a search performed with all and no other options will be displayed in the Viewer window.
local specifies that the search be performed using only Stata's keyword database. The results of a search performed with local and no other options will be displayed in the Viewer window.
net specifies that the search be performed across the materials available via Stata's net command. Using search word [word ...], net is equivalent to typing net search word [word ...] (without options); see [R] net search. The results of a search performed with net and no other options will be displayed in the Viewer window.
author specifies that the search be performed on the basis of the author's name rather than keywords. A search with the author option is performed on the local keyword database only, and the results are displayed in the Results window.
entry specifies that the search be performed on the basis of entry IDs rather than keywords. A search with the entry option is performed on the local keyword database only, and the results are displayed in the Results window.
exact prevents matching on abbreviations. A search with the exact option is performed across both the local keyword database and the net materials, and the results are displayed in the Results window.
faq limits the search to the FAQs on the Stata and other select websites. A search with the faq option is performed on the local keyword database only, and the results are displayed in the Results window.
historical adds to the search entries that are of historical interest only. By default, such entries are not listed. Past entries are classified as historical if they discuss a feature that later became an official part of Stata. Updates to historical entries will always be found, even if historical is not specified. A search with the historical option is performed on the local keyword database only, and the results are displayed in the Results window.
or specifies that an entry be listed if any of the words typed after search are associated with the entry. The default is to list the entry only if all the words specified are associated with the entry. A search with the or option is performed on the local keyword database only, and the results are displayed in the Results window.
manual limits the search to entries in the Stata Documentation; that is, the search is limited to the User's Guide and all the reference manuals. A search with the manual option is performed on the local keyword database only, and the results are displayed in the Results window.
sj limits the search to entries in the Stata Journal; see [R] sj. A search with the sj option is performed on the local keyword database only, and the results are displayed in the Results window.

## Option for set searchdefault

permanently specifies that, in addition to making the change right now, the searchdefault setting be remembered and become the default setting when you invoke Stata.

## Remarks and examples

Remarks are presented under the following headings:
Introduction
Internet searches
Author searches
Entry ID searches
Return codes

## Introduction

See [U] 4 Stata's help and search facilities for a tutorial introduction to search. search is one of Stata's most useful commands. To understand the advanced features of search, you need to know how it works.
search has a database-files - containing the titles, etc., of every entry in the Stata Documentation, undocumented help files, NetCourses, Stata Press books, FAQs posted on the Stata website, videos posted on the Stata YouTube channel, selected articles on StataCorp's official blog, selected communitycontributed FAQs and examples, and the articles in the Stata Journal. In these files is a list of words, called keywords, associated with each entry.

When you type search $x y z$, search reads the database and compares the list of keywords with $x y z$. If it finds $x y z$ in the list or a keyword that allows an abbreviation of $x y z$, it displays the entry.

When you type search $x y z a b c$, search does the same thing but displays an entry only if it contains both keywords. The order does not matter, so you can search linear regression or search regression linear.

Obviously, how many entries search finds depends on how the search database was constructed. We have included a plethora of keywords under the theory that, for a given request, it is better to list too much rather than risk listing nothing at all. Still, you are in the position of guessing the keywords. Do you look up normality test, normality tests, or tests of normality? Well, normality test would be best, but all would work. In general, use the singular, and strike the unnecessary words. For guidelines for specifying keywords, see [U] 4.6 More on search.
set searchdefault allows you to specify where search searches. set searchdefault all, the default, indicates that both the keyword database and the Internet are to be searched. set searchdef ault local restricts search to using only Stata's keyword database. set searchdefault net restricts search to searching only the Internet.

## Internet searches

search with the net option searches the Internet for community-contributed additions to Stata, including, but not limited to, community-contributed additions published in the Stata Journal (SJ). search keywords, net performs the same search as the command net search (with no options); see $[R]$ net search.

```
. search random effect, net
Search of web resources from Stata and other users
(contacting https://www.stata.com)
# packages found (Stata Journal listed first)
------------------------------------------------------------
    (output omitted)
st0468_1 from http://www.stata-journal.com/software/sj18-4
    SJ18-4 st0468_1. Update: Estimate hybrid and... / Update: Estimate hybrid
    and correlated random- / effects and Mundlak mixed-effects models for /
    linear and nonlinear outcomes / by Reinhard Schunck, GESIS --
    Leibniz-Institute / for the Social Sciences, Cologne, Germany / Francisco
st0543 from http://www.stata-journal.com/software/sj18-4
    SJ18-4 st0543. Fit dynamic random-effects probit... / Fit dynamic
    random-effects probit models with / unobserved heterogeneity / by Raffaele
    Grotti, / Department of Political and Social Sciences / European
    University Institute / San Domenico di Fiesole, Italy / Giorgio Cutuli,
    (output omitted)
```

(end of search)

## Author searches

search ordinarily compares the words following search with the keywords for the entry. If you specify the author option, however, it compares the words with the author's name. In the search database, we have filled in author names for all SJ inserts.

For instance, in the Acknowledgments of [R] kdensity, you will discover the name Isaías $H$. Salgado-Ugarte. You want to know if he has written any articles in the SJ. To find out, you type

```
. search Salgado-Ugarte, author
    (output omitted)
```

Names like Salgado-Ugarte are confusing to many people. search does not require you to specify the entire name; what you type is compared with each "word" of the name and, if any part matches, the entry is listed. The hyphen is a special character, and you can omit it. Thus, you can obtain the same list by looking up Salgado, Ugarte, or Salgado Ugarte without the hyphen.

## Entry ID searches

If you specify the entry option, search compares what you have typed with the entry ID. The entry ID is not the title-it is the reference listed to the left of the title that tells you where to look. For instance, in
[R]

```
regress . . . . . . . . . . . . . . . . . . . . . . Linear regression
(help regress)
```

[R] regress is the entry ID. This is a reference, of course, to this manual. In
FAQ

```
. . . . . . . . . . . Analysis of multiple failure-time survival data
. . . . . . . . . . . . . . . . . . . . . . . M. Cleves and I. Canette
07/09 How do I analyze multiple failure-time data using Stata?
    https://www.stata.com/support/faqs/statistics/multiple-failure-
    type-data/
```

"FAQ" is the entry ID. In

```
SJ-7-1 st0118 . . A survey on survey stat.: What is and can be done in Stata
    . . . . . . . . . . . . . . . . . . . . . . F. Kreuter and R. Valliant
    Q1/07 SJ7(1):1--21
        (no commands)
    discusses survey issues in analyzing complex survey
    data and describes some of Stata's capabilities for
    such analyses
```

"SJ-7-1" is the entry ID.
search with the entry option searches these entry IDs.
Thus you could generate a table of contents for the User's Guide by typing

```
. search [U], entry
(output omitted)
```

You could generate a table of contents for Stata Journal, Volume 1, Issue 1, by typing

```
. search sj-1-1, entry
    (output omitted)
```


## Return codes

In addition to indexing the entries in the User's Guide and all the Reference manuals, search also can be used to search return codes.

To see information on return code 131, type
. search rc 131
[P] error
not possible with test;
You requested a test of a hypothesis that is nonlinear in the variables. test tests only linear hypotheses. Use testnl.
(output omitted)
If you want a list of all Stata return codes, type
. search error, entry
(output omitted)

## Acknowledgment

We thank Nicholas J. Cox of the Department of Geography at Durham University, UK, who is coeditor of the Stata Journal and author of Speaking Stata Graphics for his contributions to the search command.

## Also see

[R] help - Display help in Stata
$[R]$ net search - Search the Internet for installable packages
[U] 4 Stata's help and search facilities

## Title

serrbar - Graph standard error bar chart

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Acknowledgment | Also see |

## Description

serrbar is typically used with a dataset containing means, standard deviations or standard errors, and an xvar. serrbar uses these data to create a standard error bar chart. The means are plotted against xvar, and error bars around the means have a width determined by the standard deviation or standard error. While it is most common to use serrbar with this type of data, serrbar may also be used to create a scatterplot with error bars for other types of data.

## Quick start

Plot of $y$ versus $x$ with error bars representing $y \pm s$
serrbar y s x
Same as above, but with error bars for $\mathrm{y} \pm 2 \times \mathrm{s}$
serrbar y s x, scale(2)

## Menu

Statistics $>$ Other $>$ Quality control $>$ Standard error bar chart

## Syntax

serrbar mvar svar xvar [if] [in] [, options]
options Description
Main
scale(\#) scale length of graph bars; default is scale(1)
Error bars
rcap_options affect rendition of capped spikes
Plotted points
mvopts (scatter_options) affect rendition of plotted points
Add plots
addplot (plot) add other plots to generated graph
Y axis, X axis, Titles, Legend, Overall
twoway_options any options other than by () documented in [G-3] twoway_options

## Options

$\qquad$ Main
scale(\#) controls the length of the bars. The upper and lower limits of the bars will be mvar + scale () $\times$ svar and mvar - scale ( $) \times$ svar. The default is scale (1).

Error bars
rcap_options affect the rendition of the plotted error bars (the capped spikes). See [G-2] graph twoway rcap.
$\int \sqrt{\text { Plotted points }}$
mvopts (scatter_options) affects the rendition of the plotted points (mvar versus xvar). See [G-2] graph twoway scatter.
$\leadsto$ Add plots
addplot (plot) provides a way to add other plots to the generated graph; see [G-3] addplot_option.
$\qquad$ $\sqrt{\mathrm{Y} \text { axis, } \mathrm{X} \text { axis, Titles, Legend, Overall }}$
twoway_options are any of the options documented in [G-3] twoway_options, excluding by (). These include options for titling the graph (see [G-3] title_options) and for saving the graph to disk (see [G-3] saving_option).

## Remarks and examples

$>$ Example 1
In quality-control applications, the three most commonly used variables with this command are the process mean, process standard deviation, and time. For instance, we have data on the average weights and standard deviations from an assembly line in San Francisco for the period January 8 to January 16. Our data are

```
. use https://www.stata-press.com/data/r18/assembly
. list, sep(0) divider
```

| date | mean | std |
| ---: | ---: | ---: |
| 108 | 192.22 | 3.94 |
| 109 | 192.64 | 2.83 |
| 110 | 192.37 | 4.58 |
| 113 | 194.76 | 3.25 |
| 114 | 192.69 | 2.89 |
| 115 | 195.02 | 1.73 |
| 116 | 193.40 | 2.62 |

We type serrbar mean std date, scale(2) but, after seeing the result, decide to make it fancier:

```
. serrbar mean std date, scale(2) title("Observed weight variation")
> sub("San Francisco plant, 1/8 to 1/16") yline(195) yaxis(1 2)
> ylab(195, axis(2)) ytitle("", axis(2))
```

Observed weight variation


## Acknowledgment

serrbar was written by Nicholas J. Cox of the Department of Geography at Durham University, UK, who is coeditor of the Stata Journal and author of Speaking Stata Graphics.

## Also see

[R] QC - Quality control charts

## Title

set - Overview of system parameters

## Description Syntax Remarks and examples Also see

## Description

This entry provides a reference to Stata's set commands. For many entries, more thorough information is provided elsewhere; see the Reference field in each entry below for the location of this information.

To reset system parameters to factory defaults, see [R] set_defaults.

## Syntax

set $[$ setcommand $\ldots$. $]$
set typed without arguments is equivalent to query typed without arguments.

## Remarks and examples

```
set adosize
    Syntax: set adosize # [, permanently]
    Default: 1,000
    Description: sets the maximum amount of memory that automatically loaded do-files
        may consume. 10\leq#\leq10000.
    Reference: [P] sysdir
set autotabgraphs (Windows only)
    Syntax: set autotabgraphs {on|off } [, permanently]
    Default: off
    Description: determines whether graphs are created as tabs within one window or as separate
            windows.
    Reference: help autotabgraphs
set cformat
    Syntax: set clevel # [, permanently]
    Default: 95
```

Syntax:
Description: specifies the output format of coefficients, standard errors, and confidence limits in coefficient tables. fmt is a numerical format; see [D] format.
Reference: [R] set cformat

```
set clevel
```

```
set clevel
```

Description: sets the default credible level for credible intervals for all commands that report credible intervals. $10.00 \leq \# \leq 99.99$, and \# can have at most two digits after the decimal point.
Reference: [BAYES] set clevel
set coeftabresults
Syntax: set coeftabresults $\{$ on $\mid$ off $\}$
Default: on
Description: determines whether coefficient table results are stored in $r()$. There is no permanently option because permanently is implied.
Reference: help coeftabresults
set collect_double
Syntax: set collect_double $\{$ on $\mid$ off $\} \quad$ [, permanently $]$
Default: on
Description: controls the storage type for numeric values that are saved in collections when using collect save.
Reference: [TABLES] set collect_double
set collect_label
Syntax: set collect_label $\{$ default $\mid$ label $\} \quad[$, permanently $]$
Default: default
Description: controls the default labels used in tables created by collect.
Reference: [TABLES] set collect_label
set collect_style
Syntax: set collect_style $\{$ default $\mid$ style $\} \quad[$, permanently $]$
Default: default
Description: controls the default styles used in tables created by collect.
Reference: [TABLES] set collect_style
set collect_warn
Syntax: set collect_warn $\{$ on $\mid$ off $\} \quad[$, permanently $]$
Default: on
Description: controls whether collect shows notes warning about unrecognized tags.
Reference: [TABLES] set collect_warn

```
set conren (Unix console only)
    Syntax 1: set conren
    Syntax 2: set conren clear
    Syntax 3: set conren [sf|bf |it]
        {\underline{result | [txt | text] | input | error | link | hilite }}
        [char[char...]]
    Syntax 4: set conren {ulon | uloff} [char [char...]]
    Syntax 5: set conren reset [char [char...]]
Description: can possibly make the output on your screen appear prettier.
                            set conren displays a list of the currently defined display codes.
        set conren clear clears all codes.
        set conren followed by a font type (bf, sf, or it) and display context (result,
        error, link, or hilite) and then followed by a series of space-separated
        characters sets the code for the specified font type and display context. If the font
        type is omitted, the code is set to the same specified code for all three font types.
        set conren ulon and set conren uloff set the codes for turning on and off
        underlining.
        set conren reset sets the code that will turn off all display and underlining codes.
    Reference: [GSU] conren
set copycolor (Mac and Windows only)
```



```
    Default: automatic
    Description: determines how colors are handled when graphs are copied to the Clipboard.
    Reference: [G-2] set printcolor
set dockable (Windows only)
    Syntax: set dockable {on|off } [, permanently]
    Default: on
    Description: determines whether to enable the use of dockable window characteristics,
        including the ability to dock or tab a window into another window.
    Reference: help dockable
```

set docx_hardbreak
Syntax: set docx_hardbreak $\{$ on $\mid$ off $\}$
Default: off
Description: determines whether spaces are added after hard line breaks within text blocks.
Reference: [RPT] set docx
set docx_maxtable
Syntax: set docx_maxtable \# [, permanently]
Default: 500
Description: sets the maximum number of tables allowed in putdocx.
Reference: [RPT] putdocx table

```
set docx_paramode
    Syntax: set docx_paramode {on|off}
    Default: off
    Description: determines whether empty lines in a text block signal the beginning of a
    new paragraph.
    Reference: [RPT] set docx
```

set doeditbackup
Syntax: set doeditbackup $\{$ on $\mid$ off $\}$
Default: on
Description: determines whether to enable automatic backups of documents in the Do-file Editor.
Reference: help doeditbackup
set dots
Syntax: set dots $\{$ on $\mid$ off $\} \quad[$, permanently]
Default: on
Description: enables or disables commands that support the dots() option from reporting a
dot each time statistics are computed from a sample or resample of the dataset.
Reference: help dots
set doublebuffer (Windows only)
Syntax: set doublebuffer $\{$ on $\mid$ off $\}$ [, permanently $]$
Default: on
Description: enables or disables double buffering of the Results, Viewer, and Data Editor
windows. Double buffering prevents the windows from flickering when redrawn
or resized. Users who encounter performance problems such as the Results window
outputting very slowly should disable double buffering.
Reference: help doublebuffer
set $d p$
Syntax: set dp \{comma|period $\}$ [, permanently]
Default: period
Description: determines whether a period or a comma is to be used as the decimal point.
Reference: [D] format
set dtable_style
Syntax: set dtable_style $\{$ dtable $\mid$ style $\}$ [, permanently]
Default: dtable
Description: controls the default styles used in tables created by dtable.
Reference: [TABLES] set dtable_style
set dtascomplevel
Syntax: set dtascomplevel \#
Default: 1
Description: sets the compression level for frames save, where \# can be any integer
between 0 and 9 .
Reference: help dtascomplevel

```
set emptycells
    Syntax: set emptycells {keep|drop} [, permanently]
    Default: keep
    Description: sets what to do with empty cells in interactions.
    Reference: [R] set emptycells
set etable_style
    Syntax: set etable_style {etable|style} [, permanently]
    Default: etable
    Description: controls the default styles used in tables created by etable.
    Reference: [TABLES] set etable_style
```

set fastscroll (Unix and Windows only)
Syntax: set fastscroll $\{$ on $\mid$ off $\}$ [, permanently $]$
Default: on
Description: sets the scrolling method for new output in the Results window. Setting
fastscroll to on is faster but can be jumpy. Setting fastscroll to off
is slower but smoother.
Reference: help fastscroll
set floatwindows (Windows only)
Syntax: set floatwindows $\{$ on $\mid$ off $\}$
Default: off
Description: determines whether to enable floating window behavior for dialog boxes and dockable
window. The term "float" in this context means that a window will always float
over the main Stata window; these windows cannot be placed behind the main Stata
window. There is no permanently option because permanently is implied.
Reference: help floatwindows
set fredkey
Syntax: set fredkey key [, permanently]
Description: sets the API key for importing data from the Federal Reserve Economic Data.
Reference: [D] import fred
set fvbase
Syntax: $\quad$ set fvbase $\{$ on | off $\}$
Description: specifies whether to automatically determine the default base level for factor variables.
Reference: help fvbase
set fvlabel
Syntax: set fvlabel \{ on |off $\}$ [, permanently]
Description: specifies whether to display factor-variable value labels in coefficient tables.
Reference: [R] set showbaselevels

```
Syntax: set fvtrack \{term|factor \(\}\) [, permanently]
```

Description: allows you to control how Stata keeps track of factor levels when you use factor-variables notation.
Reference: help fvtrack
set fvwrap
Syntax: set fvwrap \# [, permanently]
Description: specifies that long value labels wrap \# lines in coefficient tables.
Reference: [R] set showbaselevels
set fvwrapon
Syntax:
Description:
set fvwrapon \{word|width \} [, permanently]
specifies whether value labels that wrap will break at word boundaries or break based on available space.
Reference: $[\mathrm{R}]$ set showbaselevels
set graphics
Syntax: set graphics $\{$ on | off $\}$
Default: on; default is off for console Stata
Description: determines whether graphs are displayed on your monitor.
Reference: [G-2] set graphics
set haverdir
Syntax: set haverdir "path" [, permanently]
Description: specifies the directory where the Haver databases are stored.
Reference: [D] import haver
set httpproxy
Syntax: set httpproxy $\{$ on $\mid$ off $\}$ [, init ]
Default: off
Description: turns on/off the use of a proxy server. There is no permanently option because permanently is implied.
Reference: $[\mathrm{R}]$ netio
set httpproxyauth
Syntax: set httpproxyauth $\{$ on $\mid$ off $\}$
Default: off
Description: determines whether authorization is required for the proxy server. There is no permanently option because permanently is implied.
Reference: [R] netio
set httpproxyhost
Syntax: set httpproxyhost ["] name ["]
Description: sets the name of a host to be used as a proxy server. There is no permanently option because permanently is implied.
Reference: $[\mathrm{R}]$ netio

## set httpproxyport

Syntax: set httpproxyport \#
Default: 8080 if Stata cannot autodetect the proper setting for your computer.
Description: sets the port number for a proxy server. There is no permanently option because permanently is implied.
Reference: [R] netio

```
set httpproxypw
```

Syntax: set httpproxypw ["] password ["]
Description: sets the appropriate password. There is no permanently option because permanently is implied.
Reference: [R] netio
set httpproxyuser
Syntax: set httpproxyuser ["] name["]
Description: sets the appropriate user ID. There is no permanently option because permanently is implied.
Reference: [R] netio
set include_bitmap (Mac only)
Syntax: set include_bitmap $\{$ on $\mid$ off $\}$ [, permanently]
Default: on
Description: sets the output behavior when copying an image to the Clipboard.
Reference: help include_bitmap
set iterlog
Syntax: set iterlog $\{$ on |off $\}$ [, permanently]
Description: specifies whether to display an iteration log.
Reference: [R] set iter
set java_heapmax
Syntax: set java_heapmax $\{$ default $\mid \#[\mathrm{~m} \mid \mathrm{g}]\}$
Description: sets the maximum amount of heap memory allocated for the Java Virtual Machine.
Reference: set java_heapmax is a synonym for java set heapmax; see [P] Java utilities
set java_home
Syntax: set java_home \{default|"path_to_java_home_dir" $\}$
Description: sets the path to the Java Runtime Environment.
Reference: set java_home is a synonym for java set home; see [P] Java utilities
set kmp_blocktime
Syntax: set kmp_blocktime \# [, permanently]
Default: -1
Description: controls the behavior of the OpenMP runtime library regarding the suspension of threads in Stata when they are idle. This can be changed only in Stata/MP.
Reference: help kmp_blocktime

```
set lapack_mkl
    Syntax: set lapack_mkl { on|off } [, permanently]
    Default: on
    Description: specifies whether to use Intel MKL LAPACK routines.
    Reference: [M-1] LAPACK
set lapack_mkl_cnr
    Syntax: set lapack_mkl_cnr {default|auto| compatible| off }
    Default: on
    Description: sets the conditional numerical reproducibility mode for Intel MKL LAPACK routines.
    Reference: [M-1] LAPACK
set level
    Syntax: set level # [, permanently]
    Default: 95
    Description: sets the default confidence level for confidence intervals for all commands
                        that report confidence intervals. 10.00\leq#\leq99.99, and # can have at
                        most two digits after the decimal point.
    Reference: [R] level
set linegap
    Syntax: set linegap #
    Default: 1
    Description: sets the space between lines, in pixels, in the Results window. There is no
    permanently option because permanently is implied.
    Reference: help linegap
set linesize
    Syntax: set linesize #
    Default: 1 less than the full width of the screen
    Description: sets the line width, in characters, for both the screen and the log file.
    Reference: [R] log
set locale_functions
    Syntax: set locale_functions locale
    Default: en_US
    Description: sets the locale to be used by functions that take locale as an optional argument.
    Reference: [P] set locale_functions
set locale_ui
    Syntax: set locale_ui locale
    Default: en_US
    Description: sets the locale that Stata uses for the user interface.
    Reference: [P] set locale_ui
```

```
set locksplitters (Windows only)
    Syntax: set locksplitters {on|off} [, permanently]
    Default: off
    Description: determines whether splitters should be locked so that docked windows
    cannot be resized.
    Reference: help locksplitters
```

set logmsg
Syntax:
Default: on
Description: specifies whether the default message is displayed at the top and bottom of log files.
Reference: [R] log
set logtype
Syntax: set logtype $\{\underline{\text { text }} \mid \underline{\text { smcl }}\}$ [, permanently]
Default: smcl
Description: sets the default log filetype.
Reference: [R] log
set lstretch
Syntax: set lstretch [on|off] [, permanently]
Default: on
Description: specifies whether to automatically widen the coefficient table up to the width of
the Results window to accommodate longer variable names.
Reference: help lstretch
set matacache, set matafavor, set matalibs, set matalnum, set matamofirst,
set mataoptimize, set matasolvetol, and set matastrict; see [M-3] mata set.
set maxbezierpath (Mac only)
Syntax: set maxbezierpath \# [, permanently]
Default: 0
Description: sets the maximum number of lines that can be added to a Bézier path
when rendering a Stata graph to the screen.
Reference: help maxbezierpath
set maxdb
Syntax: set maxdb \# [, permanently]
Default: 50
Description: sets the maximum number of dialog boxes whose contents are remembered
from one invocation to the next during a session. $5 \leq \# \leq 1000$
Reference: $[\mathrm{R}] \mathbf{d b}$
set maxiter

```
Syntax: set maxiter \# [, permanently]
```

Default: 300
Description: sets the default maximum number of iterations for estimation commands. $0 \leq \# \leq 16000$
Reference: [R] set iter
set max_memory
Syntax: set max_memory $\#[b|k| m \mid g][$, permanently $]$
Default: . (all the memory the operating system will supply)
Description: specifies the maximum amount of memory Stata can use to store your data.
$2 \times$ segmentsize $\leq \# \leq$.
Reference: [D] memory
set max_preservemem
Syntax: set max_preservemem \#[b|k|m|g][, permanently]
Default: $\quad 1 \mathrm{~g}$ (1 gigabyte)
Description: controls the maximum amount of memory preserve will use to store preserved datasets in memory.
Reference: [P] preserve
set maxvar
Syntax: set maxvar \# [, permanently $]$
Default: 5000 for Stata/MP and Stata/SE and 2048 for Stata/BE
Description: sets the maximum number of variables. This can be changed only in Stata/MP and Stata/SE. $2048 \leq \# \leq 32767$
Reference: [D] memory
set min_memory
Syntax: $\quad$ set min_memory $\#[b|k| m \mid g] \quad[$, permanently $]$
Default: 0
Description: specifies an amount of memory Stata will not fall below. This setting affects efficiency, not the size of datasets you can analyze. $0 \leq \# \leq$ max_memory
Reference: [D] memory
set more
Syntax: set more $\{$ on $\mid$ off $\}$ [, permanently]
Default: off
Description: pauses when -more- is displayed, continuing only when the user presses a key.
Reference: [R] more
set niceness
Syntax: set niceness \# [, permanently]
Default: 5
Description: affects how soon Stata gives back unused segments to the operating system. $0 \leq \# \leq 10$
Reference: [D] memory

```
set notifyuser (Mac only)
    Syntax: set notifyuser {on|off} [, permanently]
    Default: on
    Description: sets the default Notification Manager behavior in Stata.
    Reference: help notifyuser
set obs
    Syntax: set obs #
    Default: current number of observations
    Description: changes the number of observations in the current dataset. # must be at least
                as large as the current number of observations. If there are variables in memory,
                the values of all new observations are set to missing.
    Reference: [D] obs
set odbcdriver
    Syntax: set odbcdriver {unicode|ansi} [, permanently]
    Defaul: unicode
    Description: determines whether Unicode or ANSI is your ODBC driver.
    Reference: [D] odbc
set odbcmgr (Mac and Unix only)
    Syntax: set odbcmgr {iodbc|unixodbc} [, permanently]
    Default: iodbc
    Description: determines whether iODBC or unixODBC is your ODBC driver manager.
    Reference: [D] odbc
set output
    Syntax: set output {proc|inform| error}
    Default: proc
    Description: specifies the output to be displayed. proc means display all output; inform
        suppresses procedure output but displays informative messages and error messages;
        error suppresses all output except error messages. set output is seldom used.
    Reference: [P] quietly
set pagesize
    Syntax: set pagesize #
    Default: 2 less than the physical number of lines on the screen
    Description: sets the number of lines between -more- messages.
    Reference: [R] more
set pdf_maxtable
    Syntax: set pdf_maxtable # [, permanently]
    Default: 500
    Description: sets the maximum number of tables allowed in putpdf.
    Reference: [RPT] putpdf table
```

```
set pformat
    Syntax: set pformat [fmt] [, permanently]
    Description: specifies the output format of p-values in coefficient tables.
    fmt is a numerical format; see [D] format.
    Reference: [R] set cformat
set pinnable (Windows only)
    Syntax: set pinnable {on|off} [, permanently]
    Default: on
    Description: determines whether to enable the use of pinnable window characteristics for certain
    windows in Stata.
    Reference: help pinnable
set playsnd (Mac only)
    Syntax: set playsnd {on|off} [, permanently]
    Default: on
    Description: sets the sound behavior for the Notification Manager behavior in Stata.
    Reference: help playsnd
set printcolor
```



```
    Default: automatic
    Description: determines how colors are handled when graphs are printed.
    Reference: [G-2] set printcolor
set processors
    Syntax: set processors #
    Description: sets the number of processors or cores that Stata/MP will use. The default
                        is the number of processors available on the computer, or the number of
                        processors allowed by Stata/MP's license, whichever is less.
    Reference: help processors
set python_exec
    Syntax: set python_exec pyexecutable [, permanently]
    Description: sets which version of Python to use.
    Reference: set python_exec is a synonym for python set exec;
        see [P] PyStata integration
set python_userpath
    Syntax: set python_userpath path [path ...] [, permanently prepend]
    Description: sets the user's own module search paths in addition to the system search paths.
    Reference: set python_userpath is a synonym for python set userpath;
        see [P] PyStata integration
set reshape_favor
    Syntax: set reshape_favor {default |memory| speed}[, permanently]
    Description: sets the default method for reshaping data.
    Reference: [D] reshape
```

```
set reventries
    Syntax: set reventries # [, permanently]
    Default: 5000
    Description: sets the number of scrollback lines available in the History window.
                5 < # \leq 32000.
    Reference: help reventries
set revkeyboard (Mac only)
    Syntax: set revkeyboard {on|off } [, permanently]
    Default: on
    Description: sets the keyboard navigation behavior for the History window. on indicates
                that you can use the keyboard to navigate and enter items from the History
                window into the Command window. off indicates that all keyboard input be
                directed at the Command window; items can be entered from the History
                window only by using the mouse.
    Reference: help revkeyboard
set rmsg
    Syntax: set rmsg {on|off } [, permanently]
    Default: off
    Description: indicates whether a return message telling the execution time is to be displayed at
        the completion of each command.
    Reference: [P] rmsg
set rng
    Syntax: set rng {default |mt64|mt64s|kiss32}
    Default: default
    Description: determines which random-number generator Stata's random-number functions
        and commands will use.
    Reference: [R] set rng
set rngstate
    Syntax: set rngstate statecode
    Description: resets the state of the random-number generator to the value specified.
    Reference: [R] set seed
set rngstream
    Syntax: set rngstream #
    Description: specifies the stream from which Stata's stream random-number generator
        should draw random numbers.
    Reference: [R] set rngstream
set scheme
    Syntax: set scheme schemename [, permanently]
    Default: s2color
    Description: determines the overall look for graphs.
    Reference: [G-2] set scheme
```

```
set scrollbufsize
    Syntax: set scrollbufsize #
    Default: 200000
    Description: sets the scrollback buffer size, in bytes, for the Results window;
                may be set between 10,000 and 2,000,000.
    Reference: help scrollbufsize
set searchdefault
    Syntax: set searchdefault {local|net|all}[, permanently]
    Default: local
    Description: sets the default behavior of the search command. set searchdefault local
                restricts search to use only Stata's keyword database. set searchdefault net
                restricts search to searching only the Internet. set searchdefault all
                indicates that both the keyword database and the Internet are to be searched.
    Reference: [R] search
set seed
    Syntax: set seed #
    Default: 123456789
    Description: specifies initial value of the random-number seed used by the
        random-number functions, such as runiform() and rnormal().
    Reference: [R] set seed
set segmentsize
    Syntax: set segmentsize #[b|k|m|g] [, permanently]
    Default: 32m for 64-bit machines
    Description: Stata allocates memory for data in units of segmentsize. This setting changes the
                        amount of memory in a single segment.
                        1m}\leq#\leq32g\mathrm{ for 64-bit machines
    Reference: [D] memory
```

set sformat
Syntax: set sformat $[f m t]$ [, permanently $]$
Description: specifies the output format of test statistics in coefficient tables.
fmt is a numerical format; see [D] format.
Reference: [R] set cformat
set showbaselevels
Syntax: set showbaselevels $\{$ on $\mid$ off $\mid$ all $\}$ [, permanently]
Description: specifies whether to display base levels of factor variables and their interactions
in coefficient tables.
Reference: [R] set showbaselevels
set showemptycells
Syntax: set showemptycells \{on|off \} [, permanently]
Description: specifies whether to display empty cells in coefficient tables.
Reference: [R] set showbaselevels

```
set showomitted
    Syntax: set showomitted { on| off } [, permanently]
    Description: specifies whether to display omitted coefficients in coefficient tables.
    Reference: [R] set showbaselevels
set smoothfonts (Mac only)
    Syntax: set smoothfonts {on| off }
    Default: on
    Description: determines whether to use font smoothing (antialiased text) in the Results, Viewer,
                and Data Editor windows.
    Reference: help smoothfonts
set sortmethod
    Syntax: set sortmethod {default|fsort|qsort}
    Default: default
    Description: determines which sorting method will be used by sort, gsort, and any other
                        commands that use sorting as part of their computation.
    Reference: [P] set sortmethod
set sortrngstate
    Syntax: set sortrngstate #
    Default: 1001XZA112210f4b16c1cb10507a1f38cb440c40003c9a83566fa1201b69...
    Description: specifies the initial value of the state used for the random-number generator that
                        randomizes data before they are sorted. This value is used by sort, gsort, and
                        any other commands that use sorting as part of their computation.
    Reference: [P] set sortrngstate
set table_style
    Syntax: set table_style {table|style} [, permanently]
    Default: table
    Description: controls the default styles used in tables created by table.
    Reference: [TABLES] set table_style
set taskbargroups (Windows only)
    Syntax: set taskbargroups {on| off }
    Default: on
    Description: determines whether to enable taskbar grouping of windows in Stata.
    Reference: help taskbargroups
set trace
    Syntax: set trace {on|off}
    Defaul: off
    Description: determines whether to trace the execution of programs for debugging.
    Reference: [P] trace
```

```
set tracedepth
    Syntax: set tracedepth #
    Default: }\quad32000\mathrm{ (equivalent to }\infty\mathrm{ )
    Description: if trace is set on, traces execution of programs and nested programs up to
                tracedepth. For example, if tracedepth is 2, the current program and any
                subroutine called would be traced, but subroutines of subroutines would not
                be traced.
    Reference: [P] trace
set traceexpand
    Syntax: set traceexpand {on|off} [, permanently]
    Default: on
    Description: if trace is set on, shows lines both before and after macro expansion. If
                traceexpand is set off, only the line before macro expansion is shown.
    Reference: [P] trace
set tracehilite
    Syntax: set tracehilite "pattern" [, word]
    Default: ""
    Description: highlights pattern in the trace output.
    Reference: [P] trace
set traceindent
    Syntax: set traceindent {on|off} [, permanently]
    Default: on
    Description: if trace is set on, indents displayed lines according to their nesting level. The
                                    lines of the main program are not indented. Two spaces of indentation are used for
                                    each level of nested subroutine.
    Reference: [P] trace
set tracenumber
    Syntax: set tracenumber {on|off } [, permanently]
    Default: off
    Description: if trace is set on, shows the nesting level numerically in front of the line.
        Lines of the main program are preceded by 01, lines of subroutines called by the
        main program are preceded by 02, etc.
    Reference: [P] trace
set tracesep
    Syntax: set tracesep {on|off} [, permanently]
    Default: on
    Description: if trace is set on, displays a horizontal separator line that displays the name
        of the subroutine whenever a subroutine is called or exits.
    Reference: [P] trace
```

set type
Syntax: set type $\{$ float $\mid$ double $\}[$, permanently]
Default: float
Description: specifies the default storage type assigned to new variables.
Reference: [D] generate
set update_interval (Mac and Windows only)
Syntax: set update_interval \#
Default: 7
Description: sets the number of days to elapse before performing the next automatic update query.
Reference: [R] update
set update_prompt (Mac and Windows only)
Syntax: set update_prompt $\{$ on $\mid$ off $\}$
Default: on
Description: determines whether a dialog is to be displayed before performing an automatic update query. There is no permanently option because permanently is implied.
Reference: [R] update
set update_query (Mac and Windows only)
Syntax: set update_query $\{$ on $\mid$ off $\}$
Default: on
Description: determines whether update query is to be automatically performed when Stata is launched. There is no permanently option because permanently is implied.
Reference: [R] update
set varabbrev
Syntax: set varabbrev $\{$ on $\mid$ off $\}$ [, permanently $]$
Default: on
Description: indicates whether Stata should allow variable abbreviations.
Reference: [ P ] varabbrev
set varkeyboard (Mac only)
Syntax: set varkeyboard $\{$ on $\mid$ off $\}$ [, permanently $]$
Default: on
Description: sets the keyboard navigation behavior for the Variables window. on indicates that you can use the keyboard to navigate and enter items from the Variables window into the Command window. off indicates that all keyboard input be directed at the Command window; items can be entered from the Variables window only by using the mouse.
Reference: help varkeyboard

## Also see

[R] query - Display system parameters
[R] set_defaults - Reset system parameters to original Stata defaults
[M-3] mata set - Set and display Mata system parameters
[P] creturn - Return c-class values

## Title

set cformat - Format settings for coefficient tables

## Description Syntax Option Remarks and examples Also see

## Description

set cformat specifies the output format of coefficients, standard errors, and confidence limits in coefficient tables.
set pformat specifies the output format of $p$-values in coefficient tables.
set sformat specifies the output format of test statistics in coefficient tables.

## Syntax

```
set cformat [fmt] [, permanently]
set pformat [fmt] [, permanently]
set sformat [fmt] [, permanently]
```

where fint is a numerical format.

## Option

permanently specifies that, in addition to making the change right now, the setting be remembered and become the default setting when you invoke Stata.

## Remarks and examples

The formatting of the numbers in the coefficient table can be controlled by using the set cformat, set pformat, and set sformat commands or by using the cformat (\%fmt), pformat (\%fmt), and sformat (\%fmt) options at the time of estimation or on replay of the estimation command. See [R] Estimation options.

The maximum format widths for set cformat, set pformat, and set sformat in coefficient tables are 9,5 , and 8 , respectively.

## Example 1

We use auto.dta to illustrate.

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. regress mpg weight displacement
```

| Source | SS | df | MS | Number of obs | = | 74 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | $\mathrm{F}(2,71)$ | = | 66.79 |
| Model | 1595.40969 | 2 | 797.704846 | Prob > F | = | 0.0000 |
| Residual | 848.049768 | 71 | 11.9443629 | R -squared | = | 0.6529 |
|  |  |  |  | Adj R-squared | = | 0.6432 |
| Total | 2443.45946 | 73 | 33.4720474 | Root MSE | = | 3.4561 |


| mpg | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | ---: | ---: |
| weight | -.0065671 | .0011662 | -5.63 | 0.000 | -.0088925 | -.0042417 |
| displacement | .0052808 | .0098696 | 0.54 | 0.594 | -.0143986 | .0249602 |
| _cons | 40.08452 | 2.02011 | 19.84 | 0.000 | 36.05654 | 44.11251 |

. set cformat \%9.2f
. regress mpg weight displacement

| Source | SS | df | MS | Number of obs | = | 74 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | $\mathrm{F}(2,71)$ | = | 66.79 |
| Model | 1595.40969 | 2 | 797.704846 | Prob > F | = | 0.0000 |
| Residual | 848.049768 | 71 | 11.9443629 | R -squared | = | 0.6529 |
|  |  |  |  | Adj R-squared | = | 0.6432 |
| Total | 2443.45946 | 73 | 33.4720474 | Root MSE | = | 3.4561 |


| mpg | Coefficient | Std. err. | $t$ | P>\|t| | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| weight | -0.01 | 0.00 | -5.63 | 0.000 | -0.01 | -0.00 |
| displacement | 0.01 | 0.01 | 0.54 | 0.594 | -0.01 | 0.02 |
| _cons | 40.08 | 2.02 | 19.84 | 0.000 | 36.06 | 44.11 |

. regress mpg weight displacement, cformat(\%9.3f)

| Source | SS | df | MS | Number of obs | = | 74 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | $\mathrm{F}(2,71)$ |  | 66.79 |
| Model | 1595.40969 | 2 | 797.704846 | Prob > F | = | 0.0000 |
| Residual | 848.049768 | 71 | 11.9443629 | R -squared |  | 0.6529 |
|  |  |  |  | Adj R-squared |  | 0.6432 |
| Total | 2443.45946 | 73 | 33.4720474 | Root MSE | = | 3.4561 |


| mpg | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| weight | -0.007 | 0.001 | -5.63 | 0.000 | -0.009 | -0.004 |
| displacement | 0.005 | 0.010 | 0.54 | 0.594 | -0.014 | 0.025 |
| _cons | 40.085 | 2.020 | 19.84 | 0.000 | 36.057 | 44.113 |

To reset the cformat setting to its command-specific default, type
. set cformat
. regress mpg weight displacement

| Source | SS | df | MS | Number of obs | = | 74 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | F (2, 71) | = | 66.79 |
| Model | 1595.40969 | 2 | 797.704846 | Prob > F | = | 0.0000 |
| Residual | 848.049768 | 71 | 11.9443629 | R -squared | = | 0.6529 |
|  |  |  |  | Adj R-squared | = | 0.6432 |
| Total | 2443.45946 | 73 | 33.4720474 | Root MSE | = | 3.4561 |


| mpg | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| ---: | ---: | :---: | :---: | :---: | ---: | ---: |
| weight | -.0065671 | .0011662 | -5.63 | 0.000 | -.0088925 | -.0042417 |
| displacement | .0052808 | .0098696 | 0.54 | 0.594 | -.0143986 | .0249602 |
| _cons | 40.08452 | 2.02011 | 19.84 | 0.000 | 36.05654 | 44.11251 |

## Also see

[R] Estimation options - Estimation options
[R] query - Display system parameters
[R] set - Overview of system parameters
[U] 20.9 Formatting the coefficient table

## Title

set_defaults - Reset system parameters to original Stata defaults

Description Syntax Option Remarks and examples Also see

## Description

set_defaults resets settings made by set to the original default settings that were shipped with Stata.
set_defaults may not be used with java, lapack, putdocx, python, rng, or sort systemparameter categories.

## Syntax

$$
\text { set_defaults }\{\text { category } \mid \text { _all }\}[\text {, permanently }]
$$

where category is one of memory | output | interface | graphics

```
network | update | trace | mata | unicode | other
```


## Option

permanently specifies that, in addition to making the change right now, the settings be remembered and become the default settings when you invoke Stata.

## Remarks and examples

$>$ Example 1
To assist us in debugging a new command, we modified some of the trace settings. To return them to their original values, we type

```
. set_defaults trace
-> set trace off
-> set tracedepth }3200
-> set traceexpand on
-> set tracesep on
-> set traceindent on
-> set tracenumber off
-> set tracehilite ""
(preferences reset)
```


## Also see

[R] query - Display system parameters
[R] set - Overview of system parameters
[M-3] mata set - Set and display Mata system parameters

## Title

set emptycells - Set what to do with empty cells in interactions

Description Syntax Option Remarks and examples Also see

## Description

set emptycells allows you to control how Stata handles interaction terms with empty cells. Stata can keep empty cells or drop them. The default is to keep empty cells.

## Syntax

```
set emptycells { keep|drop} [, permanently]
```


## Option

permanently specifies that, in addition to making the change right now, the setting be remembered and become the default setting when you invoke Stata.

## Remarks and examples

By default, Stata keeps empty cells so they can be reported in the coefficient table. For example, type
. use https://www.stata-press.com/data/r18/auto
. regress mpg rep78\#foreign, baselevels
and you will see a regression of mpg on 10 indicator variables because rep78 takes on 5 values and foreign takes on 2 values in the auto dataset. Two of those cells will be reported as empty because the data contain no observations of foreign cars with a rep78 value of 1 or 2.

Many real datasets contain a large number of empty cells, and this could cause the "unable to allocate matrix" error message, $\mathrm{r}(915)$. In that case, type
. set emptycells drop
to get Stata to drop empty cells from the list of coefficients. If you commonly fit models with empty cells, you can permanently set Stata to drop empty cells by typing the following:

```
. set emptycells drop, permanently
```


## Also see

[R] set - Overview of system parameters

## Title

set iter - Control iteration settings

> Description Syntax Option Remarks and examples Also see

## Description

set iterlog and set maxiter control the display of the iteration log and the maximum number of iterations, respectively, for estimation commands that iterate and for the Mata optimization functions moptimize(), optimize(), and solvenl().
set iterlog specifies whether to display the iteration log. The default setting is on, which displays the log. You can specify set iterlog off to suppress it. To change whether the iteration $\log$ is displayed for a particular estimation command, you need not reset iterlog; you can specify the log or nolog option with that command. If you do not specify log or nolog, the iterlog setting is used. To view the current setting of iterlog, type display c(iterlog).
set maxiter specifies the default maximum number of iterations. To change the maximum number of iterations performed by a particular estimation command, you need not reset maxiter; you can specify the iterate(\#) option with that command. If you do not specify iterate(\#), the maxiter value is used. To view the current setting of maxiter, type display c (maxiter).

## Syntax

Set whether to display the iteration log

```
set iterlog {on|off } [, permanently]
```

Set default maximum iterations
set maxiter \# [, permanently ]
\# is any number between 0 and 16,000 ; the initial value is set to 300 .

## Option

permanently specifies that, in addition to making the change right now, the setting be remembered and become the default setting when you invoke Stata.

## Remarks and examples

The iterlog setting is particularly useful in combination with the nolog and log options; see example 1 below. Also see [R] Maximize for details about the options. The iterlog setting has no effect on commands that suppress the iteration log by default, for example, commands prefixed with svy. To display the log with those commands, you need to use the log option.

You will rarely need to modify the maxiter setting to change the maximum number of iterations used by Stata's iterative commands. Instead, you may want to specify the iterate () option with these commands. For example, specifying iterate (0) is useful for viewing results evaluated at the initial value of the coefficient vector.

The iterlog and maxiter settings also control the default output displayed by the Mata optimization functions moptimize(), optimize(), and solvenl().

## > Example 1: Display and suppress the iteration log

Stata estimation commands that iterate usually display the iteration log by default:

```
. sysuse auto
(1978 automobile data)
. logit foreign mpg
Iteration 0: Log likelihood = -45.03321
Iteration 1: Log likelihood = -39.380959
Iteration 2: Log likelihood = -39.288802
Iteration 3: Log likelihood = -39.28864
Iteration 4: Log likelihood = -39.28864
```

Logistic regression $\quad$ Number of obs $=\quad 74$
LR chi2(1) $=11.49$
Prob > chi2 $=0.0007$
Pseudo R2 $=0.1276$
Log likelihood = -39.28864

| foreign | Coefficient | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| mpg | .1597621 | .0525876 | 3.04 | 0.002 | .0566922 | .262832 |
| _cons | -4.378866 | 1.211295 | -3.62 | 0.000 | -6.752961 | -2.004771 |

You can suppress the log by specifying the nolog option:
. logit foreign mpg, nolog
$\begin{array}{ll}\text { Logistic regression } & \text { Number of obs }= \\ & \text { LR chi2 }(1) \\ & =11.49 \\ \text { Log likelihood }=-39.28864 & \text { Prob }>\text { chi2 }=0.0007 \\ & \text { Pseudo R2 }=0.1276\end{array}$

| foreign | Coefficient | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| mpg | .1597621 | .0525876 | 3.04 | 0.002 | .0566922 | .262832 |
| _cons | -4.378866 | 1.211295 | -3.62 | 0.000 | -6.752961 | -2.004771 |

If you want to suppress the iteration $\log$ from all estimation commands every time they are run within the current Stata session, type

```
. set iterlog off
```

We can run logit again but now without the nolog option, and the iteration log will not be displayed:

| . logit foreign mpg |  |
| :--- | :--- |
| Logistic regression | Number of obs $=$ |
|  | LR chi2 $(1)$ |
|  | $=11.49$ |
| Log likelihood $=-39.28864$ | Prob $>$ chi2 $=0.0007$ |
|  | Pseudo R2 |


| foreign | Coefficient | Std. err. | z | P>\|z| | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| mpg | .1597621 | .0525876 | 3.04 | 0.002 | .0566922 | .262832 |
| _cons | -4.378866 | 1.211295 | -3.62 | 0.000 | -6.752961 | -2.004771 |

Or we can run a different command, for example, mlogit, and the log will still be suppressed:
. mlogit rep78 mpg
$\begin{array}{llr}\text { Multinomial logistic regression } & \text { Number of obs }= & 69 \\ & \text { LR chi2 }(4) & =15.88 \\ \text { Log likelihood }=-85.752375 & \text { Prob }>\text { chi2 }=0.0032 \\ & \text { Pseudo R2 } & =0.0847\end{array}$

|  | rep78 | Coefficient | Std. err. | z | $P>\|z\|$ | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 1 |  |  |  |  |  |  |  |
|  | mpg | . 0708122 | . 1471461 | 0.48 | 0.630 | -. 2175888 | . 3592132 |
|  | _cons | -4.137144 | 3.15707 | -1.31 | 0.190 | -10.32489 | 2.0506 |
| 2 |  |  |  |  |  |  |  |
|  | mpg | -. 0164251 | . 0926724 | -0.18 | 0.859 | -. 1980597 | . 1652096 |
|  | _cons | -1.005118 | 1.822129 | -0.55 | 0.581 | -4.576426 | 2.56619 |
| 3 |  | (base outc | me) |  |  |  |  |
| 4 |  |  |  |  |  |  |  |
|  | mpg | . 0958626 | . 0633329 | 1.51 | 0.130 | -. 0282676 | . 2199927 |
|  | _cons | -2.474187 | 1.341131 | -1.84 | 0.065 | -5.102756 | . 1543813 |
| 5 |  |  |  |  |  |  |  |
|  | mpg | . 2477469 | . 0764076 | 3.24 | 0.001 | . 0979908 | . 397503 |
|  | _cons | -6.653164 | 1.841794 | -3.61 | 0.000 | -10.26301 | -3.043314 |

With the iterlog setting off, we can display the iteration log for specific commands by specifying the $\log$ option:

```
. mlogit rep78 mpg, log
Iteration 0: Log likelihood = -93.692061
Iteration 1: Log likelihood = -86.581485
Iteration 2: Log likelihood = -85.767758
Iteration 3: Log likelihood = -85.752385
Iteration 4: Log likelihood = -85.752375
Iteration 5: Log likelihood = -85.752375
```

| Multinomial logistic regression | Number of obs $=$ | 69 |
| :--- | :--- | ---: |
|  | LR chi2 $(4)$ | $=15.88$ |
| Log likelihood $=-85.752375$ | Prob $>$ chi2 $=0.0032$ |  |
|  | Pseudo R2 | $=0.0847$ |


|  | rep78 | Coefficient | Std. err. | z | $P>\|z\|$ | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| 1 |  |  |  |  |  |  |  |
|  | mpg | . 0708122 | . 1471461 | 0.48 | 0.630 | -. 2175888 | . 3592132 |
|  | _cons | -4.137144 | 3.15707 | -1.31 | 0.190 | -10.32489 | 2.0506 |
| 2 |  |  |  |  |  |  |  |
|  | mpg | -. 0164251 | . 0926724 | -0.18 | 0.859 | -. 1980597 | . 1652096 |
|  | _cons | -1.005118 | 1.822129 | -0.55 | 0.581 | -4.576426 | 2.56619 |
| 3 |  | (base outcome) |  |  |  |  |  |
| 4 |  |  |  |  |  |  |  |
|  | mpg | . 0958626 | . 0633329 | 1.51 | 0.130 | -. 0282676 | . 2199927 |
|  | _cons | -2.474187 | 1.341131 | -1.84 | 0.065 | -5.102756 | . 1543813 |
| 5 |  |  |  |  |  |  |  |
|  | mpg | . 2477469 | . 0764076 | 3.24 | 0.001 | . 0979908 | . 397503 |
|  | _cons | -6.653164 | 1.841794 | -3.61 | 0.000 | -10.26301 | -3.043314 |

You can switch back to displaying iteration logs by typing

```
. set iterlog on
```

The default setting will be restored automatically the next time you invoke Stata. If you want the setting to be remembered for future Stata sessions, specify the permanently option with set iterlog.

## Also see

[R] Maximize - Details of iterative maximization
[R] set - Overview of system parameters
[M-5] moptimize( ) - Model optimization
[M-5] optimize( ) - Function optimization
[M-5] solvenl() - Solve systems of nonlinear equations

## Title

set rng - Set which random-number generator (RNG) to use

Description Syntax Remarks and examples Reference Also see

## Description

set rng determines which random-number generator (RNG) Stata's random-number functions and commands will use.

## Syntax

```
set rng {default|mt64|mt64s|kiss32}
```


## Remarks and examples

Remarks are presented under the following headings:
Introduction
Random-number generators in Stata

## Introduction

By default, Stata uses the 64-bit Mersenne Twister (mt64) RNG. mt64s is a stream RNG based on the 64-bit Mersenne Twister. Earlier versions of Stata used the 32-bit KISS (keep it simple stupid) (kiss32) RNG.

With set rng default (the default), code running under version control will automatically use the appropriate RNG-mt64 in Stata 14 and later and kiss32 for earlier code.

The scope of set rng is the Stata session, do-file, or program in which rng is set.
Unless you want to simultaneously draw random numbers in separate instances of Stata, we recommend that you do not change Stata's default behavior for its RNGs. See [R] set rngstream for an introduction to simultaneously drawing random numbers in separate instances of Stata.

See [FN] Random-number functions, [R] set seed, and [R] set rngstream for more information.

## Random-number generators in Stata

The default RNG in Stata is the 64-bit Mersenne Twister. See Matsumoto and Nishimura (1998) for more details. The default RNG in Stata 13 and earlier versions was George Marsaglia's 32-bit KISS generator (G. Marsaglia, 1994, pers. comm.). The KISS generator is still available under version control or via set rng. Multiple independent random-number streams (based on the 64-bit Mersenne Twister) are also supported for use in multiple simultaneous instances of Stata; see $[R]$ set rngstream for more information on this. The abbreviations mt64, kiss32, and mt64s are used, respectively, to specify these three generators in Stata commands and functions.

So far, we have discussed two ways you can specify the RNG: with set rng and through version control. Another way to specify the RNG is with functions and system parameters explicitly named after the generators. In fact, all random-number functions have variants that are explicitly named after each generator, using the generator abbreviation as the suffix. For example, runiform_mt64(), runiform_kiss32(), and runiform_mt64s() are variants of runiform() for each generator. Similarly, we have rnormal_mt64(), rnormal_kiss32(), rnormal_mt64s(), etc.

The system parameters seed and rngstate also have variants explicitly named after each generator: seed_mt64, seed_kiss32, seed_mt64s, rngstate_mt64, rngstate_kiss32, and rngstate_mt64s.

For example, here is how you can use functions and parameters specific to mt64 to set the seed, generate random numbers, preserve a state, generate more numbers, and restore the previously preserved state:

```
. set seed_mt64 482637
. generate u = runiform_mt64()
. local state = c(rngstate_mt64)
. generate l = rlogistic_mt64()
. set rngstate_mt64 'state'
```

Note that calling functions and setting parameters specific to, say, kiss32, will not change the current RNG, the seed of the current RNG, or the state of the current RNG-unless the current RNG is kiss32.

## Reference

Matsumoto, M., and T. Nishimura. 1998. Mersenne Twister: A 623-dimensionally equidistributed uniform pseudorandom number generator. ACM Transactions on Modeling and Computer Simulation 8: 3-30. https://doi.org/10.1145/272991.272995.

## Also see

$[R]$ set - Overview of system parameters
$[R]$ set rngstream - Specify the stream for the stream random-number generator
$[R]$ set seed - Specify random-number seed and state
[FN] Random-number functions
[P] version - Version control

## Title

set rngstream - Specify the stream for the stream random-number generator

Description Syntax Remarks and examples<br>References Also see

## Description

set rngstream specifies the subsequence, known as a stream, from which Stata's stream randomnumber generator should draw random numbers. When performing a bootstrap estimation or a Monte Carlo simulation in parallel on multiple machines, you should set the same seed on all machines but set a different stream on each machine. This will ensure that random numbers drawn on different machines are independent. We strongly recommend that you set the seed and the stream only once in each Stata session.

## Syntax

```
set rngstream #
```

\# is any integer between 1 and 32,767 .

## Remarks and examples

Stata's stream random-number generator, the stream 64-bit Mersenne Twister (mt64s), allows separate instances of Stata to simultaneously draw independent random numbers. This feature enables you to use bootstrap and to run Monte Carlo simulations in parallel on multiple machines.

What we call random numbers are elements in a sequence of deterministic numbers that appear to be random. A seed specifies a starting value in this sequence. In figure 1, each tick is an element in a random sequence and setting the seed to 12345 means that the tick identified by the arrow below is the first number drawn.


Figure 1. Seed specifies first number in random sequence
A stream random-number generator partitions a sequence of random numbers into nonoverlapping subsequences known as streams. The random numbers in each stream are independent of those in other streams because they come from distinct nonoverlapping subsets of the original sequence.

Figure 2 depicts a stream version of the generator depicted in figure 1. The stream version also starts at the place implied by seed 12345 , but it additionally partitions the random numbers into 4 streams and a set of unused numbers.


Figure 2. A stream version of figure 1 generator
In contrast to nonstream random-number generators, setting the seed for a random-number generator controls not just where the sequence starts but also how the sequence is partitioned. Compare figure 2 with figure 3 for an illustration.


Figure 3. Changing the seed changes the streams
Seed 123456 specifies the first random number, and the streams of random numbers in figure 3 differ from those in figure 2.

The mt64s generator is a stream version of Stata's default generator, the 64-bit Mersenne Twister implemented in mt64; see Matsumoto and Nishimura (1998) and Random-number generators in Stata in $[R]$ set rng for more details. Our implementation of the method discussed in Haramoto et al. (2008) partitions the mt64 sequence into 32,767 streams, each containing $2^{128}$ random numbers. The remaining numbers are unused. The mt64s seed determines the starting point of every stream in the Mersenne Twister sequence.

Stream 1 of mt64s has the same starting point as the mt64 generator. That is, given the same seed, mt64s with rngstream set to 1 will generate the same random numbers as mt64.

The mt64s generator is designed to simultaneously draw independent random numbers on different machines. To draw from different streams that guarantee independence, use the same seed and change the stream. For example, to draw some uniform $(0,1)$ random numbers from stream 10 of the mt64s generator under seed 123, type

```
. set rng mt64s
. set rngstream 10
. set seed 123
. generate u = runiform()
```

If we wanted to simultaneously draw some uniform $(0,1)$ random numbers on another machine from stream 11 of the mt64s generator, we would type
. set rng mt64s
. set rngstream 11
. set seed 123
. generate $u=$ runiform()
Again, each seed creates a different partition of the mt64 sequence into nonoverlapping subsets.
We strongly recommend that you set the stream and the seed once in each Stata session and draw numbers only from this stream.
c (rngstream) returns the current stream number. c (rngseed_mt64s) returns the last seed that was set for mt 64 s . See $[\mathrm{P}]$ creturn for more details. See $[\mathrm{R}]$ set seed for details about storing and restoring the current position in the random sequence.

As with the single-stream generators, use local state $=c$ (rngstate) to store the current position in the current random stream; see $[R]$ set seed for details. The mt64s state encodes the seed used in addition to the stream number, because the seed determines the position of every random number in every stream. Unlike the case of single-stream generators, restoring the state also restores the seed. For example, suppose you save an mt64s state with local state $=c$ (rngstate) change the seed and the stream, and later restore that state with set rngstate 'state'. The current mt64s seed is changed to the one encoded in state. In addition to changing the current stream to the one encoded in state, the current mt64s seed is changed to the one encoded in state. This behavior ensures any subsequent stream changes draw from nonoverlapping subsets.
set rngstream also sets the random-number generator to mt64s.

## > Example 1: Using stream random numbers to parallelize a bootstrap

We illustrate how to simultaneously perform 100 bootstrap replications on machine 1 and 100 bootstrap replications on machine 2 . We focus on the mechanics of distributing the draws over machines using stream random numbers; see $[R]$ bootstrap for an introduction to the bootstrap.

On machine 1, we type

```
. clear all
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. set rng mt64s
. set rngstream 1
. set seed 12345
. bootstrap, reps(100) saving(machine1, replace): regress mpg weight gear foreign
(running regress on estimation sample)
(file machine1.dta not found)
Bootstrap replications (100): .........10..........20...........30...........40........
> ...50........60........70........80.........90............... 100 done
Linear regression Number of obs = 74
                                    Replications = 100
                                    Wald chi2(3) = 191.66
                                    Prob > chi2 = 0.0000
                                    R-squared = 0.6670
                                    Adj R-squared = 0.6527
                                    Root MSE = 3.4096
```

|  | Observed <br> coefficient | Bootstrap <br> std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | Normal-based <br> [95\% conf. interval] |  |  |
| ---: | ---: | :--- | :---: | ---: | ---: | ---: | :---: |
| weight | -.006139 | .0005462 | -11.24 | 0.000 | -.0072095 | -.0050685 |  |
| gear_ratio | 1.457113 | 1.271301 | 1.15 | 0.252 | -1.03459 | 3.948817 |  |
| foreign | -2.221682 | 1.090115 | -2.04 | 0.042 | -4.358267 | -.0850957 |  |
| _cons | 36.10135 | 4.720623 | 7.65 | 0.000 | 26.8491 | 45.3536 |  |

On machine 2, we type

```
. clear all
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. set rng mt64s
. set rngstream 2
. set seed 12345
. bootstrap, reps(100) saving(machine2, replace): regress mpg weight gear foreign
(running regress on estimation sample)
(file machine2.dta not found)
Bootstrap replications (100): .........10.........20..........30.........40......
> ...50.........60.........70.........80.........90......... 100 done
Linear regression Number of obs = 74
                                    Replications = 100
                                    Wald chi2(3) = 121.48
                                    Prob > chi2 = 0.0000
                                    R-squared =0.6670
                                    Adj R-squared = 0.6527
                                    Root MSE = 3.4096
\begin{tabular}{r|rrrrrr}
\hline mpg & \begin{tabular}{c} 
Observed \\
coefficient
\end{tabular} & \begin{tabular}{l} 
Bootstrap \\
std. err.
\end{tabular} & \(z\) & \(\mathrm{P}>|\mathrm{z}|\) & \begin{tabular}{r} 
Normal-based \\
[95\% conf. interval]
\end{tabular} \\
\hline weight & -.006139 & .0005909 & -10.39 & 0.000 & -.0072972 & -.0049809 \\
gear_ratio & 1.457113 & 1.267439 & 1.15 & 0.250 & -1.027022 & 3.941249 \\
foreign & -2.221682 & 1.253503 & -1.77 & 0.076 & -4.678502 & .2351393 \\
_cons & 36.10135 & 4.419797 & 8.17 & 0.000 & 27.43871 & 44.764 \\
\hline
\end{tabular}
```

After copying machine $2 . \mathrm{dta}$ from machine 2 to the working directory on machine 1 , we produce the combined results by typing

```
. clear all
. use machine1
(bootstrap: regress)
. append using machine2
. bstat
Bootstrap results Number of obs = 74
Replications = 200
```

Command: regress mpg weight gear foreign

|  | Observed <br> coefficient | Bootstrap <br> std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ |  | Normal-based <br> [95\% conf. interval] |  |
| ---: | ---: | :--- | :---: | ---: | ---: | ---: | :---: |
| weight | -.006139 | .0005678 | -10.81 | 0.000 | -.0072519 | -.0050262 |  |
| gear_ratio | 1.457113 | 1.266586 | 1.15 | 0.250 | -1.02535 | 3.939577 |  |
| foreign | -2.221682 | 1.187847 | -1.87 | 0.061 | -4.549819 | .1064562 |  |
| _cons | 36.10135 | 4.562644 | 7.91 | 0.000 | 27.15873 | 45.04397 |  |

We used regress in this example, but the divide-and-conquer strategy reduces computation time for any command that works with bootstrap. In fact, problems that take longer produce more noticeable speed improvements. For computationally intensive problems, the two-machine time will be about one-half the one-machine time. Using distinct streams on many different machines can dramatically reduce the time required for computationally intensive problems.

## > Example 2: Using stream random numbers to parallelize a Monte Carlo simulation

We want to simultaneously perform 100 Monte Carlo replications on machine 3 and 100 Monte Carlo replications on machine 4. Again, we focus entirely on the mechanics of distributing the draws over machines. See Drukker (2015) for an introduction to Monte Carlo simulations using Stata.

As discussed in [R] simulate, the simulate command uses an ado-file that draws from the population of interest; it then computes and returns the estimates. Our program chi2sim draws from a $\chi^{2}$ distribution with one degree of freedom.

```
program define chi2sim, rclass
    version 18.0 // (or version 18.5 for StataNow)
    drop _all
    set obs 200
    tempvar z
    generate ' }z\mathrm{ ' = rchi2(1)
    summarize 'z'
    return scalar mean = r(mean)
    return scalar Var =r(Var)
end
```

On machine 3, we type

```
set rng mt64s
set rngstream 3
set seed 12345
simulate mean=r(mean) var=r(Var), reps(500) saving(machine3, replace): chi2sim
        Command: chi2sim
            mean: r(mean)
            var: r(Var)
(file machine3.dta not found)
Simulations (500): .........10.........20..........30.........40.......................
> ...60........70.........80.........90......... . . 100..........110.......... . . 120.....
> .....130........140.........150.........160..........170.......... . . 180.......... . . 1
90.........200..........210........ . .220. . . . . . . .230. . . . . . . . 240. . . . . . . . . 250. . . .
> .....260.........270.........280.........290..........300..........310........... . . 3
> 20.........330.........340.........350..........360..........370......... . . . 380.....
> . . . . 390. . . . . . .400. . . . . . . .410. . . . . . . . 420 . . . . . . . .430 . . . . . . . . 440. . . . . . . . . 4
50........460........470.........480.........490.......... . 500 done
```

On machine 4, we run a do-file that performs

```
. set rng mt64s
. set rngstream 4
. set seed 12345
. simulate mean=r(mean) var=r(Var), reps(500) saving(machine4, replace): chi2sim
    Command: chi2sim
        mean: r(mean)
        var: r(Var)
(file machine4.dta not found)
Simulations (500): .........10.........20.......... 30..........40............ 50.......
> ...60........70.........80.........90.........100..........110........... . . 120....
> .....130.........140..........150.. . . . . . . 160. . . . . . . .170. . . . . . . . . 180. . . . . . . . . }
> 90.........200. . . . . . . .210. . . . . . . .220. . . . . . . . 230. . . . . . . . . 240. . . . . . . . . 250. . . .
> . . . .260......... 270. . . . . . . 280. . . . . . . . 290. . . . . . . . . 300. . . . . . . . . 310 . . . . . . . . . }
> 20........330........340.........350.........360..........370............380....
> .....390.........400.........410..........420..........430......... . .440. . . . . . . . . 4
> 50........460........470........440.........490.......... }500\mathrm{ done
```

After copying machine4.dta from machine 4 to the working directory on machine 3 , we combine the results by typing

```
. clear all
. use machine3
(simulate: chi2sim)
```

. append using machine4
. summarize mean var

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| mean | 1,000 | 1.00296 | .1011507 | .7098877 | 1.330305 |
| var | 1,000 | 2.01955 | .5194171 | .8931134 | 4.381884 |

As in example 1, more machines enable further parallelization.

## - Technical note

While mt64s has been made robust to switching between streams within a Stata session, convoluted combinations of set rngstream \# and set seed \# can lead to drawing the same random numbers, just as it can in the case of single-stream generators. We strongly recommend that you do not switch between streams within a session.

## > Example 3: Position within a stream is stored

This example illustrates that the sequence picks up where it left off when the stream is switched, and it illustrates that clear rngstream resets all streams to their beginning positions. These features facilitate advanced programming techniques, and we recommend against using this feature in standard use.

```
. clear all
. set obs 10
Number of observations (_N) was 0, now 10.
. set rng mt64s
. set rngstream 5
. set seed 12345
. generate x = runiform() in 1/5
(5 missing values generated)
. set rngstream 6
. generate y = runiform()
. set rngstream 5
. replace x = runiform() in 6/10
(5 real changes made)
. clear rngstream
. set rngstream 5
. generate z = runiform()
. list
```

| $x$ | $y$ | $z$ |
| ---: | ---: | ---: |
| .5095264 | .8838338 | .5095264 |
| .9766202 | .7677673 | .9766202 |
| .3933811 | .5665985 | .3933811 |
| .950057 | .3141659 | .950057 |
| .5862163 | .6635106 | .5862163 |
| .4837167 | .6781911 | .4837167 |
| .1752382 | .7169843 | .1752382 |
| .2302023 | .7554966 | .2302023 |
| .4927879 | .8685812 | .4927879 |
| .9114158 | .5634732 | .9114158 |

After setting the rngstream to 5 and setting the seed, we put the first 5 draws from stream 5 into observations $1-5$ of $x$, switch to rngstream 6 , put the first 10 random draws from stream 6 into $y$, return to rngstream 5, and put the next 5 draws from stream 5 into observations $6-10$ of x . Then we use clear rngstream to initialize each rngstream at its initial position for seed 12345 and put the first 10 draws from stream 5 into $z$.

That the random numbers in x match those in z illustrates that the sequence picks up where it left off when the stream is switched and the seed has not been changed.

## References

Drukker, D. M. 2015. Monte Carlo simulations using Stata. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2015/10/06/monte-carlo-simulations-using-stata/.
Haramoto, H., M. Matsumoto, T. Nishimura, F. Panneton, and P. L'Ecuyer. 2008. Efficient jump ahead for $\mathrm{F}_{2}$-linear random number generators. INFORMS Journal on Computing 20: 385-390. https://doi.org/10.1287/ijoc.1070.0251.

Matsumoto, M., and T. Nishimura. 1998. Mersenne Twister: A 623-dimensionally equidistributed uniform pseudorandom number generator. ACM Transactions on Modeling and Computer Simulation 8: 3-30. https://doi.org/10.1145/272991.272995.
Taylor, M. A. 2018. Simulating the central limit theorem. Stata Journal 18: 345-356.
Vega Yon, G. G., and B. Quistorff. 2019. parallel: A command for parallel computing. Stata Journal 19: 667-684.

## Also see

[R] set - Overview of system parameters
$[R]$ set rng - Set which random-number generator (RNG) to use
[R] set seed - Specify random-number seed and state
[D] clear - Clear memory
[FN] Random-number functions

## Title

set seed - Specify random-number seed and state

Description Syntax Remarks and examples Reference<br>Also see

## Description

set seed \# specifies the initial value of the random-number seed used by the random-number functions, such as runiform() and rnormal().
set rngstate statecode resets the state of the random-number generator to the value specified, which is a state previously obtained from creturn value c(rngstate).
set seed \# and set rngstate statecode apply to the current random-number generator. Every random-number generator in Stata has its own seed and state encoding.

## Syntax

```
    set seed #
    set rngstate statecode
```

\# is any number between 0 and $2^{31}-1$ (or $2,147,483,647$ ).
statecode is a random-number state previously obtained from creturn value c(rngstate).

## Remarks and examples

Remarks are presented under the following headings:

```
Examples
Setting the seed
How to choose a seed
Do not set the seed too often
Preserving and restoring the random-number generator state
```


## Examples

1. Specify initial value of random-number seed
. set seed 339487731
2. Create variable $u$ containing uniformly distributed pseudorandom numbers on the interval $(0,1)$ . generate $u=$ runiform()
3. Create variable $z$ containing normally distributed random numbers with mean 0 and standard deviation 1
```
. generate z = rnormal()
```

4. Obtain state of pseudorandom-number generator and store it in a local macro named state
. local state $=c$ (rngstate)
5. Restore pseudorandom-number generator state to that previously stored in local macro named state
. set rngstate 'state'

## Setting the seed

Stata's random-number generation functions, such as runiform() and rnormal(), do not really produce random numbers. These functions are deterministic algorithms that produce numbers that can pass for random. runiform() produces numbers that can pass for independent draws from a rectangular distribution over $(0,1)$; rnormal () produces numbers that can pass for independent draws from $N(0,1)$. Stata's random-number functions are formally called pseudorandom-number functions. The default pseudorandom-number generator introduced in Stata 14 is the 64-bit Mersenne Twister. See Matsumoto and Nishimura (1998) and Random-number generators in Stata in [R] set rng for more details.

The sequences the random-number functions produce are determined by the seed, which is just a number and which is set to 123456789 every time Stata is launched. This means that runiform() produces the same sequence each time you start Stata. The first time you use runiform() after Stata is launched, runiform() returns 0.348871704556195 . The second time you use it, runiform() returns 0.266885709753138 . The third time you use it, ....

To obtain different sequences, you must specify different seeds using the set seed command. You might specify the seed 472195 :

```
. set seed 472195
```

If you were now to use runiform(), the first call would return 0.713028143573182 , the second call would return 0.920524469911484 , and so on. Whenever you set seed 472195 , runiform() will return those numbers the first two times you use it.

Thus, you set the seed to obtain different pseudorandom sequences from the pseudorandom-number functions.

If you record the seed you set, pseudorandom results such as results from a simulation or imputed values from mi impute can be reproduced later. Whatever you do after setting the seed, if you set the seed to the same value and repeat what you did, you will obtain the same results.

## How to choose a seed

Your best choice for the seed is an element chosen randomly from the set $\left\{0,1, \ldots, 2^{31}-1\right\}$ (where $2^{31}-1=2,147,483,647$ ). We recommend that, but that is difficult to achieve because finding easy-to-access, truly random sources is difficult.

One person we know uses digits from the serial numbers from dollar bills he finds in his wallet. Of course, the numbers he obtains are not really random, but they are good enough, and they are probably a good deal more random than the seeds most people choose. Some people use dates and times, although we recommend against that because, over the day, it just gets later and later, and that is a pattern. Others try to make up a random number, figuring if they include enough digits, the result just has to be random. This is a variation on the five-second rule for dropped food, and we admit to using both of these rules.

It does not really matter how you set the seed, as long as there is no obvious pattern in the seeds that you set and as long as you do not set the seed too often during a session.

Nonetheless, here are two methods that we have seen used but you should not use:

1. The first time you set the seed, you set the number 1 . The next time, you set 2 , and then 3 , and so on. Variations on this included setting 1001, 1002, 1003, $\ldots$, or setting 1001, 2001, 3001, and so on.

Do not follow any of these procedures. The seeds you set must not exhibit a pattern.
2. To set the seed, you obtain a pseudorandom number from runiform() and then use the digits from that to form the seed.
This is a bad idea because the pseudorandom-number generator can converge to a cycle. If you obtained the pseudorandom-number generator unrelated to those in Stata, this would work well, but then you would have to find a rule to set the first generator's seed.

Choosing seeds that do not exhibit a pattern is of great importance. That the seeds satisfy the other properties of randomness is minor by comparison.

## Do not set the seed too often

We cannot emphasize this enough: Do not set the seed too often.
To see why this is such a bad idea, consider the limiting case: You set the seed, draw one pseudorandom number, reset the seed, draw again, and so continue. The pseudorandom numbers you obtain will be nothing more than the seeds you run through a mathematical function. The results you obtain will not pass for random unless the seeds you choose pass for random. If you already had such numbers, why are you even bothering to use the pseudorandom-number generator?

The definition of too often is more than once per problem.
If you are running a simulation of 10,000 replications, set the seed at the start of the simulation and do not reset it until the 10,000 th replication is finished. The pseudorandom-number generators provided by Stata have long periods. The longer you go between setting the seed, the more random-like are the numbers produced.

It is sometimes useful later to be able to reproduce in isolation any one of the replications, and so you might be tempted to set the seed to a known value for each of the replications. We negatively mentioned setting the seed to $1,2, \ldots$, and it is in exactly such situations that we have seen this done. The advantage, however, is that you could reproduce the fifth replication merely by setting the seed to 5 and then repeating whatever it is that is to be replicated. If this is your goal, you do not need to reset the seed. You can record the state of the random-number generator, save the state with your replication results, and then use the recorded states later to reproduce whichever of the replications that you wish. This will be discussed in Preserving and restoring the random-number generator state.

There is another reason you might be tempted to set the seed more than once per problem. It sometimes happens that you run a simulation, let's say for 5,000 replications, and then you decide you should have run it for 10,000 replications. Instead of running all 10,000 replications afresh, you decide to save time by running another 5,000 replications and then combining those results with your previous 5,000 results. That is okay. We at StataCorp do this kind of thing. If you do this, it is important that you set the seed especially well, particularly if you repeat this process to add yet another 5,000 replications. It is also important that in each run there be a large enough number of replications, which is say thousands of them.

Even so, do not do this: You want 500,000 replications. To obtain them, you run in batches of 1,000 , setting the seed 500 times. Unless you have a truly random source for the seeds, it is unlikely
you can produce a patternless sequence of 500 seeds. The fact that you ran 1,000 replications in between choosing the seeds does not mitigate the requirement that there be no pattern to the seeds you set.

In all cases, the best solution is to set the seed only once and then use the method we suggest in the next section.

## Preserving and restoring the random-number generator state

In the previous section, we discussed the case in which you might be tempted to set the seed more frequently than otherwise necessary, either to save time or to be able to rerun any one of the replications. In such cases, there is an alternative to setting a new seed: recording the state of the pseudorandom-number generator and then restoring the state later should the need arise.

The state of the default random-number generator in Stata, the 64-bit Mersenne Twister, is a string of about 5,000 characters. The state can be displayed by typing display c(rngstate). It is more practical to save the state, say, in a local macro named state:

```
. local state = c(rngstate)
```

The state can later be restored by typing
. set rngstate 'state'
The state string specifies an entry point into the sequence produced by the pseudorandom-number generator. Let us explain.

The best way to use a pseudorandom-number generator would be to choose a seed once, draw random numbers until you use up the generator, and then get a new generator and choose a new key. Pseudorandom-number generators have a period, after which they repeat the original sequence. That is what we mean by using up a generator. The period of the 64-bit Mersenne Twister, the default pseudorandom-number generator in Stata, is $2^{19937}-1$. This is roughly $10^{6000}$. It is difficult to imagine that you could ever use up this generator.

The string reported by $c$ (rngstate) is an encoded form of the information necessary for Stata to reestablish exactly where it is located in the pseudorandom-number generator's sequence.

We are not seriously suggesting you choose only one seed over your entire lifetime, but let's look at how you might do that. Sometime after birth, when you needed your first random number, you would set your seed,
. set seed 1073741823
On that day, you would draw, say, 10,000 pseudorandom numbers, perhaps to impute some missing values. Being done for the day, you can save the state, and then later restore it.

When you type set rngstate followed by a saved state string, Stata reestablishes the previous state. Thus, the next time you draw a pseudorandom number, Stata will produce the 10,001 st result after setting seed 1073741823. Let's assume that you draw 100,000 numbers this day. Done for the day, you save the state string.

On the third day, after setting the state to the saved state string above, you will be in a position to draw the 110,001 st pseudorandom number.

In this way, you would eat your way through the $2^{19937}-1$ random numbers, but you would be unlikely ever to make it to the end.

We do not expect you to set the seed just once in your life, but using the state string makes it easy to set the seed just once for a problem.

When we do simulations at StataCorp, we record c(rngstate) for each replication. Just like everybody else, we record results from replications as observations in datasets; we just happen to have an extra variable in the dataset, namely, a string variable named state. That string is filled in observation by observation from the then-current values of $c$ (rngstate), which is a function and so can be used in any context that a function can be used in Stata.

Anytime we want to reproduce a particular replication, we thus have the information we need to reset the pseudorandom-number generator, and having it in the dataset is convenient because we had to go there anyway to determine which replication we wanted to reproduce. If we want to add more replications later, we have a state string that we can use to continue from where we left off.

## Reference

Matsumoto, M., and T. Nishimura. 1998. Mersenne Twister: A 623-dimensionally equidistributed uniform pseudorandom number generator. ACM Transactions on Modeling and Computer Simulation 8: 3-30. https://doi.org/10.1145/272991.272995.

## Also see

[R] set - Overview of system parameters
[R] set rng - Set which random-number generator (RNG) to use
$[R]$ set rngstream - Specify the stream for the stream random-number generator
[FN] Random-number functions
[P] version - Version control

## Title

set showbaselevels - Display settings for coefficient tables

## Description Syntax Option Remarks and examples Also see

## Description

set showbaselevels specifies whether to display base levels of factor variables and their interactions in coefficient tables. set showbaselevels on specifies that base levels be reported for factor variables and for interactions whose bases cannot be inferred from their component factor variables. set showbaselevels all specifies that all base levels of factor variables and interactions be reported.
set showemptycells specifies whether to display empty cells in coefficient tables.
set showomitted specifies whether to display omitted coefficients in coefficient tables.
set fvlabel specifies whether to display factor-variable value labels in coefficient tables. set fvlabel on, the default, specifies that the labels be displayed. set fvlabel off specifies that the levels of factor variables rather than the labels be displayed.
set fvwrap \# specifies that long value labels wrap \# lines in the coefficient table. The default is set fvwrap 1, which means that long value labels will be abbreviated to fit on one line.
set fvwrapon specifies whether value labels that wrap will break at word boundaries or break based on available space. set fvwrapon word, the default, specifies that value labels break at word boundaries. set fvwrapon width specifies that value labels break based on available space.

## Syntax

```
set showbaselevels {on|off|all} [, permanently]
set showemptycells {on|off } [, permanently]
set showomitted {on|off } [, permanently]
set fvlabel {on|off} [, permanently]
set fvwrap # [, permanently]
set fvwrapon {word|width} [, permanently]
```


## Option

permanently specifies that, in addition to making the change right now, the setting be remembered and become the default setting when you invoke Stata.

## Remarks and examples

## Example 1

We illustrate the first three set commands using cholesterol2.dta.

```
. use https://www.stata-press.com/data/r18/cholesterol2
(Artificial cholesterol data, empty cells)
. generate x = race
. regress chol race##agegrp x
note: 2.race#2.agegrp identifies no observations in the sample.
note: x omitted because of collinearity.
```

| Source | SS | df | MS | Number of obs | = | 70 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | $\mathrm{F}(13,56)$ | = | 13.51 |
| Model | 15751.6113 | 13 | 1211.66241 | Prob > F | = | 0.0000 |
| Residual | 5022.71559 | 56 | 89.6913498 | R -squared | = | 0.7582 |
|  |  |  |  | Adj R-squared | = | 0.7021 |
| Total | 20774.3269 | 69 | 301.077201 | Root MSE | = | 9.4706 |


| chol | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| :---: | ---: | :---: | ---: | ---: | ---: | ---: |
| race |  |  |  |  |  |  |
| White | 12.84185 | 5.989703 | 2.14 | 0.036 | .8430383 | 24.84067 |
| Other | -.167627 | 5.989703 | -0.03 | 0.978 | -12.16644 | 11.83119 |
|  |  |  |  |  |  |  |
| agegrp |  |  |  |  |  |  |
| 20-29 | 17.24681 | 5.989703 | 2.88 | 0.006 | 5.247991 | 29.24562 |
| 30-39 | 31.43847 | 5.989703 | 5.25 | 0.000 | 19.43966 | 43.43729 |
| 40-59 | 34.86613 | 5.989703 | 5.82 | 0.000 | 22.86732 | 46.86495 |
| 60-79 | 44.43374 | 5.989703 | 7.42 | 0.000 | 32.43492 | 56.43256 |
|  |  |  |  |  |  |  |
| race\#agegrp |  |  |  |  |  |  |
| White\#20-29 |  | (empty) |  |  |  |  |
| White\#30-39 | -22.83983 | 8.470719 | -2.70 | 0.009 | -39.80872 | -5.870939 |
| White\#40-59 | -14.67558 | 8.470719 | -1.73 | 0.089 | -31.64447 | 2.293306 |
| White\#60-79 | -10.51115 | 8.470719 | -1.24 | 0.220 | -27.48004 | 6.457735 |
| Other\#20-29 | -6.054425 | 8.470719 | -0.71 | 0.478 | -23.02331 | 10.91446 |
| Other\#30-39 | -11.48083 | 8.470719 | -1.36 | 0.181 | -28.44971 | 5.488063 |
| Other\#40-59 | -.6796112 | 8.470719 | -0.08 | 0.936 | -17.6485 | 16.28928 |
| Other\#60-79 | -1.578052 | 8.470719 | -0.19 | 0.853 | -18.54694 | 15.39084 |
|  |  |  |  |  |  |  |
| x | 0 | (omitted) |  |  |  |  |
| _cons | 175.2309 | 4.235359 | 41.37 | 0.000 | 166.7464 | 183.7153 |

[^69]. set showomitted off
. set showbaselevels all
. regress chol race\#\#agegrp x
note: 2.race\#2.agegrp identifies no observations in the sample.
note: $x$ omitted because of collinearity.

| Source | SS | df | MS | Number of obs | = | 70 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | F (13, 56) | = | 13.51 |
| Model | 15751.6113 | 13 | 1211.66241 | Prob > F | = | 0.0000 |
| Residual | 5022.71559 | 56 | 89.6913498 | R -squared | = | 0.7582 |
| Total | 20774.3269 | 69 | 301.077201 | Adj R-squared Root MSE | = | 0.7021 9.4706 |


| chol | Coefficient | Std. err. | t | $P>\|t\|$ | [95\% conf. interval] |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| race |  |  |  |  |  |  |
| Black | 0 | (base) |  |  |  |  |
| White | 12.84185 | 5.989703 | 2.14 | 0.036 | . 8430383 | 24.84067 |
| Other | -. 167627 | 5.989703 | -0.03 | 0.978 | -12.16644 | 11.83119 |
| agegrp |  |  |  |  |  |  |
| 10-19 | 0 | (base) |  |  |  |  |
| 20-29 | 17.24681 | 5.989703 | 2.88 | 0.006 | 5.247991 | 29.24562 |
| 30-39 | 31.43847 | 5.989703 | 5.25 | 0.000 | 19.43966 | 43.43729 |
| 40-59 | 34.86613 | 5.989703 | 5.82 | 0.000 | 22.86732 | 46.86495 |
| 60-79 | 44.43374 | 5.989703 | 7.42 | 0.000 | 32.43492 | 56.43256 |
| race\#agegrp |  |  |  |  |  |  |
| Black\#10-19 | 0 | (base) |  |  |  |  |
| Black\#20-29 | 0 | (base) |  |  |  |  |
| Black\#30-39 | 0 | (base) |  |  |  |  |
| Black\#40-59 | 0 | (base) |  |  |  |  |
| Black\#60-79 | 0 | (base) |  |  |  |  |
| White\#10-19 | 0 | (base) |  |  |  |  |
| White\#30-39 | -22.83983 | 8.470719 | -2.70 | 0.009 | -39.80872 | -5.870939 |
| White\#40-59 | -14.67558 | 8.470719 | -1.73 | 0.089 | -31.64447 | 2.293306 |
| White\#60-79 | -10.51115 | 8.470719 | -1.24 | 0.220 | -27.48004 | 6.457735 |
| Other\#10-19 | 0 | (base) |  |  |  |  |
| Other\#20-29 | -6.054425 | 8.470719 | -0.71 | 0.478 | -23.02331 | 10.91446 |
| Other\#30-39 | -11.48083 | 8.470719 | -1.36 | 0.181 | -28.44971 | 5.488063 |
| Other\#40-59 | -. 6796112 | 8.470719 | -0.08 | 0.936 | -17.6485 | 16.28928 |
| Other\#60-79 | -1.578052 | 8.470719 | -0.19 | 0.853 | -18.54694 | 15.39084 |
| _cons | 175.2309 | 4.235359 | 41.37 | 0.000 | 166.7464 | 183.7153 |

To restore the display of empty cells, omitted predictors, and baselevels to their command-specific default behavior, type
. set showemptycells
. set showomitted
. set showbaselevels
. regress chol race\#\#agegrp x
note: 2.race*2.agegrp identifies no observations in the sample.
note: $x$ omitted because of collinearity.

| Source | SS | df | MS | Number of obs | = | 70 |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  |  |  | $\mathrm{F}(13,56)$ | = | 13.51 |
| Model | 15751.6113 | 13 | 1211.66241 | Prob > F | = | 0.0000 |
| Residual | 5022.71559 | 56 | 89.6913498 | R-squared | = | 0.7582 |
|  |  |  |  | Adj R-squared | = | 0.7021 |
| Total | 20774.3269 | 69 | 301.077201 | Root MSE | = | 9.4706 |


| chol | Coefficient | Std. err. | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| :---: | ---: | :---: | :---: | :---: | :---: | :---: |
| race |  |  |  |  |  |  |
| White | 12.84185 | 5.989703 | 2.14 | 0.036 | .8430383 | 24.84067 |
| Other | -.167627 | 5.989703 | -0.03 | 0.978 | -12.16644 | 11.83119 |
|  |  |  |  |  |  |  |
| agegrp |  |  |  |  |  |  |
| 20-29 | 17.24681 | 5.989703 | 2.88 | 0.006 | 5.247991 | 29.24562 |
| 30-39 | 31.43847 | 5.989703 | 5.25 | 0.000 | 19.43966 | 43.43729 |
| 40-59 | 34.86613 | 5.989703 | 5.82 | 0.000 | 22.86732 | 46.86495 |
| 60-79 | 44.43374 | 5.989703 | 7.42 | 0.000 | 32.43492 | 56.43256 |
|  |  |  |  |  |  |  |
| race\#agegrp |  |  |  |  |  |  |
| White\#20-29 |  | (empty) |  |  |  |  |
| White\#30-39 | -22.83983 | 8.470719 | -2.70 | 0.009 | -39.80872 | -5.870939 |
| White\#40-59 | -14.67558 | 8.470719 | -1.73 | 0.089 | -31.64447 | 2.293306 |
| White\#60-79 | -10.51115 | 8.470719 | -1.24 | 0.220 | -27.48004 | 6.457735 |
| Other\#20-29 | -6.054425 | 8.470719 | -0.71 | 0.478 | -23.02331 | 10.91446 |
| Other\#30-39 | -11.48083 | 8.470719 | -1.36 | 0.181 | -28.44971 | 5.488063 |
| Other\#40-59 | -.6796112 | 8.470719 | -0.08 | 0.936 | -17.6485 | 16.28928 |
| Other\#60-79 | -1.578052 | 8.470719 | -0.19 | 0.853 | -18.54694 | 15.39084 |
|  |  |  |  |  |  |  |
| x | 0 | (omitted) |  |  |  |  |
| _cons | 175.2309 | 4.235359 | 41.37 | 0.000 | 166.7464 | 183.7153 |

## Example 2

We illustrate the last three set commands using jaw.dta.

| Equation | Obs | Parms | RMSE | "R-sq" | F | $\mathrm{P}>\mathrm{F}$ |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| y1 | 27 | 3 | 10.42366 | 0.2966 | 5.060804 | 0.0147 |
| y2 | 27 | 3 | 6.325398 | 0.1341 | 1.858342 | 0.1777 |
| y3 | 27 | 3 | 5.976973 | 0.1024 | 1.368879 | 0.2735 |


|  | Coefficient | Std. err. | t | $p>\|t\|$ | [95\% con | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| y1 |  |  |  |  |  |  |
| fracture |  |  |  |  |  |  |
| Two compou.. | -8.833333 | 4.957441 | -1.78 | 0.087 | -19.06499 | 1.398322 |
| One simple.. | 6 | 5.394759 | 1.11 | 0.277 | -5.134235 | 17.13423 |
| _cons | 37 | 3.939775 | 9.39 | 0.000 | 28.8687 | 45.1313 |
| y2 |  |  |  |  |  |  |
| fracture |  |  |  |  |  |  |
| Two compou.. | -5.761905 | 3.008327 | -1.92 | 0.067 | -11.97079 | . 446977 |
| One simple.. | -3.053571 | 3.273705 | -0.93 | 0.360 | -9.810166 | 3.703023 |
| _cons | 38.42857 | 2.390776 | 16.07 | 0.000 | 33.49425 | 43.36289 |
| y3 |  |  |  |  |  |  |
| fracture <br> Two compou.. | 4.261905 | 2.842618 | 1.50 | 0.147 | -1.60497 | 10.12878 |
| One simple.. | . 9285714 | 3.093377 | 0.30 | 0.767 | -5.455846 | 7.312989 |
| _cons | 58.57143 | 2.259083 | 25.93 | 0.000 | 53.90891 | 63.23395 |

. set fvwrap 2
. mvreg y1 y2 y3 = i.fracture

| Equation | Obs | Parms | RMSE | "R-sq" | F | P>F |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: |
| y1 | 27 | 3 | 10.42366 | 0.2966 | 5.060804 | 0.0147 |
| y2 | 27 | 3 | 6.325398 | 0.1341 | 1.858342 | 0.1777 |
| y3 | 27 | 3 | 5.976973 | 0.1024 | 1.368879 | 0.2735 |


. set fvwrapon width
. mvreg y1 y2 y3 = i.fracture

| Equation | Obs | Parms | RMSE | "R-sq" | F | P>F |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: |
| y1 | 27 | 3 | 10.42366 | 0.2966 | 5.060804 | 0.0147 |
| y2 | 27 | 3 | 6.325398 | 0.1341 | 1.858342 | 0.1777 |
| y3 | 27 | 3 | 5.976973 | 0.1024 | 1.368879 | 0.2735 |


. set fvlabel off
. mvreg y1 y2 y3 = i.fracture

| Equation | Obs | Parms | RMSE | "R-sq" | F | P>F |
| :--- | ---: | ---: | ---: | ---: | ---: | ---: |
| y1 | 27 | 3 | 10.42366 | 0.2966 | 5.060804 | 0.0147 |
| y2 | 27 | 3 | 6.325398 | 0.1341 | 1.858342 | 0.1777 |
| y3 | 27 | 3 | 5.976973 | 0.1024 | 1.368879 | 0.2735 |


|  | Coefficient | Std. err | t | $P>\|t\|$ | [95\% conf | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| y1 |  |  |  |  |  |  |
| fracture |  |  |  |  |  |  |
| 2 | -8.833333 | 4.957441 | -1.78 | 0.087 | -19.06499 | 1.398322 |
| 3 | 6 | 5.394759 | 1.11 | 0.277 | -5.134235 | 17.13423 |
| _cons | 37 | 3.939775 | 9.39 | 0.000 | 28.8687 | 45.1313 |
| y2 |  |  |  |  |  |  |
| fracture |  |  |  |  |  |  |
| 2 | -5.761905 | 3.008327 | -1.92 | 0.067 | -11.97079 | . 446977 |
| 3 | -3.053571 | 3.273705 | -0.93 | 0.360 | -9.810166 | 3.703023 |
| _cons | 38.42857 | 2.390776 | 16.07 | 0.000 | 33.49425 | 43.36289 |
| y3 |  |  |  |  |  |  |
| fracture |  |  |  |  |  |  |
| 2 | 4.261905 | 2.842618 | 1.50 | 0.147 | -1.60497 | 10.12878 |
| 3 | . 9285714 | 3.093377 | 0.30 | 0.767 | -5.455846 | 7.312989 |
| _cons | 58.57143 | 2.259083 | 25.93 | 0.000 | 53.90891 | 63.23395 |

To restore these last three set commands to their defaults, type

```
. set fvlabel on
. set fvwrap 1
. set fvwrapon word
. mvreg y1 y2 y3 = i.fracture
\begin{tabular}{lrrrrrr} 
Equation & Obs & Parms & RMSE & "R-sq" & F & P>F \\
\hline y1 & 27 & 3 & 10.42366 & 0.2966 & 5.060804 & 0.0147 \\
y2 & 27 & 3 & 6.325398 & 0.1341 & 1.858342 & 0.1777 \\
y3 & 27 & 3 & 5.976973 & 0.1024 & 1.368879 & 0.2735
\end{tabular}
```

|  | Coefficient | Std. err. | t | $P>\|t\|$ | [95\% con | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| y1 |  |  |  |  |  |  |
| fracture |  |  |  |  |  |  |
| Two compou.. | -8.833333 | 4.957441 | -1.78 | 0.087 | -19.06499 | 1.398322 |
| One simple.. | 6 | 5.394759 | 1.11 | 0.277 | -5.134235 | 17.13423 |
| _cons | 37 | 3.939775 | 9.39 | 0.000 | 28.8687 | 45.1313 |
| y2 |  |  |  |  |  |  |
| fracture |  |  |  |  |  |  |
| Two compou.. | -5.761905 | 3.008327 | -1.92 | 0.067 | -11.97079 | . 446977 |
| One simple.. | -3.053571 | 3.273705 | -0.93 | 0.360 | -9.810166 | 3.703023 |
| _cons | 38.42857 | 2.390776 | 16.07 | 0.000 | 33.49425 | 43.36289 |
| y3 |  |  |  |  |  |  |
| fracture |  |  |  |  |  |  |
| Two compou.. | 4.261905 .9285714 | 2.842618 3.093377 | 1.50 0.30 | 0.147 0.767 | -1.60497 -5.455846 | 10.12878 7.312989 |
| _cons | 58.57143 | 2.259083 | 25.93 | 0.000 | 53.90891 | 63.23395 |

## Also see

[R] set - Overview of system parameters
[R] query - Display system parameters

## Title

signrank - Equality tests on matched data

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Option for signrank | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

signrank tests the equality of matched pairs of observations by using the Wilcoxon matched-pairs signed-rank test (Wilcoxon 1945). The null hypothesis is that both distributions are the same.
signtest also tests the equality of matched pairs of observations (Arbuthnott [1710], but better explained by Snedecor and Cochran [1989]) by calculating the differences between varname and the expression. The null hypothesis is that the median of the differences is zero; no further assumptions are made about the distributions. This, in turn, is equivalent to the hypothesis that the true proportion of positive (negative) signs is one-half.

For equality tests on unmatched data, see $[R]$ ranksum.

## Quick start

Wilcoxon matched-pairs signed-rank test for v1 and v2
signrank v1 = v2
Compute an exact $p$-value for the signed-rank test
signrank $v 1=\mathrm{v} 2$, exact
Conduct signed-rank test separately for groups defined by levels of catvar
by catvar: signrank v1 = v2
Test that the median of differences between matched pairs v 1 and v 2 is 0
signtest v1 = v2

## Menu

## signrank

Statistics $>$ Nonparametric analysis $>$ Tests of hypotheses $>$ Wilcoxon matched-pairs signed-rank test

## signtest

Statistics $>$ Nonparametric analysis $>$ Tests of hypotheses $>$ Test equality of matched pairs

## Syntax

Wilcoxon matched-pairs signed-rank test
signrank varname $=\exp [$ if $][$ in $]$ [, exact $]$

Sign test of matched pairs
signtest varname $=\exp [$ if $][$ in $]$
by and collect are allowed with signrank and signtest; see [U] 11.1.10 Prefix commands.

## Option for signrank

$\qquad$ Main
exact specifies that the exact $p$-value be computed in addition to the approximate $p$-value. The exact $p$-value is based on the actual randomization distribution of the test statistic. The approximate $p$-value is based on a normal approximation to the randomization distribution. By default, the exact $p$-value is computed for sample sizes $n \leq 200$ because the normal approximation may not be precise in small samples. The exact computation can be suppressed by specifying noexact. For sample sizes larger than 200, you must specify exact to compute the exact $p$-value. The exact computation is available for sample sizes $n \leq 2000$.

## Remarks and examples

## > Example 1: signrank

We are testing the effectiveness of a new fuel additive. We run an experiment with 12 cars. We first run each car without the fuel treatment and measure the mileage. We then add the fuel treatment and repeat the experiment. The results of the experiment are

| Without <br> treatment | With <br> treatment | Without <br> treatment | With <br> treatment |
| :---: | :---: | :---: | :---: |
| 20 | 24 | 18 | 17 |
| 23 | 25 | 24 | 28 |
| 21 | 21 | 20 | 24 |
| 25 | 22 | 24 | 27 |
| 18 | 23 | 23 | 21 |
| 17 | 18 | 19 | 23 |

We create two variables called mpg1 and mpg2, representing mileage without and with the treatment, respectively. We can test the null hypothesis that the treatment had no effect by typing
. use https://www.stata-press.com/data/r18/fuel
. signrank mpg1 = mpg2
Wilcoxon signed-rank test

| Sign | Obs | Sum ranks | Expected |
| ---: | ---: | ---: | ---: |
| Positive | 3 | 13.5 | 38.5 |
| Negative | 8 | 63.5 | 38.5 |
| Zero | 1 | 1 | 1 |
| All | 12 | 78 | 78 |

Unadjusted variance
162.50

Adjustment for ties -1.62
Adjustment for zeros
$-0.25$
Adjusted variance
160.62

H0: mpg1 = mpg2
$z=-1.973$
Prob $>|z|=0.0485$
Exact prob $=0.0479$
Despite the small sample size, the $p$-value computed using a normal approximation, 0.0485 , is similar to the exact $p$-value, 0.0479 . These results indicate that we can reject the null hypothesis at a significance level of 0.05 .

## > Example 2: signtest

signtest tests that the median of the differences is zero, making no further assumptions, whereas signrank assumed that the distributions are equal as well. Using the data above, we type

```
. signtest mpg1 = mpg2
Sign test
\begin{tabular}{r|rr} 
Sign & Observed & Expected \\
\hline Positive & 3 & 5.5 \\
Negative & 8 & 5.5 \\
Zero & 1 & 1 \\
\hline All & 12 & 12
\end{tabular}
One-sided tests:
    HO: median of mpg1 - mpg2 = 0 vs.
    Ha: median of mpg1 - mpg2 > 0
            Pr(#positive >= 3) =
            Binomial(n = 11, x >= 3, p = 0.5) = 0.9673
    H0: median of mpg1 - mpg2 = 0 vs.
    Ha: median of mpg1 - mpg2 < 0
            Pr(#negative >= 8) =
                Binomial(n = 11, x >= 8, p = 0.5) = 0.1133
Two-sided test:
    HO: median of mpg1 - mpg2 = 0 vs.
    Ha: median of mpg1 - mpg2 != 0
            Pr(#positive >= 8 or #negative >= 8) =
            min(1, 2*Binomial(n = 11, x >= 8, p = 0.5)) = 0.2266
```

The summary table indicates that there were three comparisons for which mpg1 exceeded mpg2, eight comparisons for which mpg2 exceeded mpg1, and one comparison for which they were the same.

The $p$-values displayed below the summary table are based on a $\operatorname{binomial}(n, p=1 / 2)$ distribution for the test statistic, which is the number of positive or negative signs. Zeros are ignored. The $n$ for the test is the number of nonzero differences. Because no approximation is used, the $p$-values are "exact" $p$-values. The $p$-value for the one-sided test, where the alternative hypothesis is that the median of $\operatorname{mpg} 1-\operatorname{mpg} 2$ is smaller than zero, is 0.1133 . The $p$-value for the two-sided test, where the alternative hypothesis is simply that the median of the differences is different from zero, is $0.2266=2 \times 0.1133$.

## Stored results

signrank stores the following in $r()$ :
Scalars

```
r(N) sample size
    r(N_pos) number of positive comparisons
    r(N_neg) number of negative comparisons
    r(N_tie) number of tied comparisons
    r(z) z statistic
    r(Var_a) adjusted variance
    r(sum_pos) sum of the positive ranks
    r(sum_neg) sum of the negative ranks
    r(p) two-sided p-value from normal approximation
    r(p_l) lower one-sided p-value from normal approximation
    r(p_u) upper one-sided p-value from normal approximation
    r(p_exact) two-sided exact p-value
    r(p_l_exact) lower one-sided exact p-value
    r(p_u_exact) upper one-sided exact p-value
```

signtest stores the following in $r()$ :
Scalars
$r(N) \quad$ sample size
$r$ (N_pos) number of positive comparisons
$r$ ( $N \_$neg) number of negative comparisons
$r\left(N_{-}\right.$tie) number of tied comparisons
$r(\mathrm{p}) \quad$ two-sided $p$-value
$r(p-1) \quad$ lower one-sided $p$-value
$r$ (p_u) upper one-sided $p$-value

## Methods and formulas

For a practical introduction to these techniques with an emphasis on examples rather than theory, see Bland (2015) or Sprent and Smeeton (2007). For a summary of these tests, see Snedecor and Cochran (1989).

Methods and formulas are presented under the following headings:

signrank<br>signtest

## signrank

Both the sign test and Wilcoxon signed-rank tests test the null hypothesis that the distribution of a random variable $D=$ varname - exp has median zero. The sign test makes no additional assumptions, but the Wilcoxon signed-rank test makes the additional assumption that the distribution of $D$ is symmetric. If $D=X_{1}-X_{2}$, where $X_{1}$ and $X_{2}$ have the same distribution, then it follows that the distribution of $D$ is symmetric about zero. Thus, the Wilcoxon signed-rank test is often described as a test of the hypothesis that two distributions are the same, that is, $X_{1} \sim X_{2}$.

Let $d_{j}$ denote the difference for any matched pair of observations,

$$
d_{j}=x_{1 j}-x_{2 j}=\text { varname }-\exp
$$

for $j=1,2, \ldots, n$.
Rank the absolute values of the differences, $\left|d_{j}\right|$, and assign any tied values the average rank. Consider the signs of $d_{j}$, and let

$$
r_{j}=\operatorname{sign}\left(d_{j}\right) \operatorname{rank}\left(\left|d_{j}\right|\right)
$$

be the signed ranks. The test statistic is

$$
T_{\mathrm{obs}}=\sum_{j=1}^{n} r_{j}=(\text { sum of ranks for }+ \text { signs })-(\text { sum of ranks for }- \text { signs })
$$

The distribution of the test statistic is based on Fisher's principle of randomization (Fisher 1935). Fisher's idea (stated in a modern way) was to look at a family of transformations of the observed data such that the a priori likelihood (under the null hypothesis) of the transformed data is the same as the likelihood of the observed data. The distribution of the test statistic is then produced by calculating its value for each of the transformed "randomization" datasets, assuming that each dataset is equally likely.

The null hypothesis is that the distribution of $d_{j}$ is symmetric about 0 . Hence, the likelihood is unchanged if we flip signs on the $d_{j}$. The randomization distribution of our test statistic, $T$, is all of its values resulting from the $2^{n}$ possible sign changes for the $d_{j}$. Namely, the distribution is all the $2^{n}$ possible values of

$$
T=\sum_{j=1}^{n} S_{j} r_{j}
$$

where $r_{j}$ are the observed signed ranks (considered fixed) and $S_{j}$ is either +1 or -1 . When the exact option is specified (or implied for $n \leq 200$ ), this distribution is computed using a recursive algorithm whose computational time is proportional to $n^{3}$. (See Fisher [1935] for the principle of randomization; Wilcoxon, Katti, and Wilcox [1970] for the computation with untied ranks; and Baker and Tilbury [1993] for the general recursive algorithm.)
$p$-values can also be computed using a normal approximation to the randomization distribution. For the randomization distribution, the mean and variance are given by

$$
E(T)=0 \quad \text { and } \quad \operatorname{Var}_{\text {adj }}(T)=\sum_{j=1}^{n} r_{j}^{2}
$$

The test statistic for the Wilcoxon signed-rank test is often expressed (equivalently) as the sum of the positive signed ranks, $T_{+}$, where

$$
E\left(T_{+}\right)=\frac{n(n+1)}{4} \quad \text { and } \quad \operatorname{Var}_{\text {adj }}\left(T_{+}\right)=\frac{1}{4} \sum_{j=1}^{n} r_{j}^{2}
$$

Zeros and ties do not affect the theory above, and the exact variance is still given by the above formula for $\operatorname{Var}_{\text {adj }}\left(T_{+}\right)$. When $d_{j}=0$ is observed, $r_{j}$ will always be zero in each of the randomization datasets, using $\operatorname{sign}(0)=0$. (This method of handling zeros is based on the theoretical arguments made by Pratt [1959].) When there are ties, averaged ranks are assigned for each group of ties and then treated the same as other ranks.

The "unadjusted variance" reported by signrank is the variance that the randomization distribution would have had if there had been no ties or zeros:

$$
\operatorname{Var}_{\text {unadj }}\left(T_{+}\right)=\frac{1}{4} \sum_{j=1}^{n} j^{2}=\frac{n(n+1)(2 n+1)}{24}
$$

The adjustment for zeros is the change in the variance when the ranks for the zeros are signed to make $r_{j}=0$,

$$
\Delta \operatorname{Var}_{\text {zero adj }}\left(T_{+}\right)=-\frac{1}{4} \sum_{j=1}^{n_{0}} j^{2}=-\frac{n_{0}\left(n_{0}+1\right)\left(2 n_{0}+1\right)}{24}
$$

where $n_{0}$ is the number of zeros. The adjustment for ties is the change in the variance when the ranks (for nonzero observations) are replaced by averaged ranks:

$$
\Delta \operatorname{Var}_{\text {ties adj }}\left(T_{+}\right)=\operatorname{Var}_{\text {adj }}\left(T_{+}\right)-\operatorname{Var}_{\text {unadj }}\left(T_{+}\right)-\Delta \operatorname{Var}_{\text {zero adj }}\left(T_{+}\right)
$$

A normal approximation is used to calculate

$$
z=\frac{T_{+}-E\left(T_{+}\right)}{\sqrt{\operatorname{Var}_{\mathrm{adj}}\left(T_{+}\right)}}
$$

## signtest

The test statistic for the sign test is the number $n_{+}$of differences

$$
d_{j}=x_{1 j}-x_{2 j}=\text { varname }-\exp
$$

greater than zero.
Again, the distribution of the test statistic is based on Fisher's principle of randomization, which we described above for signrank. For the sign test, the "data" are simply the set of signs of the differences. Under the null hypothesis of the sign test, the probability that $d_{j}$ is less than zero is equal to the probability that $d_{j}$ is greater than zero. Thus, you can transform the observed signs by flipping any number of them, and the set of signs will have the same likelihood. The $2^{n}$ possible sign changes form the family of randomization datasets. If you have no zeros, this procedure leads to $n_{+} \sim \operatorname{binomial}(n, p=1 / 2)$.

But what if some differences are zero? If you do have zeros, changing their signs leaves them as zeros. So, if you observe $n_{0}$ zeros, each of the $2^{n}$ sign-change datasets will also have $n_{0}$ zeros. Hence, the values of $n_{+}$calculated over the sign-change datasets range from 0 to $n-n_{0}$, and the "randomization" distribution of $n_{+}$is $\operatorname{binomial}\left(n-n_{0}, p=1 / 2\right)$.

The work of Arbuthnott (1710) and later eighteenth-century contributions is discussed by Hald (2003, chap. 17).

Frank Wilcoxon (1892-1965) was born in Ireland to American parents. After working in various occupations (including merchant seaman, oil-well pump attendant, and tree surgeon), he settled in chemistry, gaining degrees from Rutgers and Cornell and employment from various companies. Working mainly on the development of fungicides and insecticides, Wilcoxon became interested in statistics in 1925 and made several key contributions to nonparametric methods. After retiring from industry, he taught statistics at Florida State until his death.

## References

Arbuthnott, J. 1710. An argument for divine providence, taken from the constant regularity observed in the births of both sexes. Philosophical Transaction of the Royal Society 27: 186-190. https://doi.org/10.1098/rstl.1710.0011.
Baker, R. D., and J. B. Tilbury. 1993. Algorithm AS 283: Rapid computation of the permutation paired and grouped $t$-tests. Applied Statistics 42: 432-441. https://doi.org/10.2307/2986254.

Bland, M. 2015. An Introduction to Medical Statistics. 4th ed. Oxford: Oxford University Press.
Bradley, R. A. 2001. Frank Wilcoxon. In Statisticians of the Centuries, ed. C. C. Heyde and E. Seneta, 420-424. New York: Springer.

Chatfield, M. D., and A. P. Mander. 2009. The Skillings-Mack test (Friedman test when there are missing data). Stata Journal 9: 299-305.
Fisher, R. A. 1935. The Design of Experiments. Edinburgh: Oliver and Boyd.
Hald, A. 2003. A History of Probability and Statistics and Their Applications before 1750. New York: Wiley.
Harris, T., and J. W. Hardin. 2013. Exact Wilcoxon signed-rank and Wilcoxon Mann-Whitney ranksum tests. Stata Journal 13: 337-343.
Kaiser, J. 2007. An exact and a Monte Carlo proposal to the Fisher-Pitman permutation tests for paired replicates and for independent samples. Stata Journal 7: 402-412.
Newson, R. B. 2006. Confidence intervals for rank statistics: Somers' D and extensions. Stata Journal 6: 309-334.
Pratt, J. W. 1959. Remarks on zeros and ties in the Wilcoxon signed rank procedures. Journal of the American Statistical Association 54: 655-667. https://doi.org/10.1080/01621459.1959.10501526.

Snedecor, G. W., and W. G. Cochran. 1989. Statistical Methods. 8th ed. Ames, IA: Iowa State University Press.
Sprent, P., and N. C. Smeeton. 2007. Applied Nonparametric Statistical Methods. 4th ed. Boca Raton, FL: Chapman and Hall/CRC.

Wilcoxon, F. 1945. Individual comparisons by ranking methods. Biometrics 1: 80-83. https://doi.org/10.2307/3001968.
Wilcoxon, F., S. K. Katti, and R. A. Wilcox. 1970. Critical values and probability levels for the Wilcoxon rank sum test and the Wilcoxon signed rank test. In Selected Tables in Mathematical Statistics, ed. I. of Mathematical Statistics, vol. 1, 171-259. Providence, RI: American Mathematical Society.

## Also see

[R] ranksum - Equality tests on unmatched data
$[\mathrm{R}]$ ttest $-t$ tests (mean-comparison tests)

## Title

simulate - Monte Carlo simulations

| Description | Quick start | Syntax | Options |
| :--- | :--- | :--- | :--- |
| Remarks and examples | References | Also see |  |

## Description

simulate eases the programming task of performing Monte Carlo-type simulations. Typing . simulate exp_list, reps(\#): command
runs command for \# replications and collects the results in exp_list.
command defines the command that performs one simulation. Most Stata commands and userwritten programs can be used with simulate, as long as they follow standard Stata syntax; see [U] 11 Language syntax. The by prefix may not be part of command.
exp_list specifies the expression to be calculated from the execution of command. If no expressions are given, exp_list assumes a default, depending upon whether command changes results in e() or $\mathrm{r}\left(\right.$ ). If command changes results in e() , the default is $\_\mathrm{b}$. If command changes results in $\mathrm{r}($ ) (but not e() ), the default is all the scalars posted to $r()$. It is an error not to specify an expression in exp_list otherwise.

## Quick start

Simple program for use with simulate
Define program myreg to generate data and fit a linear regression

```
program myreg, eclass
    drop _all
    set obs 25
    generate x = rnormal()
    generate y = 3*x + 1 + rnormal()
    regress y x
end
```


## Perform simulation

Record coefficients and SEs from 1,000 simulated replications of program myreg simulate _b _se, reps(1000): myreg

Same as above, and set random-number seed to 5,762 for reproducible results simulate _b _se, reps(1000) seed(5762): myreg

## Syntax

```
simulate \([\) exp_list], reps(\#) [options] : command
```

options

| nodots | suppress replication dots |
| :--- | :--- |
| dots(\#) | display dots every \# replications |
| noisily | display any output from command |
| trace command |  |
| trace | saving (filename, ...) |
| save results to filename |  |
| nolegend | suppress table legend <br> verbose <br> seed(\#) |

All weight types supported by command are allowed; see [U] 11.1.6 weight.

| exp_list contains | (name: elist) elist |
| :---: | :---: |
|  | eexp |
| elist contains | $\begin{aligned} & \text { newvar }=(\text { exp }) \\ & (\text { exp }) \end{aligned}$ |
| eexp is | specname |
|  | [eqno]specname |
| specname is | _b |
|  | _b [] |
|  | _se |
|  | _se [] |
| eqno is | \#\# |
|  | name |

$\exp$ is a standard Stata expression; see [U] 13 Functions and expressions.
Distinguish between [], which are to be typed, and [], which indicate optional arguments.

## Options

reps (\#) is required-it specifies the number of replications to be performed.
nodots and dots(\#) specify whether to display replication dots. By default, one dot character is displayed for each successful replication. An "x" is displayed if command returns an error or if any value in exp_list is missing. You can also control whether dots are displayed using set dots; see $[R]$ set.
nodots suppresses display of the replication dots.
dots (\#) displays dots every \# replications. dots(0) is a synonym for nodots.
noisily requests that any output from command be displayed. This option implies the nodots option.
trace causes a trace of the execution of command to be displayed. This option implies the noisily option.
saving (filename [, suboptions]) creates a Stata data file (.dta file) consisting of (for each statistic in exp_list) a variable containing the replicates.
double specifies that the results for each replication be saved as doubles, meaning 8-byte reals. By default, they are saved as floats, meaning 4-byte reals.
every (\#) specifies that results be written to disk every \#th replication. every () should be specified only in conjunction with saving() when command takes a long time for each replication. This will allow recovery of partial results should some other software crash your computer. See [P] postfile.
replace specifies that filename be overwritten if it exists.
nolegend suppresses display of the table legend. The table legend identifies the rows of the table with the expressions they represent.
verbose requests that the full table legend be displayed. By default, coefficients and standard errors are not displayed.
seed(\#) sets the random-number seed. Specifying this option is equivalent to typing the following command before calling simulate:
. set seed \#

## Remarks and examples

For an introduction to Monte Carlo methods, see Cameron and Trivedi (2022, chap. 5). White (2010) provides a command for analyzing results of simulation studies.
$>$ Example 1: Simulating basic summary statistics
We have a dataset containing means and variances of 100 -observation samples from a lognormal distribution (as a first step in evaluating, say, the coverage of a $95 \%, t$-based confidence interval). Then we perform the experiment 1,000 times.

The following command definition will generate 100 independent observations from a lognormal distribution and compute the summary statistics for this sample.

```
program lnsim, rclass
    version 18.0 // (or version 18.5 for StataNow)
    drop _all
    set obs 100
    generate z = exp(rnormal())
    summarize z
    return scalar mean = r(mean)
    return scalar Var = r(Var)
end
```

We can save 1,000 simulated means and variances from lnsim by typing

```
. set seed 1234
. simulate mean=r(mean) var=r(Var), reps(1000) nodots: lnsim
    Command: lnsim
        mean: r(mean)
            var: r(Var)
```

. describe *

| Variable <br> name | Storage <br> type | Display <br> format | Value <br> label | Variable label |
| :--- | ---: | :--- | :--- | :--- |
| mean | float | $\% 9.0 \mathrm{~g}$ |  | $r($ mean $)$ |
| var | float | $\% 9.0 \mathrm{~g}$ |  | $r($ Var $)$ |

. summarize

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| mean | 1,000 | 1.630648 | .2173062 | 1.106372 | 2.612052 |
| var | 1,000 | 4.60798 | 4.502166 | .966087 | 70.5597 |

## Technical note

Before executing our lnsim simulator, we can verify that it works by executing it interactively.

```
. set seed 1234
```

. lnsim
Number of observations (_N) was 0, now 100.

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| z | 100 | 1.534256 | 1.584568 | .0400387 | 9.818309 |

. return list
scalars:

$$
\begin{aligned}
r(\text { Var }) & =2.510857086217961 \\
r(\text { mean }) & =1.53425569280982
\end{aligned}
$$

## > Example 2: Simulating a regression model

Consider a more complicated problem. Let's experiment with fitting $y_{j}=a+b x_{j}+u_{j}$ when the true model has $a=1, b=2, u_{j}=z_{j}+c x_{j}$, and when $z_{j}$ is $N(0,1)$. We will save the parameter estimates and standard errors and experiment with varying $c . x_{j}$ will be fixed across experiments but will originally be generated as $N(0,1)$. We begin by interactively making the true data:

```
. drop _all
. set obs 100
Number of observations (_N) was 0, now 100.
. set seed 54321
. generate x = rnormal()
. generate true_y = 1+2*x
. save truth
file truth.dta saved
```

Our program is

```
program hetero1
        version 18.0 // (or version 18.5 for StataNow)
        args c
        use truth, clear
        generate y = true_y + (rnormal() + 'c'*x)
        regress y x
end
```

Note the use of ' $c$ ' in our statement for generating y. c is a local macro generated from args c and thus refers to the first argument supplied to hetero1. If we want $c=3$ for our experiment, we type

```
. simulate _b _se, reps(10000): hetero1 3
    (output omitted)
```

Our program hetero1 could, however, be more efficient because it rereads the file truth once every replication. It would be better if we could read the data just once. In fact, if we read in the data right before running simulate, we really should not have to reread for each subsequent replication. A faster version reads

```
program hetero2
    version 18.0 // (or version 18.5 for StataNow)
    args c
        capture drop y
        generate y = true_y + (rnormal() + 'c'*x)
        regress y x
end
```

Requiring that the current dataset has the variables true_y and x may become inconvenient. Another improvement would be to require that the user supply variable names, such as in

```
program hetero3
    version 18.0 // (or version 18.5 for StataNow)
    args truey x c
    capture drop y
    generate y = 'truey' + (rnormal() + 'c'*'x')
    regress y x
end
```

Thus, we can type

```
. simulate _b _se, reps(10000): hetero3 true_y x 3
    (output omitted)
```

$>$ Example 3: Simulating a ratio of statistics
Now, let's consider the problem of simulating the ratio of two medians. Suppose that each sample of size $n_{i}$ comes from a normal population with a mean $\mu_{i}$ and standard deviation $\sigma_{i}$, where $i=1,2$. We write the program below and save it as a text file called myratio. ado (see [U] $\mathbf{1 7}$ Ado-files). Our program is an rclass command that requires six arguments as input, identified by the local macros n 1 , mu1, sigma1, n 2 , mu2, and sigma2, which correspond to $n_{1}, \mu_{1}, \sigma_{1}, n_{2}, \mu_{2}$, and $\sigma_{2}$, respectively. With these arguments, myratio will generate the data for the two samples, use summarize to compute the two medians and store the ratio of the medians in $r$ (ratio).

```
program myratio, rclass
        version 18.0 // (or version 18.5 for StataNow)
        args n1 mu1 sigma1 n2 mu2 sigma2
        // generate the data
        drop _all
        local N = 'n1'+'n2'
        set obs 'N'
        tempvar y
        generate 'y' = rnormal()
        replace 'y' = cond(_n<='n1','mu1'+'y'*'sigma1','mu2'+'y'*'sigma2')
        // calculate the medians
        tempname m1
        summarize 'y' if _n<='n1', detail
```

```
            scalar 'm1' = r(p50)
            summarize 'y' if _n>'n1', detail
            // store the results
return scalar ratio = 'm1' / r(p50)
end
```

The result of running our simulation is

```
. set seed 19192
. simulate ratio=r(ratio), reps(1000) nodots: myratio 5 3 1 10 3 2
    Command: myratio 5 3 1 10 3 2
        ratio: r(ratio)
. summarize
\begin{tabular}{r|rrrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline ratio & 1,000 & 1.10875 & .5219166 & .3606606 & 9.857285
\end{tabular}
```


## Technical note

Stata lets us do simulations of simulations and simulations of bootstraps. Stata's bootstrap command (see [R] bootstrap) works much like simulate, except that it feeds the user-written program a bootstrap sample. Say that we want to evaluate the bootstrap estimator of the standard error of the median when applied to lognormally distributed data. We want to perform a simulation, resulting in a dataset of medians and bootstrap estimated standard errors.

As background, summarize (see [R] summarize) calculates summary statistics, leaving the mean in $r$ (mean) and the standard deviation in $r(s d)$. summarize with the detail option also calculates summary statistics, but more of them, and leaves the median in $r(p 50)$.

Thus, our plan is to perform simulations by randomly drawing a dataset: we calculate the median of our random sample, we use bootstrap to obtain a dataset of medians calculated from bootstrap samples of our random sample, the standard deviation of those medians is our estimate of the standard error, and the summary statistics are stored in the results of summarize.

Our simulator is

```
program define bsse, rclass
    version 18.0 // (or version 18.5 for StataNow)
    drop _all
    set obs 100
    generate x = rnormal()
    tempfile bsfile
    bootstrap midp=r(p50), rep(100) saving('bsfile'): summarize x, detail
    use 'bsfile', clear
    summarize midp
    return scalar mean = r(mean)
    return scalar sd = r(sd)
end
```

We can obtain final results, running our simulation 1,000 times, by typing

```
set seed 48901
simulate med=r(mean) bs_se=r(sd), reps(1000): bsse
    Command: bsse
        med: r(mean)
        bs_se: r(sd)
```




```
> .......130.........140.........150..........160..........170...... . . . . 180. . . . . . . .
> .190.........200.........210.........220..........230..........240........... . . 250..
> .......260.........270......... .280.......... . 290. . . . . . . . 300. . . . . . . . 310. . . . . . . 
> .320.........330......... . 340.......... 350. . . . . . . . 360. . . . . . . . 370 . . . . . . . . . 380.
> . . ....390.........400.........410. . . . . . . .420. . . . . . . .430. . . . . . . . 440. . . . . . . .
> .450........440. . . . . . .470. . . . . . . 480. . . . . . . .490. . . . . . . . 500. . . . . . . . 510. 
> .......520.........530.........540..........550..........560.......... . . 570..........
```



```
> . . . . . 650. . . . . . . .660. . . . . . . .670. . . . . . . .680. . . . . . . .690. . . . . . . . . 700 . . . . . . . .
> .710........720.........730..........740. . . . . . . . 750. . . . . . . . 760. . . . . . . . . 770. 
> ......780........790.........800.........810..........820.......... . . 830..........
> .840........850.........860.........870.........880..........890........... . . 900.
> .......910.........920.........930.........940..........950.......... . . 960......... . 
> .970.........980.........990......... . 1,000 done
summarize
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline med & 1,000 & -.0013359 & .1221602 & -.3795549 & .3656219 \\
bs_se & 1,000 & .1278773 & .0303109 & .0614031 & .2484805
\end{tabular}
```

This is a case where the simulation dots (drawn by default, unless the nodots option is specified) will give us an idea of how long this simulation will take to finish as it runs.

## References

Cameron, A. C., and P. K. Trivedi. 2022. Microeconometrics Using Stata. 2nd ed. College Station, TX: Stata Press. Hilbe, J. M. 2010. Creating synthetic discrete-response regression models. Stata Journal 10: 104-124.

Taylor, M. A. 2018. Simulating the central limit theorem. Stata Journal 18: 345-356.
White, I. R. 2010. simsum: Analyses of simulation studies including Monte Carlo error. Stata Journal 10: 369-385.

## Also see

$[\mathrm{R}]$ bootstrap - Bootstrap sampling and estimation
[R] jackknife - Jackknife estimation
[R] permute - Permutation tests
$[R]$ set rngstream - Specify the stream for the stream random-number generator

## Title

sj - Stata Journal installation instructions

## Description Remarks and examples Also see

## Description

The Stata Journal (SJ) is a quarterly journal containing articles about statistics, data analysis, teaching methods, and effective use of Stata's language. The SJ publishes reviewed articles with shorter notes and comments, regular columns, tips, book reviews, and other material of interest to researchers applying statistics in a variety of disciplines. The SJ began publication in the fourth quarter of 2001 with SJ Volume 1 Number 1. You can read all about the SJ at https://www.stata-journal.com.

The SJ is a printed and electronic journal with corresponding software. You must subscribe to the SJ; see https://www.stata-journal.com/subscription/ to purchase a subscription or an individual article. Electronic copies of SJ articles that are older than three years are available at no charge from https://www.stata-journal.com/archives.html.

The software is available at no charge; see below for installation instructions.

## Remarks and examples

Remarks are presented under the following headings:

```
Installing the Stata Journal software
    Obtaining from the Internet by pointing and clicking
    Obtaining from the Internet via command mode
```


## Installing the Stata Journal software

Each issue of the Stata Journal is labeled Volume \#, Number \#. Volume 1 refers to the first year of publication, Volume 2 to the second, and so on. Issues are numbered 1, 2, 3, and 4 within each year. The first issue of the Journal was published in the fourth quarter of 2001, and that issue is numbered Volume 1, Number 1. For installation purposes, we refer to this issue as sj1-1.

The articles, columns, notes, and comments that make up the Stata Journal are assigned a letter-and-number code, called an insert tag, such as st0001, an0034, or ds0011. The letters represent a category: st is the statistics category, an is the announcements category, etc. The numbers are assigned sequentially, so st0001 is the first article in the statistics category.

Sometimes inserts are subsequently updated, either to fix bugs or to add new features. A number such as st0001_1 indicates that this article, column, note, or comment is an update to the original st0001 article. Updates are complete; that is, installing st0001_1 provides all the features of the original article and more.

The Stata Journal software may be obtained by pointing and clicking or by using command mode.
The sections below detail how to install an insert. In all cases, pretend that you wish to install insert st0274 from sj12-4.

## Obtaining from the Internet by pointing and clicking

1. Select Help > SJ and community-contributed features.
2. Click on Stata Journal.
3. Click on sj12-4.
4. Click on st0274.
5. Click on (click here to install).

## Obtaining from the Internet via command mode

Type the following:
. net from https://www.stata-journal.com/software
. net cd sj12-4
. net describe st0274
. net install st0274
The above could be shortened to
. net from https://www.stata-journal.com/software/sj12-4
. net describe st0274
. net install st0274
Alternatively, you could type
. net sj 12-4
. net describe st0274
. net install st0274
but going about it the long way is more entertaining, at least the first time.

[^70]
## Title

sktest - Skewness and kurtosis tests for normality

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Option | Remarks and examples | Stored results | Methods and formulas |
| Acknowledgments | References | Also see |  |

## Description

For each variable in varlist, sktest presents a test for normality based on skewness and another based on kurtosis and then combines the two tests into an overall test statistic. sktest requires a minimum of eight observations to make its calculations. See [MV] mvtest normality for multivariate tests of normality.

## Quick start

Test for normality of v 1 based on skewness and kurtosis
sktest v1
Separate tests for v1 and v2
sktest v1 v2
With frequency weights in wvar
sktest v1 v2 [fweight=wvar]
Suppress adjustment to the overall $\chi^{2}$ test
sktest v1 v2, noadjust

## Menu

Statistics $>$ Summaries, tables, and tests $>$ Distributional plots and tests $>$ Skewness and kurtosis normality tests

## Syntax

sktest varlist [if][in] [weight] [, options]
options
Description

Main
noadjust do not adjust the overall $\chi^{2}$ test statistic and its $p$-value
nolstretch
do not automatically widen table for long variable names
collect is allowed; see [U] 11.1.10 Prefix commands.
aweights and fweights are allowed; see [U] 11.1.6 weight.
nolstretch does not appear in the dialog box.

## Option

$\qquad$ Main
noadjust suppresses the empirical adjustment made by Royston (1991c) to the overall $\chi^{2}$ test statistic and its $p$-value and presents the unaltered test as described by D'Agostino, Belanger, and D'Agostino (1990).

The following option is available with sktest but is not shown in the dialog box: nolstretch; see [R] Estimation options.

## Remarks and examples

Also see [R] swilk for the Shapiro-Wilk and Shapiro-Francia tests for normality. Those tests are, in general, preferred for nonaggregated data (Gould and Rogers 1991; Gould 1992; Royston 1991c). Moreover, a normal quantile plot should be used with any test for normality; see [R] Diagnostic plots for more information.

## > Example 1

Using our automobile dataset, we will test whether the variables mpg and trunk are normally distributed:

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. sktest mpg trunk
Skewness and kurtosis tests for normality
\begin{tabular}{r|rrrrr} 
Variable & Obs & \(\operatorname{Pr}\) (skewness) & \(\operatorname{Pr}\) (kurtosis) & Adj chi2(2) & Prob>chi2 \\
\hline mpg & 74 & 0.0015 & 0.0804 & 10.95 & 0.0042 \\
trunk & 74 & 0.9115 & 0.0445 & 4.19 & 0.1228
\end{tabular}
```

We can reject the hypothesis that mpg is normally distributed, but we cannot reject the hypothesis that trunk is normally distributed, at least at the $12 \%$ level. The kurtosis for trunk is 2.19 , as can be verified by issuing the command

```
. summarize trunk, detail
(output omitted)
```

and the $p$-value of 0.0445 shown in the table above indicates that it is significantly different from the kurtosis of a normal distribution at the $5 \%$ significance level. However, on the basis of skewness alone, we cannot reject the hypothesis that trunk is normally distributed.

## - Technical note

sktest implements the test as described by D'Agostino, Belanger, and D'Agostino (1990) but with the adjustment made by Royston (1991c). In the above example, if we had specified the noadjust option, the $\chi^{2}$ values would have been 13.13 for mpg and 4.05 for trunk. With the adjustment, the $\chi^{2}$ value might show as ' $\because$ '. This result should be interpreted as an absurdly large number; the data are most certainly not normal.

## Stored results

sktest stores the following in $r()$ :
Scalars
$r(N) \quad$ number of observations
$r$ (chi2) overall $\chi^{2}$
r (p_skew) $\quad p$-value for test based on skewness
$r$ (p_kurt) $p$-value for test based on kurtosis
$r$ (p_chi2) $\quad p$-value for overall $\chi^{2}$ test
Matrices
$r$ (table) matrix of displayed results, one row per variable

## Methods and formulas

sktest implements the test described by D'Agostino, Belanger, and D'Agostino (1990) with the empirical correction developed by Royston (1991c).

Let $g_{1}$ denote the coefficient of skewness and $b_{2}$ denote the coefficient of kurtosis as calculated by summarize, and let $n$ denote the sample size. If weights are specified, then $g_{1}, b_{2}$, and $n$ denote the weighted coefficients of skewness and kurtosis and weighted sample size, respectively. See $[R]$ summarize for the formulas for skewness and kurtosis.

To perform the test of skewness, we compute
and

$$
\begin{aligned}
Y & =g_{1}\left\{\frac{(n+1)(n+3)}{6(n-2)}\right\}^{1 / 2} \\
\beta_{2}\left(g_{1}\right) & =\frac{3\left(n^{2}+27 n-70\right)(n+1)(n+3)}{(n-2)(n+5)(n+7)(n+9)} \\
W^{2} & =-1+\left[2\left\{\beta_{2}\left(g_{1}\right)-1\right\}\right]^{1 / 2} \\
\alpha & =\left\{2 /\left(W^{2}-1\right)\right\}^{1 / 2}
\end{aligned}
$$

Then, the distribution of the test statistic

$$
Z_{1}=\frac{1}{\sqrt{\ln W}} \ln \left[Y / \alpha+\left\{(Y / \alpha)^{2}+1\right\}^{1 / 2}\right]
$$

is approximately standard normal under the null hypothesis that the data are distributed normally.

To perform the test of kurtosis, we compute
and

$$
\begin{aligned}
E\left(b_{2}\right) & =\frac{3(n-1)}{n+1} \\
\operatorname{var}\left(b_{2}\right) & =\frac{24 n(n-2)(n-3)}{(n+1)^{2}(n+3)(n+5)} \\
X & =\left\{b_{2}-E\left(b_{2}\right)\right\} / \sqrt{\operatorname{var}\left(b_{2}\right)} \\
\sqrt{\beta_{1}\left(b_{2}\right)} & =\frac{6\left(n^{2}-5 n+2\right)}{(n+7)(n+9)}\left\{\frac{6(n+3)(n+5)}{n(n-2)(n-3)}\right\}^{1 / 2} \\
A & =6+\frac{8}{\sqrt{\beta_{1}\left(b_{2}\right)}}\left[\frac{2}{\sqrt{\beta_{1}\left(b_{2}\right)}}+\left\{1+\frac{4}{\beta_{1}\left(b_{2}\right)}\right\}^{1 / 2}\right]
\end{aligned}
$$

Then, the distribution of the test statistic

$$
Z_{2}=\frac{1}{\sqrt{2 /(9 A)}}\left[\left(1-\frac{2}{9 A}\right)-\left\{\frac{1-2 / A}{1+X \sqrt{2 /(A-4)}}\right\}^{1 / 3}\right]
$$

is approximately standard normal under the null hypothesis that the data are distributed normally.
D'Agostino, Balanger, and D'Agostino Jr.'s omnibus test of normality uses the statistic

$$
K^{2}=Z_{1}^{2}+Z_{2}^{2}
$$

which has approximately a $\chi^{2}$ distribution with 2 degrees of freedom under the null of normality.
Royston (1991c) proposed the following adjustment to the test of normality, which sktest uses by default. Let $\Phi(x)$ denote the cumulative standard normal distribution function for $x$, and let $\Phi^{-1}(p)$ denote the inverse cumulative standard normal function [that is, $\left.x=\Phi^{-1}\{\Phi(x)\}\right]$. Define the following terms:
and

$$
\begin{aligned}
Z_{c} & =-\Phi^{-1}\left\{\exp \left(-\frac{1}{2} K^{2}\right)\right\} \\
Z_{t} & =0.55 n^{0.2}-0.21 \\
a_{1} & =(-5+3.46 \ln n) \exp (-1.37 \ln n) \\
b_{1} & =1+(0.854-0.148 \ln n) \exp (-0.55 \ln n) \\
a_{2} & =a_{1}-\{2.13 /(1-2.37 \ln n)\} Z_{t} \\
b_{2} & =2.13 /(1-2.37 \ln n)+b 1
\end{aligned}
$$

If $Z_{c}<-1$ set $Z=Z_{c}$; else if $Z_{c}<Z_{t}$ set $Z=a_{1}+b_{1} Z_{c}$; else set $Z=a_{2}+b_{2} Z_{c}$. Define $P=1-\Phi(Z)$. Then, $K^{2}=-2 \ln P$ is approximately distributed $\chi^{2}$ with 2 degrees of freedom.

The relative merits of the skewness and kurtosis test versus the Shapiro-Wilk and Shapiro-Francia tests have been a subject of debate. Our recommendation is to use the Shapiro-Francia test whenever possible, that is, whenever dealing with nonaggregated or ungrouped data (Gould and Rogers 1991; Gould 1992); see [R] swilk. If normality is rejected, use sktest to determine the source of the problem.

As both D'Agostino, Belanger, and D'Agostino (1990) and Royston (1991d) mention, researchers should also examine the normal quantile plot to determine normality rather than blindly relying on a few test statistics. See the qnorm command documented in $[R]$ Diagnostic plots for more information on normal quantile plots.
sktest is similar in spirit to the Jarque-Bera (1987) test of normality. The Jarque-Bera test statistic is also calculated from the sample skewness and kurtosis, though it is based on asymptotic standard errors with no corrections for sample size. In effect, sktest offers two adjustments for sample size, that of Royston (1991c) and that of D'Agostino, Belanger, and D'Agostino (1990).

## Acknowledgments

sktest has benefited greatly by the comments and work of Patrick Royston of the MRC Clinical Trials Unit, London, and coauthor of the Stata Press book Flexible Parametric Survival Analysis Using Stata: Beyond the Cox Model. At this point, the program should be viewed as due as much to Royston as to us, except, of course, for any errors. We are also indebted to Nicholas J. Cox of the Department of Geography at Durham University, UK, and coeditor of the Stata Journal and author of Speaking Stata Graphics for his helpful comments.

## References

Alejo, J., A. K. Bera, A. F. Galvao, G. Montes-Rojas, and Z. Xiao. 2016. Tests for normality based on the quantile-mean covariance. Stata Journal 16: 1039-1057.
D'Agostino, R. B., A. J. Belanger, and R. B. D'Agostino, Jr. 1990. A suggestion for using powerful and informative tests of normality. American Statistician 44: 316-321. https://doi.org/10.2307/2684359.
__. 1991. sg3.3: Comment on tests of normality. Stata Technical Bulletin 3: 20. Reprinted in Stata Technical Bulletin Reprints, vol. 1, pp. 105-106. College Station, TX: Stata Press.
Gould, W. W. 1991. sg3: Skewness and kurtosis tests of normality. Stata Technical Bulletin 1: 20-21. Reprinted in Stata Technical Bulletin Reprints, vol. 1, pp. 99-101. College Station, TX: Stata Press.
——. 1992. sg11.1: Quantile regression with bootstrapped standard errors. Stata Technical Bulletin 9: 19-21. Reprinted in Stata Technical Bulletin Reprints, vol. 2, pp. 137-139. College Station, TX: Stata Press.
Gould, W. W., and W. H. Rogers. 1991. sg3.4: Summary of tests of normality. Stata Technical Bulletin 3: 20-23. Reprinted in Stata Technical Bulletin Reprints, vol. 1, pp. 106-110. College Station, TX: Stata Press.

Jarque, C. M., and A. K. Bera. 1987. A test for normality of observations and regression residuals. International Statistical Review 2: 163-172. https://doi.org/10.2307/1403192.
Marchenko, Y. V., and M. G. Genton. 2010. A suite of commands for fitting the skew-normal and skew-t models. Stata Journal 10: 507-539.

Royston, P. 1991a. sg3.1: Tests for departure from normality. Stata Technical Bulletin 2: 16-17. Reprinted in Stata Technical Bulletin Reprints, vol. 1, pp. 101-104. College Station, TX: Stata Press.
__. 1991b. sg3.2: Shapiro-Wilk and Shapiro-Francia tests. Stata Technical Bulletin 3: 19. Reprinted in Stata Technical Bulletin Reprints, vol. 1, p. 105. College Station, TX: Stata Press.
—. 1991c. sg3.5: Comment on sg3.4 and an improved D'Agostino test. Stata Technical Bulletin 3: 23-24. Reprinted in Stata Technical Bulletin Reprints, vol. 1, pp. 110-112. College Station, TX: Stata Press.

1991d. sg3.6: A response to sg3.3: Comment on tests of normality. Stata Technical Bulletin 4: 8-9. Reprinted in Stata Technical Bulletin Reprints, vol. 1, pp. 112-114. College Station, TX: Stata Press.

## Also see

[R] Diagnostic plots - Distributional diagnostic plots
$[R]$ ladder - Ladder of powers
[R] lv - Letter-value displays
[R] swilk - Shapiro-Wilk and Shapiro-Francia tests for normality
[MV] mvtest normality - Multivariate normality tests

## Title

slogit — Stereotype logistic regression

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Options | Remarks and examples | Stored results | Methods and formulas |
| References | Also see |  |  |

## Description

slogit fits Anderson's (1984) maximum-likelihood stereotype logistic regression model for categorical dependent variables. Stereotype logistic models can be used when the relevance of the ordering is unclear. These models do not impose the proportional-odds assumption.

## Quick start

One-dimensional model of y as a function of x 1 and x 2
slogit y x1 x2
Add indicators for categorical variable a and set $\mathrm{y}=1$ as the base category slogit y x1 x2 i.a, baseoutcome(1)

Multidimensional model reparameterizing a multinomial logit when y has 4 categories slogit y x1 x2 i.a, dimensions(3) baseoutcome(1)

## Menu

Statistics $>$ Categorical outcomes $>$ Stereotype logistic regression

## Syntax

> slogit depvar [indepvars] [if] [in] [weight] [, options]
options
Model
dimension(\#)
baseoutcome (\#|lbl)
constraints (numlist)
nocorner
SE/Robust
vce(vcetype)

Reporting
level(\#)
nocnsreport
display_options

## Maximization

maximize_options
initialize(initype)
nonormalize
collinear
coeflegend

Description
dimension of the model; default is dimension(1)
set the base outcome to \# or lbl; default is the last outcome
apply specified linear constraints
do not generate the corner constraints
vcetype may be oim, robust, cluster clustvar, opg, bootstrap, or jackknife
set confidence level; default is level(95)
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
control the maximization process; seldom used
method of initializing scale parameters; initype can be constant,
random, or svd; see Options for details
do not normalize the numeric variables
keep collinear variables
display legend instead of statistics
indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
bootstrap, by, collect, fp, jackknife, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands.
Weights are not allowed with the bootstrap prefix; see $[\mathrm{R}]$ bootstrap.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

## Options

$\qquad$ Model
dimension(\#) specifies the dimension of the model, which is the number of equations required to describe the relationship between the dependent variable and the independent variables. The maximum dimension is $\min (m-1, p)$, where $m$ is the number of categories of the dependent variable and $p$ is the number of independent variables in the model. The stereotype model with maximum dimension is a reparameterization of the multinomial logistic model.
baseoutcome (\#|lbl) specifies the outcome level whose scale parameters and intercept are constrained to be zero. The base outcome may be specified as a number or a label. By default, slogit assumes
that the outcome levels are ordered and uses the largest level of the dependent variable as the base outcome.
constraints (numlist); see [R] Estimation options.
By default, the linear equality constraints suggested by Anderson (1984), termed the corner constraints, are generated for you. You can add constraints to these as needed, or you can turn off the corner constraints by specifying nocorner. These constraints are in addition to the constraints placed on the $\phi$ parameters corresponding to baseoutcome(\#).
nocorner specifies that slogit not generate the corner constraints. If you specify nocorner, you must specify at least dimension() $\times$ dimension() constraints for the model to be identified.

## SE/Robust

vce(vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
If specifying vce(bootstrap) or vce(jackknife), you must also specify baseoutcome().

## Reporting

level(\#), nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\% fint), pformat (\% fit) , sformat (\% fint), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate (\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from(init_specs); see [R] Maximize. These options are seldom used.

Setting the optimization type to technique (bhhh) resets the default vcetype to vce (opg).
initialize (constant $|\underline{\text { random }}|$ svd) specifies how initial estimates are computed. The default, initialize (constant), is to set the scale parameters to the constant $\min (1 / 2,1 / d)$, where $d$ is the dimension specified in dimension().
initialize(random) requests that uniformly distributed random numbers between 0 and 1 be used as initial values for the scale parameters. If you specify this option, you should also use set seed to ensure that you can replicate your results; see $[R]$ set seed.
initialize(svd) requests that a singular value decomposition (SVD) be performed on the matrix of regression estimates from mlogit to reduce its rank to the dimension specified in dimension(). slogit uses the reduced-rank components of the SVD as initial estimates for the scale and regression coefficients. For details, see Methods and formulas.
nonormalize specifies that the numeric variables not be normalized. Normalization of the numeric variables improves numerical stability but consumes more memory in generating temporary doubleprecision variables. Variables that are of type byte are not normalized, and if initial estimates are specified using the from() option, normalization of variables is not performed. See Methods and formulas for more information.

The following options are available with slogit but are not shown in the dialog box: collinear, coeflegend; see [R] Estimation options.

## Remarks and examples

Remarks are presented under the following headings:
Introduction
One-dimensional model
Higher-dimension models

## Introduction

Like multinomial logistic and ordered logistic models, stereotype logistic models are used with categorical dependent variables. They are often used when subjects are requested to assess or judge something. In a multinomial logistic model, the categories cannot be ranked. By contrast, in an ordered logistic model, the categories follow a natural ranking scheme and are subject to the proportional-odds assumption. Stereotype logistic regression can be viewed as a compromise between these two models and is primarily used when you are unsure of the relevance of the ordering of the outcome.

A common case is when subjects are asked to assess or judge something. For example, consider a survey in which consumers are asked to rate the quality of a product on a scale from 1 to 5 , with 1 indicating poor quality and 5 indicating excellent quality. If the categories are monotonically related to one underlying latent variable, the ordered logistic model is appropriate. However, suppose that consumers weigh two or three latent factors when assessing quality. The stereotype logistic model is preferred to the ordered logistic model in this case because it allows you to specify multiple equations to capture the effects of the latent variables. Unlike multinomial logit models, the number of equations you specify could be fewer than $m-1$, where $m$ is the number of categories of the dependent variable. Stereotype logistic models are also used when categories may be indistinguishable. Suppose that a consumer must choose among A, B, C, or D. Multinomial logistic modeling assumes that the four choices are distinct in the sense that a consumer choosing one of the goods can distinguish its characteristics from the others. If goods A and B are in fact similar, consumers may be randomly picking between the two. One alternative is to combine the two categories and fit a three-category multinomial logistic model. A more flexible alternative is to use a stereotype logistic model.

In the multinomial logistic model, you estimate $m-1$ parameter vectors $\widetilde{\boldsymbol{\beta}}_{k}, k=1, \ldots, m-1$, where $m$ is the number of categories of the dependent variable. The stereotype logistic model is a restriction on the multinomial model in the sense that there are $d$ parameter vectors, where $d$ is between one and $\min (m-1, p)$, and $p$ is the number of regressors. The relationship between the stereotype model's coefficients $\boldsymbol{\beta}_{j}, j=1, \ldots, d$, and the multinomial model's coefficients is $\tilde{\boldsymbol{\beta}}_{k}=-\sum_{j=1}^{d} \phi_{j k} \boldsymbol{\beta}_{j}$. The $\phi \mathrm{s}$ are scale parameters to be estimated along with the $\boldsymbol{\beta}_{j} \mathrm{~s}$.

Given a row vector of covariates $\mathbf{x}$, let $\eta_{k}=\theta_{k}-\sum_{j=1}^{d} \phi_{j k} \mathbf{x} \boldsymbol{\beta}_{j}$. The probability of observing outcome $k$ is

$$
\operatorname{Pr}\left(Y_{i}=k\right)= \begin{cases}\frac{\exp \left(\eta_{k}\right)}{1+\sum_{l=1}^{m-1} \exp \left(\eta_{l}\right)} & k<m \\ \frac{1}{1+\sum_{l=1}^{m-1} \exp \left(\eta_{l}\right)} & k=m\end{cases}
$$

This model includes a set of $\theta$ parameters so that each equation has an unrestricted constant term. If $d=m-1$, the stereotype model is just a reparameterization of the multinomial logistic model. To identify the $\phi \mathrm{s}$ and the $\boldsymbol{\beta}$ s, you must place at least $d^{2}$ restrictions on the parameters. By default, slogit uses the "corner constraints" $\phi_{j j}=1$ and $\phi_{j k}=0$ for $j \neq k, k \leq d$, and $j \leq d$.

For a discussion of the stereotype logistic model, see Lunt (2005).

## One-dimensional model

## > Example 1

We have 2 years of repair rating data on the make, price, mileage rating, and gear ratio of 104 foreign and 44 domestic automobiles (with 13 missing values on repair rating). We wish to fit a stereotype logistic model to discriminate between the levels of repair rating using mileage, price, gear ratio, and origin of the manufacturer. Here is an overview of our data:


The variable repair can take five values, $1, \ldots, 5$, which represent the subjective rating of the car model's repair record as Poor, Fair, Average, Good, and Excellent.

We wish to fit the one-dimensional stereotype logistic model

$$
\eta_{k}=\theta_{k}-\phi_{k}\left(\beta_{1} \text { foreign }+\beta_{2} \mathrm{mpg}+\beta_{3} \text { price }+\beta_{4} \text { gratio }\right)
$$

for $k<5$ and $\eta_{5}=0$. To fit this model, we type

| Iteration 0: Log likelihood = -237.99802 (not concave) |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Iteration 1: Log likelihood = -204.98644 (not conca |  |  |  |  |  |  |
| Iteration 2: Log likelihood $=-169.79473$ (not conca |  |  |  |  |  |  |
| Iteration 3: Log likelihood $=-167.53649$ |  |  |  |  |  |  |
| Iteration 4: Log likelihood $=-164.63477$ |  |  |  |  |  |  |
| Iteration 5: Log likelihood $=-163.20867$ (not concave) |  |  |  |  |  |  |
| Iteration 6: Log likelihood $=-160.67522$ |  |  |  |  |  |  |
| Iteration 7: Log likelihood $=-159.69646$ |  |  |  |  |  |  |
| Iteration 8: Log likelihood $=-159.35057$ |  |  |  |  |  |  |
| Iteration 9: Log likelihood $=-159.28495$ |  |  |  |  |  |  |
| Iteration 10: Log likelihood $=-159.25748$ |  |  |  |  |  |  |
| Iteration 11: Log likelihood $=-159.25691$ |  |  |  |  |  |  |
| Iteration 12: Log likelihood = -159.25691 |  |  |  |  |  |  |
| Stereotype logistic regression |  |  |  |  | Number of obs | $=135$ |
|  |  |  |  |  | Wald chi2(4) | 9.33 |
| Log likelihood = -159.25691 $\quad$ Prob > chi2 $=0.0535$ |  |  |  |  |  |  |
| ( 1) [phi1_1]_cons = 1 |  |  |  |  |  |  |
| repair | Coefficient | Std. err. | z | $P>\|z\|$ | [95\% conf. | interval] |
| foreign | 5.947382 | 2.094126 | 2.84 | 0.005 | 1.84297 | 10.05179 |
| mpg | . 1911968 | . 08554 | 2.24 | 0.025 | . 0235414 | . 3588521 |
| price | -. 0000576 | . 0001357 | -0.42 | 0.671 | -. 0003236 | . 0002083 |
| gratio | -4.307571 | 1.884713 | -2.29 | 0.022 | -8.00154 | -. 6136017 |
| /phi1_1 | 1 | (constraine |  |  |  |  |
| /phi1_2 | 1.262268 | . 3530565 | 3.58 | 0.000 | . 5702904 | 1.954247 |
| /phi1_3 | 1.17593 | . 3169397 | 3.71 | 0.000 | . 5547394 | 1.79712 |
| /phi1_4 | . 8657195 | . 2411228 | 3.59 | 0.000 | . 3931275 | 1.338311 |
| /phi1_5 | 0 | (base outco |  |  |  |  |
| /theta1 | -6.864749 | 4.21252 | -1.63 | 0.103 | -15.12114 | 1.391639 |
| /theta2 | -7.613977 | 4.861803 | -1.57 | 0.117 | -17.14294 | 1.914981 |
| /theta3 | -5.80655 | 4.987508 | -1.16 | 0.244 | -15.58189 | 3.968786 |
| /theta4 | -3.85724 | 3.824132 | -1.01 | 0.313 | -11.3524 | 3.637922 |
| /theta5 | 0 | (base outco |  |  |  |  |

(repair=Excellent is the base outcome)
The coefficient associated with the first scale parameter, $\phi_{11}$, is 1 , and its standard error and other statistics are missing. This is the corner constraint applied to the one-dimensional model; in the header, this constraint is listed as [phi1_1]_cons $=1$. Also, the $\phi$ and $\theta$ parameters that are associated with the base outcome are identified. Keep in mind, though, that there are no coefficient estimates for [phi1_5]_cons or [theta5]_cons in the ereturn matrix e(b). The Wald statistic is for a test of the joint significance of the regression coefficients on foreign, mpg, price, and gratio.

The one-dimensional stereotype model restricts the multinomial logistic regression coefficients $\widetilde{\boldsymbol{\beta}}_{k}, k=1, \ldots, m-1$ to be parallel; that is, $\widetilde{\boldsymbol{\beta}}_{k}=-\phi_{k} \boldsymbol{\beta}$. As Lunt (2001) discusses, in the one-dimensional stereotype model, one linear combination $\mathbf{x}_{i} \boldsymbol{\beta}$ best discriminates the outcomes of the dependent variable, and the scale parameters $\phi_{k}$ measure the distance between the outcome levels and the linear predictor. If $\phi_{1} \geq \phi_{2} \geq \cdots \phi_{m-1} \geq \phi_{m} \equiv 0$, the model suggests that the subjective assessment of the dependent variable is indeed ordered. Here the maximum likelihood estimates of the $\phi$ 's are not monotonic, as would be assumed in an ordered logit model.

We test that $\phi_{1}=\phi_{2}$ by typing

```
. test [phi1_2]_cons = [phi1_1]_cons
( 1) - [phi1_1]_cons + [phi1_2]_cons = 0
    chi2( 1) = 0.55
    Prob > chi2 = 0.4576
```

Because the two parameters are not statistically different, we decide to add a constraint to force $\phi_{1}=\phi_{2}$ :

| Stereotype logistic regression |  |  |  |  | Number of obs Wald chi2(4) | $\begin{aligned} & =135 \\ & =\quad 21.28 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Log likelihood $=-159.65769$ |  |  |  |  | Prob > chi2 | $=0.0003$ |
| [phi1_1]_cons = 1 |  |  |  |  |  |  |
| ( 2) - [phi1_1]_cons + [phi1_2]_cons = 0 |  |  |  |  |  |  |
| repair | Coefficient | Std. err. | z | $P>\|z\|$ | [95\% conf. | interval] |
| foreign | 7.166515 | 1.690177 | 4.24 | 0.000 | 3.853829 | 10.4792 |
| mpg | . 2340043 | . 0807042 | 2.90 | 0.004 | . 0758271 | . 3921816 |
| price | -. 000041 | . 0001618 | -0.25 | 0.800 | -. 0003581 | . 000276 |
| gratio | -5.218107 | 1.798717 | -2.90 | 0.004 | -8.743528 | -1.692686 |
| /phi1_1 |  | (constrained) |  |  |  |  |
| /phi1_2 | 1 | (constrained) |  |  |  |  |
| /phi1_3 | . 9751096 | . 1286563 | 7.58 | 0.000 | . 7229478 | 1.227271 |
| /phi1_4 | . 7209343 | . 1220353 | 5.91 | 0.000 | . 4817494 | . 9601191 |
| /phi1_5 | 0 | (base outc |  |  |  |  |
| /theta1 | -8.293452 | 4.645182 | -1.79 | 0.074 | -17.39784 | . 8109368 |
| /theta2 | -6.958451 | 4.629292 | -1.50 | 0.133 | -16.0317 | 2.114795 |
| /theta3 | -5.620232 | 4.953981 | -1.13 | 0.257 | -15.32986 | 4.089392 |
| /theta4 | -3.745624 | 3.809189 | -0.98 | 0.325 | -11.2115 | 3.720249 |
| /theta5 | 0 | (base outc |  |  |  |  |

(repair=Excellent is the base outcome)
The $\phi$ estimates are now monotonically decreasing and the standard errors of the $\phi$ 's are small relative to the size of the estimates, so we conclude that, with the exception of outcomes Poor and Fair, the groups are distinguishable for the one-dimensional model and that the quality assessment can be ordered.

## Higher-dimension models

The stereotype logistic model is not limited to ordered categorical dependent variables; you can use it on nominal data to reduce the dimension of the regressions. Recall that a multinomial model fit to a categorical dependent variable with $m$ levels will have $m-1$ sets of regression coefficients. However, a model with fewer dimensions may fit the data equally well, suggesting that some of the categories are indistinguishable.

## - Example 2

As discussed in [R] mlogit, we have data on the type of health insurance available to 616 psychologically depressed subjects in the United States (Tarlov et al. 1989; Wells et al. 1989). Patients may have either an indemnity (fee-for-service) plan or a prepaid plan, such as an HMO, or may be uninsured. Demographic variables include age, gender, race, and site.

First, we fit the saturated, two-dimensional model that is equivalent to a multinomial logistic model. We choose the base outcome to be 1 (indemnity insurance) because that is the default for mlogit.

(insure=Indemnity is the base outcome)

For comparison, we also fit the model by using mlogit:

| . mlogit insure age male nonwhite i.site, nolog |  |
| :--- | :--- |
| Multinomial logistic regression | Number of obs $=$615 |
|  | LR chi2 (10) $=$ |
|  | Prob $>$ chi2 $=0.99$ |
| Log likelihood $=-534.36165$ | Pseudo R2 |


| insure | Coefficient | Std. err. | z | $P>\|z\|$ | [95\% con | interval] |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Indemnity | (base outcome) |  |  |  |  |  |
| Prepaid |  |  |  |  |  |  |
| age | -. 011745 | . 0061946 | -1.90 | 0.058 | -. 0238862 | . 0003962 |
| male | . 5616934 | . 2027465 | 2.77 | 0.006 | . 1643175 | . 9590693 |
| nonwhite | . 9747768 | . 2363213 | 4.12 | 0.000 | . 5115955 | 1.437958 |
| site |  |  |  |  |  |  |
| 2 | . 1130359 | . 2101903 | 0.54 | 0.591 | -. 2989296 | . 5250013 |
| 3 | -. 5879879 | . 2279351 | -2.58 | 0.010 | -1.034733 | -. 1412433 |
| _cons | . 2697127 | . 3284422 | 0.82 | 0.412 | -. 3740222 | . 9134476 |
| Uninsure |  |  |  |  |  |  |
| age | -. 0077961 | . 0114418 | -0.68 | 0.496 | -. 0302217 | . 0146294 |
| male | . 4518496 | . 3674867 | 1.23 | 0.219 | -. 268411 | 1.17211 |
| nonwhite | . 2170589 | . 4256361 | 0.51 | 0.610 | -. 6171725 | 1.05129 |
| site |  |  |  |  |  |  |
| 2 | -1.211563 | . 4705127 | -2.57 | 0.010 | -2.133751 | -. 2893747 |
| 3 | -. 2078123 | . 3662926 | -0.57 | 0.570 | -. 9257327 | . 510108 |
| _cons | -1.286943 | . 5923219 | -2.17 | 0.030 | -2.447872 | -. 1260134 |

Apart from having opposite signs, the coefficients from the stereotype logistic model are identical to those from the multinomial logit model. Recall the definition of $\eta_{k}$ given in the Remarks and examples, particularly the minus sign in front of the summation. One other difference in the output is that the constant estimates labeled /theta in the slogit output are the constants labeled _cons in the mlogit output.

Next, we examine the one-dimensional model.

| Stereotype logistic regression |  |  |  |  | Number of obs Wald chi2(5) | $\begin{aligned} & =615 \\ & =28.20 \end{aligned}$ |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| Log likelihood = -539.75205 $\text { Prob }>\text { chi2 }=0.0000$ <br> ( 1) [phi1_2]_cons = 1 |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
| insure | Coefficient | Std. err. | z | $\mathrm{P}>\|\mathrm{z}\|$ | [95\% conf. | interval] |
| age | . 0108366 | . 0061918 | 1.75 | 0.080 | -. 0012992 | . 0229723 |
| male | -. 5032537 | . 2078171 | -2.42 | 0.015 | -. 9105678 | -. 0959396 |
| nonwhite | -. 9480351 | . 2340604 | -4.05 | 0.000 | -1.406785 | -. 489285 |
| site |  |  |  |  |  |  |
| 2 | -. 2444316 | . 2246366 | -1.09 | 0.277 | -. 6847113 | . 1958481 |
| 3 | . 556665 | . 2243799 | 2.48 | 0.013 | . 1168886 | . 9964415 |
| /phi1_1 | 0 (base outcome) |  |  |  |  |  |
| /phi1_2 | 1 (constrained) |  |  | 0.925 | -. 7612535 | . 8379613 |
| /phi1_3 | . 0383539 | . 4079705 | 0.09 |  |  |  |
| /theta1 | 0 (base outcome) |  |  |  |  |  |
| /theta2 | . 187542 | . 3303847 | 0.57 | 0.570 | -. 4600001 | . 835084 |
| /theta3 | -1.860134 | . 2158898 | -8.62 | 0.000 | -2.28327 | -1.436997 |

(insure=Indemnity is the base outcome)
We have reduced a two-dimensional multinomial model to one dimension, reducing the number of estimated parameters by four and decreasing the model likelihood by $\approx 5.4$.
slogit does not report a model likelihood-ratio test. The test of $d=1$ (a one-dimensional model) versus $d=0$ (the null model) does not have an asymptotic $\chi^{2}$ distribution because the unconstrained $\phi$ parameters (/phi1_3 in this example) cannot be identified if $\boldsymbol{\beta}=\mathbf{0}$. More generally, this problem precludes testing any hierarchical model of dimension $d$ versus $d-1$. Of course, the likelihood-ratio test of a full-dimension model versus $d=0$ is valid because the full model is just multinomial logistic, and all the $\phi$ parameters are fixed at 0 or 1 .

## - Technical note

The stereotype model is a special case of the reduced-rank vector generalized linear model discussed by Yee and Hastie (2003). If we define $\eta_{i k}=\theta_{k}-\sum_{j=1}^{d} \phi_{j k} \mathbf{x}_{i} \boldsymbol{\beta}_{j}$, for $k=1, \ldots, m-1$, we can write the expression in matrix notation as

$$
\boldsymbol{\eta}_{i}=\boldsymbol{\theta}+\boldsymbol{\Phi}\left(\mathbf{x}_{i} \mathbf{B}\right)^{\prime}
$$

where $\boldsymbol{\Phi}$ is a $(m-1) \times d$ matrix containing the $\phi_{j k}$ parameters and $\mathbf{B}$ is a $p \times d$ matrix with columns containing the $\boldsymbol{\beta}_{j}$ parameters, $j=1, \ldots, d$. The factorization $\boldsymbol{\Phi} \mathbf{B}^{\prime}$ is not unique because $\mathbf{\Phi B}^{\prime}=\boldsymbol{\Phi} \mathbf{M M}^{-1} \mathbf{B}^{\prime}$ for any nonsingular $d \times d$ matrix $\mathbf{M}$. To avoid this identifiability problem, we choose $\mathbf{M}=\boldsymbol{\Phi}_{1}^{-1}$, where

$$
\boldsymbol{\Phi}=\binom{\boldsymbol{\Phi}_{1}}{\boldsymbol{\Phi}_{2}}
$$

and $\boldsymbol{\Phi}_{1}$ is $d \times d$ of rank $d$ so that

$$
\boldsymbol{\Phi M}=\binom{\mathbf{I}_{d}}{\boldsymbol{\Phi}_{2} \boldsymbol{\Phi}_{1}^{-1}}
$$

and $\mathbf{I}_{d}$ is a $d \times d$ identity matrix. Thus, the corner constraints used by slogit are $\phi_{j j} \equiv 1$ and $\phi_{j k} \equiv 0$ for $j \neq k$ and $k, j \leq d$.

## Stored results

slogit stores the following in e():
Scalars
e(N)
e(k)
e(k_indvars)
e(k_out)
$e\left(k \_e q\right)$
e(k_eq_model)
$e\left(d f \_m\right)$
e(df_0)
e(k_dim)
e(i_base)
e(11)
e(11_0)
e(N_clust)
e(chi2)
e (p)
e(ic)
e(rank)
e(rc)
e(converged)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(indvars)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(out\#)
e(chi2type)
e(vce)
e(vcetype)
e(opt)
e(which)
e(ml_method)
e(user)
e(technique)
e(properties)
e(predict)
e(marginsnotok)
$e$ (marginsdefault)
$e$ (footnote)
e(asbalanced)
e(asobserved)
Matrices
e(b)
e(outcomes)
e(Cns)
e(ilog)
e(gradient)
e(V)
e(V_modelbased)
number of observations
number of parameters
number of independent variables
number of outcomes
number of equations in $e(b)$
number of equations in overall model test
Wald test degrees of freedom
null model degrees of freedom
model dimension
base outcome index
$\log$ likelihood
null model log likelihood
number of clusters
$\chi^{2}$
$p$-value for model test
number of iterations
rank of e(V)
return code
1 if converged, 0 otherwise

## slogit

command as typed
name of dependent variable
independent variables
weight type
weight expression
title in estimation output
name of cluster variable
outcome labels, $\#=1, \ldots, \mathrm{e}$ (k_out)
Wald; type of model $\chi^{2}$ test
vcetype specified in vce()
title used to label Std. err.
type of optimization
max or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement predict
predictions disallowed by margins
default predict() specification for margins
program used to implement the footnote display
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
outcome values
constraints matrix
iteration $\log$ (up to 20 iterations)
gradient vector
variance-covariance matrix of the estimators
model-based variance

Functions

```
    e(sample) marks estimation sample
```

In addition to the above, the following is stored in r() :

```
Matrices
```

    \(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values,
    and confidence intervals
    Note that results stored in r() are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

## Methods and formulas

slogit obtains the maximum likelihood estimates for the stereotype logistic model by using ml; see $[\mathrm{R}] \mathrm{ml}$. Each set of regression estimates, one set of $\boldsymbol{\beta}_{j} \mathrm{~s}$ for each dimension, constitutes one ml model equation. The $d \times(m-1) \phi \mathrm{s}$ and the $(m-1) \theta \mathrm{s}$ are ml ancillary parameters.

Without loss of generality, let the base outcome level be the $m$ th level of the dependent variable. Define the row vector $\phi_{k}=\left(\phi_{1 k}, \ldots, \phi_{d k}\right)$ for $k=1, \ldots, m-1$, and define the $p \times d$ matrix $\mathbf{B}=\left(\boldsymbol{\beta}_{1}, \ldots, \boldsymbol{\beta}_{d}\right)$. For observation $i$, the $\log$ odds of outcome level $k$ relative to level $m, k=1$, $\ldots, m-1$ is the index

$$
\begin{aligned}
\ln \left\{\frac{\operatorname{Pr}\left(Y_{i}=k\right)}{\operatorname{Pr}\left(Y_{i}=m\right)}\right\}=\eta_{i k} & =\theta_{k}-\phi_{k}\left(\mathbf{x}_{i} \mathbf{B}\right)^{\prime} \\
& =\theta_{k}-\boldsymbol{\phi}_{k} \boldsymbol{\nu}_{i}^{\prime}
\end{aligned}
$$

The row vector $\nu_{i}$ can be interpreted as a latent variable reducing the $p$-dimensional vector of covariates to a more interpretable $d<p$ dimension.

The probability of the $i$ th observation having outcome level $k$ is then

$$
\operatorname{Pr}\left(Y_{i}=k\right)=p_{i k}= \begin{cases}\frac{e^{\eta_{i k}}}{1+\sum_{j=1}^{m-1} e^{\eta_{i j}}}, \text { if } & k<m \\ \frac{1}{1+\sum_{j=1}^{m-1} e^{\eta_{i j}}}, \text { if } & k=m\end{cases}
$$

from which the log-likelihood function is computed as

$$
\begin{equation*}
L=\sum_{i=1}^{n} w_{i} \sum_{k=1}^{m} I_{k}\left(y_{i}\right) \ln \left(p_{i k}\right) \tag{1}
\end{equation*}
$$

Here $w_{i}$ is the weight for observation $i$ and

$$
I_{k}\left(y_{i}\right)=\left\{\begin{array}{l}
1, \text { if observation } y_{i} \text { has outcome } k \\
0, \text { otherwise }
\end{array}\right.
$$

Numeric variables are normalized for numerical stability during optimization where a new doubleprecision variable $\widetilde{x}_{j}$ is created from variable $x_{j}, j=1, \ldots, p$, such that $\widetilde{x}_{j}=\left(x_{j}-\bar{x}_{j}\right) / s_{j}$. This feature is turned off if you specify nonormalize, or if you use the from() option for initial estimates. Normalization is not performed on byte variables, including the indicator variables generated by $[R]$ xi. The linear equality constraints for regression parameters, if specified, must be scaled also. Assume that a constraint is applied to the regression parameter associated with variable $j$ and dimension $i$, $\beta_{j i}$, and the corresponding element of the constraint matrix (see [ P$]$ makecns) is divided by $s_{j}$.

After convergence, the parameter estimates for variable $j$ and dimension $i-\widetilde{\beta}_{j i}$, say-are transformed back to their original scale, $\beta_{j i}=\widetilde{\beta}_{j i} / s_{j}$. For the intercepts, you compute

$$
\theta_{k}=\widetilde{\theta}_{k}+\sum_{i=1}^{d} \phi_{i k} \sum_{j=1}^{p} \frac{\widetilde{\beta}_{j i} \bar{x}_{j}}{s_{j}}
$$

Initial values are computed using estimates obtained using mlogit to fit a multinomial logistic model. Let the $p \times(m-1)$ matrix $\widetilde{\mathbf{B}}$ contain the multinomial logistic regression parameters less the $m-1$ intercepts. Each $\phi$ is initialized with constant values $\min (1 / 2,1 / d)$, the initialize (constant) option (the default), or, with uniform random numbers, the initialize (random) option. Constraints are then applied to the starting values so that the structure of the $(m-1) \times d$ matrix $\boldsymbol{\Phi}$ is

$$
\Phi=\left(\begin{array}{c}
\phi_{1} \\
\phi_{2} \\
\vdots \\
\phi_{m-1}
\end{array}\right)=\binom{\mathbf{I}_{d}}{\boldsymbol{\Phi}}
$$

where $\mathbf{I}_{d}$ is a $d \times d$ identity matrix. Assume that only the corner constraints are used, but any constraints you place on the scale parameters are also applied to the initial scale estimates, so the structure of $\Phi$ will change accordingly. The $\phi$ parameters are invariant to the scale of the covariates, so initial estimates in $[0,1]$ are reasonable. The constraints guarantee that the rank of $\boldsymbol{\Phi}$ is at least $d$, so the initial estimates for the stereotype regression parameters are obtained from $\mathbf{B}=\widetilde{\mathbf{B}} \Phi\left(\Phi^{\prime} \boldsymbol{\Phi}\right)^{-1}$.

One other approach for initial estimates is provided: initialize(svd). It starts with the mlogit estimates and computes $\widetilde{\mathbf{B}}^{\prime}=\mathbf{U D V}$, where $\mathbf{U}_{m-1 \times p}$ and $\mathbf{V}_{p \times p}$ are orthonormal matrices and $\mathbf{D}_{p \times p}$ is a diagonal matrix containing the singular values of $\widetilde{\mathbf{B}}$. The estimates for $\boldsymbol{\Phi}$ and $\mathbf{B}$ are the first $d$ columns of $\mathbf{U}$ and VD, respectively (Yee and Hastie 2003).

The score for regression coefficients is

$$
\mathbf{u}_{i}\left(\boldsymbol{\beta}_{j}\right)=\frac{\partial L_{i k}}{\partial \boldsymbol{\beta}_{j}}=\mathbf{x}_{i}\left(\sum_{l=1}^{m-1} \phi_{j l} p_{i l}-\phi_{j k}\right)
$$

the score for the scale parameters is

$$
u_{i}\left(\phi_{j l}\right)=\frac{\partial L_{i k}}{\partial \phi_{j l}}= \begin{cases}\mathbf{x}_{i} \boldsymbol{\beta}_{j}\left(p_{i k}-1\right), & \text { if } l=k \\ \mathbf{x}_{i} \boldsymbol{\beta}_{j} p_{i l}, & \text { if } l \neq k\end{cases}
$$

for $l=1, \ldots, m-1$; and the score for the intercepts is

$$
u_{i}\left(\theta_{l}\right)=\frac{\partial L_{i k}}{\partial \theta_{l}}=\left\{\begin{array}{cl}
1-p_{i k}, & \text { if } l=k \\
-p_{i l}, & \text { if } l \neq k
\end{array}\right.
$$

This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce (robust) and vce(cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.
slogit also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

## References

Anderson, J. A. 1984. Regression and ordered categorical variables (with discussion). Journal of the Royal Statistical Society, Series B 46: 1-30. https://doi.org/10.1111/j.2517-6161.1984.tb01270.x.

Lunt, M. 2001. sg163: Stereotype ordinal regression. Stata Technical Bulletin 61: 12-18. Reprinted in Stata Technical Bulletin Reprints, vol. 10, pp. 298-307. College Station, TX: Stata Press.
-_. 2005. Prediction of ordinal outcomes when the association between predictors and outcome differs between outcome levels. Statistics in Medicine 24: 1357-1369. https://doi.org/10.1002/sim.2009.
Tarlov, A. R., J. E. Ware, Jr., S. Greenfield, E. C. Nelson, E. Perrin, and M. Zubkoff. 1989. The medical outcomes study. An application of methods for monitoring the results of medical care. Journal of the American Medical Association 262: 925-930. https://doi.org/10.1001/jama.1989.03430070073033.
Wells, K. B., R. D. Hays, M. A. Burnam, W. H. Rogers, S. Greenfield, and J. E. Ware, Jr. 1989. Detection of depressive disorder for patients receiving prepaid or fee-for-service care. Results from the Medical Outcomes Survey. Journal of the American Medical Association 262: 3298-3302. https://doi.org/10.1001/jama.1989.03430230083030.

Yee, T. W., and T. J. Hastie. 2003. Reduced-rank vector generalized linear models. Statistical Modelling 3: 15-41. https://doi.org/10.1191/1471082X03st045oa.

## Also see

[R] slogit postestimation - Postestimation tools for slogit
[R] logistic - Logistic regression, reporting odds ratios
[R] mlogit - Multinomial (polytomous) logistic regression
$[\mathrm{R}]$ ologit - Ordered logistic regression
[R] oprobit - Ordered probit regression
$[R]$ roc - Receiver operating characteristic (ROC) analysis
[SVY] svy estimation - Estimation commands for survey data
[U] 20 Estimation and postestimation commands

## Title

slogit postestimation - Postestimation tools for slogit

| Postestimation commands | predict | margins |
| :--- | :--- | :--- |
| Remarks and examples | Methods and formulas | Also see |

## Postestimation commands

The following postestimation commands are available after slogit:

| Command | Description |
| :---: | :---: |
| contrast | contrasts and ANOVA-style joint tests of estimates |
| estat ic | Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) |
| estat summarize | summary statistics for the estimation sample |
| estat vce | variance-covariance matrix of the estimators (VCE) |
| estat (svy) | postestimation statistics for survey data |
| estimates | cataloging estimation results |
| etable | table of estimation results |
| * hausman | Hausman's specification test |
| lincom | point estimates, standard errors, testing, and inference for linear combinations of coefficients |
| * lrtest | likelihood-ratio test |
| margins | marginal means, predictive margins, marginal effects, and average marginal effects |
| marginsplot | graph the results from margins (profile plots, interaction plots, etc.) |
| nlcom | point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients |
| predict | probabilities, linear predictions and their SEs, etc. |
| predictnl | point estimates, standard errors, testing, and inference for generalized predictions |
| pwcompare | pairwise comparisons of estimates |
| suest | seemingly unrelated estimation |
| test | Wald tests of simple and composite linear hypotheses |
| testnl | Wald tests of nonlinear hypotheses |

[^71]
## predict

## Description for predict

predict creates a new variable containing predictions such as probabilities, indexes for the $k$ th outcome, and standard errors.

## Menu for predict

Statistics $>$ Postestimation

## Syntax for predict

predict [type] \{stub*|newvar|newvarlist $\}[$ if ] [in] [, statistic outcome(outcome)]
predict [type]stub* [if] [in], scores
statistic Description
Main
pr probability of one of or all the dependent variable outcomes; the default
$\mathrm{xb} \quad$ index for the $k$ th outcome
stdp standard error of the index for the $k$ th outcome
You specify one or $k$ new variables with pr, where $k$ is the number of outcomes. If you specify one new variable and you do not specify outcome(), then outcome(\#1) is assumed.
You specify one new variable with xb and stdp. If you do not specify outcome(), then outcome(\#1) is assumed.
These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

## Options for predict

$\qquad$ Main
pr , the default, computes the predicted probabilities for all outcomes or for a specific outcome. To compute probabilities for all outcomes, you specify $k$ new variables, where $k$ is the number of categories of the dependent variable. Alternatively, you can specify stub*; in which case, pr will store predicted probabilities in variables stub1, stub2, ..., stubk. To compute the probability for a specific outcome, you specify one new variable and, optionally, the outcome value in option outcome(); if you omit outcome(), the first outcome value, outcome(\#1), is assumed.

Say that you fit a model by typing estimation_cmd y x1 x2, and y takes on four values. Then, you could type predict p1 p2 p3 p4 to obtain all four predicted probabilities; alternatively, you could type predict $p *$ to generate the four predicted probabilities. To compute specific probabilities one at a time, you can type predict p1, outcome(\#1) (or simply predict p1), predict p2, outcome(\#2), and so on. See option outcome() for other ways to refer to outcome values.
xb calculates the index, $\theta_{k}-\sum_{j=1}^{d} \phi_{\mathrm{jk}} \mathbf{x}_{i} \boldsymbol{\beta}_{j}$, for outcome level $k \neq \mathrm{e}$ (i_base) and dimension $d=\mathrm{e}$ (k_dim). It returns a vector of zeros if $k=\mathrm{e}$ (i_base). A synonym for xb is index. If outcome() is not specified, outcome(\#1) is assumed.
stdp calculates the standard error of the index. A synonym for stdp is seindex. If outcome() is not specified, outcome(\#1) is assumed.
outcome(outcome) specifies for which outcome the predicted probabilities are to be calculated. outcome() should contain either one value of the dependent variable or one of \#1, \#2, ..., with \#1 meaning the first category of the dependent variable, \#2 meaning the second category, etc. outcome() is not allowed with scores.
scores calculates the equation-level score variables. For models with $d$ dimensions and $m$ levels, $d+(d+1)(m-1)$ new variables are created. Assume $j=1, \ldots, d$ and $k=1, \ldots, m$ in the following.
The first $d$ new variables will contain $\partial \ln L / \partial\left(\mathbf{x} \boldsymbol{\beta}_{j}\right)$.
The next $d(m-1)$ new variables will contain $\partial \ln L / \partial \phi_{j k}$.
The last $m-1$ new variables will contain $\partial \ln L / \partial \theta_{k}$.

## margins

## Description for margins

margins estimates margins of response for probabilities and indexes for the $k$ th outcome.

## Menu for margins

Statistics $>$ Postestimation

## Syntax for margins

```
margins [marginlist] [, options]
```


statistic Description
default probabilities for each outcome
$\mathrm{pr} \quad$ probability of one of or all the dependent variable outcomes
xb index for the $k$ th outcome
stdp not allowed with margins
pr and xb default to the first outcome.
Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

## Remarks and examples

Once you have fit a stereotype logistic model, you can obtain the predicted probabilities by using the predict command for both the estimation sample and other samples; see [U] 20 Estimation and postestimation commands and $[R]$ predict.
predict without arguments (or with the pr option) calculates the predicted probability of each outcome of the dependent variable. You must therefore give a new variable name for each of the outcomes. To compute the estimated probability of one outcome, you use the outcome (outcome) option where outcome is the level encoding the outcome. If the dependent variable's levels are labeled, the outcomes can also be identified by the label values (see [D] label).

The xb option in conjunction with outcome (outcome) specifies that the index be computed for the outcome encoded by level outcome. Its approximate standard error is computed if the stdp option is specified. Only one of the $\mathrm{pr}, \mathrm{xb}$, or stdp options can be specified with a call to predict.

## > Example 1

In example 2 of [R] slogit, we fit the one-dimensional stereotype model, where the depvar is insure with levels $k=1$ for outcome Indemnity, $k=2$ for Prepaid, and $k=3$ for Uninsure. The base outcome for the model is Indemnity, so for $k \neq 1$ the vector of indices for the $k$ th level is

$$
\boldsymbol{\eta}_{k}=\theta_{k}-\phi_{k}\left(\beta_{1} \text { age }+\beta_{2} \text { male }+\beta_{3} \text { nonwhite }+\beta_{4} 2 . \text { site }+\beta_{5} 3 . \text { site }\right)
$$

We estimate the group probabilities by calling predict after slogit.

```
. use https://www.stata-press.com/data/r18/sysdsn1
(Health insurance data)
. slogit insure age male nonwhite i.site, dim(1) base(1) nolog
    (output omitted)
. predict pIndemnity pPrepaid pUninsure, p
. list pIndemnity pPrepaid pUninsure insure in 1/10
```

| pIndem~y | pPrepaid | pUnins~e | insure |
| ---: | ---: | ---: | ---: |
| .5419344 | .3754875 | .0825782 | Indemnity |
| .4359638 | .496328 | .0677081 | Prepaid <br> .5111583 |
| .4105107 | .0783309 | Indemnity |  |
| .3941132 | .5442234 | .0616633 | Prepaid |
| .4655651 | .4625064 | .0719285 | . |
| .4401779 | .4915102 | .0683118 | Prepaid |
| .4632122 | .4651931 | .0715948 | Prepaid |
| .3772302 | .5635696 | .0592002 | . |
| .4867758 | .4383018 | .0749225 | Uninsure |
| .5823668 | .3295802 | .0880531 | Prepaid |

Observations 5 and 8 are not used to fit the model because insure is missing at these points, but predict estimates the probabilities for these observations because none of the independent variables is missing. You can use if e (sample) in the call to predict to use only those observations that are used to fit the model.

## Methods and formulas <br> predict

Let level $b$ be the base outcome that is used to fit the stereotype logistic regression model of dimension $d$. The index for observation $i$ and level $k \neq b$ is $\eta_{i k}=\theta_{k}-\sum_{j=1}^{d} \phi_{j k} \mathbf{x}_{i} \boldsymbol{\beta}_{j}$. This is the $\log$ odds of outcome encoded as level $k$ relative to that of $b$ so that we define $\eta_{i b} \equiv 0$. The outcome probabilities for this model are defined as $\operatorname{Pr}\left(Y_{i}=k\right)=e^{\eta_{i k}} / \sum_{j=1}^{m} e^{\eta_{i j}}$. Unlike in mlogit, ologit, and oprobit, the index is no longer a linear function of the parameters. The standard error of index $\eta_{i k}$ is thus computed using the delta method (see also [R] predictnl).

The equation-level score for regression coefficients is

$$
\frac{\partial \ln L_{i k}}{\partial \mathbf{x}_{i} \boldsymbol{\beta}_{j}}=\left(\sum_{l=1}^{m-1} \phi_{j l} p_{i l}-\phi_{j k}\right)
$$

the equation-level score for the scale parameters is

$$
\frac{\partial \ln L_{i k}}{\partial \phi_{j l}}= \begin{cases}\mathbf{x}_{i} \boldsymbol{\beta}_{j}\left(p_{i k}-1\right), & \text { if } l=k \\ \mathbf{x}_{i} \boldsymbol{\beta}_{j} p_{i l}, & \text { if } l \neq k\end{cases}
$$

for $l=1, \ldots, m-1$; and the equation-level score for the intercepts is

$$
\frac{\partial \ln L_{i k}}{\partial \theta_{l}}= \begin{cases}1-p_{i k}, & \text { if } l=k \\ -p_{i l}, & \text { if } l \neq k\end{cases}
$$

## Also see

[R] slogit - Stereotype logistic regression
[U] 20 Estimation and postestimation commands

## Title

smooth — Robust nonlinear smoother

| Description | Quick start | Menu | Syntax |
| :--- | :--- | :--- | :--- |
| Option | Remarks and examples | Methods and formulas | Acknowledgments |
| References | Also see |  |  |

## Description

smooth applies the specified resistant, nonlinear smoother to varname and stores the smoothed series in newvar.

## Quick start

Running median smoother of span 3 for v , placing smoothed values in the new variable newv smooth 3 v , generate (newv)

Same as above, but with a compound smoother of running medians, first of span 3, then span 5 smooth 35 v , generate (newv)

Same as above, but repeating the span-5 running median until convergence smooth 35 R v, generate (newv)

Same as above, splitting any repeated values after the span-3 running median smooth 3 S5R v, generate (newv)

Same as above, but apply the compound smoother to the resulting rough smooth 3 S 5 R ,twice v , generate(newv)

## Menu

Statistics $>$ Nonparametric analysis $>$ Robust nonlinear smoother

## Syntax

smooth smoother [, twice] varname [if] [in], generate(newvar) where smoother is specified as $\operatorname{Sm}[\operatorname{Sm}[\ldots]]$ and $S m$ is one of
$\{1|2| 3|4| 5|6| 7|8| 9\}[R]$
$3[R] S[S \mid R][S \mid R] \ldots$
E
H
Letters may be specified in lowercase if preferred. Examples of smoother [, twice] include

$$
\begin{array}{lllll}
\text { 3RSSH } & \text { 3RSSH,twice } & 4253 \mathrm{H} & \text { 4253H,twice } & \text { 43RSR2H,twice } \\
\text { 3rssh } & \text { 3rssh,twice } & 4253 \mathrm{~h} & 4253 \mathrm{~h}, \text { twice } & \text { 43rsr2h,twice }
\end{array}
$$

## Option

generate (newvar) is required; it specifies the name of the new variable that will contain the smoothed values.

## Remarks and examples

Smoothing is an exploratory data-analysis technique for making the general shape of a series apparent. In this approach (Tukey 1977), the observed data series is assumed to be the sum of an underlying process that evolves smoothly (the smooth) and of an unsystematic noise component (the rough); that is,

$$
\text { data }=\text { smooth }+ \text { rough }
$$

Smoothed values $z_{t}$ are obtained by taking medians (or some other location estimate) of each point in the original data $y_{t}$ and a few of the points around it. The number of points used is called the span of the smoother. Thus, a span- 3 smoother produces $z_{t}$ by taking the median of $y_{t-1}, y_{t}$, and $y_{t+1}$. smooth provides running median smoothers of spans 1 to 9 -indicated by the digit that specifies their span. Median smoothers are resistant to isolated outliers, so they provide robustness to spikes in the data. Because the median is also a nonlinear operator, such smoothers are known as robust (or resistant) nonlinear smoothers.
smooth also provides the Hanning linear, nonrobust smoother, indicated by the letter H. Hanning is a span-3 smoother with binomial weights. Repeated applications of $\mathrm{H}-\mathrm{HH}$, HHH , etc.- provide binomial smoothers of span 5, 7, etc. See Cox $(1997,2004)$ for a graphical application of this fact.

Because one smoother usually cannot adequately separate the smooth from the rough, compound smoothers - multiple smoothers applied in sequence-are used. The smoother 35 H , for instance, then smooths the data with a span- 3 median smoother, smooths the result with a span- 5 median smoother, and finally smooths that result with the Hanning smoother. smooth allows you to specify any number of smoothers in any sequence.

Three refinements can be combined with the running median and Hanning smoothers. First, the endpoints of a smooth can be given special treatment. This is specified by the E operator. Second, smoothing by 3 , the span- 3 running median, tends to produce flat-topped hills and valleys. The splitting operator, S, "splits" these repeated values, applies the endpoint operator to them, and then "rejoins" the series. Finally, it is sometimes useful to repeat an odd-span median smoother or the splitting operator until the smooth no longer changes. Following a digit or an S with an R specifies this type of repetition.

Even the best smoother may fail to separate the smooth from the rough adequately. To guard against losing any systematic components of the data series, after smoothing, the smoother can be reapplied to the resulting rough, and any recovered signal can be added back to the original smooth. The twice operator specifies this procedure. More generally, an arbitrary smoother can be applied to the rough (using a second smooth command), and the recovered signal can be added back to the smooth. This more general procedure is called reroughing (Tukey 1977).

The details of each of the smoothers and operators are explained in Methods and formulas below.

## Example 1

smooth is designed to recover the general features of a series that has been contaminated with noise. To demonstrate this, we construct a series, add noise to it, and then smooth the noisy version to recover an estimate of the original data. First, we construct and display the data:

```
. drop _all
. set obs 10
. set seed 123456789
. generate time = _n
. label variable time "Time"
. generate x = _n^3 - 10*_n^2 + 5*_n
. label variable x "Signal"
. generate z = x + 50*rnormal()
. label variable z "Observed series"
. scatter x z time, c(l .) m(i o) ytitle("")
```



Now we smooth the noisy series, $z$, assumed to be the only data we would observe:

- smooth 4253 eh, twice $z$, gen(sz)
. label variable sz "Smoothed series"
. scatter $x \quad z$ time, $c(l .1) m(i \quad o ~ i) ~ y t i t l e(" ")|\mid s c a t t e r ~ s z ~ t i m e, ~$ > c(l . l) m(i o i) ytitle("") clpattern(dash_dot)



## > Example 2

Salgado-Ugarte and Curts-García (1993) provide data on the frequencies of observed fish lengths. In this example, the series to be smoothed - the frequencies - is ordered by fish length rather than by time.

```
. use https://www.stata-press.com/data/r18/fishdata, clear
. smooth 4253eh,twice freq, gen(sfreq)
. scatter sfreq freq length, c(l .) m(i o)
> title("Smoothed frequencies of fish lengths") ytitle("") xlabel(#4)
```



- Technical note
smooth allows missing values at the beginning and end of the series, but missing values in the middle are not allowed. Leading and trailing missing values are ignored. If you wish to ignore missing values in the middle of the series, you must drop the missing observations before using smooth. Doing so, of course, would violate smooth's assumption that observations are equally spaced-each observation represents a year, a quarter, or a month (or a 1-year birth-rate category). In practice, smooth produces good results as long as the spaces between adjacent observations do not vary too much.

Smoothing is usually applied to time series, but any variable with a natural order can be smoothed. For example, a smoother might be applied to the birth rate recorded by the age of the mothers (birthrate for 17 -year-olds, birthrate for 18 -year-olds, and so on).

## Methods and formulas

Methods and formulas are presented under the following headings:
Running median smoothers of odd span
Running median smoothers of even span
Repeat operator
Endpoint rule
Splitting operator
Hanning smoother
Twicing

## Running median smoothers of odd span

The smoother 3 defines

$$
z_{t}=\operatorname{median}\left(y_{t-1}, y_{t}, y_{t+1}\right)
$$

The smoother 5 defines

$$
z_{t}=\operatorname{median}\left(y_{t-2}, y_{t-1}, y_{t}, y_{t+1}, y_{t+2}\right)
$$

and so on. The smoother 1 defines $z_{t}=\operatorname{median}\left(y_{t}\right)$, so it does nothing.
Endpoints are handled by using smoothers of shorter, odd span. Thus, for 3,

$$
\begin{aligned}
z_{1} & =y_{1} \\
z_{2} & =\operatorname{median}\left(y_{1}, y_{2}, y_{3}\right) \\
\vdots & \\
z_{N-1} & =\operatorname{median}\left(y_{N-2}, y_{N-1}, y_{N}\right) \\
Z_{N} & =y_{N}
\end{aligned}
$$

For 5,

$$
\begin{aligned}
z_{1} & =y_{1} \\
z_{2} & =\operatorname{median}\left(y_{1}, y_{2}, y_{3}\right) \\
z_{3} & =\operatorname{median}\left(y_{1}, y_{2}, y_{3}, y_{4}, y_{5}\right) \\
z_{4} & =\operatorname{median}\left(y_{2}, y_{3}, y_{4}, y_{5}, y_{6}\right) \\
\vdots & \\
z_{N-2} & =\operatorname{median}\left(y_{N-4}, y_{N-3}, y_{N-2}, y_{N-1}, y_{N}\right) \\
z_{N-1} & =\operatorname{median}\left(y_{N-2}, y_{N-1}, y_{N}\right) \\
Z_{N} & =y_{N}
\end{aligned}
$$

and so on.

## Running median smoothers of even span

Define the median() function as returning the linearly interpolated value when given an even number of arguments. Thus, the smoother 2 defines

$$
z_{t+0.5}=\left(y_{t}+y_{t+1}\right) / 2
$$

The smoother 4 defines $z_{t+0.5}$ as the linearly interpolated median of $\left(y_{t-1}, y_{t}, y_{t+1}, y_{t+2}\right)$, and so on. Endpoints are always handled using smoothers of shorter, even span. Thus, for 4,

$$
\begin{aligned}
z_{0.5} & =y_{1} \\
z_{1.5} & =\operatorname{median}\left(y_{1}, y_{2}\right)=\left(y_{1}+y_{2}\right) / 2 \\
z_{2.5} & =\operatorname{median}\left(y_{1}, y_{2}, y_{3}, y_{4}\right) \\
\vdots & \\
z_{N-2.5} & =\operatorname{median}\left(y_{N-4}, y_{N-3}, y_{N-2}, y_{N}\right) \\
z_{N-1.5} & =\operatorname{median}\left(y_{N-2}, y_{N-1}\right) \\
z_{N-0.5} & =\operatorname{median}\left(y_{N-1}, y_{N}\right) \\
z_{N+0.5} & =y_{N}
\end{aligned}
$$

As defined above, an even-span smoother increases the length of the series by 1 observation. However, the series can be recentered on the original observation numbers, and the "extra" observation can be eliminated by smoothing the series again with another even-span smoother. For instance, the smooth of 4 illustrated above could be followed by a smooth of 2 to obtain

$$
\begin{aligned}
z_{1}^{*} & =\left(z_{0.5}+z_{1.5}\right) / 2 \\
z_{2}^{*} & =\left(z_{1.5}+z_{2.5}\right) / 2 \\
z_{3}^{*} & =\left(z_{2.5}+z_{3.5}\right) / 2 \\
\vdots & \\
z_{N-2}^{*} & =\left(z_{N-2.5}+z_{N-1.5}\right) / 2 \\
z_{N-1}^{*} & =\left(z_{N-1.5}+z_{N-0.5}\right) / 2 \\
z_{N}^{*} & =\left(z_{N-0.5}+z_{N+0.5}\right) / 2
\end{aligned}
$$

smooth keeps track of the number of even smoothers applied to the data and expands and shrinks the length of the series accordingly. To ensure that the final smooth has the same number of observations as varname, smooth requires you to specify an even number of even-span smoothers. However, the pairs of even-span smoothers need not be contiguous; for instance, 4253 and 4523 are both allowed.

## Repeat operator

$R$ indicates that a smoother is to be repeated until convergence, that is, until repeated applications of the smoother produce the same series. Thus, 3 applies the smoother of running medians of span 3. 33 applies the smoother twice. $3 R$ produces the result of repeating 3 an infinite number of times. $R$ should be used only with odd-span smoothers because even-span smoothers are not guaranteed to converge.

The smoother 453R2 applies a span-4 smoother, followed by a span-5 smoother, followed by repeated applications of a span-3 smoother, followed by a span-2 smoother.

## Endpoint rule

The endpoint rule E modifies the values $z_{1}$ and $z_{N}$ according to the following formulas:

$$
\begin{aligned}
z_{1} & =\operatorname{median}\left(3 z_{2}-2 z_{3}, z_{1}, z_{2}\right) \\
z_{N} & =\operatorname{median}\left(3 z_{N-2}-2 z_{N-1}, z_{N}, z_{N-1}\right)
\end{aligned}
$$

When the endpoint rule is not applied, endpoints are typically "copied in"; that is, $z_{1}=y_{1}$ and $z_{N}=y_{N}$.

## Splitting operator

The smoothers 3 and 3R can produce flat-topped hills and valleys. The split operator attempts to eliminate such hills and valleys by splitting the sequence, applying the endpoint rule E, rejoining the series, and then resmoothing by $3 R$.

The $S$ operator may be applied only after $3,3 R$, or $S$.
We recommend that the $S$ operator be repeated once (SS) or until no further changes take place (SR).

## Hanning smoother

H is the Hanning linear smoother:

$$
z_{t}=\left(y_{t-1}+2 y_{t}+y_{t+1}\right) / 4
$$

Endpoints are copied in: $z_{1}=y_{1}$ and $z_{N}=y_{N} . \mathrm{H}$ should be applied only after all nonlinear smoothers.

## Twicing

A smoother divides the data into a smooth and a rough:

$$
\text { data }=\text { smooth }+ \text { rough }
$$

If the smoothing is successful, the rough should exhibit no pattern. Twicing refers to applying the smoother to the observed, calculating the rough, and then applying the smoother to the rough. The resulting "smoothed rough" is then added back to the smooth from the first step.

## Acknowledgments

smooth was originally written by William Gould (1992) —at which time it was named nlsm—and was inspired by Salgado-Ugarte and Curts-García (1992). Salgado-Ugarte and Curts-García (1993) subsequently reported anomalies in nlsm's treatment of even-span median smoothers. smooth corrects these problems and incorporates other improvements but otherwise is essentially the same as originally published.

## References

Cox, N. J. 1997. gr22: Binomial smoothing plot. Stata Technical Bulletin 35: 7-9. Reprinted in Stata Technical Bulletin Reprints, vol. 6, pp. 36-38. College Station, TX: Stata Press.
__. 2004. gr22_1: Software update: Binomial smoothing plot. Stata Journal 4: 490.
-_. 2005. Speaking Stata: Smoothing in various directions. Stata Journal 5: 574-593.
Gould, W. W. 1992. sg11.1: Quantile regression with bootstrapped standard errors. Stata Technical Bulletin 9: 19-21. Reprinted in Stata Technical Bulletin Reprints, vol. 2, pp. 137-139. College Station, TX: Stata Press.
Royston, P., and N. J. Cox. 2005. A multivariable scatterplot smoother. Stata Journal 5: 405-412.
Salgado-Ugarte, I. H., and J. Curts-García. 1992. sed7: Resistant smoothing using Stata. Stata Technical Bulletin 7: 8-11. Reprinted in Stata Technical Bulletin Reprints, vol. 2, pp. 99-103. College Station, TX: Stata Press.
——. 1993. sed7.2: Twice reroughing procedure for resistant nonlinear smoothing. Stata Technical Bulletin 11: 14-16. Reprinted in Stata Technical Bulletin Reprints, vol. 2, pp. 108-111. College Station, TX: Stata Press.

Tukey, J. W. 1977. Exploratory Data Analysis. Reading, MA: Addison-Wesley.
Velleman, P. F. 1977. Robust nonlinear data smoothers: Definitions and recommendations. Proceedings of the National Academy of Sciences 74: 434-436. https://doi.org/10.1073/pnas.74.2.434.
——. 1980. Definition and comparison of robust nonlinear data smoothing algorithms. Journal of the American Statistical Association 75: 609-615. https://doi.org/10.2307/2287657.
Velleman, P. F., and D. C. Hoaglin. 1981. Applications, Basics, and Computing of Exploratory Data Analysis. Boston: Duxbury.

## Also see

[R] lowess - Lowess smoothing
[R] lpoly - Kernel-weighted local polynomial smoothing
[TS] tssmooth — Smooth and forecast univariate time-series data

## Title

spearman - Spearman's and Kendall's correlations

Description
Options for spearman
Methods and formulas

Quick start<br>Options for ktau<br>Acknowledgment

Menu
Remarks and examples
References

Syntax
Stored results
Also see

## Description

spearman displays Spearman's rank correlation coefficients for all pairs of variables in varlist or, if varlist is not specified, for all the variables in the dataset. When there are two variables, an exact $p$-value can be calculated optionally using permutations.
ktau displays Kendall's rank correlation coefficients between the variables in varlist or, if varlist is not specified, for all the variables in the dataset. ktau is intended for use on small- and moderate-sized datasets; it requires considerable computation time for larger datasets.

## Quick start

Spearman's rank correlation coefficient with approximate $p$-value for v 1 and v 2 spearman v1 v2

As above, but report an exact $p$-value calculated using Monte Carlo permutations spearman v 1 v 2 , exact

As above, but perform 100,000 Monte Carlo permutations rather than the default of 10,000 , and set the random-number seed for reproducibility spearman v1 v2, exact(montecarlo, reps(100000) rseed(1234))

Display Spearman's rank correlation coefficients in a matrix for all pairs of v 1 , v 2 , and v 3 spearman v1 v2 v3

Display $p$-values as well as correlation coefficients
spearman v1 v2 v3, stats (rho p)
Same as above, but perform Bonferroni's adjustment to $p$-values
spearman v1 v2 v3, stats (rho p) bonferroni
Kendall's rank correlation coefficients, scores, and standard errors of the scores for pairs of v1, v2, and v3
ktau v1 v2 v3, stats (taua taub score se)
Same as above, but use pairwise instead of casewise deletion
ktau v1 v2 v3, stats (taua taub score se) pw

## Menu

## spearman

Statistics $>$ Nonparametric analysis $>$ Tests of hypotheses $>$ Spearman's rank correlation

## ktau

Statistics $>$ Nonparametric analysis $>$ Tests of hypotheses $>$ Kendall's rank correlation

## Syntax

Spearman's rank correlation coefficients
spearman [varlist] [if][in] [, spearman_options]

Kendall's rank correlation coefficients
ktau [varlist] [if] [in] [, ktau_options]
spearman_options Description
Main
stats (spearman_list) list of statistics; select up to three statistics; default is stats (rho)
print(\#)
$p$-value cutoff for displaying coefficients
star(\#) $\quad p$-value cutoff for displaying a star
bonferroni report Bonferroni-adjusted $p$-values
sidak report Šidák-adjusted $p$-values
pw calculate each pairwise correlation coefficient using all available data
matrix display output in matrix form
exact [(exact_specs)] report an exact $p$-value (available only when varlist is two variables)

## ktau_options <br> Description

Main
stats (ktau_list) list of statistics; select up to six statistics; default is stats (taua)
print(\#)
star(\#) $\quad p$-value cutoff for displaying a star
bonferroni
sidak
pw
matrix
$p$-value cutoff for displaying coefficients
report Bonferroni-adjusted $p$-values
report Šidák-adjusted $p$-values
calculate each pairwise correlation coefficient using all available data display output in matrix form
by and collect are allowed with spearman and ktau; see [U] 11.1.10 Prefix commands.
where the elements of spearman_list may be
rho correlation coefficient
obs number of observations
p $\quad p$-value
and the elements of ktau_list may be

$$
\begin{array}{ll}
\text { taua } & \text { correlation coefficient } \tau_{a} \\
\text { taub } & \text { correlation coefficient } \tau_{b} \\
\text { score } & \text { score } \\
\text { se } & \text { standard error of score } \\
\text { obs } & \text { number of observations } \\
\text { p } & p \text {-value }
\end{array}
$$

## Options for spearman

## Main

stats (spearman_list) specifies the statistics to be displayed in the matrix of output. stats (rho) is the default. Up to three statistics may be specified; stats (rho obs p) would display the correlation coefficient, number of observations, and $p$-value. If varlist contains only two variables, all statistics are shown in tabular form, and stats(), print(), and star() have no effect unless the matrix option is specified.
print (\#) specifies the $p$-value cutoff for correlation coefficients to be printed. Correlation coefficients with larger $p$-values are left blank in the matrix. Typing spearman, print (.10) would display only those correlation coefficients that have $p$-values less than or equal to 0.10 .
star(\#) specifies the $p$-value cutoff for correlation coefficients to be marked with a star. Typing spearman, star(.05) would star all correlation coefficients that have $p$-values less than or equal to 0.05 .
bonferroni makes the Bonferroni adjustment to $p$-values. This adjustment affects displayed $p$-values and the print() and star() options. Thus, spearman, print(.05) bonferroni prints coefficients with Bonferroni-adjusted $p$-values of 0.05 or less.
sidak makes the Šidák adjustment to $p$-values. This adjustment affects displayed $p$-values and the print() and star() options. Thus, spearman, print(.05) sidak prints coefficients with Sidák-adjusted $p$-values of 0.05 or less.
pw specifies that correlations be calculated using pairwise deletion of observations with missing values. By default, spearman uses casewise deletion, where observations are ignored if any of the variables in varlist are missing.
matrix forces spearman to display the statistics as a matrix, even if varlist contains only two variables. matrix is implied if more than two variables are specified.
exact and exact (exact_specs) specify that an exact $p$-value be reported. This option is available only when varlist contains only two variables.
exact specifies that an exact $p$-value from a Monte Carlo permutation test be reported. exact is a synonym for exact (montecarlo).
exact (montecarlo [, options] | enumerate [, options]) specifies that an exact $p$-value be reported in addition to the approximate $p$-value. Specifying exact (montecarlo) does a Monte Carlo permutation test. Specifying exact (enumerate) does an enumeration of all possible permutations. Because the number of all possible permutations is typically extremely large, enumeration is feasible only for very small datasets. The number of permutations will be displayed, and you can click on Break to stop the computation. The exact $p$-value is computed by permute.
exact(montecarlo[, options]) allows options show, reps(\#), rseed(\#), saving (filename [, sav_options]), level(\#), dots(\#), nodots, and eps (\#). The show option specifies that the table produced by permute also be displayed. By default, 10,000 Monte Carlo permutations are done. That is, the default is the same as specifying exact (montecarlo, reps(10000)). The default for dots() is dots(100) when reps() is $\geq 10,000$; otherwise, it is dots (1). See Options in [R] permute.
exact (enumerate [, options]) allows options show, saving (filename [, sav_options]), dots(\#), nodots, and eps(\#). The show option specifies that the table produced by permute also be displayed. The default for dots() is dots(100). See Options in [R] permute.

## Options for ktau

$\qquad$ Main
stats (ktau_list) specifies the statistics to be displayed in the matrix of output. stats(taua) is the default. Up to six statistics may be specified; stats (taua taub score se obs p) would display the correlation coefficients $\tau_{a}$ and $\tau_{b}$, score, standard error of score, number of observations, and $p$-value. If varlist contains only two variables, all statistics are shown in tabular form and stats(), print(), and star() have no effect unless the matrix option is specified.
print (\#) specifies the $p$-value cutoff for correlation coefficients to be printed. Correlation coefficients with larger $p$-values are left blank in the matrix. Typing ktau, print (.10) would display only those correlation coefficients that have $p$-values less than or equal to 0.10 .
star(\#) specifies the $p$-value cutoff for correlation coefficients to be marked with a star. Typing ktau, star(.05) would star all correlation coefficients that have $p$-values less than or equal to 0.05.
bonferroni makes the Bonferroni adjustment to $p$-values. This adjustment affects displayed $p$-values and the print() and star() options. Thus, ktau, print(.05) bonferroni prints coefficients with Bonferroni-adjusted $p$-values of 0.05 or less.
sidak makes the Šidák adjustment to $p$-values. This adjustment affects displayed $p$-values and the print() and star() options. Thus, ktau, print(.05) sidak prints coefficients with Šidákadjusted $p$-values of 0.05 or less.
pw specifies that correlations be calculated using pairwise deletion of observations with missing values. By default, ktau uses casewise deletion, where observations are ignored if any of the variables in varlist are missing.
matrix forces ktau to display the statistics as a matrix, even if varlist contains only two variables. matrix is implied if more than two variables are specified.

## Remarks and examples

## > Example 1

We wish to calculate the correlation coefficients among marriage rate (mrgrate), divorce rate (divorce_rate), and median age (medage) in state data. We can calculate the standard Pearson correlation coefficients and $p$-values by typing

| . pwcorr mrgrate divorce_rate medage, sig |  |  |  |
| :---: | :---: | :---: | :---: |
|  | mrgrate | divorc~e | medage |
| mrgrate | 1.0000 |  |  |
| divorce_rate | 0.7895 | 1.0000 |  |
|  | 0.0000 |  |  |
| medage | 0.0011 | -0.1526 | 1.0000 |
|  | 0.9941 | 0.2900 |  |

We can calculate Spearman's rank correlation coefficients by typing

```
. spearman mrgrate divorce_rate medage, stats(rho p)
Number of observations = 50
```

| Key |
| :--- |
| rho <br> $p$-value |


|  | mrgrate divorc~e | medage |  |
| ---: | ---: | ---: | ---: |
| mrgrate | 1.0000 |  |  |
|  | . |  |  |
| divorce_rate | 0.6933 | 1.0000 |  |
|  | 0.0000 | . |  |
| medage | -0.4869 | -0.2455 | 1.0000 |
|  | 0.0004 | 0.0857 | . |

The large difference in the results is caused by one observation. Nevada's marriage rate is almost 10 times higher than the state with the next-highest marriage rate. An important feature of the Spearman rank correlation is its reduced sensitivity to extreme values compared with the Pearson correlation.

We can calculate Kendall's rank correlations by typing

```
. ktau mrgrate divorce_rate medage, stats(taua taub p)
Number of observations = 50
```

| Key |
| :--- |
| tau_a |
| tau_b |
| $p$-value |


|  | mrgrate | divorc~e | medage |
| ---: | ---: | ---: | ---: |
| mrgrate | 0.9829 |  |  |
|  | 1.0000 |  |  |
| divorce_rate | 0.5110 | 0.9804 |  |
|  | 0.5206 | 1.0000 |  |
|  | 0.0000 | . |  |
|  |  |  |  |
|  | -0.3486 | -0.1698 | 0.9845 |
|  | -0.3544 | -0.1728 | 1.0000 |
|  | 0.0004 | 0.0828 |  |

There are tied values for variables mrgrate, divorce_rate, and medage, so average ranks are used for the tied values. As a result, $\tau_{a}<1$ on the diagonal (see Methods and formulas for the definition of $\tau_{a}$ ).

According to Conover (1999, 323), "Spearman's $\rho$ tends to be larger than Kendall's $\tau$ in absolute value. However, as a test of significance, there is no strong reason to prefer one over the other because both will produce nearly identical results in most cases."

Newson (2000a, 2000b, 2000c, 2001, 2003, 2005, 2006) introduces confidence intervals for Kendall's $\tau_{a}$. The community-contributed somersd command provides these confidence intervals along with additional rank statistics such as Somers' $D$ and Harrell's $C$ and their corresponding confidence intervals.

See Seed (2001) for confidence intervals for Spearman's rank correlation.

## > Example 2

We illustrate spearman and ktau with the auto data, which contains some missing values.

```
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. spearman mpg rep78
Number of observations = 69
    Spearman's rho = 0.3098
Test of H0: mpg and rep78 are independent
    Prob = 0.0098
```

Because we specified two variables, spearman displayed the sample size, correlation, and $p$-value in tabular form. To obtain just the correlation coefficient displayed in matrix form, we type

```
. spearman mpg rep78, stats(rho) matrix
Number of observations = 69
\begin{tabular}{r|rr} 
& mpg & rep78 \\
\hline mpg & 1.0000 & \\
rep78 & 0.3098 & 1.0000
\end{tabular}
```

We can specify the pw option with spearman and ktau so that all nonmissing observations between a pair of variables when calculating their correlation coefficient are used. In the output below, some correlations are based on 74 observations, whereas others are based on 69 because 5 observations contain a missing value for rep78.

```
. spearman mpg price rep78, pw stats(rho obs p) star(0.01)
Number of observations:
```

```
                            min = 69
```

                            min = 69
                            avg = 71
                            avg = 71
    max = 74
max = 74

| Key |
| :--- |
| rho <br> Number of obs <br> $p$-value |

```
\begin{tabular}{r|rrr} 
& mpg & price & rep78 \\
\hline mpg & 1.0000 & & \\
& 74 & & \\
& \(\cdot\) & & \\
price & \(-0.5419 *\) & 1.0000 & \\
& 74 & 74 & \\
& 0.0000 & \(\cdot\) & \\
& \(0.3098 *\) & 0.1028 & 1.0000 \\
& 69 & 69 & 69 \\
& 0.0098 & 0.4000 &.
\end{tabular}

The bonferroni and sidak options provide adjusted \(p\)-values:
. ktau mpg price rep78, stats(taua taub score se p) bonferroni
Number of observations \(=69\)
\begin{tabular}{|l|}
\hline Key \\
\hline tau_a \\
tau_b \\
score \\
se of score \\
\(p\)-value \\
\hline
\end{tabular}
\begin{tabular}{r|rrr} 
& mpg & price & rep78 \\
\hline \multirow{3}{*}{mpg} & 0.9471 & & \\
& 1.0000 & & \\
& 2222.0000 & & \\
& 191.8600 & & \\
&. & & \\
& -0.3973 & 1.0000 & \\
& -0.4082 & 1.0000 & \\
& -932.0000 & 2346.0000 & \\
& 192.4561 & 193.0682 & \\
& 0.0000 & &. \\
& 0.2076 & 0.0648 & 0.7136 \\
& 0.2525 & 0.0767 & 1.0000 \\
& 487.0000 & 152.0000 & 1674.0000 \\
& 181.7024 & 182.2233 & 172.2161 \\
& 0.0224 & 1.0000 &
\end{tabular}

\section*{> Example 3}

We continue with the auto data and show an example of spearman with the exact option.
```

. set seed 1234
. spearman mpg gear_ratio if foreign == 1, exact
Permutations (10,000): .........1,000.........2,000.........3,000.........4,000.
> ........5,000.........6,000. . . . . . .7,000. . . . . . . . 8,000 . . . . . . . 9,000 . . . . . . . .
> 10,000 done
Number of observations = 22
Spearman's rho = 0.4881
Test of HO: mpg and gear_ratio are independent
Prob = 0.0222
Exact prob = 0.0214 (10,000 Monte Carlo permutations)

```

By default, exact does a Monte Carlo permutation test with 10,000 permutations. Because it is a Monte Carlo test, we set the random-number generator seed before running spearman for reproducibility.

The exact \(p\)-value from the Monte Carlo permutation test is 0.0214 , which is close to the approximate \(p\)-value of 0.0222 . Note that the approximate \(p\)-value is not based on the normal distribution or the \(t\) distribution. It is calculated using a beta distribution fit to the first four moments of the null distribution of Spearman's rank correlation, and these four moments are calculated exactly for any value of \(N\), the number of observations. See Methods and formulas below.

Let's increase the number of Monte Carlo permutations to \(1,000,000\) to see how close the approximate \(p\)-value is to the exact \(p\)-value. We specify dots (10000) to see a dot every 10,000 th permutation, rather than the default of every 1,000 th permutation. This time we set the random-number generator seed using a suboption. The exact \(p\)-value is computed by permute, and we can see permute's output by specifying the show suboption.
```

. spearman mpg gear_ratio if foreign == 1,
> exact(montecarlo, reps(1000000) dots(10000) rseed(1234) show)
Permutations (1,000,000): .........100,000......... 200,000......... 300,000......
> . . 400,000 . . . . . . . 500,000 . . . . . . . . 600,000 . . . . . . . . 700,000 . . . . . . . . 800,000 . . . .
> ....900,000........1,000,000 done
Monte Carlo permutation results Number of observations = 22
Permutation variable: mpg Number of permutations = 1,000,000

```
\begin{tabular}{|c|c|c|c|c|c|c|c|c|}
\hline \multirow[b]{2}{*}{T} & \multirow[b]{2}{*}{T(obs)} & \multirow[b]{2}{*}{Test} & \multirow[b]{2}{*}{c} & \multirow[b]{2}{*}{n} & \multirow[b]{2}{*}{p} & \multicolumn{3}{|l|}{Monte Carlo error} \\
\hline & & & & & & SE(p) & [95\% & \(\mathrm{CI}(\mathrm{p})]\) \\
\hline rho & . 4880642 & lower & 988530 & 1000000 & . 9885 & . 0001 & . 9883 & . 9887 \\
\hline & & upper & 11510 & 1000000 & . 0115 & . 0001 & . 0113 & . 0117 \\
\hline & & -sided & & & . 0230 & . 0001 & . 0227 & . 0233 \\
\hline
\end{tabular}

Notes: For lower one-sided test, \(c=\#\{T<=T(o b s)\}\) and \(p=p \_l o w e r=c / n\).
    For upper one-sided test, \(c=\#\{T>=T(o b s)\}\) and \(p=p \_u p p e r=c / n\).

Number of observations \(=22\)
    Spearman's rho \(=0.4881\)
Test of HO: mpg and gear_ratio are independent
                        Prob \(=0.0222\)
            Exact prob \(=0.0230\) (1,000,000 Monte Carlo permutations)

With this increase in the number of permutations, the exact \(p\)-value is now calculated as 0.0230 . From the table produced by permute, we see that this \(p\)-value has a confidence interval of [ \(0.0227,0.0233\) ], accounting for the Monte Carlo error. The approximate \(p\)-value of 0.0222 falls outside this confidence interval, but it is still very close. Not bad for \(N=22\).

\section*{> Example 4}

For very small sample sizes, an exact \(p\)-value can be computed by enumerating the full permutation distribution. Here is an example using the auto data with sample size \(N=11\).
```

. spearman mpg gear_ratio if foreign == 1 \& mpg <= 24,
> exact(enumerate, dots(10000))
(enumerating all 831,600 possible permutations)
Permutations (831,600): ..........100,000.......... 200,000..........300,000
> .400,000 ........500,000.........600,000..........700,000..........800,000.... d
> one
Number of observations = 11
Spearman's rho = 0.5636
Test of HO: mpg and gear_ratio are independent
Prob = 0.0722
Exact prob = 0.0747 (enumerated all 831,600 permutations)

```

The exact \(p\)-value is 0.0747 . The approximate \(p\)-value is 0.0722 , which is quite close to the exact \(p\)-value in this case, even with only 11 observations.

For \(N=11\), the permutation distribution consists of \(11!=39,916,800\) permutations. However, the output above says that there are only 831,600 permutations. This is because the values of mpg in this sample are not all unique.
\begin{tabular}{l}
. tabulate mpg if foreign \(==1 \& \mathrm{mpg}<=24\) \\
\begin{tabular}{r} 
Mileage \\
(mpg)
\end{tabular} \\
\hline 14 \\
17
\end{tabular}

The multiplicities in the values of mpg yield permutations that give identical results, and we need to enumerate only the permutations that are distinct. From the values of mpg, we see that each distinct permutation has a multiplicity of \(2!2!2!3!=48\), and \(39,916,800 / 48=831,600\), which reduces considerably the number of permutations that need to be computed.
permute, which computes the permutations, permutes the first variable of the two in the spearman varlist command. So to best exploit this, the variable that produces the most multiplicities in the permutations should be placed first in varlist when doing an enumeration.

Charles Edward Spearman (1863-1945) was a British psychologist who made contributions to correlation, factor analysis, test reliability, and psychometrics. After several years' military service, he obtained a PhD in experimental psychology at Leipzig and became a professor at University College London, where he sustained a long program of work on the interpretation of intelligence tests. Ironically, the rank correlation version bearing his name is not the formula he advocated.

Maurice George Kendall (1907-1983) was a British statistician who contributed to rank correlation, time series, multivariate analysis, among other topics, and wrote many statistical texts. Most notably, perhaps, his advanced survey of the theory of statistics went through several editions, later ones with Alan Stuart; the baton has since passed to others. Kendall was employed in turn as a government and business statistician, as a professor at the London School of Economics, as a consultant, and as director of the World Fertility Survey. He was knighted in 1974.

\section*{Stored results}
spearman stores the following in r() :
Scalars
\[
\begin{aligned}
& r(N) \quad \text { number of observations (last variable pair) } \\
& r \text { (rho) } \quad \rho \text { (last variable pair) } \\
& r(\mathrm{p}) \quad \text { two-sided } p \text {-value (last variable pair) } \\
& r\left(p \_1\right) \quad \text { lower one-sided } p \text {-value (last variable pair) } \\
& \mathrm{r} \text { (p_u) upper one-sided } p \text {-value (last variable pair) } \\
& r \text { (p_exact) two-sided exact } p \text {-value } \\
& r \text { (p_l_exact) lower one-sided exact } p \text {-value } \\
& r \text { (p_u_exact) upper one-sided exact } p \text {-value } \\
& r \text { (n_perm) number of permutations performed }
\end{aligned}
\]

Macros
```

r(exact) "montecarlo" or "enumerate"
r(rngstate) random-number state used for Monte Carlo permutations

| $r$ (Nobs) | number of observations |
| :--- | :--- |
| $r$ (Rho) | $\rho$ |
| $r(\mathrm{P})$ | two-sided $p$-value |

```

Matrices

If exact (..., show) is specified, the stored results from permute are returned as well; see Stored results in \([R]\) permute.
ktau stores the following in r() :
Scalars \(r(N) \quad\) number of observations (last variable pair)
\(r\) (tau_a) \(\quad \tau_{\mathrm{a}}\) (last variable pair)
\(r\left(\right.\) tau_b) \(\quad \tau_{\mathrm{b}}\) (last variable pair)
\(r\) (score) Kendall's score (last variable pair)
\(r\) (se_score) standard error of score (last variable pair)
\(r(\mathrm{p}) \quad\) two-sided \(p\)-value (last variable pair)
Matrices
\(r\) (Nobs) number of observations
r(Tau_a) \(\quad \tau_{a}\)
r(Tau_b) \(\quad \tau_{\mathrm{b}}\)
\(r\) (Score) Kendall's score
\(r\) (Se_Score) standard error of score
\(r(\mathrm{P}) \quad\) two-sided \(p\)-value

\section*{Methods and formulas}

Methods and formulas are presented under the following headings:
Spearman's rank correlation
Exact \(p\)-values
Kendall's tau

\section*{Spearman's rank correlation}

Spearman's (1904) rank correlation is calculated as Pearson's correlation computed on the ranks (averaged for ties) (Conover 1999, 314-315). Ranks are as calculated by egen; see [D] egen.

If \(x_{i}\) and \(y_{i}\), where \(i=1,2, \ldots, n\), are the ranks of one variable pair, and \(n\) is the number of observations, then Spearman's rank correlation is
\[
\rho=\frac{\sum_{i}\left(x_{i}-\bar{x}\right)\left(y_{i}-\bar{y}\right)}{\sqrt{\sum_{i}\left(x_{i}-\bar{x}\right)^{2}} \sqrt{\sum_{i}=\left(y_{i}-\bar{y}\right)^{2}}}
\]
where \(\bar{x}=\left(\sum_{i} x_{i}\right) / n\) is the mean of \(x\) and \(\bar{y}\) is defined similarly.
Under the null hypothesis of independence (or, more generally, exchangeability), the distribution of \(\rho\) is given by all the possible permutations of \(x_{i}\) with \(y_{i}\) fixed. For the permutation distribution, an equivalent statistic to \(\rho\) is
\[
T=\sum_{i} x_{i} y_{i}
\]

The moments of \(T\) for the permutation distribution can be computed exactly. Its mean is
\[
E(T)=\frac{1}{n}\left\{\sum_{i} x_{i}\right\}\left\{\sum_{i} y_{i}\right\}
\]

Assume now that the ranks, \(x_{i}\) and \(y_{i}\), are adjusted so that their means are zero. With this assumption, we have (Stuart, Ord, and Arnold 1999, eqs. 27.42 and 27.43)
\[
\begin{aligned}
E\left(T^{2}\right)= & \frac{1}{n-1}\left\{\sum_{i} x_{i}^{2}\right\}\left\{\sum_{i} y_{i}^{2}\right\} \\
E\left(T^{3}\right)= & \frac{1}{n}\left\{\sum_{i} x_{i}^{3}\right\}\left\{\sum_{i} y_{i}^{3}\right\} \\
& +\frac{3}{n(n-1)}\left\{\sum_{i \neq j} x_{i}^{2} x_{j}\right\}\left\{\sum_{i \neq j} y_{i}^{2} y_{j}\right\} \\
& +\frac{36}{n(n-1)(n-2)}\left\{\sum_{i<j<k} x_{i} x_{j} x_{k}\right\}\left\{\sum_{i<j<k} y_{i} y_{j} y_{k}\right\} \\
E\left(T^{4}\right)= & \frac{1}{n}\left\{\sum_{i} x_{i}^{4}\right\}\left\{\sum_{i} y_{i}^{4}\right\} \\
& +\frac{4}{n(n-1)}\left\{\sum_{i \neq j} x_{i}^{3} x_{j}\right\}\left\{\sum_{i \neq j} y_{i}^{3} y_{j}\right\} \\
& +\frac{12}{n(n-1)}\left\{\sum_{i<j} x_{i}^{2} x_{j}^{2}\right\}\left\{\sum_{i<j} y_{i}^{2} y_{j}^{2}\right\} \\
& +\frac{24}{n(n-1)(n-2)}\left\{\sum_{i \neq j, k ; j<k} x_{i}^{2} x_{j} x_{k}\right\}\left\{\sum_{i \neq j, k ; j<k} y_{i}^{2} y_{j} y_{k}\right\} \\
& +\frac{576}{n(n-1)(n-2)(n-3)}\left\{\sum_{i<j<k<l} x_{i} x_{j} x_{k} x_{l}\right\}\left\{\sum_{i<j<k<l} y_{i} y_{j} y_{k} y_{l}\right\}
\end{aligned}
\]

Note that Stuart, Ord, and Arnold (1999, eq. 27.43) express \(E\left(T^{3}\right)\) and \(E\left(T^{4}\right)\) in terms of \(k\) statistics, but that formulation is exactly equivalent to the equations given above.

An approximate \(p\)-value for Spearman's rank correlation is calculated by fitting a four-parameter beta distribution to the first four moments of \(T\). (See Lord [1965] and Hanson [1991] for a description of the technique of fitting a four-parameter beta distribution to moments of another distribution.) The four-parameter beta distribution with domain \([l, u]\) is
\[
f(x)=\frac{(-l+x)^{\alpha-1}(u-x)^{\beta-1}}{(u-l)^{\alpha+\beta-1} B(\alpha, \beta)}
\]
where \(0 \leq l<u \leq 1\) and \(B(\alpha, \beta)\) is the beta function with \(\alpha>0\) and \(\beta>0\).
The parameters for the beta distribution are calculated as follows. Let \(m\) be the mean of \(T, v\) its variance, \(g_{3}\) its skewness, and \(g_{4}\) its kurtosis (see Methods and formulas in [R] summarize). Define
\[
r=\frac{6\left(g_{4}-g_{3}^{2}-1\right)}{6+3 g_{3}^{2}-2 g_{4}}
\]
and
\[
d=1-\frac{24(r+1)}{(r+2)(r+3) g_{4}-3(r-6)(r+1)}
\]

Let
\[
\begin{aligned}
& a=\frac{r(1-\sqrt{d})}{2} \\
& b=\frac{r(1+\sqrt{d})}{2}
\end{aligned}
\]

If \(g_{3}>0\), then \(\alpha=a\) and \(\beta=b\); otherwise, \(\alpha=b\) and \(\beta=a\). The domain boundaries are given by
\[
\begin{aligned}
& l=m-\alpha \sqrt{\frac{v(\alpha+\beta+1)}{\alpha \beta}} \\
& u=m+\beta \sqrt{\frac{v(\alpha+\beta+1)}{\alpha \beta}}
\end{aligned}
\]

To calculate the \(p\)-values, we first scale the observed value of \(T\) to the domain of beta:
\[
s=\frac{T_{\mathrm{obs}}-l}{u-l}
\]

Then the lower and upper one-sided \(p\)-values are given by
\[
\begin{aligned}
p_{\text {lower }} & =\operatorname{ibeta}(\alpha, \beta, s) \\
p_{\text {upper }} & =\operatorname{ibetatail}(\alpha, \beta, s)
\end{aligned}
\]
where ibeta is Stata's two-parameter cumulative beta distribution and ibetatail is the function for its upper tail; see [FN] Statistical functions. The two-sided \(p\)-value is given by
\[
p=\min \left\{1,2 \min \left(p_{\text {lower }}, p_{\text {upper }}\right)\right\}
\]

\section*{Exact p-values}

Exact \(p\)-values for Spearman's rank correlation are computed by permute. For details on the permutation computation, see \([R]\) permute.

\section*{Kendall's tau}

Kendall's \(\tau\) is calculated in the following manner. For any two pairs of ranks \(\left(x_{i}, y_{i}\right)\) and \(\left(x_{j}, y_{j}\right)\), \(1 \leq i, j \leq n\), define them as concordant if
\[
\left(x_{i}-x_{j}\right)\left(y_{i}-y_{j}\right)>0
\]
and discordant if this product is less than zero.
Kendall's (1938; also see Kendall and Gibbons [1990] or Bland [2015], 187-188) score \(S\) is defined as \(C-D\), where \(C(D)\) is the number of concordant (discordant) pairs. Let \(N=n(n-1) / 2\) be the total number of pairs, so \(\tau_{\mathrm{a}}\) is given by
\[
\tau_{\mathrm{a}}=\frac{S}{N}
\]
and \(\tau_{\mathrm{b}}\) is given by
\[
\tau_{\mathrm{b}}=\frac{S}{\sqrt{N-U} \sqrt{N-V}}
\]
where
\[
\begin{aligned}
U & =\sum_{i=1}^{N_{1}} u_{i}\left(u_{i}-1\right) / 2 \\
V & =\sum_{j=1}^{N_{2}} v_{j}\left(v_{j}-1\right) / 2
\end{aligned}
\]
and where \(N_{1}\) is the number of sets of tied \(x\) values, \(u_{i}\) is the number of tied \(x\) values in the \(i\) th set, \(N_{2}\) is the number of sets of tied \(y\) values, and \(v_{j}\) is the number of tied \(y\) values in the \(j\) th set.

Under the null hypothesis of independence, the variance of \(S\) is exactly (Kendall and Gibbons 1990, 66)
\[
\begin{aligned}
\operatorname{Var}(S)= & \frac{1}{18}\left\{n(n-1)(2 n+5)-\sum_{i=1}^{N_{1}} u_{i}\left(u_{i}-1\right)\left(2 u_{i}+5\right)-\sum_{j=1}^{N_{2}} v_{j}\left(v_{j}-1\right)\left(2 v_{j}+5\right)\right\} \\
& +\frac{1}{9 n(n-1)(n-2)}\left\{\sum_{i=1}^{N_{1}} u_{i}\left(u_{i}-1\right)\left(u_{i}-2\right)\right\}\left\{\sum_{j=1}^{N_{2}} v_{j}\left(v_{j}-1\right)\left(v_{j}-2\right)\right\} \\
& +\frac{1}{2 n(n-1)}\left\{\sum_{i=1}^{N_{1}} u_{i}\left(u_{i}-1\right)\right\}\left\{\sum_{j=1}^{N_{2}} v_{j}\left(v_{j}-1\right)\right\}
\end{aligned}
\]

Using a normal approximation with a continuity correction,
\[
z=\frac{|S|-1}{\sqrt{\operatorname{Var}(S)}}
\]

For the hypothesis of independence, the statistics \(S, \tau_{\mathrm{a}}\), and \(\tau_{\mathrm{b}}\) produce equivalent tests and give the same \(p\)-value.

For Kendall's \(\tau\), the normal approximation is surprisingly accurate for sample sizes as small as 8, at least for calculating \(p\)-values under the null hypothesis for continuous variables. See Kendall and Gibbons [1990, chap. 4], who also present some tables for calculating exact \(p\)-values for \(n<10\).

Let \(v\) be the number of variables specified so that \(k=v(v-1) / 2\) correlation coefficients are to be estimated. If bonferroni is specified, the adjusted \(p\)-value is \(p^{\prime}=\min (1, k p)\). If sidak is specified, \(p^{\prime}=\min \left\{1,1-(1-p)^{n}\right\}\). See Methods and formulas in [R] oneway for a more complete description of the logic behind these adjustments.

Early work on rank correlation is surveyed by Kruskal (1958).

\section*{Acknowledgment}

The original version of ktau was written by Sean Becketti, author of the Stata Press book Introduction to Time Series Using Stata, Revised Edition.

\section*{References}

Barnard, G. A. 1997. Kendall, Maurice George. In Leading Personalities in Statistical Sciences: From the Seventeenth Century to the Present, ed. N. L. Johnson and S. Kotz, 130-132. New York: Wiley.

Bland, M. 2015. An Introduction to Medical Statistics. 4th ed. Oxford: Oxford University Press.
Conover, W. J. 1999. Practical Nonparametric Statistics. 3rd ed. New York: Wiley.
David, H. A., and W. A. Fuller. 2007. Sir Maurice Kendall (1907-1983): A centenary appreciation. American Statistician 61: 41-46. https://doi.org/10.1198/000313007X169055.
Hanson, B. A. 1991. Method of Moments Estimates of the Four-Paramenter Beta Compound Binominal Model and the Calculation of Classification Consistency Indexes. ACT Research Report Series 91-5, American College Testing Program, Iowa City, IA.

Jeffreys, H. 1961. Theory of Probability. 3rd ed. Oxford: Oxford University Press.
Kendall, M. G. 1938. A new measure of rank correlation. Biometrika 30: 81-93. https://doi.org/10.2307/2332226.
Kendall, M. G., and J. D. Gibbons. 1990. Rank Correlation Methods. 5th ed. New York: Oxford University Press.
Kruskal, W. H. 1958. Ordinal measures of association. Journal of the American Statistical Association 53: 814-861. https://doi.org/10.2307/2281954.
Lord, F. M. 1965. A strong true-score theory, with applications. Psychometrika 30: 239-270. https://doi.org/10.1007/BF02289490.
Lovie, P., and A. D. Lovie. 1996. Charles Edward Spearman, F.R.S. (1863-1945). Notes and Records of the Royal Society of London 50: 75-88. https://doi.org/10.1098/rsnr.1996.0007.
Newson, R. B. 2000a. snp15: somersd-Confidence intervals for nonparametric statistics and their differences. Stata Technical Bulletin 55: 47-55. Reprinted in Stata Technical Bulletin Reprints, vol. 10, pp. 312-322. College Station, TX: Stata Press.
__. 2000b. snp15.1: Update to somersd. Stata Technical Bulletin 57: 35. Reprinted in Stata Technical Bulletin Reprints, vol. 10, pp. 322-323. College Station, TX: Stata Press.
——. 2000c. snp15.2: Update to somersd. Stata Technical Bulletin 58: 30. Reprinted in Stata Technical Bulletin Reprints, vol. 10, p. 323. College Station, TX: Stata Press.

\footnotetext{
——. 2001. snp15.3: Update to somersd. Stata Technical Bulletin 61: 22. Reprinted in Stata Technical Bulletin Reprints, vol. 10, p. 324. College Station, TX: Stata Press.
}
__. 2003. snp15_4: Software update for somersd. Stata Journal 3: 325.
__. 2005. snp15_5: Software update for somersd. Stata Journal 5: 470.
2006. Confidence intervals for rank statistics: Percentile slopes, differences, and ratios. Stata Journal 6: 497-520.

Seed, P. T. 2001. sg159: Confidence intervals for correlations. Stata Technical Bulletin 59: 27-28. Reprinted in Stata Technical Bulletin Reprints, vol. 10, pp. 267-269. College Station, TX: Stata Press.
Spearman, C. E. 1904. The proof and measurement of association between two things. American Journal of Psychology 15: 72-101. https://doi.org/10.2307/1412159.

Stuart, A., J. K. Ord, and S. F. Arnold. 1999. Kendall's Advanced Theory of Statistics: Classical Inference and the Linear Model, Vol. 2A. 6th ed. London: Arnold.

\section*{Also see}
[R] correlate - Correlations of variables
[R] nptrend - Tests for trend across ordered groups
[R] permute - Permutation tests

\section*{Title}
spikeplot - Spike plots and rootograms
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Acknowledgments & References \\
Also see & & &
\end{tabular}

\section*{Description}
spikeplot produces a frequency plot for a variable in which the frequencies are depicted as vertical lines from zero. The frequency may be a count, a fraction, or the square root of the count (Tukey's rootogram, circa 1965). The vertical lines may also originate from a baseline other than zero at the user's option.

\section*{Quick start}

Spike plot of v1
spikeplot v1
Same as above, but apply frequency weights wvar
spikeplot v1 [fweight = wvar]
Plot proportions of the total number of observations instead of frequencies
spikeplot v1, fraction
Tukey's rootogram of v2
spikeplot v2, root

\section*{Menu}

Graphics \(>\) Distributional graphs \(>\) Spike plot and rootogram

\section*{Syntax}
spikeplot varname \([\) if \(][\) in \(][\) weight \(][\), options \(]\)
options Description
Main
round (\#) round varname to nearest multiple of \# (bin width)
fraction make vertical scale the proportion of total values; default is frequencies
root make vertical scale show square roots of frequencies
Plot
spike_options affect rendition of plotted spikes
Add plots
addplot (plot) add other plots to generated graph
Y axis, X axis, Titles, Legend, Overall, By
twoway_options any options documented in [G-3] twoway_options
fweights, aweights, and iweights are allowed; see [U] 11.1.6 weight.

\section*{Options}
\(\qquad\) Main
round(\#) rounds the values of varname to the nearest multiple of \#. This action effectively specifies the bin width.
fraction specifies that the vertical scale be the proportion of total values (percentage) rather than the count.
root specifies that the vertical scale show square roots. This option may not be specified if fraction is specified.
\(\qquad\) Plot
spike_options affect the rendition of the plotted spikes; see [G-2] graph twoway spike.
\(\qquad\) Add plots
addplot (plot) provides a way to add other plots to the generated graph. See [G-3] addplot_option.
\(\qquad\) Y axis, X axis, Titles, Legend, Overall, By
twoway_options are any of the options documented in [G-3] twoway_options. These include options for titling the graph (see [G-3] title_options), options for saving the graph to disk (see [G-3] saving_option), and the by () option (see [G-3] by_option).

\section*{Remarks and examples}

\section*{\(>\) Example 1}

Cox and Brady (1997a) present an illustrative example using the age structure of the population of Ghana from the 1960 census (rounded to the nearest 1,000 ). The dataset has ages from 0 (less than 1 year) to 90 . To view the distribution of ages, we would like to use each integer from 0 to 90 as the bins for the dataset.
```

. use https://www.stata-press.com/data/r18/ghanaage
(Age structure of population of Ghana)
. spikeplot age [fw=pop], ytitle("Population in 1,000s") xlab(0(10)90)
> xmtick(5(10)85)

```


The resulting graph shows a "heaping" of ages at the multiples of 5 . Also, ages ending in even numbers are more frequent than ages ending in odd numbers (except for 5). This preference for reporting ages is well known in demography and other social sciences.

Note also that we used the ytitle() option to override the default title of "Frequency" and that we used the xlab() and xmtick() options with numlists to further customize the resulting graph. See [U] 11.1.8 numlist for details on specifying numlists.

\section*{> Example 2}

The rootogram is a plot of the square-root transformation of the frequency counts. The square root of a normal distribution is a multiple of another normal distribution.
```

. clear
. set seed 1234567
. set obs 5000
Number of observations (_N) was 0, now 5,000.
. generate normal = rnormal()
. label variable normal "Gaussian(0,1) random numbers"

```
. spikeplot normal, round(.10) xlab(-4(1)4)

. spikeplot normal, round(.10) xlab(-4(1)4) root


Interpreting a histogram in terms of normality is thus similar to interpreting the rootogram for normality.

This example also shows how the round() option is used to bin the values for a spike plot of a continuous variable.

\section*{> Example 3}
spikeplot can also be used to produce time-series plots. varname should be the time variable, and weights should be specified as the values for those times. To get a plot of daily rainfalls, we type . spikeplot day [fw=rain] if rain, ytitle("Daily rainfall in mm")

The base() option of graph twoway spike may be used to set a different baseline, such as when we want to show variations relative to an average or to some other measure of level.

\section*{Acknowledgments}

The original version of spikeplot was written by Nicholas J. Cox of the Department of Geography at Durham University, UK, who is coeditor of the Stata Journal and author of Speaking Stata Graphics and by Anthony R. Brady founder of Sealed Envelope, London (1997a, 1997b).

\section*{References}

Cox, N. J., and A. R. Brady. 1997a. gr25: Spike plots for histograms, rootograms, and time-series plots. Stata Technical Bulletin 36: 8-11. Reprinted in Stata Technical Bulletin Reprints, vol. 6, pp. 50-54. College Station, TX: Stata Press.
——. 1997b. gr25.1: Spike plots for histograms, rootograms, and time-series plots: Update. Stata Technical Bulletin 40: 12. Reprinted in Stata Technical Bulletin Reprints, vol. 7, p. 58. College Station, TX: Stata Press.
Tukey, J. W. 1965. The future of processes of data analysis. In The Collected Works of John W. Tukey, Volume IV: Philosophy and Principles of Data Analysis: 1965-1986, ed. L. V. Jones, 123-126. Monterey, CA: Wadsworth and Brooks/Cole.

\section*{Also see}
[R] histogram - Histograms for continuous and categorical variables

\section*{Title}
sSc - Install and uninstall packages from SSC
\begin{tabular}{llll} 
Description & Quick start & Syntax & Options \\
Remarks and examples & Acknowledgments & References & Also see
\end{tabular}

\section*{Description}
ssc works with packages (and files) from the Statistical Software Components (SSC) Archive, which is often called the Boston College Archive and is provided by http://repec.org.

The SSC has become the premier Stata download site for community-contributed software on the web. ssc provides a convenient interface to the resources available there. For example, on Statalist (see https://www.statalist.org/), users will often write

The program can be found by typing ssc install newprogramname.
Typing that would load everything associated with newprogramname, including the help files.
If you are searching for what is available, type ssc new and ssc hot, and see [R] search. search searches the SSC and other places, too. search provides a GUI interface from which programs can be installed, including the programs at the SSC Archive.

You can uninstall particular packages by using ssc uninstall. For the packages that you keep, see \([R]\) ado update for an automated way of keeping those packages up to date.

\section*{Command overview}
ssc new summarizes the packages made available or updated recently. Output is presented in the Stata Viewer, and from there you may click to find out more about individual packages or to install them.
ssc hot lists the most popular packages-popular based on a moving average of the number of downloads in the past three months. By default, 10 packages are listed.
ssc describe pkgname describes, but does not install, the specified package. Use search to find packages; see [R] search. If you know the package name but do not know the exact spelling, type ssc describe followed by one letter, \(\mathrm{a}_{\mathrm{z}} \mathbf{z}\) or _ (underscore), to list all the packages starting with that letter.
ssc install pkgname installs the specified package. You do not have to describe a package before installing it. (You may also install a package by using net install; see [R] net.)
ssc uninstall pkgname removes the previously installed package from your computer. It does not matter how the package was installed. (ssc uninstall is a synonym for ado uninstall, so either may be used to installed any package.)
ssc type filename types a specific file stored at SSC. ssc cat is a synonym for ssc type, which may appeal to those familiar with Unix.
ssc copy filename copies a specific file stored at SSC to your computer. By default, the file is copied to the current directory, but you can use options to change this. ssc copy is a rarely used alternative to ssc install..., all. ssc cp is a synonym for ssc copy.

\section*{Quick start}

Describe mycommand at SSC
ssc describe mycommand
Install mycommand from SSC
ssc install mycommand
Same as above, but replace previously installed version of mycommand
ssc install mycommand, replace
See a summary of all new and recently updated packages on SSC
ssc new
See a summary of the 10 most popular SSC packages
ssc hot
Same as above, but see the top 25 packages
ssc hot, \(n(25)\)

\section*{Syntax}

Summary of packages most recently added or updated at SSC
```

ssc new [, saving(filename[, replace]) type]

```

Summary of most popular packages at SSC ssc hot \([, \mathrm{n}(\#)\) author(name)]

Describe a specified package at SSC
```

ssc describe {pkgname | letter} [, saving(filename[, replace])]

```

Install a specified package from SSC
```

ssc install pkgname [, all replace]

```

Uninstall from your computer a previously installed package from SSC ssc uninstall pkgname

Type a specific file stored at SSC
```

ssc type filename [, asis]

```

Copy a specific file from SSC to your computer
```

ssc copy filename [, plus personal replace public binary]

```
where letter in ssc describe is a-z or _.

\section*{Options}

Options are presented under the following headings:
Options for use with ssc new
Options for use with ssc hot
Option for use with ssc describe
Options for use with ssc install
Option for use with ssc type
Options for use with ssc copy

\section*{Options for use with ssc new}
saving(filename[, replace]) specifies that the "what's new" summary be saved in filename. If filename is specified without a suffix, filename.smcl is assumed. If saving() is not specified, saving(ssc_results.smcl) is assumed.
type specifies that the "what's new" results be displayed in the Results window rather than in the Viewer.

\section*{Options for use with ssc hot}
n (\#) specifies the number of packages to list; \(\mathrm{n}(10)\) is the default. Specify \(\mathrm{n}(\).\() to list all packages\) in order of popularity.
author (name) lists the 10 most popular packages by the specified author. If \(n(\#)\) is also specified, the top \# packages are listed.

\section*{Option for use with ssc describe}
saving(filename [, replace]) specifies that, in addition to the description's being displayed on your screen, it be saved in the specified file.

If filename is specified without an extension, .smcl will be assumed, and the file will be saved as a SMCL file.

If filename is specified with an extension, no default extension is added. If the extension is .log, the file will be stored as a text file.

If replace is specified, filename is replaced if it already exists.

\section*{Options for use with ssc install}
all specifies that any ancillary files associated with the package be downloaded to your current directory, in addition to the program and help files being installed. Ancillary files are files that do not end in . ado or .sthlp and typically contain datasets or examples of the use of the new command.

You can find out which files are associated with the package by typing ssc describe pkgname before or after installing. If you install without using the all option and then want the ancillary files, you can ssc install again.
replace specifies that any files being downloaded that already exist on your computer be replaced by the downloaded files. If replace is not specified and any files already exist, none of the files from the package is downloaded or installed.

It is better not to specify the replace option and wait to see if there is a problem. If there is a problem, it is usually better to uninstall the old package by using ssc uninstall or ado uninstall (which are, in fact, the same command).

\section*{Option for use with ssc type}
asis affects how files with the suffixes .smcl and .sthlp are displayed. The default is to interpret SMCL directives the file might contain. asis specifies that the file be displayed in raw, uninterpreted form.

\section*{Options for use with ssc copy}
plus specifies that the file be copied to the PLUS directory, the directory where community-contributed additions are installed. Typing sysdir will display the identity of the PLUS directory on your computer; see [P] sysdir.
personal specifies that the file be copied to your PERSONAL directory as reported by sysdir; see [ P\(]\) sysdir.
If neither plus nor personal is specified, the default is to copy the file to the current directory. replace specifies that, if the file already exists on your computer, the new file replace it.
public specifies that the new file be made readable by everyone; otherwise, the file will be created according to the default permission you have set with your operating system.
binary specifies that the file being copied is a binary file and that it is to be copied as is. The default is to assume that the file is a text file and change the end-of-line characters to those appropriate for your computer/operating system.

\section*{Remarks and examples}

Users can add new features to Stata, and some users choose to make new features that they have written available to others via the web. The files that comprise a new feature are called a package, and a package usually consists of one or more ado-files and help files. The net command (see [R] net) makes it reasonably easy to install and uninstall packages regardless of where they are on the web. One site, the SSC, has become particularly popular as a repository for additions to Stata. Command ssc is an easier to use version of net designed especially for the SSC.

Many packages are available at the SSC. Packages have names, such as oaxaca, estout, or egenmore. At SSC, capitalization is not significant, so Oaxaca, ESTOUT, and EGENmore are ways of writing the same package names.

When you type
. ssc install oaxaca
the files associated with the package are downloaded and installed on your computer. Package names usually correspond to the names of the command being added to Stata, so one would expect that installing the package oaxaca will add command oaxaca to Stata on your computer, and expect that typing help oaxaca will provide the documentation. That is the situation here, but that is not always so. Before or after installing a package, type ssc describe pkgname to obtain the details.

\section*{- Example 1}
ssc new summarizes the packages most recently made available or updated. Output is presented in the Viewer, from which you may click on a package name to find out more or install it. Try it for yourself! Type ssc new in the Command window.
ssc hot provides a list of the most popular packages at SSC.
. ssc hot
Top 10 packages at SSC
Jan 2023
\begin{tabular}{rcll} 
Rank & \# hits & Package & Author(s) \\
\hline 1 & 62060.3 & outreg2 & Roy Wada \\
2 & 52912.8 & estout & Ben Jann \\
3 & 32185.0 & winsor2 & Yujun Lian \\
4 & 31112.7 & asdoc & Attaullah Shah \\
5 & 23368.2 & grstyle & Ben Jann \\
6 & 21891.0 & fastgini & Zurab Sajaia \\
7 & 20350.6 & reghdfe & Sergio Correia \\
8 & 16610.7 & distinct & Gary Longton, Nicholas J. Cox \\
9 & 16247.2 & ftools & Sergio Correia \\
10 & 15721.5 & logout & Roy Wada
\end{tabular}
(Click on package name for description)
Use the \(n(\#)\) option to change the number of packages listed:
. ssc hot, \(\mathrm{n}(20)\)
Top 20 packages at SSC
\begin{tabular}{|c|c|c|c|}
\hline Rank & \[
\begin{gathered}
\text { Jan } 2023 \\
\text { \# hits }
\end{gathered}
\] & Package & Author (s) \\
\hline 1 & 62060.3 & outreg2 & Roy Wada \\
\hline 2 & 52912.8 & estout & Ben Jann \\
\hline 3 & 32185.0 & winsor2 & Yujun Lian \\
\hline 4 & 31112.7 & asdoc & Attaullah Shah \\
\hline 5 & 23368.2 & grstyle & Ben Jann \\
\hline 6 & 21891.0 & fastgini & Zurab Sajaia \\
\hline 7 & 20350.6 & reghdfe & Sergio Correia \\
\hline 8 & 16610.7 & distinct & Gary Longton, Nicholas J. Cox \\
\hline 9 & 16247.2 & ftools & Sergio Correia \\
\hline 10 & 15721.5 & logout & Roy Wada \\
\hline 11 & 13391.7 & coefplot & Ben Jann \\
\hline 12 & 12541.0 & ivreg29 & Christopher F Baum, Steven Stillman, Mark E Schaffer \\
\hline 13 & 12506.3 & ivreg2 & Mark E Schaffer, Steven Stillman, Christopher F Baum \\
\hline 14 & 12466.0 & ivreg28 & Mark E Schaffer, Steven Stillman, Christopher F Baum \\
\hline 15 & 12441.0 & ivreg210 & Steven Stillman, Christopher F Baum, Mark E Schaffer \\
\hline 16 & 11071.7 & psmatch2 & Barbara Sianesi, Edwin Leuven \\
\hline 17 & 10332.5 & geodist & Robert Picard \\
\hline 18 & 10031.7 & fre & Ben Jann \\
\hline 19 & 9337.1 & labutil & Nicholas J. Cox \\
\hline 20 & 8827.7 & unique & Tony Brady \\
\hline
\end{tabular}
(Click on package name for description)

The author (name) option allows you to list the most popular packages by a specific person:
. ssc hot, author(baum)
Top 10 packages at SSC by author Baum
\begin{tabular}{ccll} 
Rank & \begin{tabular}{c} 
Jan 2023 \\
\# hits
\end{tabular} & Package & Author(s) \\
\hline 12 & 12541.0 & ivreg29 & \begin{tabular}{c} 
Christopher F Baum, Steven Stillman, \\
Mark E Schaffer
\end{tabular} \\
13 & 12506.3 & ivreg2 & \begin{tabular}{c} 
Mark E Schaffer, Steven Stillman, \\
Christopher F Baum
\end{tabular} \\
14 & 12466.0 & ivreg28 & \begin{tabular}{c} 
Mark E Schaffer, Steven Stillman, \\
Christopher F Baum
\end{tabular} \\
(output omitted) & &
\end{tabular}
(Click on package name for description)
ssc describe pkgname describes, but does not install, the specified package. You must already know the name of the package. See [R] search for assistance in searching for packages. Sometimes you know the package name, but you do not know the exact spelling. Then, you can type ssc describe followed by one letter, a-z or \(\quad\), to list all the packages starting with that letter; even so, using search is better.
. ssc describe i
```

http://fmwww.bc.edu/repec/bocode/i/
(no title)
PACKAGES you could -net describe-:
(output omitted)
ivreg2
module for extended instrumental variables/2SLS and GMM
estimation
ivreg210 module for extended instrumental variables/2SLS and GMM
estimation (v10)
ivreg28 module for extended instrumental variables/2SLS and GMM
estimation (v8)
ivreg29 module for extended instrumental variables/2SLS and GMM
estimation (v9)
ivreg2h module to perform instrumental variables estimation
using heteroskedasticity-based instruments
ivreg2hdfe module to estimate an Instrumental Variable Linear
Regression Model with two High Dimensional Fixed
Effects
ivreg2m module to identify treatment-effects estimates with
potentially misreported and endogenous program
participation
ivreg_ss module to compute confidence intervals, standard errors,
and p-values in an IV regression in which the excluded
instrumental variable has a shift-share structure
ivreghdfe module for extended instrumental variable regressions
with multiple levels of fixed effects
ivregress2 module to export first and second-stage results similar
to ivregress
(output omitted)

```
(type ssc describe pkgname for more information on pkgname)

The default setting for the saving() option is for the output to be saved with the .smcl extension. You could also save the file with a log extension, and in this case, the file would be stored as a text file. For example, we could type
```

. ssc describe i, saving(i.index)

```
or the following:
```

. ssc describe ivreg2, saving(ivreg2.log)

```
ssc install pkgname installs the specified package. You do not have to describe a package before installing it. There are ways of installing packages other than ssc install, such as net; see \([R]\) net. It does not matter how a package is installed. For instance, a package can be installed using net and still be uninstalled using ssc.
```

. ssc install ivreg2
checking ivreg2 consistency and verifying not already installed...
installing into C:\ado\plus\...
installation complete.

```
ssc uninstall pkgname removes the specified, previously installed package from your computer. You can uninstall immediately after installation or at any time in the future. (Technical note: ssc uninstall is a synonym for ado uninstall, so it can uninstall any installed package, not just packages obtained from the SSC.)
```

. ssc uninstall ivreg2
package ivreg2 from http://fmwww.bc.edu/repec/bocode/i
'IVREG2': module for extended instrumental variables/2SLS and GNM
estimation
(package uninstalled)

```
ssc type filename types a specific file stored at the SSC. Although not shown in the syntax diagram, ssc cat is a synonym for ssc type, which may appeal to those familiar with Unix. To view only the ivreg2 help file from the ivreg2 package, you would type
```

. ssc type ivreg2.sthlp

```
```

help for ivreg2

```
Extended instrumental variables/2SLS, GMM and AC/HAC, LIML and k-class regression
    Full syntax
    (output omitted)
ssc copy filename copies a specific file stored at the SSC to your computer. By default, the file is copied to the current directory, but you can use options to change this. ssc copy is a rarely used alternative to ssc install ..., all. ssc cp is a synonym for ssc copy.
```

. ssc copy ivreg2.ado
(file ivreg2.ado copied to current directory)

```

For more details on the SSC Archive and for information on how to submit your own programs to the SSC, see http://repec.org/bocode/s/sscsubmit.html.

\section*{Acknowledgments}
ssc is based on archutil by Nicholas J. Cox of the Department of Geography at Durham University, UK, who is coeditor of the Stata Journal and author of Speaking Stata Graphics and by Christopher F. Baum of the Department of Economics at Boston College and author of the Stata Press books An Introduction to Modern Econometrics Using Stata and An Introduction to Stata Programming and coauthor of the Stata Press book Environmental Econometrics Using Stata. The reworking of the original was done with their blessing and their participation.

Baum maintains the Stata-related files stored at the SSC Archive. We thank him for this contribution to the Stata community.

\section*{References}

Baum, C. F., and N. J. Cox. 1999. ip29: Metadata for user-written contributions to the Stata programming language. Stata Technical Bulletin 52: 10-12. Reprinted in Stata Technical Bulletin Reprints, vol. 9, pp. 121-124. College Station, TX: Stata Press.

Choodari-Oskooei, B., and T. P. Morris. 2016. Quantifying the uptake of user-written commands over time. Stata Journal 16: 88-95.

Cox, N. J., and C. F. Baum. 2000. ip29.1: Metadata for user-written contributions to the Stata programming language. Stata Technical Bulletin 54: 21-22. Reprinted in Stata Technical Bulletin Reprints, vol. 9, pp. 124-126. College Station, TX: Stata Press.

Newson, R. B. 2022. Stata tip 147: Porting downloaded packages between machines. Stata Journal 22: 996-997.

\section*{Also see}
[R] ado update - Update community-contributed packages
[R] net - Install and manage community-contributed additions from the Internet
[R] search - Search Stata documentation and other resources
[R] sj - Stata Journal installation instructions
[P] sysdir - Query and set system directories

\section*{Title}
stem - Stem-and-leaf displays
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & References \\
Also see & & &
\end{tabular}

\section*{Description}
stem displays stem-and-leaf plots.

\section*{Quick start}

Stem-and-leaf plot for v1
stem v1
Same as above, with 2 lines per interval of 10
stem v1, lines(2)
Specify that each stem has a width of 5 (equivalent to above)
stem v1, width(5)
Stem-and-leaf plot for v2 with an interval of 100
stem v2, digits(2)
Same as above, with 4 lines per interval of 100
stem v2, digits(2) lines(4)
Display v2 rounded to the nearest hundred
stem v2, round(100)
Do not display empty stems
stem v2, prune

\section*{Menu}

Statistics \(>\) Summaries, tables, and tests \(>\) Distributional plots and tests \(>\) Stem-and-leaf display

\section*{Syntax}
stem varname [if] [in] [, options]
options Description

Main
\begin{tabular}{ll} 
prune & do not print stems that have no leaves \\
round (\#) & round data to this value; default is round(1) \\
truncate(\#) & truncate data to this value \\
digits(\#) & digits per leaf; default is digits (1) \\
lines(\#) & number of stems per interval of \(10^{\text {digits }}\) \\
width(\#) & stem width; equal to \(10^{\text {digits } / \text { width }}\)
\end{tabular}
round (\#)
truncate(\#)
digits(\#)
lines(\#)
width(\#)
do not print stems that have no leaves
round data to this value; default is round(1)
truncate data to this value
digits per leaf; default is digits (1)
number of stems per interval of \(10^{\text {digits }}\)
stem width; equal to \(10^{\text {digits }} /\) width
by and collect are allowed; see [U] 11.1.10 Prefix commands.

\section*{Options}

\section*{\(\downarrow \sqrt{\text { Main }}\)}
prune prevents printing any stems that have no leaves.
round(\#) rounds the data to this value and displays the plot in these units. If round() is not specified, noninteger data will be rounded automatically.
truncate(\#) truncates the data to this value and displays the plot in these units. digits (\#) sets the number of digits per leaf. The default is 1 .
lines (\#) sets the number of stems per every data interval of \(10^{\text {digits }}\). The value of lines () must divide \(10^{\text {digits }}\); that is, if digits (1) is specified, then lines () must divide 10. If digits (2) is specified, then lines () must divide 100, etc. Only one of lines() or width() may be specified. If neither is specified, an appropriate value will be set automatically.
width (\#) sets the width of a stem. lines () is equal to \(10^{\text {digits }} /\) width, and this option is merely an alternative way of setting lines(). The value of width() must divide \(10^{\text {digits }}\). Only one of width() or lines() may be specified. If neither is specified, an appropriate value will be set automatically.

Note: If lines() or width() is not specified, digits() may be decreased in some circumstances to make a better-looking plot. If lines() or width() is set, the user-specified value of digits() will not be altered.

\section*{Remarks and examples}
\(>\) Example 1
Stem-and-leaf displays are a compact way to present considerable information about a batch of data. For instance, using our automobile data (described in [U] 1.2.2 Example datasets):
```

. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. stem mpg
Stem-and-leaf plot for mpg (Mileage (mpg))
1t | 22
1f 44444455
1s 66667777
1. }88888888899999999
2* 00011111
2t 22222333
2f 444455555
2s 666
2. }888
3* 001
3t
3f 455
3s
3.
4* 1

```

The stem-and-leaf display provides a way to list our data. The expression to the left of the vertical bar is called the stem; the digits to the right are called the leaves. All the stems that begin with the same digit and the corresponding leaves, written beside each other, reconstruct an observation of the data. Thus, if we look at the four stems that begin with the digit 1 and their corresponding leaves, we see that we have two cars rated at \(12 \mathrm{mpg}, 6\) cars at 14,2 at 15 , and so on. The car with the highest mileage rating in our data is rated at 41 mpg .

The above plot is a five-line plot with lines() equal to 5 (five lines per interval of 10) and width() equal to 2 (two leaves per stem).

Instead, we could specify lines(2):
```

. stem mpg, lines(2)
Stem-and-leaf plot for mpg (Mileage (mpg))
1* 年2444444
1. 556666777788888888899999999
2* 00011111222223334444
2. 555556668889
3* 0014
3. 55
4* 1

```
stem mpg, width(5) would produce the same plot as above.

The stem-and-leaf display provides a crude histogram of our data, one not so pretty as that produced by histogram (see \([R]\) histogram), but one that is nonetheless informative.

\section*{> Example 2}

The miles per gallon rating fits easily into a stem-and-leaf display because, in our data, it has two digits. However, stem does not require two digits.
```

. stem price, lines(1) digits(3)
Stem-and-leaf plot for price (Price)
3*** 291,299,667,748,798,799,829,895,955,984,995
4*** 010,060,082,099,172,181,187,195,296,389,424,425,453,482,499, ... (26)
5*** 079,104,172,189,222,379,397,705,719,788,798,799,886,899
6*** 165,229,295,303,342,486,850
7*** 140,827
8*** 129,814
9*** 690,735
10*** 371,372
11*** 385,497,995
12*** 990
13*** 466,594
14*** 500
15*** 906

```

The (26) at the right of the second stem shows that there were 26 leaves on this stem-too many to display on one line.

We can make a more compact stem-and-leaf plot by rounding. To display stem in units of 100 , we could type
```

. stem price, round(100)
Stem-and-leaf plot for price (Price)
price rounded to nearest multiple of 100
plot in units of 100
3* | 33778889
4* 00001112222344455555667777899
5* 11222447788899
6* 2233359
7* 18
8* 18
9* 77
10* 44
11* 45
12* 0
13* 056
14* 5
15* 9

```
price, in our data, has four or five digits. stem presented the display in terms of units of 100 , so a car that cost \(\$ 3,291\) was treated for display purposes as \(\$ 3,300\).

\section*{- Technical note}

Stem-and-leaf diagrams have been used in Japanese railway timetables, as shown in Tufte (1990, 46-47).

\section*{Stored results}
stem stores the following in r() :
Scalars
r(width) width of a stem
\(r\) (digits) number of digits per leaf; default is 1
Macros
\(r\) (round) number specified in round()
\(r\) (truncate) number specified in truncate()

\section*{References}

Cox, N. J. 2007. Speaking Stata: Turning over a new leaf. Stata Journal 7: 413-433.
Emerson, J. D., and D. C. Hoaglin. 1983. Stem-and-leaf displays. In Understanding Robust and Exploratory Data Analysis, ed. D. C. Hoaglin, C. F. Mosteller, and J. W. Tukey, 7-32. New York: Wiley.
Tufte, E. R. 1990. Envisioning Information. Cheshire, CT: Graphics Press.
Tukey, J. W. 1972. Some graphic and semigraphic displays. In Statistical Papers in Honor of George W. Snedecor, ed. T. A. Bancroft and S. A. Brown, 293-316. Ames, IA: Iowa State University Press.
__. 1977. Exploratory Data Analysis. Reading, MA: Addison-Wesley.

\section*{Also see}
[R] histogram - Histograms for continuous and categorical variables
[R] lv - Letter-value displays

\section*{Title}
stepwise - Stepwise estimation
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
stepwise performs stepwise estimation. Typing
```

. stepwise, pr(\#): command

```
performs backward-selection estimation for command. The stepwise selection method is determined by the following option combinations:
\begin{tabular}{ll} 
options & Description \\
\hline \(\operatorname{pr}(\#)\) & backward selection \\
\(\operatorname{pr}(\#)\) hierarchical & backward hierarchical selection \\
\(\operatorname{pr}(\#)\) pe(\#) & backward stepwise \\
pe(\#) & forward selection \\
pe(\#) hierarchical & forward hierarchical selection \\
pr(\#) pe(\#) forward & forward stepwise \\
\hline
\end{tabular}
command defines the estimation command to be executed. The following Stata commands are supported by stepwise:
betareg, clogit, cloglog, glm, intreg, logistic, logit, nbreg, ologit, oprobit, poisson, probit, qreg, regress, scobit, stcox, stcrreg, stintreg, streg, tobit
stepwise expects command to have the following form:
command_name [depvar] term [term...] [if] [in] [weight] [, command_options] where term is either varname or (varlist) (a varlist in parentheses indicates that this group of variables is to be included or excluded together). depvar is not present when command_name is stcox, stcrreg, stintreg, or streg; otherwise, depvar is assumed to be present. For intreg, depvar is actually two dependent variable names (depvar \({ }_{1}\) and depvar \({ }_{2}\) ).
sw is a synonym for stepwise.
For model selection and estimation using lasso, see the Stata Lasso Reference Manual.

\section*{Quick start}

Backward selection, removing terms with \(p \geq 0.2\)
stepwise, \(p r(.2):\) regress \(y ~ x 1 ~ x 2 ~ x 3 ~ x 4 ~\)
Same as above, and select from the indicators for categorical variable a

Same as above, but force x 1 to be included in model
stepwise, pr(.2) lockterm1: regress y x1 x2 x3 x4 i.a
Consider the indicators for a as a group for inclusion in model

Add d1, d 2 , and d 3 , and force them to be included in model
stepwise, \(\operatorname{pr}(.2)\) lockterm1: regress \(y(d 1 d 2 d 3) \mathrm{x} 1 \mathrm{x} 2 \mathrm{x} 3 \mathrm{x} 4\) (i.a)
Forward selection, adding terms with \(p<0.1\)
stepwise, pe(.1): regress y x1 x2 x3 x4
Backward stepwise selection, removing terms with \(p \geq 0.2\) and adding those with \(p<0.1\)
stepwise, \(\mathrm{pr}(.2) \mathrm{pe}(.1):\) regress y x1 x 2 x 3 x 4
Forward stepwise selection, adding terms with \(p<0.1\) and removing those with \(p \geq 0.2\)
stepwise, \(\operatorname{pr}(.2) \mathrm{pe}(.1)\) forward: regress y x1 x 2 x 3 x 4
Backward hierarchical selection
stepwise, \(\operatorname{pr}(.2)\) hierarchical: regress y x1 x2 x3 x4
Forward hierarchical selection
stepwise, pe(.1) hierarchical: regress y x1 x2 x3 x4

Note: In the above examples, regress could be replaced with any estimation command allowing the stepwise prefix.

\section*{Menu}

Statistics \(>\) Other \(>\) Stepwise estimation

\section*{Syntax}
```

stepwise [, options] : command

```
options Description

Model
* pr (\#)
* pe (\#)

Model2
forward
hierarchical
lockterm1
lr
Reporting
display_options control columns and column formats and line width
*At least one of \(\operatorname{pr}(\#)\) or pe(\#) must be specified.
by is allowed; see [U] 11.1.10 Prefix commands.
Weights are allowed if command allows them; see [U] 11.1.6 weight.
All postestimation commands behave as they would after command without the stepwise prefix; see the postestimation manual entry for command.

\section*{Options}
\(\qquad\) Model
\(\mathrm{pr}(\#)\) specifies the significance level for removal from the model; terms with \(p \geq \mathrm{pr}()\) are eligible for removal.
\(\mathrm{pe}(\#)\) specifies the significance level for addition to the model; terms with \(p<\mathrm{pe}()\) are eligible for addition.
\(\qquad\) Model 2
forward specifies the forward-stepwise method and may be specified only when both pr() and pe () are also specified. Specifying both pr() and pe() without forward results in backward-stepwise selection. Specifying only pr() results in backward selection, and specifying only pe() results in forward selection.
hierarchical specifies hierarchical selection.
lockterm1 specifies that the first term be included in the model and not be subjected to the selection criteria.

Ir specifies that the test of term significance be the likelihood-ratio test. The default is the less computationally expensive Wald test; that is, the test is based on the estimated variance-covariance matrix of the estimators.

\footnotetext{
Reporting
display_options: noci, nopvalues, cformat (\%fmt), pformat(\%fmt), sformat(\%fmt), and nolstretch; see [R] Estimation options.
}

\section*{Remarks and examples}

Remarks are presented under the following headings:

\author{
Introduction \\ Search logic for a step \\ Full search logic \\ Examples \\ Estimation sample considerations \\ Messages \\ Programming for stepwise
}

\section*{Introduction}

Typing
. stepwise, pr(.10): regress y1 x1 x2 d1 d2 d3 x4 x5
performs a backward-selection search for the regression model y 1 on \(\mathrm{x} 1, \mathrm{x} 2, \mathrm{~d} 1, \mathrm{~d} 2, \mathrm{~d} 3, \mathrm{x} 4\), and \(x 5\). In this search, each explanatory variable is said to be a term. Typing
```

. stepwise, pr(.10): regress y1 x1 x2 (d1 d2 d3) (x4 x5)

```
performs a similar backward-selection search, but the variables d 1 , d 2 , and d 3 are treated as one term, as are x 4 and x 5 . That is, d 1 , d 2 , and d 3 may or may not appear in the final model, but they appear or do not appear together.

\section*{> Example 1}

Using the automobile dataset, we fit a backward-selection model of mpg:


This estimation treated each variable as its own term and thus considered each one separately. The engine displacement and gear ratio should really be considered together:

. stepwise, pr(.2): regress mpg c.weight\#\#c.weight (displ gear) turn headroom
> i.foreign price
note: Ob.foreign omitted because of estimability.
Wald test, begin with full model:
\(p=0.7116>=0.2000\), removing headroom
\(p=0.3944>=0.2000\), removing displacement gear_ratio
\(p=0.2798>=0.2000\), removing price

\section*{Search logic for a step}

Before discussing the complete search logic, consider the logic for a step-the first step-in detail. The other steps follow the same logic. If you type
```

. stepwise, pr(.20): regress y1 x1 x2 (d1 d2 d3) (x4 x5)

```
the logic is
1. Fit the model y on x 1 x 2 d 1 d 2 d 3 x 4 x 5 .
2. Consider dropping \(\times 1\).
3. Consider dropping x2.
4. Consider dropping d1 d2 d3.
5. Consider dropping \(\mathrm{x} 4 \times 5\).
6. Find the term above that is least significant. If its significance level is \(\geq 0.20\), remove that term.

If you type
```

. stepwise, pr(.20) hierarchical: regress y1 x1 x2 (d1 d2 d3) (x4 x5)

```
the logic would be different because the hierarchical option states that the terms are ordered. The initial logic would become
1. Fit the model y on x 1 x 2 d 1 d 2 d 3 x 4 x 5 .
2. Consider dropping \(x 4 \times 5\)-the last term.
3. If the significance of this last term is \(\geq 0.20\), remove the term.

The process would then stop or continue. It would stop if \(x 4 \times 5\) were not omitted, and otherwise, stepwise would continue to consider the significance of the next-to-last term, d1 d2 d3.

Specifying pe() rather than \(\operatorname{pr}()\) switches to forward estimation. If you type
```

. stepwise, pe(.20): regress y1 x1 x2 (d1 d2 d3) (x4 x5)

```
stepwise performs forward-selection search. The logic for the first step is
1. Fit a model of y on nothing (meaning a constant).
2. Consider adding \(\times 1\).
3. Consider adding x2.
4. Consider adding d1 d2 d3.
5. Consider adding \(\mathrm{x} 4 \times 5\).
6. Find the term above that is most significant. If its significance level is \(<0.20\), add that term.

As with backward estimation, if you specify hierarchical,
```

. stepwise, pe(.20) hierarchical: regress y1 x1 x2 (d1 d2 d3) (x4 x5)

```
the search for the most significant term is restricted to the next term:
1. Fit a model of \(y\) on nothing (meaning a constant).
2. Consider adding x 1 -the first term.
3. If the significance is \(<0.20\), add the term.

If x 1 were added, stepwise would next consider x 2 ; otherwise, the search process would stop.
stepwise can also use a stepwise selection logic that alternates between adding and removing terms. The full logic for all the possibilities is given below.

\section*{Full search logic}
\begin{tabular}{|c|c|}
\hline Option & Logic \\
\hline \begin{tabular}{l}
\[
\mathrm{pr}()
\] \\
(backward selection)
\end{tabular} & \begin{tabular}{l}
Fit the full model on all explanatory variables. \\
While the least-significant term is "insignificant", remove it and reestimate.
\end{tabular} \\
\hline pr() hierarchical (backward hierarchical selection) & Fit full model on all explanatory variables. While the last term is "insignificant", remove it and reestimate. \\
\hline \begin{tabular}{l}
pr() pe() \\
(backward stepwise)
\end{tabular} & \begin{tabular}{l}
Fit full model on all explanatory variables. \\
If the least-significant term is "insignificant", remove it and reestimate; otherwise, stop. \\
Do that again: if the least-significant term is "insignificant", remove it and reestimate; otherwise, stop. \\
Repeatedly, \\
if the most-significant excluded term is "significant", add it and reestimate; \\
if the least-significant included term is "insignificant", remove it and reestimate; \\
until neither is possible.
\end{tabular} \\
\hline \begin{tabular}{l}
pe() \\
(forward selection)
\end{tabular} & \begin{tabular}{l}
Fit "empty" model. \\
While the most-significant excluded term is "significant", add it and reestimate.
\end{tabular} \\
\hline pe() hierarchical (forward hierarchical selection) & \begin{tabular}{l}
Fit "empty" model. \\
While the next term is "significant", add it and reestimate.
\end{tabular} \\
\hline pr() pe() forward (forward stepwise) & \begin{tabular}{l}
Fit "empty" model. \\
If the most-significant excluded term is "significant", add it and reestimate; otherwise, stop. \\
Do that again: if the most-significant excluded term is "significant", add it and reestimate; otherwise, stop. Repeatedly, \\
if the least-significant included term is "insignificant", remove it and reestimate; \\
if the most-significant excluded term is "significant", add it and reestimate; \\
until neither is possible.
\end{tabular} \\
\hline
\end{tabular}

\section*{Examples}

The following two statements are equivalent; both include solely single-variable terms:
. stepwise, pr(.2): regress price mpg weight displ
. stepwise, \(\operatorname{pr}(.2)\) : regress price (mpg) (weight) (displ)
The following two statements are equivalent; the last term in each is \(r 1, \ldots, r 4\) :
```

. stepwise, pr(.2) hierarchical: regress price mpg weight displ (r1-r4)
. stepwise, pr(.2) hierarchical: regress price (mpg) (weight) (displ) (r1-r4)

```

To group variables weight and displ into one term, type
```

. stepwise, pr(.2) hierarchical: regress price mpg (weight displ) (r1-r4)

```
stepwise can be used with commands other than regress; for instance,
```

. stepwise, pr(.2): logit outcome (sex weight) treated1 treated2

```
. stepwise, \(\operatorname{pr}(.2)\) : logistic outcome (sex weight) treated1 treated2

Either statement would fit the same model because logistic and logit both perform logistic regression; they differ only in how they report results; see [R] logit and [R] logistic.

We use the lockterm1 option to force the first term to be included in the model. To keep treated1 and treated2 in the model no matter what, we type
```

. stepwise, pr(.2) lockterm1: logistic outcome (treated1 treated2) ...

```

After stepwise estimation, we can type stepwise without arguments to redisplay results,
. stepwise
(output from logistic appears)
or type the underlying estimation command:
. logistic
(output from logistic appears )
At estimation time, we can specify options unique to the command being stepped:
```

. stepwise, pr(.2): logit outcome (sex weight) treated1 treated2, or

```
or is logit's option to report odds ratios rather than coefficients; see \([R]\) logit.

\section*{Estimation sample considerations}

Whether you use backward or forward estimation, stepwise forms an estimation sample by taking observations with nonmissing values of all the variables specified (except for depvar \(r_{1}\) and depvar \({ }_{2}\) for intreg). The estimation sample is held constant throughout the stepping. Thus, if you type
```

. stepwise, pr(.2) hierarchical: regress amount sk edul sval

```
and variable sval is missing in half the data, that half of the data will not be used in the reported model, even if sval is not included in the final model.

The function e (sample) identifies the sample that was used. e(sample) contains 1 for observations used and 0 otherwise. For instance, if you type
and the final model is outcome on \(\mathrm{x} 1, \mathrm{x} 5, \mathrm{x} 6\), and x 7 , you could re-create the final regression by typing
```

. logistic outcome x1 x5 x6 x7 if e(sample)

```

You could obtain summary statistics within the estimation sample of the independent variables by typing
```

. summarize x1 x5 x6 x7 if e(sample)

```

If you fit another model, e(sample) will automatically be redefined. Typing
```

. stepwise, lock pr(.2): logistic outcome (x1 x2) (x3 x4) (x5 x6 x7)

```
would automatically drop e(sample) and re-create it.

\section*{Messages}
note: \(\qquad\) omitted because of estimability.
This indicates that a variable was omitted because its coefficient could not be estimated. This message is most commonly issued because the variable is collinear with other variables in the model. For instance, say that you type
```

. stepwise, pr(.2): regress y x1 x2 x3 x4

```
and \(x 2\) is collinear with \(x 3\), one of these variables will automatically be omitted. If you type
```

stepwise, pr(.2): regress y x1 x2 i.a

```
and include indicators for factor variable a in your model, the set of indicators for a are perfectly collinear, and one will be omitted with the note indicating that it was omitted because of estimability.
note: \(\qquad\) omitted because of estimability. note: \(\qquad\) obs omitted because of estimability.
You probably received this message in fitting a logistic or probit model. Regardless of estimation strategy, stepwise checks that the full model can be fit. The indicated variable had a 0 or infinite standard error.

For logistic, logit, and probit, this message is typically caused by one-way causation. Assume that you type
```

. stepwise, pr(.2): logistic outcome (x1 x2 x3) d1

```
and assume that variable d 1 is an indicator (dummy) variable. Further assume that whenever \(\mathrm{d} 1=1\), outcome \(=1\) in the data. Then the coefficient on d 1 is infinite. One (conservative) solution to this problem is to drop the d 1 variable and the \(\mathrm{d} 1==1\) observations. The underlying estimation commands probit, logit, and logistic report the details of the difficulty and solution; stepwise simply accumulates such problems and reports the above summary messages. Thus, if you see this message, you could type
```

. logistic outcome x1 x2 x3 d1

```
to see the details. Although you should think carefully about such situations, Stata's solution of dropping the offending variables and observations is, in general, appropriate.

\section*{Programming for stepwise}
stepwise requires that command_name follow standard Stata syntax and allow the if qualifier; see [U] 11 Language syntax. Furthermore, command_name must have sw or swml as a program property; see [P] program properties. If command_name has swml as a property, command_name must store the log-likelihood value in \(e(11)\) and model degrees of freedom in \(e\left(d f \_m\right)\).

\section*{Stored results}
stepwise stores whatever is stored by the underlying estimation command.
Also, stepwise stores stepwise in e(stepwise).

\section*{Methods and formulas}

Some statisticians do not recommend stepwise procedures; see Sribney (1998) for a summary.

\section*{References}

Afifi, A. A., S. May, R. A. Donatello, and V. A. Clark. 2020. Practical Multivariate Analysis. 6th ed. Boca Raton, FL: CRC Press.
Beale, E. M. L. 1970. Note on procedures for variable selection in multiple regression. Technometrics 12: 909-914. https://doi.org/10.2307/1267335.

Bendel, R. B., and A. A. Afifi. 1977. Comparison of stopping rules in forward "stepwise" regression. Journal of the American Statistical Association 72: 46-53. https://doi.org/10.2307/2286904.
Berk, K. N. 1978. Comparing subset regression procedures. Technometrics 20: 1-6. https://doi.org/10.2307/1268153.
Draper, N., and H. Smith. 1998. Applied Regression Analysis. 3rd ed. New York: Wiley.
Efroymson, M. A. 1960. Multiple regression analysis. In Mathematical Methods for Digital Computers, ed. A. Ralston and H. S. Wilf, 191-203. New York: Wiley.

Gorman, J. W., and R. J. Toman. 1966. Selection of variables for fitting equations to data. Technometrics 8: 27-51. https://doi.org/10.2307/1266260.

Hocking, R. R. 1976. The analysis and selection of variables in linear regression. Biometrics 32: 1-49. https://doi.org/10.2307/2529336.
Hosmer, D. W., Jr., S. A. Lemeshow, and R. X. Sturdivant. 2013. Applied Logistic Regression. 3rd ed. Hoboken, NJ: Wiley.
Kennedy, W. J., Jr., and T. A. Bancroft. 1971. Model-building for prediction in regression based on repeated significance tests. Annals of Mathematical Statistics 42: 1273-1284. https://doi.org/10.1214/aoms/1177693240.
Lindsey, C., and S. J. Sheather. 2010. Variable selection in linear regression. Stata Journal 10: 650-669.
Luchman, J. N. 2021. Determining relative importance in Stata using dominance analysis: domin and domme. Stata Journal 21: 510-538.
Mantel, N. 1970. Why stepdown procedures in variable selection. Technometrics 12: 621-625. https://doi.org/10.2307/1267207.
- 1971. More on variable selection and an alternative approach (letter to the editor). Technometrics 13: 455-457. https://doi.org/10.2307/1266815.
Sribney, W. M. 1998. FAQ: What are some of the problems with stepwise regression? https://www.stata.com/support/faqs/stat/stepwise.html.

Williams, R. 2007. Stata tip 46: Step we gaily, on we go. Stata Journal 7: 272-274.

\section*{Also see}
\([\mathrm{R}]\) nestreg - Nested model statistics
[LASSO] Lasso intro - Introduction to lasso

\section*{Title}

Stored results — Stored results

Description Syntax Option Remarks and examples References Also see

\section*{Description}

Results of calculations are stored by many Stata commands so that they can be easily accessed and substituted into later commands.
return list lists results stored in \(r()\).
ereturn list lists results stored in e().
sreturn list lists results stored in s() .
This entry discusses using stored results. Programmers wishing to store results should see \([\mathrm{P}]\) return and \([\mathrm{P}]\) ereturn.

\section*{Syntax}

List results from general commands, stored in \(r()\)
```

return list [, all]

```

List results from estimation commands, stored in e()
```

ereturn list [, all]

```

List results from parsing commands, stored in \(s()\)
sreturn list

\section*{Option}
all is for use with return list and ereturn list. all specifies that hidden and historical stored results be listed along with the usual stored results. This option is seldom used. See Using hidden and historical stored results and Programming hidden and historical stored results under Remarks and examples of \([\mathrm{P}]\) return for more information. These sections are written in terms of return list, but everything said there applies equally to ereturn list.
all is not allowed with sreturn list because \(s()\) does not allow hidden or historical results.

\section*{Remarks and examples}

Stata commands are classified as being
\[
\begin{array}{ll}
\mathrm{r} \text {-class } & \text { general commands that store results in } \mathrm{r}() \\
\text { e-class } & \text { estimation commands that store results in } \mathrm{e}() \\
\mathrm{s} \text {-class } & \text { parsing commands that store results in } \mathrm{s}() \\
\mathrm{n} \text {-class } & \text { commands that do not store in } \mathrm{r}(), \mathrm{e}(), \text { or } \mathrm{s}()
\end{array}
\]

There is also a c-class, c() , containing the values of system parameters and settings, along with certain constants, such as the value of pi; see [ P\(]\) creturn. A program, however, cannot be c-class.

You can look at the Stored results section of the manual entry of a command to determine whether it is \(\mathrm{r}-\), \(\mathrm{e}-\), s -, or n -class, but it is easy enough to guess.

Commands producing statistical results are either r-class or e-class. They are e-class if they present estimation results and r-class otherwise. s-class is a class used by programmers and is primarily used in subprograms performing parsing. n-class commands explicitly state where the result is to go. For instance, generate and replace are n-class because their syntax is generate varname \(=\ldots\) and replace varname = ....

After executing a command, you can type return list, ereturn list, or sreturn list to see what has been stored.

\section*{> Example 1}


Sorted by:
```

. return list

```
scalars:
            \(r(\) changed \()=0\)
                \(r(\) width \()=25\)
                        \(r(k)=6\)
                        \(r(N)=74\)
macros:
```

            r(datalabel) : "1978 automobile data"
    ```

To view all stored results, including those that are historical or hidden, specify the all option.
```

. return list, all
scalars:

$$
\begin{aligned}
r(\text { changed }) & =0 \\
r(\text { width }) & =25 \\
r(k) & =6 \\
r(N) & =74
\end{aligned}
$$

```
macros:
r(datalabel) : "1978 automobile data"

Historical; used before Stata 12, may exist only under version control
scalars:
\[
r(\text { widthmax })=1048576
\]
\(r\left(k_{\text {_max }}\right)=5000\)
\(r\left(N_{\text {_max }}\right)=2147483619\)
\(r\) (widthmax), \(r\left(k \_\max \right)\), and \(r\left(N \_\max \right)\) are historical stored results. They are no longer relevant because Stata dynamically adjusts memory beginning with Stata 12.

\section*{Technical note}

In the above example, we stated that \(r\) (widthmax) and \(r\) ( \(N \_\)max) are no longer relevant. In fact, they are not useful. Stata no longer has a fixed memory size, so the methods used to calculate \(r\) (widthmax) and \(r\) ( \(N \_\)max ) are no longer appropriate.

\section*{> Example 2}

You can use stored results in expressions.


Be careful to use results stored in \(r()\) soon because they will be replaced the next time you execute another r-class command. For instance, although r (mean) was 21.3 (approximately) after summarize mpg , it is \(-1.64 \mathrm{e}-16\) now because you just ran summarize with mpgstd.

\section*{> Example 3}
e-class is really no different from r-class, except for where results are stored and that, when an estimation command stores results, it tends to store a lot of them:
```

. regress mpg weight length
(output omitted)
. ereturn list
scalars:
e(N) = 74
e(df_m) = 2
e(df_r) = 71
e(F) = 69.34050004300227
e(r2) = .6613903979336323
e(rmse) = 3.413681741382589
e(mss) = 1616.08062422659
e(rss) = 827.3788352328695
e(r2_a) = .6518520992838754
e(ll) = -194.3267619410807
e(ll_0) = -234.3943376482347
e}(\mathrm{ rank) = 3
macros:
e(cmdline) : "regress mpg weight length"
e(title) : "Linear regression"
e(marginsok) : "XB default"
e(vce) : "ols"
e(depvar) : "mpg"
e(cmd) : "regress"
e(properties) : "b V"
e(predict) : "regres_p"
e(model) : "ols"
e(estat_cmd) : "regress_estat"
matrices:
e(b) : 1 x 3
e(V) : 3 x 3
e(beta) : 1 x 2
functions:
e(sample)

```

These e-class results will stick around until you run another estimation command. Typing return list and ereturn list is the easy way to find out what a command stores.

Both r- and e-class results come in four types: scalars, macros, matrices, and functions. (s-class results come in only one type-macros-and as earlier noted, s-class is used solely by programmers, so ignore it.)

Scalars are just that—numbers by any other name. You can subsequently refer to \(r\) (mean) or \(e\) (rmse) in numeric expressions and obtain the result to full precision.

Macros are strings. For instance, e (depvar) contains "mpg". You can refer to it, too, in subsequent expressions, but really that would be of most use to programmers, who will refer to it using constructs like "'e(depvar)'". In any case, macros are macros, and you obtain their contents just as you would a local macro, by enclosing their name in single quotes. The name here is the full name, so 'e(depvar)' is mpg.

Matrices are matrices, and all estimation commands store e(b) and e(V) containing the coefficient vector and variance-covariance matrix of the estimates (VCE).

Functions are stored by e-class commands only, and the only function existing is e(sample). \(e\) (sample) evaluates to 1 (meaning true) if the observation was used in the previous estimation and to 0 (meaning false) otherwise.

\section*{Technical note}

Say that some command set \(r\) (scalar) and \(r\) (macro), the first being stored as a scalar and the second as a macro. In theory, in subsequent use you are supposed to refer to \(r\) (scalar) and ' \(r\) (macro)'. In fact, however, you can refer to either one with or without quotes, so you could refer to 'r(scalar)' and \(r\) (macro). Programmers sometimes do this.

When you refer to \(r\) (scalar), you are referring to the full double-precision stored result. Think of \(r\) (scalar) without quotes as a function returning the value of the stored result scalar. When you refer to \(r\) (scalar) in quotes, Stata understands ' \(r\) (scalar)' to mean "substitute the printed result of evaluating \(r\) (scalar)". Pretend that \(r\) (scalar) equals the number 23. Then, ' \(r\) (scalar)' is 23 , the character 2 followed by 3.

Referring to \(r\) (scalar) in quotes is sometimes useful. Say that you want to use the immediate command cii with r(scalar). The immediate command cii requires its arguments to be numbersnumeric literals in programmer's jargon-and it will not take an expression. Thus, you could not type 'cii r(scalar) ...'. You could, however, type 'cii 'r(scalar)' ...' because 'r(scalar)' is just a numeric literal.

For \(r\) (macro), you are supposed to refer to it in quotes: ' \(r\) (macro)'. If, however, you omit the quotes in an expression context, Stata evaluates the macro and then pretends that it is the result of function-returning-string. There are side effects of this, the most important being that the result is trimmed to 80 characters.

Referring to \(r\) (macro) without quotes is never a good idea; the feature was included merely for completeness.

You can even refer to \(r\) (matrix) in quotes (assume that \(r\) (matrix) is a matrix). ' \(r\) (matrix)' does not result in the matrix being substituted; it returns the word matrix. Programmers sometimes find that useful.

\section*{References}

Jann, B. 2005. Making regression tables from stored estimates. Stata Journal 5: 288-308.
__. 2007. Making regression tables simplified. Stata Journal 7: 227-244.

\section*{Also see}
[P] ereturn - Post the estimation results
[P] return - Return stored results
[U] 18.8 Accessing results calculated by other programs
[U] 18.9 Accessing results calculated by estimation commands

\section*{Title}
suest - Seemingly unrelated estimation
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
Acknowledgment & References & Also see &
\end{tabular}

\section*{Description}
suest is a postestimation command; see [U] 20 Estimation and postestimation commands.
suest combines the estimation results-parameter estimates and associated (co)variance matricesstored under namelist into one parameter vector and simultaneous (co)variance matrix of the sandwich/robust type. This (co)variance matrix is appropriate even if the estimates were obtained on the same or on overlapping data.

Typical applications of suest are tests for intramodel and cross-model hypotheses using test or testnl, for example, a generalized Hausman specification test. lincom and nlcom may be used after suest to estimate linear combinations and nonlinear functions of coefficients. suest may also be used to adjust a standard VCE for clustering or survey design effects.

Different estimators are allowed, for example, a regress model and a probit model; the only requirement is that predict produce equation-level scores with the score option after an estimation command. The models may be estimated on different samples, due either to explicit if or in selection or to missing values. If weights are applied, the same weights (type and values) should be applied to all models in namelist. The estimators should be estimated without vce (robust) or vce (cluster clustvar) options. suest returns the robust VCE, allows the vce(cluster clustvar) option, and automatically works with results from the svy prefix command (only for vce(linearized)). See example 8 in [SVY] svy postestimation for an example using suest with svy: ologit.

Because suest posts its results like a proper estimation command, its results can be stored via estimates store. Moreover, like other estimation commands, suest typed without arguments replays the results.

\section*{Quick start}

Combined results for stored estimates m1 and m2
suest m1 m2
Same as above, but report exponentiated coefficients and label them "Odds ratio"
```

suest m1 m2, eform(Odds ratio)

```

With cluster-robust standard errors adjusting for clustering by levels of cvar suest m1 m2, vce(cluster cvar)

Use svyset data after specifying command prefix svy: to estimate m1 and m2 suest m1 m2

\section*{Menu}

Statistics \(>\) Postestimation

\section*{Syntax}
suest namelist [, options]
where namelist is a list of one or more names under which estimation results were stored via estimates store; see [R] estimates store. Wildcards may be used. * and _all refer to all stored results. A period (.) may be used to refer to the last estimation results, even if they have not (yet) been stored.
options
Description
SE/Robust
svy
vce (vcetype)
Reporting
level(\#)
dir
eform(string)
display_options
coeflegend
survey data estimation
vcetype may be robust or cluster clustvar
collect is allowed; see [U] 11.1.10 Prefix commands.
coeflegend does not appear in the dialog box.

\section*{Options}

SE/Robust
svy specifies that estimation results should be modified to reflect the survey design effects according to the svyset specifications, see [SVY] svyset.

The svy option is implied when suest encounters survey estimation results from the svy prefix; see [SVY] svy. Poststratification is allowed only with survey estimation results from the svy prefix.
vce(vcetype) specifies the type of standard error reported, which includes types that are robust to some kinds of misspecification (robust) and that allow for intragroup correlation (cluster clustvar); see [R] vce_option.

The vce() option may not be combined with the svy option or estimation results from the svy prefix.
level(\#) specifies the confidence level, as a percentage, for confidence intervals of the coefficients; see [ R ] level.
dir displays a table describing the models in namelist just like estimates dir namelist.
eform (string) displays the coefficient table in exponentiated form: for each coefficient, \(\exp (b)\) rather than \(b\) is displayed, and standard errors and confidence intervals are transformed. string is the table header that will be displayed above the transformed coefficients and must be 11 characters or fewer, for example, eform("Odds ratio").
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fmt), and nolstretch; see [R] Estimation options.

The following option is available with suest but is not shown in the dialog box:
coeflegend; see [R] Estimation options.

\section*{Remarks and examples}

Remarks are presented under the following headings:
```

Using suest
Remarks on regress
Testing the assumption of the independence of irrelevant alternatives
Testing proportionality
Testing cross-model hypotheses

```

\section*{Using suest}

If you plan to use suest, you must take precautions when fitting the original models. These restrictions are relaxed when using svy commands; see [SVY] svy postestimation.
1. suest works with estimation commands that allow predict to generate equation-level score variables when supplied with the score (or scores) option. For example, equation-level score variables are generated after running mlogit by typing
```

. predict sc*, scores

```

To ensure that suest generates the appropriate scores and therefore an appropriate VCE, you should not modify the values of the variables used in the models in between fitting the models and using suest.
2. Estimation should take place without the vce (robust) or vce(cluster clustvar) option. suest always computes the robust estimator of the (co)variance, and suest has a vce(cluster clustvar) option.

The within-model covariance matrices computed by suest are identical to those obtained by specifying a vce(robust) or vce(cluster clustvar) option during estimation. suest, however, also estimates the between-model covariances of parameter estimates.
3. Finally, the estimation results to be combined should be stored by estimates store; see [R] estimates store.

After estimating and storing a series of estimation results, you are ready to combine the estimation results with suest,
and you can subsequently use postestimation commands, such as test, to test hypotheses. Here an important issue is how suest assigns names to the equations. If you specify one model name, the original equation names are left unchanged; otherwise, suest constructs new equation names. The coefficients of a single-equation model (such as logit and poisson) that was estimate stored under name \(X\) are collected under equation \(X\). With a multiequation model stored under name \(X\), suest prefixes \(X_{-}\)to an original equation name eq, forming equation name, \(X_{-}\)eq.

\section*{- Technical note}

Earlier we said that standard errors from suest are identical to those obtained by specifying the vce (robust) option with each command individually. Thus if you fit a logistic model using logit with the vce(robust) option, you will get the same standard errors when you type
. suest .
directly after logit using the same data without the vce (robust) option.
This is not true for multiple estimation results when the estimation samples are not all the same. The standard errors from suest will be slightly smaller than those from individual model fits using the vce (robust) option because suest uses a larger number of observations to estimate the simultaneous (co)variance matrix.

\section*{- Technical note}

In rare circumstances, suest may have to truncate equation names to 32 characters. When equation names are not unique because of truncation, suest numbers the equations within models, using equations named \(X \_\#\).

\section*{Remarks on regress}
regress (see [R] regress) does not include its ancillary parameter, the residual variance, in its coefficient vector and (co)variance matrix. Moreover, while the score option is allowed with predict after regress, a score variable is generated for the mean but not for the variance parameter. suest contains special code that assigns the equation name mean to the coefficients for the mean, adds the equation Invar for the log variance, and computes the appropriate two score variables itself.

\section*{Testing the assumption of the independence of irrelevant alternatives}

The multinomial logit model and the closely related conditional logit model satisfy a probabilistic version of the assumption of the independence of irrelevant alternatives (IIA), implying that the ratio of the probabilities for two alternatives does not depend on what other alternatives are available. Hausman and McFadden (1984) proposed a test for this assumption that is implemented in the hausman command. The standard Hausman test has several limitations. First, the test statistic may be undefined because the estimated VCE does not satisfy the required asymptotic properties of the test. Second, the classic Hausman test applies only to the test of the equality of two estimators. Third, the test requires access to a fully efficient estimator; such an estimator may not be available, for example, if you are analyzing complex survey data. Using suest can overcome these three limitations.

\section*{Example 1}

In our first example, we follow the analysis of the type of health insurance reported in [R] mlogit and demonstrate the hausman command with the suest/test combination. We fit the full multinomial logit model for all three alternatives and two restricted multinomial models in which one alternative is excluded. After fitting each of these models, we give them a title with the estimates title command and then store them with the estimates store command.
```

. use https://www.stata-press.com/data/r18/sysdsn4
(Health insurance data)
. mlogit insure age male
Iteration 0: Log likelihood = -555.85446
Iteration 1: Log likelihood = -551.32973
Iteration 2: Log likelihood = -551.32802
Iteration 3: Log likelihood = -551.32802

```

\begin{tabular}{r|rlrlrl}
\hline insure & Coefficient & Std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline Indemnity & (base outcome) & & & & \\
\hline Prepaid & & & & & & \\
age & -.0100251 & .0060181 & -1.67 & 0.096 & -.0218204 & .0017702 \\
male & .5095747 & .1977893 & 2.58 & 0.010 & .1219147 & .8972346 \\
_cons & .2633838 & .2787575 & 0.94 & 0.345 & -.2829708 & .8097383 \\
\hline Uninsure & & & & & & \\
age & -.0051925 & .0113821 & -0.46 & 0.648 & -.0275011 & .0171161 \\
male & .4748547 & .3618462 & 1.31 & 0.189 & -.2343508 & 1.18406 \\
_cons & -1.756843 & .5309602 & -3.31 & 0.001 & -2.797506 & -.7161803 \\
\hline
\end{tabular}
```

. estimates title: All three insurance forms
. estimates store m1
. quietly mlogit insure age male if insure != "Uninsure":insure
. estimates title: insure != "Uninsure":insure
. estimates store m2
. quietly mlogit insure age male if insure != "Prepaid":insure
. estimates title: insure != "Prepaid":insure
. estimates store m3

```

Having performed the three estimations, we inspect the results. estimates dir provides short descriptions of the models that were stored using estimates store. Typing estimates table lists the coefficients, displaying blanks for a coefficient not contained in a model.
\begin{tabular}{|c|c|c|c|c|}
\hline Name & Command & Dependent variable & Number of param. & Title \\
\hline m1 & mlogit & insure & 9 & All three insurance forms \\
\hline m2 & mlogit & insure & 6 & insure != "Uninsure":insure \\
\hline m3 & mlogit & insure & 6 & insure != "Prepaid":insure \\
\hline
\end{tabular}
. estimates table m1 m2 m3, star stats(N ll) keep(Prepaid: Uninsure:)
\begin{tabular}{|c|c|c|c|}
\hline Variable & m1 & m2 & m3 \\
\hline \multicolumn{4}{|l|}{Prepaid} \\
\hline age & -. 01002511 & -. 01015205 & \\
\hline male & .50957468** & .51440033** & \\
\hline _cons & . 26338378 & . 26780432 & \\
\hline \multicolumn{4}{|l|}{Uninsure} \\
\hline age & -. 00519249 & & -. 00410547 \\
\hline male & . 47485472 & & . 45910738 \\
\hline _cons & -1.7568431*** & & -1.8017743*** \\
\hline \multicolumn{4}{|l|}{Statistics} \\
\hline N & 615 & 570 & 338 \\
\hline 11 & -551.32802 & -390.48643 & -131.76807 \\
\hline
\end{tabular}

Comparing the coefficients between models does not suggest substantial differences. We can formally test that coefficients are the same for the full model m 1 and the restricted models m 2 and m 3 by using the hausman command. hausman expects the models to be specified in the order "always consistent" first and "efficient under \(H_{0}\) " second.
. hausman m2 m1, alleqs constant
—— Coefficients -_
\begin{tabular}{r|cccc} 
& \begin{tabular}{c} 
(b) \\
m 2
\end{tabular} & \begin{tabular}{c}
\((\mathrm{B})\) \\
m 1
\end{tabular} & \begin{tabular}{c}
\((\mathrm{b}-\mathrm{B})\) \\
Difference
\end{tabular} & \begin{tabular}{c} 
sqrt(diag(V_b-V_B)) \\
Std.err.
\end{tabular} \\
\hline age & -.0101521 & -.0100251 & -.0001269 & \\
male & .5144003 & .5095747 & .0048256 & .0123338 \\
_cons & .2678043 & .2633838 & .0044205 &. \\
\hline
\end{tabular}
\(\mathrm{b}=\) Consistent under HO and Ha ; obtained from mlogit.
B = Inconsistent under Ha, efficient under HO; obtained from mlogit.
Test of HO: Difference in coefficients not systematic
\(\operatorname{chi2}(3)=(b-B) \cdot\left[\left(V \_b-V \_B\right)^{\wedge}(-1)\right](b-B)\)
\(=0.08\)
Prob > chi2 \(=0.9944\)
(V_b-V_B is not positive definite)
. hausman m3 m1, alleqs constant
—_ Coefficients __
\begin{tabular}{r|ccrc} 
& \begin{tabular}{c} 
(b) \\
m 3
\end{tabular} & \multicolumn{1}{c}{\begin{tabular}{c}
\((\mathrm{B})\) \\
m 1
\end{tabular}} & \begin{tabular}{c}
\((\mathrm{b}-\mathrm{B})\) \\
Difference
\end{tabular} & \begin{tabular}{c} 
sqrt(diag(V_b-V_B)) \\
Std.err.
\end{tabular} \\
\hline age & -.0041055 & -.0051925 & .001087 & .0021355 \\
male & .4591074 & .4748547 & -.0157473 & \\
_cons & -1.801774 & -1.756843 & -.0449311 & .1333421 \\
\hline
\end{tabular}

\footnotetext{
b = Consistent under HO and Ha; obtained from mlogit.
B = Inconsistent under Ha, efficient under HO; obtained from mlogit.
Test of HO: Difference in coefficients not systematic
\(\operatorname{chi2}(3)=(b-B)^{\prime}\left[\left(V \_b-V \_B\right)^{\wedge}(-1)\right](b-B)\)
\(=-0.18\)
Warning: chi2 < 0 ==> model fitted on these data
fails to meet the asymptotic assumptions
of the Hausman test; see suest for a
generalized test.
}

According to the test of \(m 1\) against \(m 2\), we cannot reject the hypothesis that the coefficients of \(m 1\) and m 2 are the same. The second Hausman test is not well defined-something that happens fairly often. The problem is due to the estimator of the variance \(V(b-B)\) as \(V(b)-V(B)\), which is a feasible estimator only asymptotically. Here it simply is not a proper variance matrix, and the Hausman test becomes undefined.
suest m 1 m 2 estimates the simultaneous (co)variance of the coefficients of models m 1 and m 2 . Although suest is technically a postestimation command, it acts like an estimation command in that it stores the simultaneous coefficients in \(e(b)\) and the full (co)variance matrix in \(e(V)\). We could have used the estat vce command to display the full (co)variance matrix to show that the cross-model covariances were indeed estimated. Typically, we would not have a direct interest in e(V).
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline \multicolumn{5}{|l|}{Simultaneous results for m1, m2} & \multicolumn{2}{|l|}{Number of obs \(=615\)} \\
\hline & Coefficient & Robust std. err. & z & \(P>|z|\) & [95\% conf & interval] \\
\hline \multicolumn{7}{|l|}{m1_Indemnity} \\
\hline \multicolumn{7}{|l|}{m1_Prepaid} \\
\hline age & -. 0100251 & . 0059403 & -1.69 & 0.091 & -. 0216679 & . 0016176 \\
\hline male & . 5095747 & . 1988159 & 2.56 & 0.010 & . 1199027 & . 8992467 \\
\hline _cons & . 2633838 & . 277307 & 0.95 & 0.342 & -. 280128 & . 8068956 \\
\hline \multicolumn{7}{|l|}{m1_Uninsure} \\
\hline age & -. 0051925 & . 0109005 & -0.48 & 0.634 & -. 0265571 & . 0161721 \\
\hline male & . 4748547 & . 3677326 & 1.29 & 0.197 & -. 2458879 & 1.195597 \\
\hline _cons & -1.756843 & . 4971383 & -3.53 & 0.000 & -2.731216 & -. 78247 \\
\hline \multicolumn{7}{|l|}{m2_Indemnity} \\
\hline \multicolumn{7}{|l|}{m2_Prepaid} \\
\hline age & -. 0101521 & . 0058988 & -1.72 & 0.085 & -. 0217135 & . 0014094 \\
\hline male & . 5144003 & . 1996133 & 2.58 & 0.010 & . 1231654 & . 9056352 \\
\hline _cons & . 2678043 & . 2744019 & 0.98 & 0.329 & -. 2700134 & . 8056221 \\
\hline
\end{tabular}
suest created equation names by combining the name under which we stored the results using estimates store with the original equation names. Thus, in the simultaneous estimation result, equation Prepaid originating in model m 1 is named m1_Prepaid. According to the McFaddenHausman specification of a test for IIA, the coefficients of the equations m1_PrePaid and m2_PrePaid should be equal. This equality can be tested easily with the test command. The cons option specifies that the intercept _cons be included in the test.
```

. test [m1_Prepaid = m2_Prepaid], cons
( 1) [m1_Prepaid]age - [m2_Prepaid]age = 0
( 2) [m1_Prepaid]male - [m2_Prepaid]male = 0
( 3) [m1_Prepaid]_cons - [m2_Prepaid]_cons = 0
chi2( 3) = 0.89
Prob > chi2 = 0.8266

```

The Hausman test via suest is comparable with that computed by hausman, but they use different estimators of the variance of the difference of the estimates. The hausman command estimates \(V(b-B)\) by \(V(b)-V(B)\), whereas suest estimates \(V(b-B)\) by \(V(b)-\operatorname{cov}(b, B)-\operatorname{cov}(B, b)+V(B)\). One advantage of the second estimator is that it is always admissible, so the resulting test is always well defined. This quality is illustrated in the Hausman-type test of IIA comparing models m1 and m3.
. suest m1 m3, noomitted
Simultaneous results for m1, m3 Number of obs \(=615\)
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline & Coefficient & Robust std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% con & interval] \\
\hline \multicolumn{7}{|l|}{m1_Indemnity} \\
\hline \multicolumn{7}{|l|}{m1_Prepaid} \\
\hline age & -. 0100251 & . 0059403 & -1.69 & 0.091 & -. 0216679 & . 0016176 \\
\hline male & . 5095747 & . 1988159 & 2.56 & 0.010 & . 1199027 & . 8992467 \\
\hline _cons & . 2633838 & . 277307 & 0.95 & 0.342 & -. 280128 & . 8068956 \\
\hline \multicolumn{7}{|l|}{m1_Uninsure} \\
\hline age & -. 0051925 & . 0109005 & -0.48 & 0.634 & -. 0265571 & . 0161721 \\
\hline male & . 4748547 & . 3677326 & 1.29 & 0.197 & -. 2458879 & 1.195597 \\
\hline _cons & -1.756843 & . 4971383 & -3.53 & 0.000 & -2.731216 & -. 78247 \\
\hline \multicolumn{7}{|l|}{m3_Indemnity} \\
\hline \multicolumn{7}{|l|}{m3_Uninsure} \\
\hline & -. 0041055 & . 0111185 & -0.37 & 0.712 & -. 0258974 & . 0176865 \\
\hline male & . 4591074 & . 3601307 & 1.27 & 0.202 & -. 2467357 & 1.164951 \\
\hline _cons & -1.801774 & . 5226351 & -3.45 & 0.001 & -2.82612 & -. 7774283 \\
\hline
\end{tabular}
```

. test [m1_Uninsure = m3_Uninsure], cons
( 1) [m1_Uninsure]age - [m3_Uninsure]age = 0
( 2) [m1_Uninsure]male - [m3_Uninsure]male = 0
( 3) [m1_Uninsure]_cons - [m3_Uninsure]_cons = 0
chi2( 3) = 1.49
Prob > chi2 = 0.6845

```

Although the classic Hausman test computed by hausman is not defined here, the suest-based test is just fine. We cannot reject the equality of the common coefficients across m 1 and m 3 .

A second advantage of the suest approach is that we can estimate the (co)variance matrix of the multivariate normal distribution of the estimators of the three models m1, m2, and m3 and test that the common coefficients are equal.
. suest m*, noomitted
Simultaneous results for \(\mathrm{m} 1, \mathrm{~m} 2, \mathrm{~m} 3 \quad\) Number of obs \(=615\)
\begin{tabular}{r|rrrrrr}
\hline & & Robust \\
& Coefficient \\
std. err. & z & P>|z| & [95\% conf. interval] \\
\hline m1_Indemnity & & & & & & \\
\hline m1_Prepaid & & & & & & \\
age & -.0100251 & .0059403 & -1.69 & 0.091 & -.0216679 & .0016176 \\
male & .5095747 & .1988159 & 2.56 & 0.010 & .1199027 & .8992467 \\
_cons & .2633838 & .277307 & 0.95 & 0.342 & -.280128 & .8068956 \\
\hline m1_Uninsure & & & & & & \\
age & -.0051925 & .0109005 & -0.48 & 0.634 & -.0265571 & .0161721 \\
male & .4748547 & .3677326 & 1.29 & 0.197 & -.2458879 & 1.195597 \\
_cons & -1.756843 & .4971383 & -3.53 & 0.000 & -2.731216 & -.78247 \\
\hline m2_Indemnity & & & & & & \\
\hline m2_Prepaid & & & & & & \\
age & -.0101521 & .0058988 & -1.72 & 0.085 & -.0217135 & .0014094 \\
male & .5144003 & .1996133 & 2.58 & 0.010 & .1231654 & .9056352 \\
_cons & .2678043 & .2744019 & 0.98 & 0.329 & -.2700134 & .8056221 \\
\hline m3_Indemnity & & & & & & \\
\hline m3_Uninsure & & & & & & \\
age & -.0041055 & .011185 & -0.37 & 0.712 & -.0258974 & .0176865 \\
male & .4591074 & .3601307 & 1.27 & 0.202 & -.2467357 & 1.164951 \\
_cons & -1.801774 & .5226351 & -3.45 & 0.001 & -2.82612 & -.7774283 \\
\hline
\end{tabular}
```

. test [m1_Prepaid = m2_Prepaid] , cons notest
( 1) [m1_Prepaid]age - [m2_Prepaid]age $=0$
( 2) [m1_Prepaid]male - [m2_Prepaid]male $=0$
( 3) [m1_Prepaid]_cons - [m2_Prepaid]_cons $=0$
. test [m1_Uninsure $=m 3 \_$Uninsure], cons acc
( 1) [m1_Prepaid]age - [m2_Prepaid]age $=0$
( 2) [m1_Prepaid]male - [m2_Prepaid]male $=0$
( 3) [m1_Prepaid]_cons - [m2_Prepaid]_cons $=0$
( 4) [m1_Uninsure]age - [m3_Uninsure]age $=0$
( 5) [m1_Uninsure]male - [m3_Uninsure]male $=0$
( 6) [m1_Uninsure]_cons - [m3_Uninsure]_cons $=0$
$\operatorname{chi2}(6)=1.95$
Prob $>$ chi2 $=0.9240$

```

Again, we do not find evidence against the correct specification of the multinomial logit for type of insurance. The classic Hausman test assumes that one of the estimators (named B in hausman) is efficient, that is, it has minimal (asymptotic) variance. This assumption ensures that \(V(b)-V(B)\) is an admissible, viable estimator for \(V(b-B)\). The assumption that we have an efficient estimator is a restrictive one. It is violated, for instance, if our data are clustered. We want to adjust for clustering via a vce (cluster clustvar) option by requesting the cluster-adjusted sandwich estimator of variance. Consequently, in such a case, hausman cannot be used. This problem does not exist with the suest version of the Hausman test. To illustrate this feature, we suppose that the data are clustered by city-we constructed an imaginary variable cityid for this illustration. If we plan to apply suest, we would not specify the vce(cluster clustvar) option the time of estimation. suest has a vce(cluster clustvar) option. Thus, we do not need to refit the models; we can call suest and test right away.
. suest m1 m2, vce(cluster cityid) noomitted
Simultaneous results for m 1 , \(\mathrm{m} 2 \quad\) Number of obs \(=615\)
(Std. err. adjusted for 260 clusters in cityid)
\begin{tabular}{c|rcrrrr}
\hline & Robust & & & \\
& Coefficient \\
std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline m1_Indemnity & & & & & & \\
\hline m1_Prepaid & & & & & & \\
age & -.0100251 & .005729 & -1.75 & 0.080 & -.0212538 & .0012035 \\
male & .5095747 & .1910496 & 2.67 & 0.008 & .1351244 & .884025 \\
_cons & .2633838 & .2698797 & 0.98 & 0.329 & -.2655708 & .7923384 \\
\hline m1_Uninsure & & & & & & \\
age & -.0051925 & .0104374 & -0.50 & 0.619 & -.0256495 & .0152645 \\
male & .4748547 & .3774021 & 1.26 & 0.208 & -.2648399 & 1.214549 \\
_cons & -1.756843 & .4916613 & -3.57 & 0.000 & -2.720481 & -.7932048 \\
\hline m2_Indemnity & & & & & & \\
\hline m2_Prepaid & & & & & & \\
age & -.0101521 & .0057164 & -1.78 & 0.076 & -.0213559 & .0010518 \\
male & .5144003 & .1921385 & 2.68 & 0.007 & .1378158 & .8909848 \\
_cons & .2678043 & .2682193 & 1.00 & 0.318 & -.2578959 & .7935045 \\
\hline
\end{tabular}
```

. test [m1_Prepaid = m2_Prepaid], cons
( 1) [m1_Prepaid]age - [m2_Prepaid]age = 0
( 2) [m1_Prepaid]male - [m2_Prepaid]male = 0
( 3) [m1_Prepaid]_cons - [m2_Prepaid]_cons = 0
chi2( 3) = 0.79
Prob > chi2 = 0.8529

```
suest provides some descriptive information about the clustering on cityid. Like any other estimation command, suest informs us that the standard errors are adjusted for clustering. The Hausman-type test obtained from the test command uses a simultaneous (co)variance of m1 and m2 appropriately adjusted for clustering. In this example, we still do not have reason to conclude that the multinomial logit model in this application is misspecified, that is, that IIA is violated.

The multinomial logistic regression model is a special case of the conditional logistic regression model; see [R] clogit. Like the multinomial logistic regression model, the conditional logistic regression model also makes the IIA assumption. Consider an example, introduced in [CM] cmclogit, in which the demand for American, Japanese, and European cars is modeled in terms of the number of local dealers of the respective brands and of some individual attributes incorporated in interaction with the nationality of cars. We want to perform a Hausman-type test for IIA comparing the decision between all nationalities with the decision between non-American cars. The following code fragment demonstrates how to conduct a Hausman test for IIA via suest in this case.
```

. clogit choice japan europe maleJap maleEur incJap incEur dealer, group(id)
. estimates store allcars
. clogit choice japan maleJap incJap dealer if car!=1 , group(id)
. estimates store foreign
. suest allcars foreign
. test [allcars_choice=foreign_choice], common

```

\section*{Testing proportionality}

The applications of suest that we have discussed so far concern Hausman-type tests for misspecification. To test such a hypothesis, we compared two estimators that have the same probability limit if the hypothesis holds true, but otherwise have different limits. We may also want to compare the coefficients of models (estimators) for other substantive reasons. Although we most often want to test whether coefficients differ between models or estimators, we may occasionally want to test other constraints (see Hausman and Ruud [1987]).
\(>\) Example 2
In this example, using simulated labor market data for siblings, we consider two dependent variables, income (inc) and whether a person was promoted in the last year (promo). We apply familiar economic arguments regarding human capital, according to which employees have a higher income and a higher probability of being promoted, by having more human capital. Human capital is acquired through formal education (edu) and on-the-job training experience (exp). We study whether income and promotion are "two sides of the same coin", that is, whether they reflect a common latent variable, "human capital". Accordingly, we want to compare the effects of different aspects of human capital on different outcome variables.

We estimate fairly simple labor market equations. The income model is estimated with regress, and the estimation results are stored under the name Inc.
```

. use https://www.stata-press.com/data/r18/income
. regress inc edu exp male

```
\begin{tabular}{r|crc} 
Source & SS & df & MS \\
\hline Model & 2058.44672 & 3 & 686.148908 \\
Residual & 4424.05183 & 273 & 16.2053181 \\
\hline Total & 6482.49855 & 276 & 23.4873136
\end{tabular}
\begin{tabular}{llr} 
Number of obs & \(=\) & 277 \\
F(3, 273) & \(=\) & 42.34 \\
Prob > F & \(=\) & 0.0000 \\
R-squared & \(=\) & 0.3175 \\
Adj R-squared & \(=\) & 0.3100 \\
Root MSE & \(=\) & 4.0256
\end{tabular}
\begin{tabular}{r|rccccc}
\hline inc & Coefficient & Std. err. & t & \(\mathrm{P}>|\mathrm{t}|\) & [95\% conf. interval] \\
\hline edu & 2.213707 & .243247 & 9.10 & 0.000 & 1.734828 & 2.692585 \\
exp & 1.47293 & .231044 & 6.38 & 0.000 & 1.018076 & 1.927785 \\
male & .5381153 & .4949466 & 1.09 & 0.278 & -.436282 & 1.512513 \\
_cons & 1.255497 & .3115808 & 4.03 & 0.000 & .642091 & 1.868904 \\
\hline
\end{tabular}
```

. est store Inc

```

Being sibling data, the observations are clustered on family of origin, famid. In the estimation of the regression parameters, we did not specify a vce (cluster famid) option to adjust standard errors for clustering on family (famid). Thus, the standard errors reported by regress are potentially flawed. This problem will, however, be corrected by specifying a vce(cluster clustvar) option with suest.

Next, we estimate the promotion equation with probit and again store the results under an appropriate name.
\begin{tabular}{|c|c|c|}
\hline \multirow[t]{3}{*}{Probit regression} & Number of obs & \(=277\) \\
\hline & LR chi2(3) & \(=49.76\) \\
\hline & Prob > chi2 & \(=0.0000\) \\
\hline Log likelihood = -158.43888 & Pseudo R2 & \(=0.1357\) \\
\hline
\end{tabular}
\begin{tabular}{r|rrrrrr}
\hline promo & Coefficient & Std. err. & z & P>|z| & [95\% conf. interval] \\
\hline edu & .4593002 & .0898537 & 5.11 & 0.000 & .2831901 & .6354102 \\
exp & .3593023 & .0805774 & 4.46 & 0.000 & .2013735 & .5172312 \\
male & .2079983 & .1656413 & 1.26 & 0.209 & -.1166527 & .5326494 \\
_cons & -.464622 & .1088166 & -4.27 & 0.000 & -.6778985 & -.2513454 \\
\hline
\end{tabular}
est store Promo

The coefficients in the income and promotion equations definitely seem to be different. However, because the scales of the two variables are different, we would not expect the coefficients to be equal. The correct hypothesis here is that the proportionality of the coefficients of the two models, apart from the constant, are equal. This formulation would still reflect that the relative effects of the different aspects of human capital do not differ between the dependent variables. We can obtain a nonlinear Wald test for the hypothesis of proportionality by using the testnl command on the combined estimation results of the two estimators. Thus, we first have to form the combined estimation results. At this point, we specify the vce(cluster famid) option to adjust for the clustering of observations on famid.
```

. suest Inc Promo, vce(cluster famid)
Simultaneous results for Inc, Promo Number of obs = 277

```
(Std. err. adjusted for 135 clusters in famid)
\begin{tabular}{r|rrrrrr}
\hline & \multicolumn{4}{|c}{\begin{tabular}{c} 
Robust \\
std. err.
\end{tabular}} & \(\mathbf{z}\) & \(\mathrm{P}>|\mathrm{z}|\) \\
& Coefficient & [95\% conf. interval] \\
\hline Inc_mean & & & & & & \\
edu & 2.213707 & .2483907 & 8.91 & 0.000 & 1.72687 & 2.700543 \\
exp & 1.47293 & .1890583 & 7.79 & 0.000 & 1.102383 & 1.843478 \\
male & .5381153 & .4979227 & 1.08 & 0.280 & -.4377952 & 1.514026 \\
_cons & 1.255497 & .3374977 & 3.72 & 0.000 & .594014 & 1.916981 \\
\hline Inc_lnvar & & & & & & \\
\hline Promo_promo & & & & & & \\
edu & .4593002 & .0886982 & 5.18 & 0.000 & .2854549 & .6331454 \\
exp & .3593023 & .079772 & 4.50 & 0.000 & .2029522 & .5156525 \\
male & .2079983 & .1691053 & 1.23 & 0.219 & -.1234419 & .5394386 \\
_cons & -.464622 & .1042169 & -4.46 & 0.000 & -.6688833 & -.2603607 \\
\hline
\end{tabular}

The standard errors reported by suest are identical to those reported by the respective estimation commands when invoked with the vce(cluster famid) option. We are now ready to test for proportionality:
\[
H_{0}: \frac{\beta_{\text {edu }}^{\text {Income }}}{\beta_{\text {edu }}^{\text {Promotion }}}=\frac{\beta_{\text {exp }}^{\text {Income }}}{\beta_{\text {exp }}^{\text {Promotion }}}=\frac{\beta_{\text {male }}^{\text {Income }}}{\beta_{\text {male }}^{\text {Protion }}}
\]

It is straightforward to translate this into syntax suitable for testnl, recalling that the coefficient of variable \(v\) in equation \(e q\) is denoted by [eq] \(v\).
```

. testnl [Inc_mean]edu/[Promo_promo]edu =
[Inc_mean] exp/[Promo_promo] exp =
[Inc_mean]male/[Promo_promo]male
(1) [Inc_mean] edu/[Promo_promo]edu = [Inc_mean] exp/[Promo_promo] exp
(2) [Inc_mean] edu/[Promo_promo]edu = [Inc_mean]male/[Promo_promo]male
chi2(2) = 0.61
Prob > chi2 = 0.7385

```

From the evidence, we fail to reject the hypotheses that the coefficients of the income and promotion equations are proportional. Thus, it is not unreasonable to assume that income and promotion can be explained by the same latent variable, "labor market success".

A disadvantage of the nonlinear Wald test is that it is not invariant with respect to representation: a Wald test for a mathematically equivalent formulation of the nonlinear constraint usually leads to a different test result. An equivalent formulation of the proportionality hypothesis is
\[
\begin{aligned}
H_{0}: & \beta_{\mathrm{edu}}^{\text {Income }} \beta_{\mathrm{exp}}^{\text {Promotion }}=\beta_{\mathrm{edu}}^{\text {Promotion }} \beta_{\mathrm{exp}}^{\text {Income }} \quad \text { and } \\
& \beta_{\mathrm{edu}}^{\text {Income }} \beta_{\text {male }}^{\text {Promotion }}=\beta_{\mathrm{edu}}^{\text {Promotion }} \beta_{\text {male }}^{\text {Income }}
\end{aligned}
\]

This formulation is "more linear" in the coefficients. The asymptotic \(\chi^{2}\) distribution of the nonlinear Wald statistic can be expected to be more accurate for this representation.
```

. testnl ([Inc_mean] edu*[Promo_promo] exp = [Inc_mean] exp*[Promo_promo]edu)
> ([Inc_mean] edu*[Promo_promo]male = [Inc_mean]male*[Promo_promo]edu)
(1) [Inc_mean] edu*[Promo_promo] exp = [Inc_mean] exp*[Promo_promo]edu
(2) [Inc_mean]edu*[Promo_promo]male = [Inc_mean]male*[Promo_promo]edu
chi2(2) = 0.46
Prob > chi2 = 0.7936

```

Here the two representations lead to similar test statistics and \(p\)-values. As before, we fail to reject the hypothesis of proportionality of the coefficients of the two models.

\section*{Testing cross-model hypotheses}
> Example 3
In this example, we demonstrate how some cross-model hypotheses can be tested using the facilities already available in most estimation commands. This demonstration will explain the intricate relationship between the cluster adjustment of the robust estimator of variance and the suest command. It will also be made clear that a new facility is required to perform more general cross-model testing.

We want to test whether the effect of \(x_{1}\) on the binary variable \(y_{1}\) is the same as the effect of \(x_{2}\) on the binary \(y_{2}\); see Clogg, Petkova, and Haritou (1995). In this setting, \(x_{1}\) may equal \(x_{2}\), and \(y_{1}\) may equal \(y_{2}\). We assume that logistic regression models can be used to model the responses, and for simplicity, we ignore further predictor variables in these models. If the two logit models are fit on independent samples so that the estimators are (stochastically) independent, a Wald test for _b [x1] \(=\_\mathrm{b}[\mathrm{x} 2]\) rejects the null hypothesis if
\[
\frac{\widehat{b}\left(x_{1}\right)-\widehat{b}\left(x_{2}\right)}{\left[\widehat{\sigma}^{2}\left\{\widehat{b}\left(x_{1}\right)\right\}+\widehat{\sigma}^{2}\left\{\widehat{b}\left(x_{2}\right)\right\}\right]^{1 / 2}}
\]
is larger than the appropriate \(\chi_{1}^{2}\) threshold. If the models are fit on the same sample (or on dependent samples), so that the estimators are stochastically dependent, the above test that ignores the covariance between the estimators is not appropriate.

It is instructive to see how this problem can be tackled by "stacking" data. In the stacked format, we doubled the number of observations. The dependent variable is \(y_{1}\) in the first half of the data and is \(y_{2}\) in the second half of the data. The predictor variable \(z_{1}\) is set to \(x_{1}\) in the first half of the expanded data and to 0 in the rest. Similarly, \(z_{2}\) is 0 in the first half and \(x_{2}\) in the second half. The following diagram illustrates the transformation, in the terminology of the reshape command, from wide to long format.
\[
\left(\begin{array}{ccccc}
\text { id } & y_{1} & y_{2} & x_{1} & x_{2} \\
\hline 1 & y_{11} & y_{21} & x_{11} & x_{21} \\
2 & y_{12} & y_{22} & x_{12} & x_{22} \\
3 & y_{13} & y_{23} & x_{13} & x_{23}
\end{array}\right) \Longrightarrow\left(\begin{array}{ccccc}
\hline 1 & y_{11} & x_{11} & 0 & 1 \\
2 & y_{12} & x_{12} & 0 & 1 \\
3 & y_{13} & x_{13} & 0 & 1 \\
1 & y_{21} & 0 & x_{21} & 2 \\
2 & y_{22} & 0 & x_{22} & 2 \\
3 & y_{23} & 0 & x_{23} & 2
\end{array}\right)
\]

The observations in the long-format data organization are clustered on the original subjects and are identified with the identifier id. The clustering on id has to be accounted for when fitting a simultaneous model. The simplest way to deal with clustering is to use the cluster adjustment of the robust or sandwich estimator; see \([\mathrm{P}]\) _robust. The data manipulation can be accomplished easily with the stack command; see [D] stack. Subsequently, we fit a simultaneous logit model and perform a Wald test for the hypothesis that the coefficients of \(z 1\) and \(z 2\) are the same. A full setup to obtain the cross-model Wald test could then be as follows:
```

. generate zero = 0 // a variable that is always 0
. generate one = 1 // a variable that is always 1
. generate two = 2 // a variable that is always 2
. stack id y1 x1 zero one id y2 zero x2 two, into(id y z1 z2 model)
. generate model2 = (model==2)
. logit y model2 z1 z2, vce(cluster id)
. test _b[z1] = _b[z2]

```

The coefficient of \(z 1\) represents the effect of \(x 1\) on \(y 1\), and similarly, \(z 2\) for the effect of \(x 2\) on y2. The variable model2 is a dummy for the "second model", which is included to allow the intercept in the second model to differ from that in the first model. The estimates of the coefficient of \(z 1\) and its standard error in the combined model are the same as the estimates of the coefficient of \(z 1\) and its standard error if we fit the model on the unstacked data.
```

. logit y1 x1, vce(robust)

```

The vee (cluster clustvar) option specified with the logit command for the stacked data ensures that the covariances of \(\_\mathrm{b}[\mathrm{z} 1]\) and \(\_\mathrm{b}[\mathrm{z} 2]\) are indeed estimated. This estimation ensures that the Wald test for the equality of the coefficients is correct. If we had not specified the vce (cluster clustvar) option, the (co)variance matrix of the coefficients would have been block-diagonal; that is, the covariances of \(\_\mathrm{b}[\mathrm{z} 1]\) and \(\_\mathrm{b}[\mathrm{z} 2\) ] would have been 0 . Then, test would have effectively used the invalid formula for the Wald test for two independent samples.

In this example, the two logit models were fit on the same data. The same setup would apply, without modification, when the two logit models were fit on overlapping data that resulted, for instance, if the \(y\) or \(x\) variables were missing in some observations.

The suest command allows us to obtain the above Wald test more efficiently by avoiding the data manipulation, obviating the need to fit a model with twice the number of coefficients. The test statistic produced by the above code fragment is identical to that obtained via suest on the original (unstacked) data:
. logit y1 x1
. estimates store M1
. \(\operatorname{logit~y2~x2~}\)
. estimates store M2
. suest M1 M2
. test \([\mathrm{M} 1] \times 1=[\mathrm{M} 2] \mathrm{x} 2\)
The stacking method can be applied not only to the testing of cross-model hypotheses for logit models but also to any estimation command that supports the vce (cluster clustvar) option. The stacking approach clearly generalizes to stacking more than two logit or other models, testing more general linear hypotheses, and testing nonlinear cross-model hypotheses (see [R] testnl). In all of these cases, suest would yield identical statistical results but at smaller costs in terms of data management, computer storage, and computer time.

Is suest nothing but a convenience command? No, there are two disadvantages to the stacking method, both of which are resolved via suest. First, if the models include ancillary parameters (in a regression model, the residual variance; in an ordinal response model, the cutpoints; and in lognormal survival-time regression, the time scale parameter), these parameters are constrained to be equal between the stacked models. In suest, this constraint is relaxed. Second, the stacking method does not generalize to compare different statistical models, such as a probit model and a regression model. As demonstrated in the previous section, suest can deal with this situation.

\section*{Stored results}
suest stores the following in e():
Scalars
\(e(N) \quad\) number of observations
e(N_clust) number of clusters
\(e\) (rank) rank of \(e(V)\)
Macros
e(cmd)
e(eqnames\#)
suest
\(e\) (names) list of model names
\(e\) (wtype) weight type
e(wexp)
\(e\) (clustvar) name of cluster variable
e(vce)
e(vcetype)
e(properties)
Matrices
e(b) stacked coefficient vector of the models
e(V)
variance-covariance matrix of the estimators
Functions
e(sample) marks estimation sample

\section*{Methods and formulas}

The estimation of the simultaneous (co)variance of a series of \(k\) estimators is a nonstandard application of the sandwich estimator, as implemented by the command [P] _robust. You may want to read this manual entry before reading further.

The starting point is that we have fit \(k\) different models on the same data-partially overlapping or nonoverlapping data are special cases. We want to derive the simultaneous distribution of these \(k\) estimators, for instance, to test a cross-estimator hypothesis \(H_{0}\). As in the framework of Hausman testing, \(H_{0}\) will often be of the form that different estimators have the same probability limit under some hypothesis, while the estimators have different limits if the hypothesis is violated.

We consider (vector) estimators \(\widehat{\boldsymbol{\beta}}_{i}\) to be defined as "the" solution of the estimation equations \(\mathbf{G}_{i}\),
\[
\mathbf{G}_{i}\left(\mathbf{b}_{i}\right)=\sum_{j} w_{i j} \mathbf{u}_{i j}\left(\mathbf{b}_{i}\right)=\mathbf{0}, \quad i=1, \ldots, k
\]

We refer to the \(\mathbf{u}_{i j}\) as the "scores". Specifying some weights \(w_{i j}=0\) trivially accommodates for partially overlapping or even disjointed data. Under "suitable regularity conditions" (see White [1982; 1996] for details), the \(\widehat{\boldsymbol{\beta}}_{i}\) are asymptotically normally distributed, with the variance estimated consistently by the sandwich estimator
\[
V_{i}=\operatorname{Var}\left(\widehat{\boldsymbol{\beta}}_{i}\right)=\mathbf{D}_{i}^{-1} \sum_{j} w_{i j} \mathbf{u}_{i j} \mathbf{u}_{i j}^{\prime} \mathbf{D}_{i}^{-1}
\]
where \(\mathbf{D}_{i}\) is the Jacobian of \(\mathbf{G}_{i}\) evaluated at \(\widehat{\boldsymbol{\beta}}_{i}\). In the context of maximum likelihood estimation, \(\mathbf{D}_{i}\) can be estimated consistently by (minus) the Hessian of the \(\log\) likelihood or by the Fisher information matrix. If the model is also well specified, the sandwiched term \(\left(\sum_{j} w_{i j} \mathbf{u}_{i j} \mathbf{u}_{i j}^{\prime}\right)\) converges in probability to \(\mathbf{D}_{i}\), so \(V_{i}\) may be consistently estimated by \(\mathbf{D}_{i}^{-1}\).

To derive the simultaneous distribution of the estimators, we consider the "stacked" estimation equation,
\[
\mathbf{G}(\widehat{\boldsymbol{\beta}})=\left\{\begin{array}{llll}
\mathbf{G}_{1}\left(\widehat{\boldsymbol{\beta}}_{1}\right)^{\prime} & \mathbf{G}_{1}\left(\widehat{\boldsymbol{\beta}}_{2}\right)^{\prime} & \ldots & \mathbf{G}_{k}\left(\widehat{\boldsymbol{\beta}}_{k}\right)^{\prime}
\end{array}\right\}^{\prime}=\mathbf{0}
\]

Under "suitable regularity conditions" (see White [1996] for details), \(\widehat{\boldsymbol{\beta}}\) is asymptotically jointly normally distributed. The Jacobian and scores of the simultaneous equation are easily expressed in the Jacobian and scores of the separate equations. The Jacobian of \(\mathbf{G}\),
\[
\mathbf{D}(\widehat{\boldsymbol{\beta}})=\left.\frac{d \mathbf{G}(\boldsymbol{\beta})}{d \boldsymbol{\beta}}\right|_{\boldsymbol{\beta}=\widehat{\boldsymbol{\beta}}}
\]
is block diagonal with blocks \(\mathbf{D}_{1}, \ldots, \mathbf{D}_{k}\). The inverse of \(\mathbf{D}(\widehat{\boldsymbol{\beta}})\) is again block diagonal, with the inverses of \(\mathbf{D}_{i}\) on the diagonal. The scores \(\mathbf{u}\) of \(\mathbf{G}\) are simply obtained as the concatenated scores of the separate equations:
\[
\mathbf{u}_{j}=\left(\begin{array}{llll}
\mathbf{u}_{1 j}^{\prime} & \mathbf{u}_{2 j}^{\prime} & \ldots & \mathbf{u}_{k j}^{\prime}
\end{array}\right)^{\prime}
\]

Out-of-sample (that is, where \(w_{i j}=0\) ) values of the score variables are defined as 0 (thus we drop the \(i\) subscript from the common weight variable). The sandwich estimator for the asymptotic variance of \(\widehat{\boldsymbol{\beta}}\) reads
\[
V=\operatorname{Var}(\widehat{\boldsymbol{\beta}})=\mathbf{D}(\widehat{\boldsymbol{\beta}})^{-1}\left(\sum_{j} w_{j} \mathbf{u}_{j} \mathbf{u}_{j}^{\prime}\right) \mathbf{D}(\widehat{\boldsymbol{\beta}})^{-1}
\]

Taking a "partitioned" look at this expression, we see that \(V\left(\widehat{\boldsymbol{\beta}}_{i}\right)\) is estimated by
\[
\mathbf{D}_{i}^{-1}\left(\sum_{j} w_{j} \mathbf{u}_{i j} \mathbf{u}_{i j}^{\prime}\right) \mathbf{D}_{i}^{-1}
\]
which is, yet again, the familiar sandwich estimator for \(\widehat{\boldsymbol{\beta}}_{i}\) based on the separate estimation equation \(\mathbf{G}_{i}\). Thus, considering several estimators simultaneously in this way does not affect the estimators of the asymptotic variances of these estimators. However, as a bonus of stacking, we obtained a sandwich-type estimate of the covariance \(V_{i h}\) of estimators \(\widehat{\boldsymbol{\beta}}_{i}\) and \(\widehat{\boldsymbol{\beta}}_{h}\),
\[
V_{i h}=\operatorname{Cov}\left(\widehat{\boldsymbol{\beta}}_{i}, \widehat{\boldsymbol{\beta}}_{h}\right)=\mathbf{D}_{i}^{-1}\left(\sum_{j} w_{j} \mathbf{u}_{i j} \mathbf{u}_{h j}^{\prime}\right) \mathbf{D}_{h}^{-1}
\]
which is also obtained by White (1982).
This estimator for the covariance of estimators is an application of the cluster modification of the sandwich estimator proposed by Rogers (1993). Consider the stacked data format as discussed in the logit example, and assume that Stata would be able to estimate a "stacked model" in which different models apply to different observations, for example, a probit model for the first half, a regression model for the second half, and a one-to-one cluster relation between the first and second half. If there are no common parameters to both models, the score statistics of parameters for the stacked models are zero in the half of the data in which they do not occur. In Rogers's method, we have to sum the score statistics over the observations within a cluster. This step boils down to concatenating the score statistics at the level of the cluster.

We compare the sandwich estimator of the (co)variance \(V_{12}\) of two estimators with the estimator of variance \(\widetilde{V}_{12}\) applied in the classic Hausman test. Hausman (1978) showed that if \(\widehat{\boldsymbol{\beta}}_{1}\) is consistent under \(H_{0}\) and \(\widehat{\boldsymbol{\beta}}_{2}\) is efficient under \(H_{0}\), then asymptotically
\[
\operatorname{Cov}\left(\widehat{\boldsymbol{\beta}}_{1}, \widehat{\boldsymbol{\beta}}_{2}\right)=\operatorname{Var}\left(\widehat{\boldsymbol{\beta}}_{2}\right)
\]
and so \(\operatorname{var}\left(\widehat{\boldsymbol{\beta}}_{1}-\widehat{\boldsymbol{\beta}}_{2}\right)\) is consistently estimated by \(V_{1}-V_{2}\).

\section*{Acknowledgment}
suest was written by Jeroen Weesie of the Department of Sociology at Utrecht University, The Netherlands. This research is supported by grant PGS 50-370 by The Netherlands Organization for Scientific Research.

An earlier version of suest was published in the Stata Technical Bulletin (1999). The current version of suest is not backward compatible with the 1999 version because of the introduction of new ways to manage estimation results via the estimates command.

\section*{References}

Arminger, G. 1990. Testing against misspecification in parametric rate models. In Event History Analysis in Life Course Research, ed. K. U. Mayer and N. B. Tuma, 146-158. Madison: University of Wisconsin Press.

Canette, I. 2014. Using gsem to combine estimation results. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2014/08/18/using-gsem-to-combine-estimation-results/.

Clogg, C. C., E. Petkova, and A. Haritou. 1995. Statistical methods for comparing regression coefficients between models. American Journal of Sociology 100: 1261-1312. (With comments by P. D. Allison and a reply by C. C. Clogg, E. Petkova, and T. Cheng).

Gouriéroux, C. S., and A. Monfort. 1997. Time Series and Dynamic Models. Trans. ed. G. M. Gallo. Cambridge: Cambridge University Press.
Hausman, J. A. 1978. Specification tests in econometrics. Econometrica 46: 1251-1271. https://doi.org/10.2307/1913827.

Hausman, J. A., and D. L. McFadden. 1984. Specification tests for the multinomial logit model. Econometrica 52: 1219-1240. https://doi.org/10.2307/1910997.
Hausman, J. A., and P. A. Ruud. 1987. Specifying and testing econometric models for rank-ordered data. Journal of Econometrics 34: 83-104. https://doi.org/10.1016/0304-4076(87)90068-6.

Huber, P. J. 1967. The behavior of maximum likelihood estimates under nonstandard conditions. In Vol. 1 of Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, 221-233. Berkeley: University of California Press.
Oberfichtner, M., and H. Tauchmann. 2021. Stacked linear regression analysis to facilitate testing of hypotheses across OLS regressions. Stata Journal 21: 411-429.

Rogers, W. H. 1993. sg16.4: Comparison of nbreg and glm for negative binomial. Stata Technical Bulletin 16: 7. Reprinted in Stata Technical Bulletin Reprints, vol. 3, pp. 82-84. College Station, TX: Stata Press.
Weesie, J. 1999. sg121: Seemingly unrelated estimation and the cluster-adjusted sandwich estimator. Stata Technical Bulletin 52: 34-47. Reprinted in Stata Technical Bulletin Reprints, vol. 9, pp. 231-248. College Station, TX: Stata Press.

White, H. L., Jr. 1982. Maximum likelihood estimation of misspecified models. Econometrica 50: 1-25. https://doi.org/10.2307/1912526.
——. 1996. Estimation, Inference and Specification Analysis. Cambridge: Cambridge University Press.

\section*{Also see}
[R] estimates - Save and manipulate estimation results
[R] hausman - Hausman specification test
[R] lincom - Linear combinations of parameters
[R] nlcom - Nonlinear combinations of parameters
[R] test - Test linear hypotheses after estimation
[R] testnl - Test nonlinear hypotheses after estimation
[P] _robust - Robust variance estimates

\section*{Title}
summarize - Summary statistics
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
summarize calculates and displays a variety of univariate summary statistics. If no varlist is specified, summary statistics are calculated for all the variables in the dataset.

\section*{Quick start}

Basic summary statistics for continuous variable v1
summarize v1
Same as above, and include v2 and v3
summarize v1-v3
Same as above, and provide additional detail about the distribution
summarize v1-v3, detail
Summary statistics reported separately for each level of catvar
by catvar: summarize v1
With frequency weight wvar
summarize v1 [fweight=wvar]

\section*{Menu}

Statistics \(>\) Summaries, tables, and tests \(>\) Summary and descriptive statistics \(>\) Summary statistics

\section*{Syntax}
summarize [varlist] [if] [in] [weight] [, options]
options Description
Main
detail display additional statistics
meanonly suppress the display; calculate only the mean; programmer's option
format
separator(\#)
display_options use variable's display format
draw separator line after every \# variables; default is separator (5)
control spacing, line width, and base and empty cells
varlist may contain factor variables; see [U] 11.4.3 Factor variables.
varlist may contain time-series operators; see [U] 11.4.4 Time-series varlists.
by, collect, rolling, and statsby are allowed; see [U] 11.1.10 Prefix commands.
aweights, fweights, and iweights are allowed. However, iweights may not be used with the detail option; see
[U] 11.1.6 weight.

\section*{Options}

\section*{Main}
detail produces additional statistics, including skewness, kurtosis, the four smallest and four largest values, and various percentiles.
meanonly, which is allowed only when detail is not specified, suppresses the display of results and calculation of the variance. Ado-file writers will find this useful for fast calls.
format requests that the summary statistics be displayed using the display formats associated with the variables rather than the default \(g\) display format; see [U] 12.5 Formats: Controlling how data are displayed.
separator (\#) specifies how often to insert separation lines into the output. The default is separator (5), meaning that a line is drawn after every five variables. separator (10) would draw a line after every 10 variables. separator ( 0 ) suppresses the separation line.
display_options: vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), and fvwrapon(style); see [R] Estimation options.

\section*{Remarks and examples}
summarize can produce two different sets of summary statistics. Without the detail option, the number of nonmissing observations, the mean and standard deviation, and the minimum and maximum values are presented. With detail, the same information is presented along with the variance, skewness, and kurtosis; the four smallest and four largest values; and the 1st, 5th, 10th, 25th, 50th (median), 75th, 90th, 95th, and 99th percentiles.

Also see \([R]\) ci for calculating the standard error and confidence intervals of the mean.
> Example 1: summarize with the separator() option
We have data containing information on various automobiles, among which is the variable mpg, the mileage rating. We can obtain a quick summary of the mpg variable by typing
```

. use https://www.stata-press.com/data/r18/auto2
(1978 automobile data)
. summarize mpg

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| $m p g$ | 74 | 21.2973 | 5.785503 | 12 | 41 |

```

We see that we have 74 observations. The mean of mpg is 21.3 miles per gallon, and the standard deviation is 5.79 . The minimum is 12 , and the maximum is 41 .

If we had not specified the variable (or variables) we wanted to summarize, we would have obtained summary statistics on all the variables in the dataset:
\begin{tabular}{r|rrrrr}
. summarize, separator(4) & Mean & Std. dev. & Min & Max \\
Variable & Obs & Men & & & \\
\hline make & 0 & & & 3291 & 15906 \\
price & 74 & 6165.257 & 2949.496 & 12 & 41 \\
mpg & 74 & 21.2973 & 5.785503 & 1 & 5 \\
rep78 & 69 & 3.405797 & .9899323 & 1.5 & 5 \\
\hline headroom & 74 & 2.993243 & .8459948 & 5 & 23 \\
trunk & 74 & 13.75676 & 4.277404 & 1760 & 4840 \\
weight & 74 & 3019.459 & 777.1936 & 142 & 233 \\
length & 74 & 187.9324 & 22.26634 & 31 & 51 \\
\hline turn & 74 & 39.64865 & 4.399354 & 79 & 425 \\
displacement & 74 & 197.2973 & 91.83722 & 2.19 & 3.89 \\
gear_ratio & 74 & 3.014865 & .4562871 & 0 & 1
\end{tabular}

There are only 69 observations on rep78, so some of the observations are missing. There are no observations on make because it is a string variable.

The idea of the mean is quite old (Plackett 1958), but its extension to a scheme of moment-based measures was not done until the end of the 19th century. Between 1893 and 1905, Pearson discussed and named the standard deviation, skewness, and kurtosis, but he was not the first to use any of these. Thiele (1889), in contrast, had earlier firmly grasped the notion that the \(m_{r}\) provide a systematic basis for discussing distributions. However, even earlier anticipations can also be found. For example, Euler in 1778 used \(m_{2}\) and \(m_{3}\) in passing in a treatment of estimation (Hald 1998, 87), but seemingly did not build on that.

Similarly, the idea of the median is quite old. The history of the interquartile range is tangled up with that of the probable error, a long-popular measure. Extending this in various ways to a more general approach based on quantiles (to use a later term) occurred to several people in the nineteenth century. Galton (1875) is a nice example, particularly because he seems so close to the key idea of the quantiles as a function, which took another century to reemerge strongly.
Thorvald Nicolai Thiele (1838-1910) was a Danish scientist who worked in astronomy, mathematics, actuarial science, and statistics. He made many pioneering contributions to statistics, several of which were overlooked until recently. Thiele advocated graphical analysis of residuals checking for trends, symmetry of distributions, and changes of sign, and he even warned against overinterpreting such graphs.

\section*{> Example 2: summarize with the detail option}

The detail option provides all the information of a normal summarize and more. The format of the output also differs, as shown here:
\begin{tabular}{|c|c|c|c|c|}
\hline \multicolumn{5}{|c|}{Mileage (mpg)} \\
\hline & Percentiles & Smallest & & \\
\hline 1\% & 12 & 12 & & \\
\hline 5\% & 14 & 12 & & \\
\hline 10\% & 14 & 14 & Obs & 74 \\
\hline 25\% & 18 & 14 & Sum of wgt. & 74 \\
\hline 50\% & 20 & & Mean & 21.2973 \\
\hline & & Largest & Std. dev. & 5.785503 \\
\hline 75\% & 25 & 34 & & \\
\hline 90\% & 29 & 35 & Variance & 33.47205 \\
\hline 95\% & 34 & 35 & Skewness & . 9487176 \\
\hline 99\% & 41 & 41 & Kurtosis & 3.975005 \\
\hline
\end{tabular}

As in the previous example, we see that the mean of mpg is 21.3 miles per gallon and that the standard deviation is 5.79 . We also see the various percentiles. The median of mpg (the 50 th percentile) is 20 miles per gallon. The 25 th percentile is 18 , and the 75 th percentile is 25 .

When we performed summarize, we learned that the minimum and maximum were 12 and 41, respectively. We now see that the four smallest values in our dataset are \(12,12,14\), and 14 . The four largest values are \(34,35,35\), and 41 . The skewness of the distribution is 0.95 , and the kurtosis is 3.98. (A normal distribution would have a skewness of 0 and a kurtosis of 3.)

Skewness is a measure of the lack of symmetry of a distribution. If the distribution is symmetric, the coefficient of skewness is 0 . If the coefficient is negative, the median is usually greater than the mean and the distribution is said to be skewed left. If the coefficient is positive, the median is usually less than the mean and the distribution is said to be skewed right. Kurtosis (from the Greek kyrtosis, meaning curvature) is a measure of peakedness of a distribution. The smaller the coefficient of kurtosis, the flatter the distribution. The normal distribution has a coefficient of kurtosis of 3 and provides a convenient benchmark.

\section*{- Technical note}

The convention of calculating the median of an even number of values by averaging the central two order statistics is of long standing. (That is, given 8 values, average the 4 th and 5 th smallest values, or given 42, average the 21 st and 22 nd smallest.) Stigler (1977) filled a much-needed gap in the literature by naming such paired central order statistics as "comedians", although it remains unclear how far he was joking.

\section*{Example 3: summarize with the by prefix}
summarize can usefully be combined with the by varlist: prefix. In our dataset, we have a variable, foreign, that distinguishes foreign and domestic cars. We can obtain summaries of mpg and weight within each subgroup by typing
. by foreign: summarize mpg weight
\begin{tabular}{|c|c|c|c|c|c|}
\hline Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline mpg & 52 & 19.82692 & 4.743297 & 12 & 34 \\
\hline weight & 52 & 3317.115 & 695.3637 & 1800 & 4840 \\
\hline \multicolumn{6}{|l|}{-> foreign = Foreign} \\
\hline Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline mpg & 22 & 24.77273 & 6.611187 & 14 & 41 \\
\hline weight & 22 & 2315.909 & 433.0035 & 1760 & 3420 \\
\hline
\end{tabular}

Domestic cars in our dataset average 19.8 miles per gallon, whereas foreign cars average 24.8 .
Because by varlist: can be combined with summarize, it can also be combined with summarize, detail:
by foreign: summarize mpg, detail
```

-> foreign = Domestic

```
\begin{tabular}{rrrlr}
\multicolumn{4}{c}{ Mileage (mpg) } \\
& Percentiles & Smallest & \\
\(1 \%\) & 12 & 12 & & 52 \\
\(5 \%\) & 14 & 12 & Obs & 52 \\
\(10 \%\) & 14 & 14 & Sum of wgt. & \\
\(25 \%\) & 16.5 & 14 & Mean & 19.82692 \\
\(50 \%\) & 19 & & Std. dev. & 4.743297 \\
\(75 \%\) & & Largest & 28 & Variance \\
\(90 \%\) & 22 & 29 & 30 & Skewness \\
\(95 \%\) & 26 & 34 & Kurtosis & 22.49887 \\
\(99 \%\) & 29 & & & \\
& & & & \\
\hline
\end{tabular}
-> foreign = Foreign
Mileage (mpg)
\begin{tabular}{rrrlr}
\hline & Percentiles & Smallest & & \\
\(1 \%\) & 14 & 14 & & \\
\(5 \%\) & 17 & 17 & & \\
\(10 \%\) & 17 & 17 & Obs & 22 \\
\(25 \%\) & 21 & 18 & Sum of wgt. & \\
\(50 \%\) & 24.5 & & Mean & 24.77273 \\
& & Largest & Std. dev. & 6.611187 \\
\(75 \%\) & 28 & 31 & & \\
\(90 \%\) & 35 & 35 & Variance & 43.70779 \\
\(95 \%\) & 35 & 35 & Skewness & .657329 \\
\(99 \%\) & 41 & 41 & Kurtosis & 3.10734
\end{tabular}

\section*{Technical note}
summarize respects display formats if we specify the format option. When we type summarize price weight, we obtain
. summarize price weight
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline price & 74 & 6165.257 & 2949.496 & 3291 & 15906 \\
weight & 74 & 3019.459 & 777.1936 & 1760 & 4840
\end{tabular}

The display is accurate but is not as aesthetically pleasing as we may wish, particularly if we plan to use the output directly in published work. By placing formats on the variables, we can control how the table appears:
. format price weight \(\% 9.2 f c\)
. summarize price weight, format
\begin{tabular}{r|rrrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline price & 74 & \(6,165.26\) & \(2,949.50\) & \(3,291.00\) & \(15,906.00\) \\
weight & 74 & \(3,019.46\) & 777.19 & \(1,760.00\) & \(4,840.00\)
\end{tabular}

If you specify a weight (see [U] 11.1.6 weight), each observation is multiplied by the value of the weighting expression before the summary statistics are calculated so that the weighting expression is interpreted as the discrete density of each observation.
> Example 4: summarize with factor variables
You can also use summarize to obtain summary statistics for factor variables. For example, if you type
. summarize i.rep78
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline rep78 & & & & & \\
Poor & 69 & .0289855 & .1689948 & 0 & 1 \\
Fair & 69 & .115942 & .3225009 & 0 & 1 \\
Average & 69 & .4347826 & .4993602 & 0 & 1 \\
Good & 69 & .2608696 & .4423259 & 0 & 1 \\
Excellent & 69 & .1594203 & .3687494 & 0 & 1
\end{tabular}
you obtain the sample proportions for the five levels of the rep78 variable. For example, \(11.6 \%\) of the 69 cars with nonmissing values of rep78 have a fair repair record.

We could have used tabulate oneway rep78 to obtain the sample proportions along with the cumulative proportions. Alternatively, we could have used proportions rep78 to obtain the sample proportions along with the standard errors of the proportions instead of the standard deviations of the proportions.

\section*{\(>\) Example 5: summarize with weights}

We have 1980 census data on each of the 50 states. Included in our variables is medage, the median age of the population of each state. If we type summarize medage, we obtain unweighted statistics:
```

. use https://www.stata-press.com/data/r18/census, clear
(1980 Census data by state)
. summarize medage

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | :---: | :---: | :---: | :---: | :---: |
| medage | 50 | 29.54 | 1.693445 | 24.2 | 34.7 |

```

Also among our variables is pop, the population in each state. Typing summarize medage [ \(\mathrm{w}=\mathrm{pop}\) ] produces population-weighted statistics:
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline Variable & Obs & Weight & Mean & Std. Dev. & Min & Max \\
\hline medage & 50 & 225907472 & 30.11047 & 1.66933 & 24.2 & 34.7 \\
\hline
\end{tabular}

The number listed under Weight is the sum of the weighting variable, pop, indicating that there are roughly 226 million people in the United States. The pop-weighted mean of medage is 30.11 (compared with 29.54 for the unweighted statistic), and the weighted standard deviation is 1.67 (compared with 1.69).

\section*{> Example 6: summarize with weights and the detail option}

We can obtain detailed summaries of weighted data as well. When we do this, all the statistics are weighted, including the percentiles.
```

. summarize medage [w=pop], detail
(analytic weights assumed)

```

> Median age
\begin{tabular}{rcrlr}
\hline & Percentiles & Smallest & & \\
\(1 \%\) & 27.1 & 24.2 & & \\
\(5 \%\) & 27.7 & 26.1 & & 50 \\
\(10 \%\) & 28.2 & 27.1 & Obs & \\
\(25 \%\) & 29.2 & 27.4 & Sum of wgt. & 225907472 \\
\(50 \%\) & 29.9 & & Mean & 30.11047 \\
& & Largest & Std. dev. & 1.66933 \\
\(75 \%\) & 30.9 & 32 & & \\
\(90 \%\) & 32.1 & 32.1 & Variance & 2.786661 \\
\(95 \%\) & 32.2 & 32.2 & Skewness & .5281972 \\
\(99 \%\) & 34.7 & 34.7 & Kurtosis & 4.494223
\end{tabular}

\section*{Technical note}

If you are writing a program and need to access the mean of a variable, the meanonly option provides for fast calls. For example, suppose that your program reads as follows:
```

program mean
summarize '1', meanonly
display " mean = " r(mean)
end

```

The result of executing this is
```

. use https://www.stata-press.com/data/r18/auto2
(1978 automobile data)
. mean price
mean = 6165.2568

```

\section*{Video example}

Descriptive statistics in Stata

\section*{Stored results}
summarize stores the following in \(r()\) :
\begin{tabular}{|c|c|c|c|}
\hline \multicolumn{4}{|l|}{Scalars} \\
\hline r ( N ) & number of observations & \(r\) (p50) & 50th percentile (detail only) \\
\hline \(r\) (mean) & mean & r (p75) & 75th percentile (detail only) \\
\hline \(r\) (skewness) & skewness (detail only) & r (p90) & 90th percentile (detail only) \\
\hline \(r\) (min) & minimum & r (p95) & 95th percentile (detail only) \\
\hline \(r\) (max) & maximum & r (p99) & 99th percentile (detail only) \\
\hline r (sum_w) & sum of the weights & r (Var) & variance \\
\hline \(\mathrm{r}(\mathrm{p} 1)\) & 1st percentile (detail only) & r(kurtosis) & kurtosis (detail only) \\
\hline r (p5) & 5 th percentile (detail only) & r (sum) & sum of variable \\
\hline r(p10) & 10th percentile (detail only) & r(sd) & standard deviation \\
\hline r (p25) & 25 th percentile (detail only) & & \\
\hline
\end{tabular}

\section*{Methods and formulas}

Let \(x\) denote the variable on which we want to calculate summary statistics, and let \(x_{i}, i=1, \ldots, n\), denote an individual observation on \(x\). Let \(v_{i}\) be the weight, and if no weight is specified, define \(v_{i}=1\) for all \(i\).

Define the total number of observations as \(N=n\) if \(v_{i}\) is an aweight and as
\[
N=\sum_{i=1}^{n} v_{i}
\]
otherwise.
Additionally, define the normalized weight as
\[
w_{i}=\frac{n v_{i}}{\sum_{i=1}^{n} v_{i}}
\]
if \(v_{i}\) is an aweight and as \(w_{i}=v_{i}\) otherwise.
The mean, \(\bar{x}\), is defined as
\[
\bar{x}=\frac{1}{N} \sum_{i=1}^{n} w_{i} x_{i}
\]

The variance, \(s^{2}\), is defined as
\[
s^{2}=\frac{1}{N-1} \sum_{i=1}^{n} w_{i}\left(x_{i}-\bar{x}\right)^{2}
\]

The standard deviation, \(s\), is defined as \(\sqrt{s^{2}}\).
Define \(m_{r}\) as the \(r\) th moment about the mean \(\bar{x}\) :
\[
m_{r}=\frac{1}{N} \sum_{i=1}^{n} w_{i}\left(x_{i}-\bar{x}\right)^{r}
\]

The coefficient of skewness is then defined as \(m_{3} m_{2}^{-3 / 2}\). The coefficient of kurtosis is defined as \(m_{4} m_{2}^{-2}\).

Let \(x_{(i)}\) refer to the \(x\) in ascending order, and let \(w_{(i)}\) refer to the corresponding weights of \(x_{(i)}\). The four smallest values are \(x_{(1)}, x_{(2)}, x_{(3)}\), and \(x_{(4)}\). The four largest values are \(x_{(n)}, x_{(n-1)}\), \(x_{(n-2)}\), and \(x_{(n-3)}\).

To obtain the \(p\) th percentile, which we will denote as \(x_{[p]}\), let \(P=N p / 100\). Let
\[
W_{(i)}=\sum_{j=1}^{i} w_{(j)}
\]

Find the first index \(i\) such that \(W_{(i)}>P\). The \(p\) th percentile is then
\[
x_{[p]}= \begin{cases}\frac{x_{(i-1)}+x_{(i)}}{2} & \text { if } W_{(i-1)}=P \\ x_{(i)} & \text { otherwise }\end{cases}
\]

\section*{References}

Cox, N. J. 2010. Speaking Stata: The limits of sample skewness and kurtosis. Stata Journal 10: 482-495.
——. 2013. Speaking Stata: Trimming to taste. Stata Journal 13: 640-666.
David, H. A. 2001. First (?) occurrence of common terms in statistics and probability. In Annotated Readings in the History of Statistics, ed. H. A. David and A. W. F. Edwards, 209-246. New York: Springer.

Galton, F. 1875. Statistics by intercomparison, with remarks on the law of frequency of error. Philosophical Magazine 49: 33-46. https://doi.org/10.1080/14786447508641172.

Gelade, W., V. Verardi, and C. Vermandele. 2015. Time-efficient algorithms for robust estimators of location, scale, symmetry, and tail heaviness. Stata Journal 15: 77-94.
Hald, A. 1998. A History of Mathematical Statistics from 1750 to 1930. New York: Wiley.
Kirkwood, B. R., and J. A. C. Sterne. 2003. Essential Medical Statistics. 2nd ed. Malden, MA: Blackwell.
Lauritzen, S. L. 2002. Thiele: Pioneer in Statistics. Oxford: Oxford University Press.
Plackett, R. L. 1958. Studies in the history of probability and statistics: VII. The principle of the arithmetic mean. Biometrika 45: 130-135. https://doi.org/10.2307/2333051.
Pollock, P. H., III, and B. C. Edwards. 2019. A Stata Companion to Political Analysis. 4th ed. Thousand Oaks, CA: CQ Press.

Scott, L. J., and C. A. Rogers. 2015. Creating summary tables using the sumtable command. Stata Journal 15: 775-783.

Stigler, S. M. 1977. Fractional order statistics, with applications. Journal of the American Statistical Association 72: 544-550. https://doi.org/10.2307/2286215.
Stuart, A., and J. K. Ord. 1994. Kendall's Advanced Theory of Statistics: Distribution Theory, Vol. 1. 6th ed. London: Arnold.

Thiele, T. N. 1889. Forelæsringer over Almindelig Iagttagelseslære: Sandsynlighedsregning og mindste Kvadraters Methode. Kjøbenhavn: C.A. Reitzel. (English translation included in Lauritzen 2002).
Weisberg, H. F. 1992. Central Tendency and Variability. Newbury Park, CA: Sage.

\section*{Also see}
[R] ameans - Arithmetic, geometric, and harmonic means
\([R]\) centile - Report centile and confidence interval
\([\mathrm{R}]\) ci - Confidence intervals for means, proportions, and variances
[R] mean - Estimate means
[R] proportion - Estimate proportions
[ R ] ratio - Estimate ratios
\([R]\) table - Table of frequencies, summaries, and command results
[R] tabstat - Compact table of summary statistics
\([R]\) tabulate, summarize() - One- and two-way tables of summary statistics
[R] total - Estimate totals
[D] codebook - Describe data contents
[D] describe - Describe data in memory or in a file
[D] inspect - Display simple summary of data's attributes
[ST] stsum - Summarize survival-time data
[SVY] svy estimation - Estimation commands for survey data
[XT] xtsum - Summarize xt data

\section*{Title}
sunflower - Density-distribution sunflower plots
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Acknowledgments & References
\end{tabular}

\section*{Description}
sunflower draws density-distribution sunflower plots (Plummer and Dupont 2003). Dark sunflowers, light sunflowers, and marker symbols represent high-, medium-, and low-density regions of the data, respectively. These plots are useful for displaying bivariate data whose density is too great for conventional scatterplots to be effective.

\section*{Quick start}

Density-distribution sunflower plot showing the relationship between x and y
sunflower y x
Set the center of the reference bin to \(x=5\) and \(y=8\)
sunflower y x, xcenter(5) ycenter (8)
Same as above, but specify the width of the hexagonal bins to be 1.5
sunflower y x, xcenter(5) ycenter(8) binwidth(1.5)
Same as above, but set the minimum number of observations needed for a bin to be represented by a light sunflower to 4
```

sunflower y x, xcenter(5) ycenter(8) binwidth(1.5) light(4)

```

Use the s1color scheme
sunflower y x, scheme(s1color)
Specify that only the petals are shown and the hexegons are omitted
sunflower y x, flowersonly
Suppress display of the table
sunflower y x, notable

\section*{Menu}

Graphics \(>\) Smoothing and densities \(>\) Density-distribution sunflower plot

\section*{Syntax}
sunflower yvar xvar [if] [in] [weight] [, options]
options Description
Main
nograph
notable
marker_options
Bins/Petals
binwidth(\#)
binar(\#)
bin_options
light(\#)
dark(\#)
xcenter (\#)
ycenter (\#)
petalweight(\#)
petallength(\#)
petal_options
flowersonly
nosinglepetal
Add plots
addplot (plot) add other plots to generated graph
Y axis, X axis, Titles, Legend, Overall, By
twoway_options any options documented in [G-3] twoway_options
bin_options Description
\([\underline{1} \mid \mathrm{d}]\) bstyle (areastyle) overall look of hexagonal bins
\([\underline{\mathrm{I}} \mid \mathrm{d}]\) bcolor (colorstyle) outline and fill color
\([\underline{1} \mid \underline{d}] \underline{\text { bfcolor (colorstyle) } \quad \text { fill color }}\)
\([\underline{1} \mid \underline{d}]\) blstyle(linestyle) overall look of outline
[ 1 |d \(]\) blcolor (colorstyle) outline color
[ \(\underline{1} \mid \mathrm{d}]\) blwidth(linewidthstyle) thickness of outline
petal_options
Description
[1]d flstyle (linestyle)
overall style of sunflower petals
[l|d] flcolor(colorstyle) color of sunflower petals
[l|d \(]\) flwidth(linewidthstyle) thickness of sunflower petals

All options are rightmost; see [G-4] Concept: repeated options.
fweights are allowed; see [U] 11.1.6 weight.

\section*{Options}

Main
nograph prevents the graph from being generated.
notable prevents the summary table from being displayed. This option is implied when the by () option is specified.
marker_options affect the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G-3] marker_options.

Bins/Petals
binwidth(\#) specifies the horizontal width of the hexagonal bins in the same units as xvar. By default,
\[
\text { binwidth }=\max (\mathrm{rbw}, \mathrm{nbw})
\]
where
\[
\begin{gathered}
\text { rbw }=\text { range of } x v a r / 40 \\
\text { nbw }=\text { range of } x \operatorname{var} / \max (1, \mathrm{nb})
\end{gathered}
\]
and
\[
\mathrm{nb}=\operatorname{int}(\min (\operatorname{sqrt}(n), 10 * \log 10(n)))
\]
where
\[
n=\text { the number of observations in the dataset }
\]
binar(\#) specifies the aspect ratio for the hexagonal bins. The height of the bins is given by
\[
\text { binheight }=\text { binwidth } \times \# \times 2 / \sqrt{3}
\]
where binheight and binwidth are specified in the units of yvar and xvar, respectively. The default is binar \((r)\), where \(r\) results in the rendering of regular hexagons.
bin_options affect how the hexagonal bins are rendered.
lbstyle(areastyle) and dbstyle(areastyle) specify the look of the light and dark hexagonal bins, respectively. The options listed below allow you to change each attribute, but lbstyle() and dbstyle() provide the starting points. See [G-4] areastyle for a list of available area styles.
lbcolor (colorstyle) and dbcolor (colorstyle) specify one color to be used both to outline the shape and to fill the interior of the light and dark hexagonal bins, respectively. See [G-4] colorstyle for a list of color choices.
lbfcolor (colorstyle) and dbfcolor (colorstyle) specify the color to be used to fill the interior of the light and dark hexagonal bins, respectively. See [G-4] colorstyle for a list of color choices.
lblstyle(linestyle) and dblstyle (linestyle) specify the overall style of the line used to outline the area, which includes its pattern (solid, dashed, etc.), thickness, and color. The other options listed below allow you to change the line's attributes, but lblstyle() and dblstyle() are the starting points. See [G-4] linestyle for a list of choices.
lblcolor (colorstyle) and dblcolor(colorstyle) specify the color to be used to outline the light and dark hexagonal bins, respectively. See [G-4] colorstyle for a list of color choices.
lblwidth(linewidthstyle) and dblwidth(linewidthstyle) specify the thickness of the line to be used to outline the light and dark hexagonal bins, respectively. See [G-4] linewidthstyle for a list of choices.
light(\#) specifies the minimum number of observations needed for a bin to be represented by a light sunflower. The default is light (3).
dark(\#) specifies the minimum number of observations needed for a bin to be represented by a dark sunflower. The default is dark(13).
xcenter(\#) and ycenter(\#) specify the center of the reference bin. The default values are the median values of xvar and yvar, respectively. The centers of the other bins are implicitly defined by the location of the reference bin together with the common bin width and height.
petalweight(\#) specifies the number of observations represented by each petal of a dark sunflower. The default value is chosen so that the maximum number of petals on a dark sunflower is 14 .
petallength(\#) specifies the length of petals in the sunflowers. The value specified is interpreted as a percentage of half the bin width. The default is \(100 \%\).
petal_options affect how the sunflower petals are rendered.
lflstyle(linestyle) and dflstyle(linestyle) specify the overall style of the light and dark sunflower petals, respectively.
lflcolor (colorstyle) and dflcolor (colorstyle) specify the color of the light and dark sunflower petals, respectively.
lflwidth(linewidthstyle) and dflwidth(linewidthstyle) specify the width of the light and dark sunflower petals, respectively.
flowersonly suppresses rendering of the bins. This option is equivalent to specifying lbcolor (none) and dbcolor (none).
nosinglepetal suppresses flowers from being drawn in light bins that contain only 1 observation and dark bins that contain as many observations as the petal weight (see the petalweight () option).

Add plots
addplot (plot) provides a way to add other plots to the generated graph; see [G-3] addplot_option.

\footnotetext{
Y axis, X axis, Titles, Legend, Overall, By
twoway_options are any of the options documented in [G-3] twoway_options. These include options for titling the graph (see [G-3] title_options), options for saving the graph to disk (see [G-3] saving_option), and the by () option (see [G-3] by_option).
}

\section*{Remarks and examples}

A sunflower is several line segments of equal length, called petals, that radiate from a central point. There are two varieties of sunflowers: light and dark. Each petal of a light sunflower represents 1 observation. Each petal of a dark sunflower represents several observations. Dark and light sunflowers represent high- and medium-density regions of the data, and marker symbols represent individual observations in low-density regions.
sunflower divides the plane defined by the variables yvar and xvar into contiguous hexagonal bins. The number of observations contained within a bin determines how the bin will be represented.
- When there are fewer than light (\#) observations in a bin, each point is plotted using the usual marker symbols in a scatterplot.
- Bins with at least light (\#) but fewer than dark(\#) observations are represented by a light sunflower. Each petal of a light sunflower represents one observation in the bin.
- Bins with at least dark(\#) observations are represented by a dark sunflower. Each petal of a dark sunflower represents multiple observations.
See Dupont (2009, 87-92) for a discussion of sunflower plots and how to create them using Stata.

\section*{Example 1}

Using the auto dataset, we want to examine the relationship between weight and mpg. To do that, we type



The two darkly shaded sunflowers immediately catch our eyes, indicating a group of 10 cars that are heavy (nearly 4,000 pounds) and fuel inefficient and a group 16 of cars that get about 20 miles per gallon and weight in the neighborhood of 3,000 pounds. The lighter sunflowers each indicate groups of three to seven cars that share similar weight and fuel economy characteristics. To obtain the number of cars in each group, we count the number of petals in each flower and consult the graph legend to see how many observations each petal represents.

\section*{Acknowledgments}

We thank William D. Dupont and W. Dale Plummer Jr., both of the Department of Biostatistics at Vanderbilt University, who are the authors of the original sunflower command, for their assistance in producing this version.

\section*{References}

Cleveland, W. S., and R. McGill. 1984. The many faces of a scatterplot. Journal of the American Statistical Association 79: 807-822. https://doi.org/10.2307/2288711.
Dupont, W. D. 2009. Statistical Modeling for Biomedical Researchers: A Simple Introduction to the Analysis of Complex Data. 2nd ed. Cambridge: Cambridge University Press.
Dupont, W. D., and W. D. Plummer, Jr. 2005. Using density-distribution sunflower plots to explore bivariate relationships in dense data. Stata Journal 5: 371-384.

Huang, C., J. A. McDonald, and W. Stuetzle. 1997. Variable resolution bivariate plots. Journal of Computational and Graphical Statistics 6: 383-396. https://doi.org/10.1080/10618600.1997.10474749.
Levy, D. E. 1999. 50 Years of Discovery: Medical Milestones from the National Heart, Lung, and Blood Institute's Framingham Heart Study. Hoboken, NJ: Center for Bio-Medical Communication.

Plummer, W. D., Jr., and W. D. Dupont. 2003. Density distribution sunflower plots. Journal of Statistical Software 8: 1-11. https://doi.org/10.18637/jss.v008.i03.
Steichen, T. J., and N. J. Cox. 1999. flower: Stata module to draw sunflower plots. Boston College Department of Economics, Statistical Software Components S393001. https://ideas.repec.org/c/boc/bocode/s393001.html.

\section*{Title}
sureg - Zellner's seemingly unrelated regression
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
sureg fits seemingly unrelated regression models (Zellner 1962; Zellner and Huang 1962; Zellner 1963). The acronyms SURE and SUR are often used for the estimator.

\section*{Quick start}

Model of y 1 and y 2 as a function of \(\mathrm{x} 1, \mathrm{x} 2\), and x 3
sureg (y1 x1 x2 x3) (y2 x1 x2 x3)
Same as above
sureg ( \(\mathrm{y} 1 \mathrm{y} 2=\mathrm{x} 1 \mathrm{x} 2 \mathrm{x} 3\) )
Same as above, but obtain small-sample statistics and use small-sample adjustment
sureg ( \(\mathrm{y} 1 \mathrm{y} 2=\mathrm{x} 1 \mathrm{x} 2 \mathrm{x} 3\) ), small dfk
Also perform Breusch-Pagan test
sureg ( \(\mathrm{y} 1 \mathrm{y} 2=\mathrm{x} 1 \mathrm{x} 2 \mathrm{x} 3\) ), small dfk corr
Model of y 1 as a function of x 1 and x 2 and y 2 as a function of x 2 and lag of x 2 using tsset data sureg (y1 x1 x2) (y2 x2 L.x2)

\section*{Menu}

Statistics \(>\) Linear models and related \(>\) Multiple-equation models \(>\) Seemingly unrelated regression

\section*{Syntax}

\section*{Basic syntax}
\[
\begin{aligned}
& \text { sureg }\left(\text { depvar }_{1} \text { varlist }\right)_{1}\left(\text { depvar }_{2} \text { varlist }_{2}\right) \ldots\left(\text { depvar }_{N} \text { varlist }_{N}\right) \\
& \quad[\text { if }][\text { in }][\text { weight }]
\end{aligned}
\]

Full syntax
\[
\begin{aligned}
& \text { sureg }^{\left(\left[\text {eqname }_{1}:\right] \text { depvar }_{1 a}\left[\text { depvar }_{1 b} \ldots=\right] \text { varlist }_{1}[\text {, noconstant }]\right)} \begin{array}{l}
\left(\left[\text { eqname }_{2}:\right] \text { depvar }_{2 a}\left[\text { depvar }_{2 b} \ldots=\right] \text { varlist }_{2}[\text {, noconstant }]\right) \\
\quad \ldots \\
\quad\left(\left[\text { eqname }_{N}:\right] \text { depvar }_{N a}\left[\text { depvar }_{N b} \ldots=\right] \text { varlist }_{N}[\text {, noconstant }]\right) \\
{[\text { if }][\text { in }][\text { weight }][\text {, options }]}
\end{array} .
\end{aligned}
\]

Explicit equation naming (eqname:) cannot be combined with multiple dependent variables in an equation specification.
options
Description

Model
isure
constraints (constraints)
SE/Robust
vce (vcetype)
df adj.
small
dfk
dfk2
Reporting
level(\#)
corr
nocnsreport
display_options
iterate until estimates converge
apply specified linear constraints
vcetype may be unadjusted, robust, or cluster clustvar
report small-sample statistics
use small-sample adjustment
use alternate adjustment
set confidence level; default is level (95)
perform Breusch-Pagan test
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

Optimization
optimization_options
noheader
notable
coeflegend
control the optimization process; seldom used
suppress header table from above coefficient table
suppress coefficient table
display legend instead of statistics
varlist \(_{1}, \ldots\), varlist \({ }_{N}\) may contain factor variables; see [U] 11.4.3 Factor variables. You must have the same levels of factor variables in all equations that have factor variables.
depvars and the varlists may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bootstrap, by, collect, fp, jackknife, rolling, and statsby are allowed; see [U] 11.1.10 Prefix commands.
Weights are not allowed with the bootstrap prefix; see \([R]\) bootstrap.
aweights are not allowed with the jackknife prefix; see [R] jackknife.
aweights and fweights are allowed; see [U] 11.1.6 weight.
noheader, notable, and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options}

\section*{Model}
isure specifies that sureg iterate over the estimated disturbance covariance matrix and parameter estimates until the parameter estimates converge. Under seemingly unrelated regression, this iteration converges to the maximum likelihood results. If this option is not specified, sureg produces two-step estimates.
```

constraints (constraints); see [R] Estimation options.

```

SE/Robust
vce(vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (unadjusted), that are robust to some kinds of misspecification (robust), and that allow for intragroup correlation (cluster clustvar); see [R] vce_option.
vce (unadjusted), the default, specifies that an unadjusted (nonrobust) VCE matrix be used; this results in efficient estimates when assuming homoskedasticity.
\(\sqrt{\mathrm{df} \text { adj. }}\)
small specifies that small-sample statistics be computed. It shifts the test statistics from \(\chi^{2}\) and \(z\) statistics to \(F\) statistics and \(t\) statistics. Although the standard errors from each equation are computed using the degrees of freedom for the equation, the degrees of freedom for the \(t\) statistics are all taken to be those for the first equation.
dfk specifies the use of an alternative divisor in computing the covariance matrix for the equation residuals. As an asymptotically justified estimator, sureg by default uses the number of sample observations ( \(n\) ) as a divisor. When the dfk option is set, a small-sample adjustment is made and the divisor is taken to be \(\sqrt{\left(n-k_{i}\right)\left(n-k_{j}\right)}\), where \(k_{i}\) and \(k_{j}\) are the number of parameters in equations \(i\) and \(j\), respectively.
dfk 2 specifies the use of an alternative divisor in computing the covariance matrix for the equation residuals. When the dfk2 option is set, the divisor is taken to be the mean of the residual degrees of freedom from the individual equations.

\section*{Reporting}
level(\#); see [R] Estimation options.
corr displays the correlation matrix of the residuals between equations and performs a Breusch-Pagan test for independent equations; that is, the disturbance covariance matrix is diagonal.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap (\#), fvwrapon (style), cformat (\% fint), pformat (\% fit ), sformat (\% fint), and nolstretch; see [R] Estimation options.
\(\sqrt{\text { Optimization }}\)
optimization_options control the iterative process that minimizes the sum of squared errors when isure is specified. These options are seldom used.
iterate(\#) specifies the maximum number of iterations. When the number of iterations equals \#, the optimizer stops and presents the current results, even if the convergence tolerance has not been reached. The default is the number set using set maxiter, which is 300 by default.
trace adds to the iteration \(\log\) a display of the current parameter vector.
log and nolog specify whether to display the iteration log. The iteration \(\log\) is displayed by default unless you used set iterlog off to suppress it; see set iterlog in [R] set iter.
tolerance(\#) specifies the tolerance for the coefficient vector. When the relative change in the coefficient vector from one iteration to the next is less than or equal to \#, the optimization process is stopped. tolerance \((1 \mathrm{e}-6)\) is the default.

The following options are available with sureg but are not shown in the dialog box:
noheader suppresses display of the header reporting \(F\) statistics, \(R^{2}\), and root mean squared error above the coefficient table.
notable suppresses display of the coefficient table.
coeflegend; see [R] Estimation options.

\section*{Remarks and examples}

Seemingly unrelated regression models are so called because they appear to be joint estimates from several regression models, each with its own error term. The regressions are related because the (contemporaneous) errors associated with the dependent variables may be correlated. Chapter 6 of Cameron and Trivedi (2022) contains a discussion of the seemingly unrelated regression model and the feasible generalized least-squares estimator underlying it.
\(>\) Example 1
When we fit models with the same set of right-hand-side variables, the seemingly unrelated regression results (in terms of coefficients and standard errors) are the same as fitting the models separately (using, say, regress). The same is true when the models are nested. Even in such cases, sureg is useful when we want to perform joint tests. For instance, let us assume that we think
\[
\begin{aligned}
\text { price } & =\beta_{0}+\beta_{1} \text { foreign }+\beta_{2} \text { length }+u_{1} \\
\text { weight } & =\gamma_{0}+\gamma_{1} \text { foreign }+\gamma_{2} \text { length }+u_{2}
\end{aligned}
\]

Because the models have the same set of explanatory variables, we could estimate the two equations separately. Yet, we might still choose to estimate them with sureg because we want to perform the joint test \(\beta_{1}=\gamma_{1}=0\).

We use the small and dfk options to obtain small-sample statistics comparable with regress or mvreg.
```

. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. sureg (price foreign length) (weight foreign length), small dfk
Seemingly unrelated regression

```
\begin{tabular}{lrrrrrrr}
\hline Equation & Obs & Params & RMSE & "R-squared" & F & P>F \\
\hline price & 74 & 2 & 2474.593 & 0.3154 & 16.35 & 0.0000 \\
weight & 74 & 2 & 250.2515 & 0.8992 & 316.54 & 0.0000 \\
\hline
\end{tabular}
\begin{tabular}{r|rrrrrr}
\hline & Coefficient & Std. err. & t & \(\mathrm{P}>\mid \mathrm{tI}\) & [95\% conf. interval] \\
\hline price & & & & & & \\
foreign & 2801.143 & 766.117 & 3.66 & 0.000 & 1286.674 & 4315.611 \\
length & 90.21239 & 15.83368 & 5.70 & 0.000 & 58.91219 & 121.5126 \\
_cons & -11621.35 & 3124.436 & -3.72 & 0.000 & -17797.77 & -5444.93 \\
\hline weight & & & & & & \\
foreign & -133.6775 & 77.47615 & -1.73 & 0.087 & -286.8332 & 19.4782 \\
length & 31.44455 & 1.601234 & 19.64 & 0.000 & 28.27921 & 34.60989 \\
_cons & -2850.25 & 315.9691 & -9.02 & 0.000 & -3474.861 & -2225.639 \\
\hline
\end{tabular}

Note: Small-sample degrees-of-freedom adjustment applied when estimating covariance matrix of residuals.

These two equations have a common set of regressors, and we could have used a shorthand syntax to specify the equations:
```

. sureg (price weight = foreign length), small dfk

```

Here the results presented by sureg are the same as if we had estimated the equations separately:
```

. regress price foreign length
(output omitted)
. regress weight foreign length
(output omitted)

```

There is, however, a difference. We have allowed \(u_{1}\) and \(u_{2}\) to be correlated and have estimated the full variance-covariance matrix of the coefficients. sureg has estimated the correlations, but it does not report them unless we specify the corr option. We did not remember to specify corr when we fit the model, but we can redisplay the results:
```

. sureg, notable noheader corr
Note: Small-sample degrees-of-freedom adjustment applied when estimating
covariance matrix of residuals.
Correlation matrix of residuals:
rrice weight
Breusch-Pagan test of independence: chi2(1) = 25.237, Pr = 0.0000

```

The notable and noheader options prevented sureg from redisplaying the header and coefficient tables. We find that, for the same cars, the correlation of the residuals in the price and weight equations is 0.5840 and that we can reject the hypothesis that this correlation is zero.

We can test that the coefficients on foreign are jointly zero in both equations-as we set out to do-by typing test foreign; see \([R]\) test. When we type a variable without specifying the equation, that variable is tested for zero in all equations in which it appears:
```

. test foreign
( 1) [price]foreign = 0
( 2) [weight]foreign = 0
F( 2, 142) = 17.99
Prob > F = 0.0000

```

\section*{Example 2}

When the models do not have the same set of explanatory variables and are not nested, sureg may lead to more efficient estimates than running the models separately as well as allowing joint tests. This time, let us assume that we believe
\[
\begin{aligned}
\text { price } & =\beta_{0}+\beta_{1} \text { foreign }+\beta_{2} \text { mpg }+\beta_{3} \text { displ }+u_{1} \\
\text { weight } & =\gamma_{0}+\gamma_{1} \text { foreign }+\gamma_{2} \text { length }+u_{2}
\end{aligned}
\]

To fit this model, we type
. sureg (price foreign mpg displ) (weight foreign length), corr
Seemingly unrelated regression
\begin{tabular}{|c|c|c|c|c|c|c|c|}
\hline Equation & Obs & Params & RMSE & \multicolumn{2}{|l|}{"R-squared"} & chi2 & P>chi2 \\
\hline price & 74 & 3 & 2165.321 & \multicolumn{2}{|r|}{0.4537} & 49.64 & 0.0000 \\
\hline weight & 74 & 2 & 245.2916 & \multicolumn{2}{|r|}{0.8990} & 661.84 & 0.0000 \\
\hline & Coefficient & Std. err. & z & \(P>|z|\) & \multicolumn{3}{|l|}{[95\% conf. interval]} \\
\hline \multicolumn{8}{|l|}{price} \\
\hline foreign & 3058.25 & 685.7357 & 4.46 & 0.000 & & 233 & 4402.267 \\
\hline mpg & -104.9591 & 58.47209 & -1.80 & 0.073 & -219 & 5623 & 9.644042 \\
\hline displacement & 18.18098 & 4.286372 & 4.24 & 0.000 & & 㖪 & 26.58211 \\
\hline _cons & 3904.336 & 1966.521 & 1.99 & 0.047 & & 0263 & 7758.645 \\
\hline \multicolumn{8}{|l|}{weight} \\
\hline foreign & -147.3481 & 75.44314 & -1.95 & 0.051 & -295 & 2139 & . 517755 \\
\hline length & 30.94905 & 1.539895 & 20.10 & 0.000 & & 3091 & 33.96718 \\
\hline _cons & -2753.064 & 303.9336 & -9.06 & 0.000 & -33 & . 763 & -2157.365 \\
\hline
\end{tabular}

Correlation matrix of residuals:
price weight
price 1.0000
weight 0.32851 .0000
Breusch-Pagan test of independence: chi2(1) \(=7.984, \operatorname{Pr}=0.0047\)

In comparison, if we had fit the price model separately,
. regress price foreign mpg displ
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline Source & SS & df & MS & Number of obs & = & 74 \\
\hline & & & & F (3, 70) & = & 20.13 \\
\hline Model & 294104790 & 3 & 98034929.9 & Prob > F & = & 0.0000 \\
\hline Residual & 340960606 & 70 & 4870865.81 & R-squared & = & 0.4631 \\
\hline & & & & Adj R-squared & = & 0.4401 \\
\hline Total & 635065396 & 73 & 8699525.97 & Root MSE & = & 2207 \\
\hline
\end{tabular}
\begin{tabular}{r|rlccrr}
\hline price & Coefficient & Std. err. & t & P>|t| & [95\% conf. interval] \\
\hline foreign & 3545.484 & 712.7763 & 4.97 & 0.000 & 2123.897 & 4967.072 \\
mpg & -98.88559 & 63.17063 & -1.57 & 0.122 & -224.8754 & 27.10426 \\
displacement & 22.40416 & 4.634239 & 4.83 & 0.000 & 13.16146 & 31.64686 \\
_cons & 2796.91 & 2137.873 & 1.31 & 0.195 & -1466.943 & 7060.763 \\
\hline
\end{tabular}

The coefficients are slightly different, but the standard errors are uniformly larger. This would still be true if we specified the dfk option to make a small-sample adjustment to the estimated covariance of the disturbances.

\section*{- Technical note}

Constraints can be applied to SURE models using Stata's standard syntax for constraints. For a general discussion of constraints, see [R] constraint; for examples similar to seemingly unrelated regression models, see \([R]\) reg3.

\section*{Stored results}
sureg stores the following in e() :
Scalars
e(N)
e(k)
e(k_eq)
e(mss_\#)
e(df_m\#)
e(rss_\#)
e(df_r)
e(r2_\#)
e(F_\#)
e(rmse_\#)
e(dfk2_adj)
e(11)
e(N_clust)
e(chi2_\#)
e(p_\#)
e(cons_\#)
e(chi2_bp)
e(df_bp)
e(rank)
e(ic)
number of observations
number of parameters
number of equations in \(e(b)\)
model sum of squares for equation \#
model degrees of freedom for equation \#
residual sum of squares for equation \#
residual degrees of freedom
\(R^{2}\) for equation \#
\(F\) statistic for equation \# (small only)
root mean squared error for equation \#
divisor used with VCE when dfk2 specified
log likelihood
number of clusters
\(\chi^{2}\) for equation \#
\(p\)-value for equation \#
1 if equation \# has a constant, 0 otherwise
Breusch-Pagan \(\chi^{2}\)
degrees of freedom for Breusch-Pagan \(\chi^{2}\) test
rank of \(e(V)\)
number of iterations
```

Macros
e(cmd)
e(cmdline)
e(method)
e(depvar)
e(exog)
e(eqnames)
e(wtype)
e(wexp)
e(corr)
e(small)
e(dfk)
e(clustvar)
e(vce)
e(vcetype)
e(properties)
e(predict)
e(marginsok)
e(marginsnotok)
e(marginsdefault)
e(asbalanced)
e(asobserved)
Matrices
e(b)
e(Cns)
e(Sigma)
e(V)
Functions
e(sample) marks estimation sample

```

In addition to the above, the following is stored in r() :
Matrices
\(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

\section*{Methods and formulas}
sureg uses the asymptotically efficient, feasible, generalized least-squares algorithm described in Greene (2018, 328-339). The computing formulas are given on page 328-335.

The \(R^{2}\) reported is the percent of variance explained by the predictors. It may be used for descriptive purposes, but \(R^{2}\) is not a well-defined concept when GLS is used.
sureg will refuse to compute the estimators if the same equation is named more than once or the covariance matrix of the residuals is singular.

The Breusch and Pagan (1980) \(\chi^{2}\) statistic—a Lagrange multiplier statistic—is given by
\[
\lambda=T \sum_{m=2}^{M} \sum_{n=1}^{m-1} r_{m n}^{2}
\]
where \(r_{m n}\) is the estimated correlation between the residuals of the \(M\) equations and \(T\) is the number of observations. It is distributed as \(\chi^{2}\) with \(M(M-1) / 2\) degrees of freedom.

This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce(robust) and vce(cluster clustvar), respectively. See [P] _robust, particularly Introduction and Methods and formulas.

Arnold Zellner (1927-2010) was born in New York. He studied physics at Harvard and economics at Berkeley, and then he taught economics at the Universities of Washington and Wisconsin before settling in Chicago in 1966. Among his many major contributions to econometrics and statistics are his work on seemingly unrelated regression, three-stage least squares, and Bayesian econometrics.

\section*{References}

Breusch, T. S., and A. R. Pagan. 1980. The Lagrange multiplier test and its applications to model specification in econometrics. Review of Economic Studies 47: 239-253. https://doi.org/10.2307/2297111.

Cameron, A. C., and P. K. Trivedi. 2022. Microeconometrics Using Stata. 2nd ed. College Station, TX: Stata Press. Greene, W. H. 2018. Econometric Analysis. 8th ed. New York: Pearson.
McDowell, A. W. 2004. From the help desk: Seemingly unrelated regression with unbalanced equations. Stata Journal 4: 442-448.

Rossi, P. E. 1989. The ET interview: Professor Arnold Zellner. Econometric Theory 5: 287-317. https://doi.org/10.1017/S0266466600012445.
Zellner, A. 1962. An efficient method of estimating seemingly unrelated regressions and tests for aggregation bias. Journal of the American Statistical Association 57: 348-368. https://doi.org/10.2307/2281644.
__. 1963. Estimators for seemingly unrelated regression equations: Some exact finite sample results. Journal of the American Statistical Association 58: 977-992. https://doi.org/10.2307/2283326.
Zellner, A., and D. S. Huang. 1962. Further properties of efficient estimators for seemingly unrelated regression equations. International Economic Review 3: 300-313. https://doi.org/10.2307/2525396.

\footnotetext{
Also see
[R] sureg postestimation - Postestimation tools for sureg
[R] nlsur - Estimation of nonlinear systems of equations
[R] reg3 - Three-stage estimation for systems of simultaneous equations
\([R]\) regress - Linear regression
[MV] mvreg - Multivariate regression
[SEM] Example 12 - Seemingly unrelated regression
[SEM] Intro 5 - Tour of models
[TS] dfactor - Dynamic-factor models
[U] 20 Estimation and postestimation commands
}

\section*{Title}
sureg postestimation - Postestimation tools for sureg
Postestimation commands predict margins Remarks and examples

\section*{Postestimation commands}

The following postestimation commands are available after sureg:
\begin{tabular}{|c|c|}
\hline Command & Description \\
\hline contrast & contrasts and ANOVA-style joint tests of estimates \\
\hline estat ic & Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) \\
\hline estat summarize & summary statistics for the estimation sample \\
\hline estat vce & variance-covariance matrix of the estimators (VCE) \\
\hline estimates & cataloging estimation results \\
\hline etable & table of estimation results \\
\hline forecast & dynamic forecasts and simulations \\
\hline hausman & Hausman's specification test \\
\hline lincom & point estimates, standard errors, testing, and inference for linear combinations of coefficients \\
\hline margins & marginal means, predictive margins, marginal effects, and average marginal effects \\
\hline marginsplot & graph the results from margins (profile plots, interaction plots, etc.) \\
\hline nlcom & point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients \\
\hline predict & predictions and their SEs, residuals, etc. \\
\hline predictnl & point estimates, standard errors, testing, and inference for generalized predictions \\
\hline pwcompare & pairwise comparisons of estimates \\
\hline test & Wald tests of simple and composite linear hypotheses \\
\hline testnl & Wald tests of nonlinear hypotheses \\
\hline
\end{tabular}

\section*{predict}

\section*{Description for predict}
predict creates a new variable containing predictions such as linear predictions, standard errors, residuals, and differences between the linear predictions.

\section*{Menu for predict}

Statistics \(>\) Postestimation

\section*{Syntax for predict}
predict [type] newvar [if] [in] [, equation(eqno[,eqno]) statistic]
statistic Description
Main
xb
stdp
residuals
linear prediction; the default
standard error of the linear prediction
difference
stddp residuals
difference between the linear predictions of two equations
standard error of the difference in linear predictions
These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

\section*{Options for predict}

equation(eqno[,eqno]) specifies to which equation(s) you are referring.
equation() is filled in with one eqno for the xb , stdp, and residuals options. equation(\#1) would mean that the calculation is to be made for the first equation, equation(\#2) would mean the second, and so on. You could also refer to the equations by their names. equation (income) would refer to the equation named income and equation(hours) to the equation named hours. If you do not specify equation(), the results are the same as if you specified equation(\#1).
difference and stddp refer to between-equation concepts. To use these options, you must specify two equations, for example, equation(\#1,\#2) or equation(income,hours). When two equations must be specified, equation() is required.
xb , the default, calculates the linear prediction (fitted values) - the prediction of \(\mathbf{x}_{j} \mathbf{b}\) for the specified equation.
stdp calculates the standard error of the prediction for the specified equation. It can be thought of as the standard error of the predicted expected value or mean for the observation's covariate pattern. The standard error of the prediction is also referred to as the standard error of the fitted value.
residuals calculates the residuals.
difference calculates the difference between the linear predictions of two equations in the system. With equation(\#1,\#2), difference computes the prediction of equation(\#1) minus the prediction of equation(\#2).
stddp is allowed only after you have previously fit a multiple-equation model. The standard error of the difference in linear predictions ( \(\mathbf{x}_{1 j} \mathbf{b}-\mathbf{x}_{2 j} \mathbf{b}\) ) between equations 1 and 2 is calculated.

For more information on using predict after multiple-equation estimation commands, see \([R]\) predict.

\section*{margins}

\section*{Description for margins}
margins estimates margins of response for linear predictions and differences between the linear predictions.

\section*{Menu for margins}

Statistics \(>\) Postestimation

\section*{Syntax for margins}
```

margins [marginlist] [, options]

```

statistic Description
\begin{tabular}{ll} 
default & linear predictions for each equation \\
xb & linear prediction for a specified equation \\
difference & difference between the linear predictions of two equations \\
stdp & not allowed with margins \\
residuals & not allowed with margins \\
stddp & not allowed with margins
\end{tabular}
xb defaults to the first equation.
Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

\section*{Remarks and examples}

For an example of cross-equation testing of parameters using the test command, see example 1 in [R] sureg.

\section*{> Example 1}

In example 1 of \([R]\) sureg, we fit a seemingly unrelated regressions model of price and weight. Here we obtain the fitted values.
```

. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. sureg (price foreign length) (weight foreign length), small dfk
(output omitted)
. predict phat, equation(price)
(option xb assumed; fitted values)
. predict what, equation(weight)
(option xb assumed; fitted values)

```
. summarize price phat weight what
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline price & 74 & 6165.257 & 2949.496 & 3291 & 15906 \\
phat & 74 & 6165.257 & 1656.407 & 1639.872 & 9398.138 \\
weight & 74 & 3019.459 & 777.1936 & 1760 & 4840 \\
what & 74 & 3019.459 & 736.9666 & 1481.199 & 4476.331
\end{tabular}

Just as in single-equation OLS regression, in a SURE model the sample mean of the fitted values for an equation equals the sample mean of the dependent variable.

\section*{> Example 2}

Suppose that for whatever reason we were interested in the difference between the predicted values of price and weight. predict has an option to compute this difference in one step:
. predict diff, equation(price, weight) difference
diff is the same as phat - what:
\begin{tabular}{|c|c|c|c|c|c|}
\hline Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline diff & 74 & 3145.797 & 1233.26 & -132. 2275 & 5505.914 \\
\hline mydiff & 74 & 3145.797 & 1233.26 & -132.2275 & 5505.914 \\
\hline
\end{tabular}

\section*{Also see}
[R] sureg - Zellner's seemingly unrelated regression
[U] 20 Estimation and postestimation commands

\section*{Title}
swilk - Shapiro-Wilk and Shapiro-Francia tests for normality
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options for swilk & Options for sfrancia & Remarks and examples & Stored results \\
Methods and formulas & Acknowledgment & References & Also see
\end{tabular}

\section*{Description}
swilk performs the Shapiro-Wilk \(W\) test for normality for each variable in the specified varlist.
Likewise, sfrancia performs the Shapiro-Francia \(W^{\prime}\) test for normality. See [MV] mvtest normality for multivariate tests of normality.

\section*{Quick start}

Shapiro-Wilk test of normality
Shapiro-Wilk test for v1
swilk v1
Separate tests of normality for v 1 and v 2
swilk v1 v2
Generate new variable w containing \(W\) test coefficients
swilk v1, generate(w)
Specify that average ranks should not be used for tied values
swilk v1 v2, noties
Test that v3 is distributed lognormally
generate \(\operatorname{lnv} 3=\ln (v 3)\)
swilk lnv3

Shapiro-Francia test of normality
Shapiro-Francia test for v1
sfrancia v1
Separate tests of normality for v 1 and v 2
sfrancia v1 v2
Same as above, but use the Box-Cox transformation
sfrancia v1 v2, boxcox
Specify that average ranks should not be used for tied values
sfrancia v1 v2, noties

\section*{Menu}
swilk
Statistics \(>\) Summaries, tables, and tests \(>\) Distributional plots and tests \(>\) Shapiro-Wilk normality test
sfrancia
Statistics \(>\) Summaries, tables, and tests \(>\) Distributional plots and tests \(>\) Shapiro-Francia normality test

\section*{Syntax}

Shapiro-Wilk normality test
```

swilk varlist [if] [in] [, swilk_options]

```

Shapiro-Francia normality test
sfrancia varlist \([\) if \(][\) in \(][\), sfrancia_options \(]\)
swilk_options Description
Main
generate (newvar) create newvar containing \(W\) test coefficients
Innormal test for three-parameter lognormality
noties
do not use average ranks for tied values
sfrancia_options
Description
Main
boxcox use the Box-Cox transformation for \(W^{\prime}\); the default is to use the log transformation
noties
do not use average ranks for tied values
by and collect are allowed with swilk and sfrancia; see [U] 11.1.10 Prefix commands.

\section*{Options for swilk}
\(\qquad\) Main
generate (newvar) creates new variable newvar containing the \(W\) test coefficients.
Innormal specifies that the test be for three-parameter lognormality, meaning that \(\ln (X-k)\) is tested for normality, where \(k\) is calculated from the data as the value that makes the skewness coefficient zero. When simply testing \(\ln (X)\) for normality, do not specify this option. See [R] lnskew0 for estimation of \(k\).
noties suppresses use of averaged ranks for tied values when calculating the \(W\) test coefficients.

\section*{Options for sfrancia}
\(\qquad\) Main
boxcox specifies that the Box-Cox transformation of Royston (1983) for calculating \(W^{\prime}\) test coefficients be used instead of the default log transformation (Royston 1993a). Under the Box-Cox transformation, the normal approximation to the sampling distribution of \(W^{\prime}\), used by sfrancia, is valid for \(5 \leq n \leq 1000\). Under the log transformation, it is valid for \(10 \leq n \leq 5000\).
noties suppresses use of averaged ranks for tied values when calculating the \(W^{\prime}\) test coefficients.

\section*{Remarks and examples}
swilk can be used with \(4 \leq n \leq 2000\) observations. sfrancia can be used with \(10 \leq n \leq 5000\) observations; however, if the boxcox option is specified, it can be used with \(5 \leq n \leq 1000\) observations.

Also see \([\mathrm{R}]\) sktest for the skewness and kurtosis test described by D'Agostino, Belanger, and D'Agostino (1990) with the empirical correction developed by Royston (1991b). While the Shapiro-Wilk and Shapiro-Francia tests for normality are, in general, preferred for nonaggregated data (Gould and Rogers 1991; Gould 1992b; Royston 1991b), the skewness and kurtosis test will permit more observations. Moreover, a normal quantile plot should be used with any test for normality; see [ R ] Diagnostic plots for more information.

\section*{> Example 1}

Using our automobile dataset, we will test whether the variables mpg and trunk are normally distributed:
```

. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. swilk mpg trunk

```
            Shapiro-Wilk W test for normal data
\begin{tabular}{r|ccccc} 
Variable & Obs & W & V & z & Prob>z \\
\hline mpg & 74 & 0.94821 & 3.335 & 2.627 & 0.00430 \\
trunk & 74 & 0.97921 & 1.339 & 0.637 & 0.26215 \\
. sfrancia mpg trunk & & & & &
\end{tabular}
\begin{tabular}{|c|c|c|c|c|c|}
\hline \multirow[b]{2}{*}{Variable} & \multicolumn{5}{|l|}{Shapiro-Francia W' test for normal data} \\
\hline & Obs & W & V' & z & Prob>z \\
\hline mpg & 74 & 0.94872 & 3.650 & 2.510 & 0.00604 \\
\hline trunk & 74 & 0.98446 & 1.106 & 0.195 & 0.42271 \\
\hline
\end{tabular}

We can reject the hypothesis that mpg is normally distributed, but we cannot reject that trunk is normally distributed.

The values reported under \(W\) and \(W^{\prime}\) are the Shapiro-Wilk and Shapiro-Francia test statistics. The tests also report \(V\) and \(V^{\prime}\) (Royston 1991d), which are more appealing indexes for departure from normality. The median values of \(V\) and \(V^{\prime}\) are 1 for samples from normal populations. Large values indicate nonnormality. There is no more information in \(V\left(V^{\prime}\right)\) than in \(W\left(W^{\prime}\right)\)-one is just the transform of the other.

\section*{> Example 2}

We have data on a variable called studytime, which we suspect is distributed lognormally:
```

. use https://www.stata-press.com/data/r18/cancer
(Patient survival in drug trial)
. generate lnstudytime = ln(studytime)
. swilk lnstudytime
Shapiro-Wilk W test for normal data

| Variable | Obs | W | V | z | Prob>z |
| ---: | ---: | :---: | :---: | :---: | :---: |
| Instudytime | 48 | 0.92731 | 3.311 | 2.547 | 0.00543 |

```

We can reject the lognormal assumption. We do not specify the lnnormal option when testing for lognormality. The lnnormal option is for three-parameter lognormality.

\section*{> Example 3}

Having discovered that \(\ln\) (studytime) is not distributed normally, we now test that \(\ln (\) studytime \(-k)\) is normally distributed, where \(k\) is chosen so that the resulting skewness is zero. We obtain the estimate for \(k\) from lnskew0; see [R] Inskew0:
. InskewO lnstudytimek = studytime, level(95)
\begin{tabular}{r|rrrr} 
Transform & k & [95\% conf. interval] & Skewness \\
\hline \(\ln (\) studytim-k) & -11.01181 & -infinity & -.9477328 & -.0000173
\end{tabular}
. swilk lnstudytimek, lnnormal
Shapiro-Wilk W test for 3-parameter lognormal data
\begin{tabular}{r|ccccc} 
Variable & Obs & W & V & z & Prob>z \\
\hline lnstudytimek & 48 & 0.97064 & 1.337 & 1.261 & 0.10363
\end{tabular}

We cannot reject the hypothesis that \(\ln (\) studytime +11.01181\()\) is distributed normally. We do specify the lnnormal option when using an estimated value of \(k\).

\section*{Stored results}
swilk and sfrancia store the following in \(r()\) :
Scalars
\begin{tabular}{llll}
\(\mathrm{r}(\mathrm{N})\) & number of observations & \(\mathrm{r}(\mathrm{W})\) & \(W\) or \(W^{\prime}\) \\
\(\mathrm{r}(\mathrm{p})\) & \(p\)-value & \(\mathrm{r}(\mathrm{V})\) & \(V\) or \(V^{\prime}\) \\
\(\mathrm{r}(\mathrm{z})\) & \(z\) statistic & &
\end{tabular}

\section*{Methods and formulas}

The Shapiro-Wilk test is based on Shapiro and Wilk (1965) with a new approximation accurate for \(4 \leq n \leq 2000\) (Royston 1992). The calculations made by swilk are based on Royston (1982, 1992, 1993b).

The Shapiro-Francia test (Shapiro and Francia 1972; Royston 1983; Royston 1993a) is an approximate test that is similar to the Shapiro-Wilk test for very large samples.

The relative merits of the Shapiro-Wilk and Shapiro-Francia tests the versus skewness and kurtosis test have been a subject of debate. The interested reader is directed to the articles in the Stata Technical Bulletin. Our recommendation is to use the Shapiro-Francia test whenever possible, that is, whenever dealing with nonaggregated or ungrouped data (Gould and Rogers 1991; Gould 1992b); see [R] swilk. If normality is rejected, use sktest to determine the source of the problem. As both D'Agostino, Belanger, and D'Agostino (1990) and Royston (1991c) mention, researchers should also examine the normal quantile plot to determine normality rather than blindly relying on a few test statistics. See the qnorm command documented in \([\mathrm{R}]\) Diagnostic plots for more information on normal quantile plots.

Samuel Sanford Shapiro (1930-2023) earned degrees in statistics and engineering from City College of New York, Columbia, and Rutgers. After employment in the US Army and industry, he joined the faculty at Florida International University in 1972. Shapiro has coauthored various texts in statistics and published several papers on distributional testing and other statistical topics.

\section*{Acknowledgment}
swilk and sfrancia were written by Patrick Royston of the MRC Clinical Trials Unit, London and coauthor of the Stata Press book Flexible Parametric Survival Analysis Using Stata: Beyond the Cox Model.

\section*{References}

Brzezinski, M. 2012. The Chen-Shapiro test for normality. Stata Journal 12: 368-374.
D’Agostino, R. B., A. J. Belanger, and R. B. D'Agostino, Jr. 1990. A suggestion for using powerful and informative tests of normality. American Statistician 44: 316-321. https://doi.org/10.2307/2684359.
Genest, C., and G. J. Brackstone. 2010. A conversation with Martin Bradbury Wilk. Statistical Science 25: 258-273. https://doi.org/10.1214/08-STS272.

Gould, W. W. 1992a. sg3.7: Final summary of tests of normality. Stata Technical Bulletin 5: 10-11. Reprinted in Stata Technical Bulletin Reprints, vol. 1, pp. 114-115. College Station, TX: Stata Press.
__. 1992b. sg11.1: Quantile regression with bootstrapped standard errors. Stata Technical Bulletin 9: 19-21. Reprinted in Stata Technical Bulletin Reprints, vol. 2, pp. 137-139. College Station, TX: Stata Press.
Gould, W. W., and W. H. Rogers. 1991. sg3.4: Summary of tests of normality. Stata Technical Bulletin 3: 20-23. Reprinted in Stata Technical Bulletin Reprints, vol. 1, pp. 106-110. College Station, TX: Stata Press.

Royston, P. 1982. An extension of Shapiro and Wilks's \(W\) test for normality to large samples. Applied Statistics 31: 115-124. https://doi.org/10.2307/2347973.
-_ 1983. A simple method for evaluating the Shapiro-Francia \(W^{\prime}\) test of non-normality. Statistician 32: 297-300. https://doi.org/10.2307/2987935.
——. 1991a. sg3.2: Shapiro-Wilk and Shapiro-Francia tests. Stata Technical Bulletin 3: 19. Reprinted in Stata Technical Bulletin Reprints, vol. 1, p. 105. College Station, TX: Stata Press.
__. 1991b. sg3.5: Comment on sg3.4 and an improved D'Agostino test. Stata Technical Bulletin 3: 23-24. Reprinted in Stata Technical Bulletin Reprints, vol. 1, pp. 110-112. College Station, TX: Stata Press.

1991c. sg3.6: A response to sg3.3: Comment on tests of normality. Stata Technical Bulletin 4: 8-9. Reprinted in Stata Technical Bulletin Reprints, vol. 1, pp. 112-114. College Station, TX: Stata Press.
_-. 1991d. Estimating departure from normality. Statistics in Medicine 10: 1283-1293.
https://doi.org/10.1002/sim. 4780100811.
_-. 1992. Approximating the Shapiro-Wilk \(W\)-test for non-normality. Statistics and Computing 2: 117-119. https://doi.org/10.1007/BF01891203.

1993a. A pocket-calculator algorithm for the Shapiro-Francia test for non-normality: An application to medicine. Statistics in Medicine 12: 181-184. https://doi.org/10.1002/sim.4780120209.
——. 1993b. A toolkit for testing for non-normality in complete and censored samples. Statistician 42: 37-43. https://doi.org/10.2307/2348109.

Shapiro, S. S., and R. S. Francia. 1972. An approximate analysis of variance test for normality. Journal of the American Statistical Association 67: 215-216. https://doi.org/10.1080/01621459.1972.10481232.
Shapiro, S. S., and M. B. Wilk. 1965. An analysis of variance test for normality (complete samples). Biometrika 52: 591-611. https://doi.org/10.2307/2333709.

\section*{Also see}
[R] Inskew0 - Find zero-skewness log or Box-Cox transform
[R] lv - Letter-value displays
[R] sktest - Skewness and kurtosis tests for normality
[R] Diagnostic plots - Distributional diagnostic plots
[MV] mvtest normality - Multivariate normality tests

\section*{Title}
symmetry - Symmetry and marginal homogeneity tests
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
symmetry performs asymptotic symmetry and marginal homogeneity tests, as well as an exact symmetry test on \(K \times K\) tables where there is a 1 -to-1 matching of cases and controls (nonindependence). This testing is used to analyze matched-pair case-control data with multiple discrete levels of the exposure (outcome) variable. In genetics, the test is known as the transmission/disequilibrium test (TDT) and is used to test the association between transmitted and nontransmitted parental marker alleles to an affected child (Spieldman, McGinnis, and Ewens 1993). For \(2 \times 2\) tables, the asymptotic test statistics reduce to the McNemar test statistic, and the exact symmetry test produces an exact McNemar test; see [R] Epitab. For many exposure variables, symmetry can optionally perform a test for linear trend in the log relative risk.
symmetry expects the data to be in the wide format; that is, each observation contains the matched case and control values in variables casevar and controlvar. Variables can be numeric or string.
symmi is the immediate form of symmetry. The symmi command uses the values specified on the command line; rows are separated by ' \(\backslash\) ', and options are the same as for symmetry. See [U] 19 Immediate commands for a general introduction to immediate commands.

\section*{Quick start}

Symmetry and marginal homogeneity tests for 1-to-1 matched case-control studies
symmetry case control
Same as above
symmetry control case
Exact test of table symmetry
symmetry case control, exact
Report the contribution from each off-diagonal pair to the overall \(\chi^{2}\)-statistic
symmetry control case, contrib
Test for a linear trend in the \(\log\) of the relative risk
symmetry control case, trend
Request marginal homogeneity statistics that do not require the inversion of the variance-covariance matrix
symmetry case control, mh
Using frequency weight variable wvar
symmetry case control [fweight=wvar]

\section*{Menu}

\section*{symmetry}

Statistics \(>\) Epidemiology and related \(>\) Other \(>\) Symmetry and marginal homogeneity test

\section*{symmi}

Statistics \(>\) Epidemiology and related \(>\) Other \(>\) Symmetry and marginal homogeneity test calculator

\section*{Syntax}

Symmetry and marginal homogeneity tests symmetry casevar controlvar [if] [in] [weight] [, options]

Immediate form of symmetry and marginal homogeneity tests
symmi \(\#_{11} \#_{12}[\ldots] \backslash \#_{21} \#_{22}[\ldots][\backslash \ldots][\) if \(][\) in \(][\), options \(]\)
options
Description

Main
notable
contrib
exact
mh
trend
cc
suppress output of contingency table report contribution of each off-diagonal cell pair perform exact test of table symmetry perform two marginal homogeneity tests perform a test for linear trend in the (log) relative risk (RR) use continuity correction when calculating test for linear trend
collect is allowed with symmetry; see [U] 11.1.10 Prefix commands.
fweights are allowed; see [U] 11.1.6 weight.

\section*{Options}

Main
notable suppresses the output of the contingency table. By default, symmetry displays the \(n \times n\) contingency table at the top of the output.
contrib reports the contribution of each off-diagonal cell pair to the overall symmetry \(\chi^{2}\). exact performs an exact test of table symmetry. This option is recommended for sparse tables. CAUTION: The exact test requires substantial amounts of time and memory for large tables.
mh performs two marginal homogeneity tests that do not require the inversion of the variance-covariance matrix.

By default, symmetry produces the Stuart-Maxwell test statistic, which requires the inversion of the nondiagonal variance-covariance matrix, \(\mathbf{V}\). When the table is sparse, the matrix may not be of full rank, and then the command substitutes a generalized inverse \(\mathbf{V}^{*}\) for \(\mathbf{V}^{-1}\). mh calculates optional marginal homogeneity statistics that do not require the inversion of the variance-covariance matrix. These tests may be preferred in certain situations. See Methods and formulas and Bickeböller and Clerget-Darpoux (1995) for details on these test statistics.
trend performs a test for linear trend in the (log) relative risk (RR). This option is allowed only for numeric exposure (outcome) variables, and its use should be restricted to measurements on the ordinal or the interval scales.
cc specifies that the continuity correction be used when calculating the test for linear trend. This correction should be specified only when the levels of the exposure variable are equally spaced.

\section*{Remarks and examples}
symmetry and symmi may be used to analyze 1-to-1 matched case-control data with multiple discrete levels of the exposure (outcome) variable.

\section*{> Example 1}

Consider a survey of 344 individuals (BMDP 1990, 267-270) who were asked in October 1986 whether they agreed with President Reagan's handling of foreign affairs. In January 1987, after the Iran-Contra affair became public, these same individuals were surveyed again and asked the same question. We would like to know if public opinion changed over this period.

We first describe the dataset and list a few observations.


Sorted by:
. list in \(1 / 5\)
\begin{tabular}{|rr|}
\hline before & after \\
\hline Agree & Agree \\
Agree & Disagree \\
Agree & Unsure \\
Disagree & Agree \\
Disagree & Disagree \\
\hline
\end{tabular}

Each observation corresponds to one of the 344 individuals. The data are in wide form so that each observation has a before and an after measurement. We now perform the test without options.
. symmetry before after
\begin{tabular}{r|cccc}
\hline \begin{tabular}{l} 
Public \\
opinion \\
before IC
\end{tabular} & \multicolumn{4}{c}{ Public opinion after IC } \\
Agree & Disagree & Unsure & Total \\
\hline Agree & 47 & 56 & 38 & 141 \\
\begin{tabular}{r} 
Disagree \\
Unsure
\end{tabular} & 28 & 61 & 31 & 120 \\
Total & 101 & 47 & 10 & 83 \\
\hline
\end{tabular}
\begin{tabular}{l|ccr} 
& chi2 & df & Prob>chi2 \\
\hline Symmetry (asymptotic) (Stuart-Maxwell) & 14.87 & 3 & 0.0019 \\
Marginal homogeneity (Stu & 14.78 & 2 & 0.0006 \\
\hline
\end{tabular}

The test first tabulates the data in a \(K \times K\) table and then performs Bowker's (1948) test for table symmetry and the Stuart-Maxwell (Stuart 1955; Maxwell 1970) test for marginal homogeneity.

Both the symmetry test and the marginal homogeneity test are highly significant, thus indicating a shift in public opinion.

An exact test of symmetry is provided for use on sparse tables. This test is computationally intensive, so it should not be used on large tables. Because we are working on a fast computer, we will run the symmetry test again and this time include the exact option. We will suppress the output of the contingency table by specifying notable and include the contrib option so that we may further examine the cells responsible for the significant result.
\begin{tabular}{|c|c|c|c|}
\hline \begin{tabular}{rl} 
& \begin{tabular}{r} 
Contribution \\
to symmetry \\
chi-squared
\end{tabular}
\end{tabular} & & & \\
\hline n1_2 \& n2_1 9.3333 & & & \\
\hline n1_3 \& n3_1 2.2500 & & & \\
\hline n2_3 \& n3_2 3.2821 & & & \\
\hline & chi2 & df & Prob>chi2 \\
\hline Symmetry (asymptotic) & 14.87 & 3 & 0.0019 \\
\hline Marginal homogeneity (Stuart-Maxwell) & 14.78 & 2 & 0.0006 \\
\hline Marginal homogeneity (Bickenboller) & 13.53 & 2 & 0.0012 \\
\hline Marginal homogeneity (no diagonals) & 15.25 & 2 & 0.0005 \\
\hline \multicolumn{3}{|l|}{Symmetry (exact significance probability)} & 0.0018 \\
\hline
\end{tabular}

The largest contribution to the symmetry \(\chi^{2}\) is due to cells \(n_{12}\) and \(n_{21}\). These correspond to changes between the agree and disagree categories. Of the 344 individuals, 56 ( \(16.3 \%\) ) changed from the agree to the disagree response, whereas only \(28(8.1 \%)\) changed in the opposite direction.

For these data, the results from the exact test are similar to those from the asymptotic test.

\section*{> Example 2}

Breslow and Day (1980, 163) reprinted data from Mack et al. (1976) from a case-control study of the effect of exogenous estrogen on the risk of endometrial cancer. The data consist of 59 elderly women diagnosed with endometrial cancer and 59 disease-free control subjects living in the same community as the cases. Cases and controls were matched on age, marital status, and time living in the community. The data collected included information on the daily dose of conjugated estrogen therapy. Breslow and Day analyzed these data by creating four levels of the dose variable. Here are the data as entered into a Stata dataset:
. use https://www.stata-press.com/data/r18/bd163
. list, noobs divider
\begin{tabular}{|r|r|r|}
\hline case & control & count \\
\hline 0 & 0 & 6 \\
0 & \(0.1-0.299\) & 2 \\
0 & \(0.3-0.625\) & 3 \\
0 & \(0.626+\) & 1 \\
\(0.1-0.299\) & 0 & 9 \\
\hline \(0.1-0.299\) & \(0.1-0.299\) & 4 \\
\(0.1-0.299\) & \(0.3-0.625\) & 2 \\
\(0.1-0.299\) & \(0.626+\) & 1 \\
\(0.3-0.625\) & 0 & 9 \\
\(0.3-0.625\) & \(0.1-0.299\) & 2 \\
\hline \(0.3-0.625\) & \(0.3-0.625\) & 3 \\
\(0.3-0.625\) & \(0.626+\) & 1 \\
\(0.626+\) & 0 & 12 \\
\(0.626+\) & \(0.1-0.299\) & 1 \\
\(0.626+\) & \(0.3-0.625\) & 2 \\
\hline \(0.626+\) & \(0.626+\) & 1 \\
\hline
\end{tabular}

This dataset is in a different format from that of the previous example. Instead of each observation representing one matched pair, each observation represents possibly multiple pairs indicated by the count variable. For instance, the first observation corresponds to six matched pairs where neither the case nor the control was on estrogen, the second observation corresponds to two matched pairs where the case was not on estrogen and the control was on 0.1 to \(0.299 \mathrm{mg} /\) day, etc.

To use symmetry to analyze this dataset, we must specify fweight to indicate that in our data there are observations corresponding to more than one matched pair.
```

. symmetry case control [fweight=count]

```
\begin{tabular}{l|ccccc}
\hline \begin{tabular}{l} 
Dosage \\
level for \\
case
\end{tabular} & 0 & \(0.1-0.299\) & \(0.3-0.625\) & \(0.626+\) & Total \\
\hline 0 & 6 & 2 & 3 & 1 & 12 \\
\(0.1-0.299\) & 9 & 4 & 2 & 1 & 16 \\
\(0.3-0.625\) & 9 & 2 & 3 & 1 & 15 \\
\(0.626+\) & 12 & 1 & 2 & 1 & 16 \\
Total & 36 & 9 & 10 & 4 & 59 \\
\hline
\end{tabular}
\begin{tabular}{l|ccr}
\hline & chi2 & df & Prob>chi2 \\
\hline Symmetry (asymptotic) (Stuart-Maxwell) & 17.10 & 6 & 0.0089 \\
Marginal homogeneity (Stu & 16.96 & 3 & 0.0007 \\
\hline
\end{tabular}

Both the test of symmetry and the test of marginal homogeneity are highly significant, thus leading us to reject the null hypothesis that there is no effect of exposure to estrogen on the risk of endometrial cancer.

Breslow and Day perform a test for trend assuming that the estrogen exposure levels were equally spaced by recoding the exposure levels as \(1,2,3\), and 4.

We can easily reproduce their results by recoding our data in this way and by specifying the trend option. Two new numeric variables were created, ca and co, corresponding to the variables case and control, respectively. Below, we list some of the data and our results from symmetry:
```

. encode case, gen(ca)
. encode control, gen(co)
. label values ca
. label values co
. list in 1/4

```
\begin{tabular}{l|rrrrr|}
\cline { 2 - 6 } & case & control & count & ca & co \\
\cline { 2 - 6 } 1. & 0 & 0 & 6 & 1 & 1 \\
2. & 0 & \(0.1-0.299\) & 2 & 1 & 2 \\
3. & 0 & \(0.3-0.625\) & 3 & 1 & 3 \\
4. & 0 & \(0.626+\) & 1 & 1 & 4 \\
\hline
\end{tabular}
. symmetry ca co [fw=count], notable trend cc
\begin{tabular}{l|ccr} 
& chi2 & df & Prob>chi2 \\
\hline Symmetry (asymptotic) (Stuart-Maxwell) & 17.10 & 6 & 0.0089 \\
Marginal homogeneity \\
\hline Linear trend in the (log) RR & 14.96 & 3 & 0.0007 \\
\hline
\end{tabular}

We requested the continuity correction by specifying cc. Doing so is appropriate because our coded exposure levels are equally spaced.

The test for trend was highly significant, indicating an increased risk of endometrial cancer with increased dosage of conjugated estrogen.

You must be cautious: the way in which you code the exposure variable affects the linear trend statistic. If instead of coding the levels as \(1,2,3\), and 4 , we had instead used \(0,0.2,0.46\), and 0.7
(roughly the midpoint in the range of each level), we would have obtained a \(\chi^{2}\) statistic of 11.19 for these data.

\section*{Stored results}
symmetry stores the following in \(r()\) :
Scalars
\(r\) (N_pair) number of matched pairs
\(r\) (chi2) asymptotic symmetry \(\chi^{2}\)
\(r(d f) \quad\) asymptotic symmetry degrees of freedom
\(r(p) \quad\) asymptotic symmetry \(p\)-value
r(chi2_sm) MH (Stuart-Maxwell) \(\chi^{2}\)
\(r\left(d f \_s m\right) \quad\) MH (Stuart-Maxwell) degrees of freedom
\(r\left(\mathrm{p} \_\mathrm{sm}\right) \quad\) MH (Stuart-Maxwell) \(p\)-value
\(r\) (chi2_b) MH (Bickenböller) \(\chi^{2}\)
\(r\left(d f \_b\right) \quad\) MH (Bickenböller) degrees of freedom
r(p_b) MH (Bickenböller) p-value
\(r\) (chi2_nd) \(\quad\) MH (no diagonals) \(\chi^{2}\)
\(r\left(d f \_n d\right) \quad\) MH (no diagonals) degrees of freedom
\(r\) (p_nd) \(\quad\) MH (no diagonals) \(p\)-value
\(r\left(\right.\) chi2 \(\_\)t) \(\quad \chi^{2}\) for linear trend
\(r\) (p_trend) \(\quad p\)-value for linear trend
\(r\) (p_exact) exact symmetry \(p\)-value

\section*{Methods and formulas}

Methods and formulas are presented under the following headings:
Asymptotic tests
Exact symmetry test

\section*{Asymptotic tests}

Consider a square table with \(K\) exposure categories, that is, \(K\) rows and \(K\) columns. Let \(n_{i j}\) be the count corresponding to row \(i\) and column \(j\) of the table, \(N_{i j}=n_{i j}+n_{j i}\), for \(i, j=1,2, \ldots, K\), and \(n_{i}\), and let \(n_{. j}\) be the marginal totals for row \(i\) and column \(j\), respectively. Asymptotic tests for symmetry and marginal homogeneity for this \(K \times K\) table are calculated as follows:

The null hypothesis of complete symmetry \(p_{i j}=p_{j i}, i \neq j\), is tested by calculating the test statistic (Bowker 1948)
\[
T_{\mathrm{cs}}=\sum_{i<j} \frac{\left(n_{i j}-n_{j i}\right)^{2}}{n_{i j}+n_{j i}}
\]
which is asymptotically distributed as \(\chi^{2}\) with \(K(K-1) / 2-R\) degrees of freedom, where \(R\) is the number of off-diagonal cells with \(N_{i j}=0\) as discussed in Hoenig, Morgan, and Brown (1995).

The null hypothesis of marginal homogeneity, \(p_{i .}=p_{. i}\), is tested by calculating the Stuart-Maxwell test statistic (Stuart 1955; Maxwell 1970),
\[
T_{\mathrm{sm}}=\mathbf{d}^{\prime} \mathbf{V}^{-1} \mathbf{d}
\]
where \(\mathbf{d}\) is a column vector with elements equal to the differences \(d_{i}=n_{i .}-n_{. i}\) for \(i=1,2, \ldots, K\), and \(\mathbf{V}\) is the variance-covariance matrix with elements
\[
\begin{aligned}
v_{i i} & =n_{i .}+n_{. i}-2 n_{i i} \\
v_{i j} & =-\left(n_{i j}+n_{j i}\right), \quad i \neq j
\end{aligned}
\]
\(T_{\mathrm{sm}}\) is asymptotically \(\chi^{2}\) with \(K-1\) degrees of freedom.
This test statistic properly accounts for the dependence between the table's rows and columns. When the matrix \(\mathbf{V}\) is not of full rank, a generalized inverse \(\mathbf{V}^{*}\) is substituted for \(\mathbf{V}^{-1}\).

The Bickeböller and Clerget-Darpoux (1995) marginal homogeneity test statistic is calculated by
\[
T_{\mathrm{mh}}=\sum_{i} \frac{\left(n_{i .}-n_{. i}\right)^{2}}{n_{i .}+n_{. i}}
\]

This statistic is asymptotically distributed, under the assumption of marginal independence, as \(\chi^{2}\) with \(K-1\) degrees of freedom.

The marginal homogeneity (no diagonals) test statistic \(T_{\mathrm{mh}}^{0}\) is calculated in the same way as \(T_{\mathrm{mh}}\), except that the diagonal elements do not enter into the calculation of the marginal totals. Unlike the previous test statistic, \(T_{\mathrm{mh}}^{0}\) reduces to a McNemar test statistic for \(2 \times 2\) tables. The test statistic \(\{(K-1) / 2\} T_{\mathrm{mh}}^{0}\) is asymptotically distributed as \(\chi^{2}\) with \(K-1\) degrees of freedom (Cleves, Olson, and Jacobs 1997; Spieldman and Ewens 1996).

Breslow and Day's test statistic for linear trend in the (log) of RR is
\[
\frac{\left\{\sum_{i<j}\left(n_{i j}-n_{j i}\right)\left(X_{j}-X_{i}\right)-c c\right\}^{2}}{\sum_{i<j}\left(n_{i j}+n_{j i}\right)\left(X_{j}-X_{i}\right)^{2}}
\]
where the \(X_{j}\) are the doses associated with the various levels of exposure and \(c c\) is the continuity correction; it is asymptotically distributed as \(\chi^{2}\) with 1 degree of freedom.

The continuity correction option is applicable only when the levels of the exposure variable are equally spaced.

\section*{Exact symmetry test}

The exact test is based on a permutation algorithm applied to the null distribution. The distribution of the off-diagonal elements \(n_{i j}, i \neq j\), conditional on the sum of the complementary off-diagonal cells, \(N_{i j}=n_{i j}+n_{j i}\), can be written as the product of \(K(K-1) / 2\) binomial random variables,
\[
P(\mathbf{n})=\prod_{i<j}\binom{N_{i j}}{n_{i j}} \pi_{i j}^{n_{i j}}\left(1-\pi_{i j}\right)^{n_{i j}}
\]
where \(\mathbf{n}\) is a vector with elements \(n_{i j}\) and \(\pi_{i j}=E\left(n_{i j} / N_{i j} \mid N_{i j}\right)\). Under the null hypothesis of complete symmetry, \(\pi_{i j}=\pi_{j i}=1 / 2\), and thus the permutation distribution is given by
\[
P_{0}(\mathbf{n})=\prod_{i<j}\binom{N_{i j}}{n_{i j}}\left(\frac{1}{2}\right)^{N_{i j}}
\]

The exact significance test is performed by evaluating
\[
P_{\mathrm{cs}}=\sum_{n \in p} P_{0}(\mathbf{n})
\]
where \(p=\left\{n: P_{0}(\mathbf{n})<P_{0}\left(\mathbf{n}^{*}\right)\right\}\) and \(\mathbf{n}^{*}\) is the observed contingency table data vector. The algorithm evaluates \(p_{\text {cs }}\) exactly. For information about permutation tests, see Good (2005, 2006).

\section*{References}

Bickeböller, H., and F. Clerget-Darpoux. 1995. Statistical properties of the allelic and genotypic transmission/disequilibrium test for multiallelic markers. Genetic Epidemiology 12: 865-870. https://doi.org/10.1002/gepi. 1370120656.

BMDP. 1990. BMDP Statistical Software Manual. Oakland, CA: University of California Press.
Bowker, A. H. 1948. A test for symmetry in contingency tables. Journal of the American Statistical Association 43: 572-574. https://doi.org/10.2307/2280710.
Breslow, N. E., and N. E. Day. 1980. Statistical Methods in Cancer Research: Vol. 1-The Analysis of Case-Control Studies. Lyon: IARC.

Cleves, M. A., J. M. Olson, and K. B. Jacobs. 1997. Exact transmission-disequilibrium tests with multiallelic markers. Genetic Epidemiology 14: 337-347. https://doi.org/10.1002/(SICI)1098-2272(1997)14:4〈337::AID-GEPI1〉 3.0.CO;2-0.

Good, P. I. 2005. Permutation, Parametric, and Bootstrap Tests of Hypotheses: A Practical Guide to Resampling Methods for Testing Hypotheses. 3rd ed. New York: Springer.
——. 2006. Resampling Methods: A Practical Guide to Data Analysis. 3rd ed. Boston: Birkhäuser.
Hoenig, J. M., M. J. Morgan, and C. A. Brown. 1995. Analysing differences between two age determination methods by tests of symmetry. Canadian Journal of Fisheries and Aquatic Sciences 52: 364-368. https://doi.org/10.1139/f95-038.
Mack, T. M., M. C. Pike, B. E. Henderson, R. I. Pfeffer, V. R. Gerkins, M. Arthur, and S. E. Brown. 1976. Estrogens and endometrial cancer in a retirement community. New England Journal of Medicine 294: 1262-1267. https://doi.org/10.1056/NEJM197606032942304.
Maxwell, A. E. 1970. Comparing the classification of subjects by two independent judges. British Journal of Psychiatry 116: 651-655. https://doi.org/10.1192/bjp.116.535.651.

Spieldman, R. S., and W. J. Ewens. 1996. The TDT and other family-based tests for linkage disequilibrium and association. American Journal of Human Genetics 59: 983-989.

Spieldman, R. S., R. E. McGinnis, and W. J. Ewens. 1993. Transmission test for linkage disequilibrium: The insulin gene region and insulin-dependence diabetes mellitus (IDDM). American Journal of Human Genetics 52: 506-516.
Stuart, A. 1955. A test for homogeneity of the marginal distributions in a two-way classification. Biometrika 42: 412-416. https://doi.org/10.2307/2333387.

\section*{Also see}
[R] Epitab - Tables for epidemiologists

\section*{Title}
table intro - Introduction to tables of frequencies, summaries, and command results
Description Remarks and examples Reference Also see

\section*{Description}

Tables allow us to effectively communicate information about our data and results from our analyses. The table command is a flexible tool for creating tables. It allows you to create tabulations, tables of summary statistics, tables of results from hypothesis tests, tables of regression results, and more. table also allows you to customize the table so that it effectively communicates the results.

\section*{Remarks and examples}

Remarks are presented under the following headings:

\author{
Overview \\ Tabulations \\ Tables of summary statistics \\ Tables of results from other commands \\ Flexible tables combining results \\ Formatting, customizing, and exporting tables
}

\section*{Overview}

The table command allows you to create various types of tables.
1. Tabulations of one, two, or more categorical variables, reporting frequencies, percentages, or proportions.
2. Tables of summary statistics such as means, standard deviations, medians, and the likeperhaps computed across levels of one or more categorical variables.
3. Tables of results from other Stata commands, such as regression results or results of classic hypothesis tests.
4. Tables combining summary statistics and results from other Stata commands.
table obtains the statistics it reports in two ways. The table command itself can compute summary statistics. table also provides a command() option that allows you to run any Stata command and include its results in the table.
table also provides much flexibility for you to arrange the results in your table in a meaningful way. You can control how the rows, columns, and potentially even separate tables are defined. For instance, you might type
```

. table (var1) (var2) ...

```
to place the levels of var1 on the rows and the levels of var2 on the columns. The first set of parentheses is used to define the rows, and the second set is used to define the columns. The parentheses could be omitted in this case, but for clarity we will use them in our discussion here.

We could use the levels of both var1 and var2 to define the rows.
```

. table (var1 var2) ...

```

If we have an additional variable, var3, we can define rows by both var1 and var2 and the columns by var3 by typing
```

. table (var1 var2) (var3) ...

```
or we can place the levels of var2 and var3 on the columns,
```

. table (var1) (var2 var3)...

```

We could even create separate tables for the levels of var3,
```

. table (var1) (var2) (var3) ...

```

The third set of parentheses defines the separate tables.
If our table reports multiple statistics, say, means and standard deviations, we can use the keyword result to specify how they are included in the table layout. To place the statistics on separate rows, we could type
```

. table (var1 result) (var2) ...

```

The flexibility of the table layout goes beyond these examples. You can add additional variables to define rows, columns, and tables as is appropriate for the table you wish to create.

In addition to controlling the layout of your table, you can customize the results by specifying formats, including stars representing significance, and selecting from styles that determine how the headers of the table are displayed.

\section*{Tabulations}

Tabulations allow you to examine the distribution of your data across the levels of one or more categorical variables. Tabulations often report frequencies, percentages, and proportions.

With table, you can easily create one-way tabulations, two-way tabulations, three-way tabulations, and even more complex tabulations. For example, we could create a two-way tabulation of region and diabetes, reporting frequencies,
. table (region) (diabetes)
\begin{tabular}{l|rrr}
\hline & \multicolumn{3}{|c}{ Diabetes status } \\
& Not diabetic & Diabetic & Total \\
\hline Region & & & \\
NE & 1,997 & 98 & 2,095 \\
MW & 2,648 & 125 & 2,773 \\
S & 2,692 & 161 & 2,853 \\
W & 2,513 & 115 & 2,628 \\
Total & 9,850 & 499 & 10,349 \\
\hline
\end{tabular}
or we can report percentages,
\begin{tabular}{l|ccc}
\multicolumn{4}{c}{ table (region) (diabetes), statistic(percent) } \\
& \multicolumn{3}{c}{ Diabetes status } \\
& Not diabetic & Diabetic & Total \\
\hline Region & & & \\
NE & 19.30 & 0.95 & 20.24 \\
MW & 25.59 & 1.21 & 26.79 \\
S & 26.01 & 1.56 & 27.57 \\
W & 24.28 & 1.11 & 25.39 \\
Total & 95.18 & 4.82 & 100.00 \\
\hline
\end{tabular}

To learn how to create tabulations, see
[R] table oneway
[R] table twoway
[R] table multiway

One-way tabulation
Two-way tabulation
Multiway tables

In these entries, we provide simplified syntax for creating the specific type of tabulation you are interested in, and we provide a number of examples demonstrating how to build these tables and customize the results.

\section*{Tables of summary statistics}
table can compute summary statistics such as minimums, maximums, means, standard deviations, medians, and interquartile ranges. Summary statistics can be computed for one or more variables. Summary statistics can also be computed separately for groups of data. For example, we can create a table reporting the means of age, height, and bmi,
```

. table, statistic(mean age height bmi)

```
\begin{tabular}{l|r}
\hline Age (years) & 47.57965 \\
Height (cm) & 167.6509 \\
Body mass index (BMI) & 25.5376 \\
\hline
\end{tabular}
or we can compute these means separately for males and females,
. table () (sex), statistic(mean age height bmi)
\begin{tabular}{l|rrr}
\hline & Male & Sex \\
& Female & Total \\
\hline Age (years) & 47.4238 & 47.72057 & 47.57965 \\
Height (cm) & 174.7421 & 161.2393 & 167.6509 \\
Body mass index (BMI) & 25.50999 & 25.56256 & 25.5376 \\
\hline
\end{tabular}

To learn how to create tables of summary statistics, including the simplified table syntax and worked examples, see
[ R\(]\) table summary
Table of summary statistics
You can also create a table of summary statistics with dtable. The advantage of using dtable is that you can create a table with a title and notes and export it to a variety of file types, all in a single step. Unlike table, however, dtable has a predefined layout, and you will not be able to specify how the results should be arranged in the table.

\section*{Tables of results from other commands}

With table's command () option, you can run any Stata command and place statistics reported by that command in your table. For instance, you might use test to test for differences in means or use regress to fit a linear regression model. You can include any stored results from these commands in your table. For example, we can create a table to compare coefficients from linear regressions of bpsysol on age and weight fit separately for males and females.
```

. table (colname) (sex result), command(regress bpsystol age weight)

```
\begin{tabular}{l|rrr}
\hline & Male & \begin{tabular}{c} 
Sex \\
Female
\end{tabular} & Total \\
\hline Age (years) & .4789361 & .7735499 & .6379892 \\
Weight (kg) & .3346106 & .4586108 & .4069041 \\
Intercept & 84.08037 & 61.70456 & 71.27096 \\
\hline
\end{tabular}

To learn how to create tables with results from other commands, including simplified table syntax and worked examples, see
[R] table hypothesis tests
\([R]\) table regression

Table of hypothesis tests
Table of regression results

\section*{Flexible tables combining results}
table can create tables with combinations of frequencies, other summary statistics, and results from other table commands. To see the full syntax and discussion of how table automates the table layout, see
[R] table
Table of frequencies, summaries, and command results

\section*{Formatting, customizing, and exporting tables}
table allows you to customize your results by specifying the layout and with options that change the numeric formatting, add characters such as percent signs or parentheses to specific types of statistics, add stars representing significance levels, and modify the look of confidence intervals. In addition, Stata provides predefined styles that you may select from using the style() option. These styles control which labels are displayed in the headers of the tables, how the labels are aligned, how the statistics are aligned within the cells, and more. To learn about the predefined styles that you can select from, see [TABLES] Predefined styles. Examples of using the style() and other formatting options are provided in the entries for specific types of tables listed in the previous sections.

The results from table can be customized even beyond what its formatting options allow. table is unique in that it stores its results in a format that we refer to as a "collection". Stata's collect suite of commands can be used to produce highly customized tables from results in a collection and to export those tables to presentation-ready formats such as HTML, Word, LATEX, PDF, Excel, and more. Examples of using collect to modify labels and table layout are also provided in the entries for the specific types of tables listed in the previous sections. To learn more about the collect commands, see [TABLES] Intro and the entries discussed therein.

\section*{Reference}

Huber, C. 2021. Customizable tables in Stata 17, part 1: The new table command. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2021/06/07/customizable-tables-in-stata-17-part-1-the-new-table-command/.

\section*{Also see}
[R] table - Table of frequencies, summaries, and command results [TABLES] Intro - Introduction

\section*{Title}
table oneway - One-way tabulation
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & References \\
Also see & & &
\end{tabular}

\section*{Description}

In this entry, we discuss how to use table to create a one-way tabulation, including frequencies, percentages, and proportions.

\section*{Quick start}

One-way table of frequencies, with rows corresponding to the levels of a1 table a1

Same as above, but with columns corresponding to the levels of a1
table () a1
Same as above, but treat missing values like other values of a1
table () a1, missing
One-way table of frequencies, using the collection style mystyle
table a1, style(mystyle)
One-way table of frequencies and percentages
table a1, statistic(frequency) statistic (percent)

\section*{Menu}

Statistics \(>\) Summaries, tables, and tests \(>\) Tables of frequencies, summaries, and command results

\section*{Syntax}

Basic one-way tabulation
table varname [if] [in] [weight] [, options]

Customized one-way tabulation
table \([\) (rowspec) \(][\) (colspec \()][\) if \(][\) in \(][\) weight \(][\), options \(]\)
rowspec and colspec may be empty or may include varname, result, or varname and result, where result refers to the requested statistics.
options

Main
nototals

Statistics
statistic (stat)

Description
suppress the marginal totals
statistic to be reported; default is statistic (frequency) when no weights are specified and statistic (sumw) otherwise

\section*{Formats}
nformat (\% fint [results] [, basestyle]) specify numeric format sformat (sfmt [results])

Options
missing
name (cname)
append
replace
label(filename)
style(filename [, override])
markvar (newvar)
treat numeric missing values of varname like other values
collect results into a collection named cname append results to an existing collection
replace results of an existing collection
specify the collection labels
specify the collection style
create newvar that identifies observations used in the tabulation

\footnotetext{
fweights, aweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
strL variables are not allowed; see [U] 12.4.8 strL.
markvar() does not appear in the dialog box.
}

\section*{Options}

\section*{Main}
nototals prevents table from displaying the row or column totals.

Statistics
statistic (stat) specifies the statistic to be displayed. statistic() may be repeated to request multiple statistics.
Available statistics are
\begin{tabular}{ll} 
stat & Definition \\
\hline \begin{tabular}{l} 
frequency
\end{tabular} & frequency \\
sumw & sum of weights \\
proportion & proportion \\
\hline percent & percentage \\
\begin{tabular}{l} 
rawproportion \\
rawpercent
\end{tabular} & \begin{tabular}{l} 
proportion ignoring optionally specified weights \\
\hline
\end{tabular} \\
\hline
\end{tabular}

Formats
nformat ( \(\%\) fint [results][, basestyle]) changes the numeric format, such as the number of decimal places, for specified results. If results are not specified, the numeric format is changed for all results.
results may be any statistic named in option statistic() (that is, any stat).
This option is repeatable, and when multiple formats apply to one result, the rightmost specification is applied.

This option does not affect the format of numeric layout variables (rowspec and colspec). The default format of these variables is taken from the dataset.
basestyle indicates that the format be applied to results that do not already have their own format instead of overriding the format for all results.
sformat (sfmt [results]) changes the string format for specified results. You can, for instance, add symbols or text to the values reported in the table by modifying the string format.
sfint may contain a mix of text and \(\%\) s. Here \(\%\) s refers to the numeric value that is formatted as specified using nformat(). The text will be placed around the numeric values in your table as it is placed around \(\% \mathrm{~s}\) in this option. For instance, to place parentheses around the percent statistics, you can specify sformat(" (\%s)" percent).
results may be any statistic named in option statistic() (that is, any stat).
Two text characters must be specified using a special character sequence if you want them to be displayed in your table. To include \(\%\), type \(\% \%\). To include \(\backslash\), type \(\backslash \backslash\). For instance, to place a percent sign following percent statistics, you can specify sformat("\%s\%\%" percent).
This option is repeatable, and when multiple formats apply to one result, the rightmost specification is applied.

Options
missing specifies that numeric missing values of varname be treated as valid categories. By default, observations with a numeric missing value in varname are omitted.
name (cname) specifies that a collection named cname be associated with the collected statistics and results. The default is name (Table).
append specifies that table append its collection information into the collection named in name().
replace permits table to overwrite an existing collection. This option is implied for name (Table) when append is not specified.
label (filename) specifies the filename containing the collection labels to use for your table. Labels in filename will be loaded for the table, and any labels not specified in filename will be taken from the labels defined in \(c\) (collect_label). The default is to use only the collection labels set in c(collect_label); see [TABLES] set collect_label.
style(filename [, override]) specifies the filename containing the collection styles to use for your table. The default collection styles will be discarded, and only the collection styles in filename will be applied.
If you prefer the default collection styles but also want to apply any styles in filename, specify override. If there are conflicts between the default collection styles and those in filename, the ones in filename will take precedence.

The default is to use only the collection styles set in c (table_style); see [TABLES] set table_style.
The following option is available with table but is not shown in the dialog box:
markvar (newvar) generates an indicator variable that identifies the observations used in the tabulation.

\section*{Remarks and examples}

Remarks are presented under the following headings:
Tabulation of one variable
Tabulation, including percentages
Customizing results
Advanced customization

\section*{Tabulation of one variable}

To obtain a one-way tabulation that reports the number of observations for each level of a categorical variable, we need specify only the name of the categorical variable following table.

To demonstrate, we use data from the Second National Health and Nutrition Examination Survey (NHANES II) (McDowell et al. 1981). We tabulate the hlthstat variable, which contains individuals' self-reported health status categories.
```

. use https://www.stata-press.com/data/r18/nhanes2l
(Second National Health and Nutrition Examination Survey)
. table hlthstat

```
\begin{tabular}{l|r}
\hline & Frequency \\
\hline Health status & \\
Excellent & 2,407 \\
Very good & 2,591 \\
Good & 2,938 \\
Fair & 1,670 \\
Poor & 729 \\
Total & 10,335 \\
\hline
\end{tabular}

We see that more people self-reported having excellent, very good, or good health status than reported having fair or poor health status.

Above, we see frequencies for those who reported a health status. This information is not available for some individuals in the dataset. We can determine how many missing values we have for this variable by adding the missing option
. table hlthstat, missing
\begin{tabular}{l|r}
\hline & Frequency \\
\hline Health status & \\
Excellent & 2,407 \\
Very good & 2,591 \\
Good & 2,938 \\
Fair & 1,670 \\
Poor & 729 \\
Blank but applicable & 2 \\
Total & 14 \\
\hline
\end{tabular}

We find that there is missing health status data for 16 individuals- 2 with a generic missing value and 14 whose responses were labeled "Blank but applicable".

\section*{Tabulation, including percentages}

In addition to frequencies, we can report the proportion or percentage of observations in each health status category. By default, table reports frequencies, which is equivalent to including the statistic (frequency) option. Here we include that option along with the statistic (percent) option to report both frequencies and percentages.
. table hlthstat, statistic(frequency) statistic(percent)
\begin{tabular}{l|rr}
\hline & Frequency & Percent \\
\hline Health status & & \\
Excellent & 2,407 & 23.29 \\
Very good & 2,591 & 25.07 \\
Good & 2,938 & 28.43 \\
Fair & 1,670 & 16.16 \\
Poor & 729 & 7.05 \\
Total & 10,335 & 100.00 \\
\hline
\end{tabular}

Now, it is clear that \(28.43 \%\) of respondents reported having good health.

\section*{Customizing results}

There are a number of ways that you can customize the results in your table.
In some cases, you may prefer to place frequencies and percentages on the rows and the levels of the variable being tabulated on the columns. To do this, you can include both the row and column specifications in parentheses following table. Here we use result in the first set of parentheses to request that the statistics be placed on rows and the variable hlthstat in the second set of parentheses to request that the levels of this variable be placed on the columns.
. table (result) (hlthstat), statistic(frequency) statistic(percent)
\begin{tabular}{l|rrrrrrr}
\hline & & \multicolumn{6}{|c}{ Health status } \\
& Excellent & Very good & Good & Fair & Poor & Total \\
\hline Frequency & 2,407 & 2,591 & 2,938 & 1,670 & 729 & 10,335 \\
Percent & 23.29 & 25.07 & 28.43 & 16.16 & 7.05 & 100.00 \\
\hline
\end{tabular}

Alternatively, we could have omitted result and typed
```

. table () (hlthstat), statistic(freq) statistic(percent)

```

Because we requested that hlthstat be moved to the columns by specifying it in the second set of parentheses, table automatically moves the requested statistics to the rows.

If instead of a short and wide table, you prefer a tall and narrow table, you can specify that both the levels of hlthstat and the statistics be used to define the rows by including the variable name and result in the first set of parentheses.
\begin{tabular}{l|l}
\multicolumn{2}{l}{ table (hlthstat result), statistic(frequency) statistic(percent) } \\
\hline Health status & \\
Excellent & \\
Frequency & 2,407 \\
Percent & 23.29 \\
Very good & \\
Frequency & 2,591 \\
Percent & 25.07 \\
Good & \\
Frequency & 2,938 \\
Percent & 28.43 \\
Fair & \\
Frequency & 1,670 \\
Percent & 16.16 \\
Poor & \\
Frequency & 729 \\
Percent & 7.05 \\
Total & \\
Frequency & 10,335 \\
Percent & 100.00 \\
\hline
\end{tabular}

In addition to modifying the layout of the table, we may want to customize the results reported within the cells of the table. For instance, we can specify that the percentages be reported using only one decimal place by using the nformat () option. Here we return to the two-column table layout.
```

. table hlthstat, statistic(frequency) statistic(percent)
> nformat(%5.1f percent)

```
\begin{tabular}{l|rr}
\hline & Frequency & Percent \\
\hline Health status & & \\
Excellent & 2,407 & 23.3 \\
Very good & 2,591 & 25.1 \\
Good & 2,938 & 28.4 \\
Fair & 1,670 & 16.2 \\
Poor & 729 & 7.1 \\
Total & 10,335 & 100.0 \\
\hline
\end{tabular}

The table command produces its output using a default set of styles, typically those defined in the table style but could be any other style that you have set as the default by using set table_style. When customizing our tables, we can take advantage of one of the styles described in [TABLES] Predefined styles. For instance, for tables with only one or two row variables, row labels that are right-aligned may be preferred. Here we use the table-right style.
```

. table hlthstat, statistic(frequency) statistic(percent)
> nformat(%5.1f percent) style(table-right)

```
\begin{tabular}{c|rr}
\hline & Frequency & Percent \\
\hline Health status & & \\
Excellent & 2,407 & 23.3 \\
Very good & 2,591 & 25.1 \\
Good & 2,938 & 28.4 \\
Fair & 1,670 & 16.2 \\
Poor & 729 & 7.1 \\
Total & 10,335 & 100.0 \\
\hline
\end{tabular}

\section*{Advanced customization}

Customization can go beyond the predefined styles and options available to you in the table command. table creates a collection of results that can be used in combination with the collect suite of commands to produce highly customized tables and to export those tables to presentation-ready formats such as HTML, Word, LATEX, PDF, Excel, and more.

Continuing with our example above, if we want to shorten the labels on the column headings, we could use the collect label levels command to define our new labels. After a change using collect, we can use collect preview to see the results.
```

. collect label levels result frequency "Freq" percent "%", modify

```
- collect preview
\begin{tabular}{c|rr}
\hline & Freq & \(\%\) \\
\hline Health status & & \\
Excellent & 2,407 & 23.3 \\
Very good & 2,591 & 25.1 \\
Good & 2,938 & 28.4 \\
Fair & 1,670 & 16.2 \\
Poor & 729 & 7.1 \\
Total & 10,335 & 100.0 \\
\hline
\end{tabular}

We could continue making style edits to this table. When we are happy with the result, we can then export it to the format of our choice using collect export.

See [TABLES] collect label for details on the collect label command we used here, and for an overview of the collect suite, see [TABLES] Intro.

\section*{Stored results}
table stores the following in \(r()\) :
Scalars
\(r(N)\) number of observations

\section*{References}

Bruun, N. H. 2022. Interactively building table reports with basetable. Stata Journal 22: 416-429.
McDowell, A., A. Engel, J. T. Massey, and K. Maurer. 1981. Plan and operation of the Second National Health and Nutrition Examination Survey, 1976-1980. Vital and Health Statistics 1(15): 1-144.

\section*{Also see}
[R] table - Table of frequencies, summaries, and command results
\([R]\) table intro - Introduction to tables of frequencies, summaries, and command results
[R] table multiway - Multiway tables
[R] table twoway - Two-way tabulation
[R] tabulate oneway - One-way table of frequencies
[TABLES] Intro - Introduction

\section*{Title}
table twoway - Two-way tabulation
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & References \\
Also see & & &
\end{tabular}

\section*{Description}

In this entry, we discuss how to use table to create a two-way tabulation, including frequencies, percentages, and proportions.

\section*{Quick start}

Table of frequencies, with rows defined by categories of a1 and columns defined by categories of a2 table a1 a2

Same as above, but treat missing values like other values
```

table a1 a2, missing

```

Table with the percentage of observations in each cell
```

table a1 a2, statistic(percent)

```

For each category of a1, report the percentage of observations across levels of a2
table a1 a2, statistic(percent, across(a2))
Report frequencies and the proportion of observations across categories of a1, enclosed within parentheses
```

table a1 a2, statistic(frequency) ///
statistic(proportion, across(a1)) sformat("(%s)" proportion)

```

\section*{Menu}

Statistics \(>\) Summaries, tables, and tests \(>\) Tables of frequencies, summaries, and command results

\section*{Syntax}

Basic two-way tabulation
table rowvar colvar [if] [in] [weight] [, options]

Customized two-way tabulation
table \([\) (rowspec \()][\) (colspec \()][\) if \(][\) in \(][\) weight \(][\), options \(]\)
rowspec may be empty or may include rowvar, result, or rowvar and result, where result refers to the requested statistics.
colspec may be empty or may include colvar, result, or colvar and result, where result refers to the requested statistics.
options
Main
totals(totals)
nototals
Statistics
statistic (stat \([\), statopts \(]\) )

Formats
nformat (\% fint [results] [, basestyle]) sformat (sfint [results])

Options
missing
zerocounts
name (cname)
append
replace
label (filename)
style(filename [, override])
markvar (newvar)

Description
report only the specified totals
suppress the marginal totals
statistic to be reported; default is statistic (frequency) when no weights are specified and statistic (sumw) otherwise
specify numeric format specify string format
treat numeric missing values of rowvar and colvar like other values
report 0 for empty cell counts
collect results into a collection named cname append results to an existing collection replace results of an existing collection specify the collection labels specify the collection style
create newvar that identifies observations used in the tabulation

\footnotetext{
fweights, aweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
strL variables are not allowed; see [U] 12.4.8 strL.
markvar() does not appear in the dialog box.
}

\section*{Options}
\(\qquad\) Main
totals (totals) and nototals control which totals are to be displayed in the table. By default, all totals are reported.
totals (totals) specifies which margin totals to display in the reported table. totals can contain rowvar, colvar, and their interaction. Interactions can be specified by using the \# operator.
nototals prevents table from displaying any totals.
 to request multiple statistics.
Available statistics are
\begin{tabular}{ll} 
stat & Definition \\
\hline frequency & frequency \\
sumw & sum of weights \\
proportion & proportion \\
percent & percentage \\
\hline rawproportion & proportion ignoring optionally specified weights \\
\hline rawpercent & percentage ignoring optionally specified weights \\
\hline
\end{tabular}

The following options may be specified in combination with statistics proportion, percent, rawproportion, and rawpercent:
\begin{tabular}{ll} 
statopts & Definition \\
\hline across (rowvar) & percentages or proportions across rows \\
across (colvar) & \begin{tabular}{l} 
percentages or proportions across columns \\
compute overall percentages or proportions
\end{tabular} \\
total &
\end{tabular}

Formats
nformat (\% fint [results][, basestyle]) changes the numeric format, such as the number of decimal places, for specified results. If results are not specified, the numeric format is changed for all results.
results may be any statistic named in option statistic() (that is, any stat).
This option is repeatable, and when multiple formats apply to one result, the rightmost specification is applied.
This option does not affect the format of numeric layout variables (rowspec and colspec). The default format of these variables is taken from the dataset.
basestyle indicates that the format be applied to results that do not already have their own format instead of overriding the format for all results.
sformat (sfint \([\) results \(]\) ) changes the string format for specified results. You can, for instance, add symbols or text to the values reported in the table by modifying the string format.
sfint may contain a mix of text and \(\%\) s. Here \(\%\) s refers to the numeric value that is formatted as specified using nformat(). The text will be placed around the numeric values in your table as it is placed around \(\% \mathrm{~s}\) in this option. For instance, to place parentheses around the percent statistics, you can specify sformat(" (\%s)" percent).
results may be any statistic named in option statistic() (that is, any stat).
Two text characters must be specified using a special character sequence if you want them to be displayed in your table. To include \(\%\), type \(\% \%\). To include \(\backslash\), type \(\backslash \backslash\). For instance, to place a percent sign following percent statistics, you can specify sformat("\%s\%\%" percent).
This option is repeatable, and when multiple formats apply to one result, the rightmost specification is applied.

Options
missing specifies that numeric missing values of rowvar or colvar be treated as valid categories. By default, observations with a numeric missing value in rowvar or colvar are omitted.
zerocounts specifies that table report a 0 in empty cells for the frequency statistic.
name (cname) specifies that a collection named cname be associated with the collected statistics and results. The default is name (Table).
append specifies that table append its collection information into the collection named in name(). replace permits table to overwrite an existing collection. This option is implied for name (Table) when append is not specified.
label (filename) specifies the filename containing the collection labels to use for your table. Labels in filename will be loaded for the table, and any labels not specified in filename will be taken from the labels defined in \(c\) (collect_label). The default is to use only the collection labels set in c(collect_label); see [TABLES] set collect_label.
style(filename [, override]) specifies the filename containing the collection styles to use for your table. The default collection styles will be discarded, and only the collection styles in filename will be applied.
If you prefer the default collection styles but also want to apply any styles in filename, specify override. If there are conflicts between the default collection styles and those in filename, the ones in filename will take precedence.

The default is to use only the collection styles set in c (table_style) ; see [TABLES] set table_style.
The following option is available with table but is not shown in the dialog box:
markvar (newvar) generates an indicator variable that identifies the observations used in the tabulation.

\section*{Remarks and examples}

Remarks are presented under the following headings:
Tabulation of two variables Tabulation, including percentages
Customizing results

\section*{Tabulation of two variables}

To obtain a two-way tabulation that reports the number of observations across the levels of two categorical variables, we need to specify only the names of the categorical variables following table.

To demonstrate, we use data from the Second National Health and Nutrition Examination Survey (NHANES II) (McDowell et al. 1981) and create a two-way tabulation of self-reported health status (hlthstat) by region of the USA (region).
. use https://www.stata-press.com/data/r18/nhanes2l
(Second National Health and Nutrition Examination Survey)
. table hlthstat region
\begin{tabular}{l|rrrrr}
\hline & NE & MW & \begin{tabular}{c} 
Region \\
S
\end{tabular} & W & Total \\
\hline Health status & & & & & \\
\(\quad\) Excellent & 562 & 730 & 546 & 569 & 2,407 \\
Very good & 558 & 721 & 651 & 661 & 2,591 \\
Good & 631 & 735 & 807 & 765 & 2,938 \\
Fair & 257 & 419 & 532 & 462 & 1,670 \\
Poor & 77 & 167 & 317 & 168 & 729 \\
Total & 2,085 & 2,772 & 2,853 & 2,625 & 10,335 \\
\hline
\end{tabular}

We can examine the missing values as well by adding the missing option.
. table hlthstat region, missing
\begin{tabular}{l|rrrrr}
\hline & NE & MW & \begin{tabular}{c} 
Region \\
\(S\)
\end{tabular} & W & Total \\
\hline Health status & & & & & \\
Excellent & 562 & 730 & 546 & 569 & 2,407 \\
Very good & 558 & 721 & 651 & 661 & 2,591 \\
Good & 631 & 735 & 807 & 765 & 2,938 \\
Fair & 257 & 419 & 532 & 462 & 1,670 \\
Poor & 77 & 167 & 317 & 168 & 729 \\
C & 1 & 1 & & & 2 \\
Blank but applicable & 10 & 1 & & 3 & 14 \\
Total & 2,096 & 2,774 & 2,853 & 2,628 & 10,351 \\
\hline
\end{tabular}

We find that 16 individuals have a missing health status, and the majority of these are from individuals in the Northeast. The empty cells correspond to regions in which all the individuals have a nonmissing health status; we can fill in these empty cells with 0s:
. table hlthstat region, missing zerocounts
\begin{tabular}{l|rrrrr}
\hline & NE & MW & Region \\
& S & W & Total \\
\hline Health status & & & & & \\
Excellent & 562 & 730 & 546 & 569 & 2,407 \\
Very good & 558 & 721 & 651 & 661 & 2,591 \\
Good & 631 & 735 & 807 & 765 & 2,938 \\
Fair & 257 & 419 & 532 & 462 & 1,670 \\
Poor & 77 & 167 & 317 & 168 & 729 \\
Clank but applicable & 1 & 1 & 0 & 0 & 2 \\
Bla & 10 & 1 & 0 & 3 & 14 \\
Total & 2,096 & 2,774 & 2,853 & 2,628 & 10,351 \\
\hline
\end{tabular}

\section*{Tabulation, including percentages}

Instead of frequencies, we can request that table report the percentage of observations in each cell of the table by specifying the statistic (percent) option.
. table hlthstat region, statistic (percent)
\begin{tabular}{l|rrrrr}
\hline & NE & MW & \begin{tabular}{c} 
Region \\
S
\end{tabular} & W & Total \\
\hline Health status & & & & & \\
Excellent & 5.44 & 7.06 & 5.28 & 5.51 & 23.29 \\
Very good & 5.40 & 6.98 & 6.30 & 6.40 & 25.07 \\
Good & 6.11 & 7.11 & 7.81 & 7.40 & 28.43 \\
Fair & 2.49 & 4.05 & 5.15 & 4.47 & 16.16 \\
Poor & 0.75 & 1.62 & 3.07 & 1.63 & 7.05 \\
Total & 20.17 & 26.82 & 27.61 & 25.40 & 100.00 \\
\hline
\end{tabular}

We see that \(5.44 \%\) of all observations correspond to individuals in excellent health who live in the Northeast.

Rather than looking at overall percentages, we might want to examine the distribution of observations within each health status level across the four regions. To do this, we can add the across (region) option.
\begin{tabular}{l|rrrrrr}
\multicolumn{7}{c}{. table hlthstat region, statistic(percent, across (region) } \\
\hline & NE & MW & \begin{tabular}{c} 
Region \\
S
\end{tabular} & W & Total \\
\hline Health status & & & & & \\
Excellent & 23.35 & 30.33 & 22.68 & 23.64 & 100.00 \\
Very good & 21.54 & 27.83 & 25.13 & 25.51 & 100.00 \\
Good & 21.48 & 25.02 & 27.47 & 26.04 & 100.00 \\
Fair & 15.39 & 25.09 & 31.86 & 27.66 & 100.00 \\
Poor & 10.56 & 22.91 & 43.48 & 23.05 & 100.00 \\
Total & 20.17 & 26.82 & 27.61 & 25.40 & 100.00 \\
\hline
\end{tabular}

Of individuals reporting excellent health, \(23.35 \%\) live in the Northeast, while \(30.33 \%\) live in the Midwest, \(22.68 \%\) live in the South, and \(23.64 \%\) live in the West.

We can also look at the distribution of observations across health status categories within each region.
\begin{tabular}{l|rrrrr}
\multicolumn{7}{c}{. table hlthstat region, statistic(percent, across(hlthstat)) } \\
\hline & & & Region & \\
& NE & MW & S & W & Total \\
\hline Health status & & & & & \\
Excellent & 26.95 & 26.33 & 19.14 & 21.68 & 23.29 \\
Very good & 26.76 & 26.01 & 22.82 & 25.18 & 25.07 \\
Good & 30.26 & 26.52 & 28.29 & 29.14 & 28.43 \\
Fair & 12.33 & 15.12 & 18.65 & 17.60 & 16.16 \\
Poor & 3.69 & 6.02 & 11.11 & 6.40 & 7.05 \\
Total & 100.00 & 100.00 & 100.00 & 100.00 & 100.00 \\
\hline
\end{tabular}

Of individuals living in the South, \(11.11 \%\) report having poor health. This is notably higher than the percentage of individuals reporting poor health in the other regions.

It is often helpful to see both frequencies and percentages in the same table. To do this, we can add the statistic (frequency) option to our command.
```

. table hlthstat region, statistic(frequency)

```
> statistic(percent, across(hlthstat))
\begin{tabular}{l|rrrrr}
\hline & & NE & MW & \begin{tabular}{r} 
Region \\
S
\end{tabular} & W
\end{tabular} Total

\section*{Customizing results}

There are several ways that we can customize the results of our two-way tabulation.
For instance, in some cases, we may prefer to omit the row and column totals. We can specify the nototals option to suppress these totals.
\begin{tabular}{l}
. table hlthstat region, statistic (frequency) \\
> statistic (percent, across(hlthstat)) \\
\hline
\end{tabular}

Or perhaps we want to see row totals or column totals but not both. We can include the totals (region) option to display only the region totals.
\begin{tabular}{|c|c|c|c|c|}
\hline \multirow[t]{2}{*}{} & \multicolumn{4}{|c|}{Region} \\
\hline & NE & MW & S & W \\
\hline \multicolumn{5}{|l|}{Health status} \\
\hline Excellent & & & & \\
\hline Frequency & 562 & 730 & 546 & 569 \\
\hline Percent & 26.95 & 26.33 & 19.14 & 21.68 \\
\hline \multicolumn{5}{|l|}{Very good} \\
\hline Frequency & 558 & 721 & 651 & 661 \\
\hline Percent & 26.76 & 26.01 & 22.82 & 25.18 \\
\hline \multicolumn{5}{|l|}{Good} \\
\hline Frequency & 631 & 735 & 807 & 765 \\
\hline Percent & 30.26 & 26.52 & 28.29 & 29.14 \\
\hline \multicolumn{5}{|l|}{Fair} \\
\hline Frequency & 257 & 419 & 532 & 462 \\
\hline Percent & 12.33 & 15.12 & 18.65 & 17.60 \\
\hline \multicolumn{5}{|l|}{Poor} \\
\hline Frequency & 77 & 167 & 317 & 168 \\
\hline Percent & 3.69 & 6.02 & 11.11 & 6.40 \\
\hline \multicolumn{5}{|l|}{Total} \\
\hline Frequency & 2,085 & 2,772 & 2,853 & 2,625 \\
\hline Percent & 100.00 & 100.00 & 100.00 & 100.00 \\
\hline
\end{tabular}

Once we have the statistics we want in our table, we can format the way that they appear. If, for instance, we want to add a percent sign to each of our percentages, we can specify the sformat ( \(" \%\) s \(\% \%\) percent) option. The sformat() option specifies that we want to add string characters to the numbers in the table. Within it, we refer to the numeric values as \%s and place any string characters we want around this. The percent sign is unique because it already has special meaning in this context. Therefore, we must type two percent signs, \%\%, to display one. Finally, by adding percent within the sformat () option, we specify that we want to apply this format only to the percent statistics.
```

. table hlthstat region, statistic(frequency)
> statistic(percent, across(hlthstat)) totals(region)
> sformat("%s%%" percent)

```
\(\left.\begin{array}{c|rrrr}\hline & \text { NE } & \begin{array}{c}\text { Region } \\ \\ \end{array} & \text { MW } & \mathrm{S}\end{array}\right) \mathrm{W}\)

Now that we have added the percent sign, we could argue that the labels Frequency and Percent are unnecessary. If we remove these statistic names from the row labels, we might also want to rightalign the remaining labels in row headers. Finally, for readability, we could insert blank lines between levels of hlthstat. We could use the collect suite of commands to make these style changes. Fortunately, however, one of our predefined styles, table-tab2, includes these style changes, and we can select it using the style() option.
\begin{tabular}{|c|c|c|c|c|}
\hline \multirow[t]{2}{*}{} & \multicolumn{4}{|c|}{Region} \\
\hline & NE & MW & S & W \\
\hline \multirow[t]{3}{*}{Health status Excellent} & & & & \\
\hline & 562 & 730 & 546 & 569 \\
\hline & 26.95\% & 26.33\% & 19.14\% & 21.68\% \\
\hline \multirow[t]{2}{*}{Very good} & 558 & 721 & 651 & 661 \\
\hline & 26.76\% & 26.01\% & 22.82\% & 25.18\% \\
\hline \multirow[t]{2}{*}{Good} & 631 & 735 & 807 & 765 \\
\hline & 30.26\% & 26.52\% & 28.29\% & 29.14\% \\
\hline \multirow[t]{2}{*}{Fair} & 257 & 419 & 532 & 462 \\
\hline & 12.33\% & 15.12\% & 18.65\% & 17.60\% \\
\hline \multirow[t]{2}{*}{Poor} & 77 & 167 & 317 & 168 \\
\hline & 3.69\% & 6.02\% & 11.11\% & 6.40\% \\
\hline \multirow[t]{2}{*}{Total} & 2,085 & 2,772 & 2,853 & 2,625 \\
\hline & 100.00\% & 100.00\% & 100.00\% & 100.00\% \\
\hline
\end{tabular}

You can learn more about the predefined styles described at [TABLES] Predefined styles. If none of these provide the exact style you want for your table, you can further customize the results by using the collect suite of commands. To learn more, see [TABLES] Intro.

If you wish to include this table in a paper, on a webpage, or in another format, you can easily export it in \(\mathrm{LAT}_{\mathrm{E}} \mathrm{X}\), Word, Excel, HTML, and a variety of other formats by using collect export.

\section*{Stored results}
table stores the following in \(r()\) :
Scalars \(r(N)\) number of observations

\section*{References}

Huber, C. 2021. Customizable tables in Stata 17, part 1: The new table command. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2021/06/07/customizable-tables-in-stata-17-part-1-the-new-table-command/.
McDowell, A., A. Engel, J. T. Massey, and K. Maurer. 1981. Plan and operation of the Second National Health and Nutrition Examination Survey, 1976-1980. Vital and Health Statistics 1(15): 1-144.

\section*{Also see}
\([R]\) table - Table of frequencies, summaries, and command results
\([R]\) table intro - Introduction to tables of frequencies, summaries, and command results
[R] table multiway - Multiway tables
[R] table oneway - One-way tabulation
\([R]\) tabulate twoway - Two-way table of frequencies
[TABLES] Intro - Introduction

\section*{Title}
table multiway - Multiway tables
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Reference \\
Also see & & &
\end{tabular}

\section*{Description}

In this entry, we discuss how to use the table command to create tables displaying frequencies, percentages, and proportions across levels of three or more variables. Additionally, we demonstrate how a single table command can create multiple tables corresponding to levels of variables or different statistics.

\section*{Quick start}

Table of frequencies with rows defined by the levels of a1 and columns defined by the categories of a2 and a3
```

    table (a1) (a2 a3)
    ```

Same as above, but with rows defined by levels of a1 and a2 and columns defined by levels of a3 table (a1 a2) (a3)

Separate tables for each level a3 with the rows of each table defined by the levels of a1 and the columns defined by levels of a2
table (a1) (a2) (a3)
Report percentages of observations in each cell rather than frequencies
table (a1) (a2 a3), statistic (percent)
Report both frequencies and percentages
table (a1) (a2 a3), statistic(percent) statistic (frequency)
Report the percentages across levels of a3
table (a1) (a2 a3), statistic(percent, across(a3))

\section*{Menu}

Statistics \(>\) Summaries, tables, and tests \(>\) Tables of frequencies, summaries, and command results

\section*{Syntax}

Table with rows defined by multiple variables
table rowvars colvar \([\) if \(][\) in \(]\) [weight] [, options]

Table with columns defined by multiple variables table rowvar (colvars) [if] [in] [weight] [, options]

Table with rows and columns defined by multiple variables
table (rowvars) (colvars) [if] [in] [weight] [, options]

Multiple multiway tables
table (rowvars) (colvars) (tabvars) [if] [in] [weight] [, options]

Customized multiway tables
table (rowspec) (colspec) [(tabspec)] [if] [in] [weight] [, options]
rowspec, colspec, and tabspec may be empty or may include variable names or any of the following keywords:
\begin{tabular}{ll} 
keyword & Description \\
\hline result & requested statistics \\
across & index across () specifications \\
\hline
\end{tabular}
options
Main
totals (totals)
nototals
Statistics
statistic (stat \([\), statopts \(]\) )

\section*{Formats}
nformat (\% fint [results] [, basestyle]) specify numeric format sformat (sfmt \([\) results \(]\) ) specify string format

Options
listwise
missing
showcounts
zerocounts
name (cname)
append
replace
label(filename)
style(filename [, override])
markvar (newvar)

Description
report only the specified totals
suppress the marginal totals
statistic to be reported; default is
statistic (frequency) when no weights
are specified and statistic (sumw) otherwise
use listwise deletion to handle missing values treat numeric missing values like other values show sample size for all variables in statistic() option
report 0 for empty cell counts
collect results into a collection named cname
append results to an existing collection
replace results of an existing collection
specify the collection labels
specify the collection style
create newvar to identify observations used to compute the statistics
fweights, aweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
strL variables are not allowed; see [U] 12.4.8 strL.
markvar() does not appear in the dialog box.

\section*{Options}

totals (totals) and nototals control which totals are to be displayed in the table. By default, all totals are reported.
totals (totals) specifies that totals be displayed only for the variables or interactions specified. totals can contain rowvars, colvars, tabvars, and interactions between any of these variables. Interactions can be specified by using the \# operator.
nototals prevents table from displaying any totals.

Statistics
statistic (stat [, statopts]) specifies the statistic to be displayed. statistic() may be repeated to request multiple statistics.

Available statistics are
\begin{tabular}{ll} 
stat & Definition \\
\hline \begin{tabular}{l} 
frequency \\
sumw
\end{tabular} & frequency \\
proportion & sum of weights \\
percent & percentage \\
\begin{tabular}{l} 
rawproportion \\
\hline rawpercent
\end{tabular} & \begin{tabular}{l} 
proportion ignoring optionally specified weights \\
percentage ignoring optionally specified weights
\end{tabular} \\
\hline
\end{tabular}

The following options may be specified in combination with statistics proportion, percent, rawproportion, and rawpercent:
\begin{tabular}{ll} 
statopts & Definition \\
\hline across (cellspec) & \begin{tabular}{l} 
percentages or proportions across levels of variables or interactions \\
compute overall percentages or proportions
\end{tabular} \\
total &
\end{tabular}
cellspec may contain rowvars, colvars, tabvars, or an interaction between any of these variables.
Interactions can be specified by using the \# operator.
\(\qquad\) Formats
nformat ( \(\%\) fint [results][, basestyle]) changes the numeric format, such as the number of decimal places, for specified results. If results are not specified, the numeric format is changed for all results.
results may be any statistic named in option statistic() (that is, any stat).
This option is repeatable, and when multiple formats apply to one result, the rightmost specification is applied.
This option does not affect the format of numeric layout variables (rowspec, colspec, and tabspec). The default format of these variables is taken from the dataset.
basestyle indicates that the format be applied to results that do not already have their own format instead of overriding the format for all results.
sformat (sfint \([\) results \(]\) ) changes the string format for specified results. You can, for instance, add symbols or text to the values reported in the table by modifying the string format.
sfint may contain a mix of text and \(\%\) s. Here \(\%\) s refers to the numeric value that is formatted as specified using nformat(). The text will be placed around the numeric values in your table as it is placed around \(\%\) s in this option. For instance, to place parentheses around the percent statistics, you can specify sformat (" (\%s)" percent).
results may be any statistic named in option statistic() (that is, any stat).
Two text characters must be specified using a special character sequence if you want them to be displayed in your table. To include \(\%\), type \(\% \%\). To include \(\backslash\), type \(\backslash \backslash\). For instance, to place a percent sign following percent statistics, you can specify sformat("\%s\%\%" percent).
This option is repeatable, and when multiple formats apply to one result, the rightmost specification is applied.
listwise handles missing values through listwise deletion, meaning that the entire observation is omitted from the sample if any variable specified in a statistic() option is missing for that observation. By default, table will omit an observation only if all variables specified in all statistic() options are missing for that observation.
missing specifies that numeric missing values of any rowvars, colvars, or tabvars be treated as valid categories. By default, observations with a numeric missing value in rowvars, colvars, or tabvars are omitted.
showcounts specifies that table report the sample size for each variable specified in option statistic().
zerocounts specifies that table report a 0 in empty cells for the frequency statistic.
name (cname) specifies that a collection named cname be associated with the collected statistics and results. The default is name (Table).
append specifies that table append its collection information into the collection named in name().
replace permits table to overwrite an existing collection. This option is implied for name (Table) when append is not specified.
label (filename) specifies the filename containing the collection labels to use for your table. Labels in filename will be loaded for the table, and any labels not specified in filename will be taken from the labels defined in \(c\) (collect_label). The default is to use only the collection labels set in c(collect_label); see [TABLES] set collect_label.
style(filename [, override]) specifies the filename containing the collection styles to use for your table. The default collection styles will be discarded, and only the collection styles in filename will be applied.
If you prefer the default collection styles but also want to apply any styles in filename, specify override. If there are conflicts between the default collection styles and those in filename, the ones in filename will take precedence.
The default is to use only the collection styles set in c (table_style); see [TABLES] set table_style.
The following option is available with table but is not shown in the dialog box:
markvar (newvar) generates an indicator variable that identifies the observations used in the tabulation.

\section*{Remarks and examples}

Remarks are presented under the following headings:

\author{
Introduction \\ Tables with columns defined by multiple variables \\ Appending tables \\ Multiple tables with specified totals
}

\section*{Introduction}

The table command allows you to create complex tables beyond one- and two-way tabulations. In multiway tabulations, you can display frequencies across levels of two or more variables. You can have levels of one variable nested within levels of another variable in columns, in rows, or in both dimensions. And with a single command, you can create separate tables for levels of one or more variables or for different results.

\section*{Tables with columns defined by multiple variables}

We use data from the Second National Health and Nutrition Examination Survey (NHANES II) (McDowell et al. 1981). The data contain some demographic information, such as the age, sex, and race of participants. The dataset also contains some measures of health, including whether the individual has high blood pressure (highbp).

Before we create any tables, we will modify a few labels in our dataset so that they will appear as we wish in our tables.
```

. use https://www.stata-press.com/data/r18/nhanes21
(Second National Health and Nutrition Examination Survey)
. label define yesno O "No" 1 "Yes"
. label values highbp diabetes heartatk yesno
. label variable diabetes "Diabetes"

```

Suppose we want to examine how many males and females in each age group have high blood pressure. Let's place the levels of age group on the rows and the levels of high blood pressure and sex on the columns.
. table (agegrp) (sex highbp), nototals
\begin{tabular}{c|cccc}
\hline & \multicolumn{3}{c}{ Sex } & \multicolumn{2}{c}{ Fale } & \begin{tabular}{c} 
Female
\end{tabular} \\
& \begin{tabular}{c} 
Migh blood pressure \\
No
\end{tabular} & Yes & \begin{tabular}{c} 
High blood pressure \\
No
\end{tabular} & Yes \\
& & & & \\
Age group & & & & \\
\(20-29\) & 825 & 291 & 1,103 & 101 \\
\(30-39\) & 480 & 290 & 687 & 165 \\
\(40-49\) & 336 & 274 & 434 & 228 \\
\(50-59\) & 255 & 347 & 335 & 354 \\
\(60-69\) & 568 & 801 & 625 & 866 \\
\(70+\) & 147 & 301 & 180 & 358 \\
\hline
\end{tabular}

By default, table includes the totals for each category; we added the nototals option to suppress them here.

To better compare the occurrence of high blood pressure, let's now compute percentages of highbp. Below, we create the same table, but within each sex and age group combination, we report the percentage of individuals with and without high blood pressure.
```

. table (agegrp) (sex highbp), nototals statistic(percent, across(highbp))

```
\begin{tabular}{c|cccr}
\hline & \multicolumn{3}{c}{ Male } \\
& \begin{tabular}{c} 
Males \\
High blood pressure \\
No
\end{tabular} & \begin{tabular}{c} 
Yes
\end{tabular} & \begin{tabular}{c} 
Female \\
High blood pressure \\
No
\end{tabular} & Yes
\end{tabular}

Here we see that \(26.08 \%\) of males in their 20s have high blood pressure and only \(8.39 \%\) of females in their 20s have high blood pressure.

If we had simply typed statistic (percent), then we would see the percentage of observations in each cell. With the suboption across(), we can compute the percentage within each sex and age group combination (across levels of highbp).

Next, let's request that percentages be calculated across the categories of high blood pressure and sex. We can alternatively think of this as being percentages within age group.
\begin{tabular}{|c|c|c|c|c|}
\hline \multirow[t]{2}{*}{} & \multicolumn{4}{|c|}{Sex} \\
\hline & \multicolumn{2}{|l|}{\begin{tabular}{l}
Male \\
High blood pressure \\
No \\
Yes
\end{tabular}} & \multicolumn{2}{|l|}{\begin{tabular}{l}
Female \\
High blood pressure No Yes
\end{tabular}} \\
\hline Age group & & & & \\
\hline 20-29 & 35.56 & 12.54 & 47.54 & 4.35 \\
\hline 30-39 & 29.59 & 17.88 & 42.36 & 10.17 \\
\hline 40-49 & 26.42 & 21.54 & 34.12 & 17.92 \\
\hline 50-59 & 19.75 & 26.88 & 25.95 & 27.42 \\
\hline 60-69 & 19.86 & 28.01 & 21.85 & 30.28 \\
\hline 70+ & 14.91 & 30.53 & 18.26 & 36.31 \\
\hline
\end{tabular}

Here we see that \(12.54 \%\) of individuals in their 20s are males with high blood pressure and \(4.35 \%\) are females with high blood pressure.

We can reverse the order of our column variables so that we have the levels of sex nested within levels of high blood pressure. We will again request percentages across the categories of highbp.
```

. table (agegrp) (highbp sex), nototals statistic(percent, across(highbp))

```
\begin{tabular}{l|lllr}
\hline & \multicolumn{3}{|c}{\begin{tabular}{c} 
High blood pressure \\
No \\
Sex
\end{tabular}} & \multicolumn{2}{c}{\begin{tabular}{c} 
Yes \\
Sex
\end{tabular}} \\
& \multicolumn{2}{c}{ Male } & Female & Male \\
& & Female \\
\hline Age group & & & & \\
\(20-29\) & 73.92 & 91.61 & 26.08 & 8.39 \\
\(30-39\) & 62.34 & 80.63 & 37.66 & 19.37 \\
\(40-49\) & 55.08 & 65.56 & 44.92 & 34.44 \\
\(50-59\) & 42.36 & 48.62 & 57.64 & 51.38 \\
\(60-69\) & 41.49 & 41.92 & 58.51 & 58.08 \\
\(70+\) & 32.81 & 33.46 & 67.19 & 66.54 \\
\hline
\end{tabular}

The last two columns represent the percentage of males with high blood pressure in the age group and the percentage of females with high blood pressure in the age group. We can clearly see that, across all age groups, the percent of males who have high blood pressure is greater than the percentage of females with high blood pressure.

Perhaps we want a table that includes only the two columns on the right. To create a table with the percentages of individuals with high blood pressure, we can take advantage of a unique feature of table-its results are automatically stored in a "collection" and can be easily customized. Specifically, when we create a table using the table command, the results are stored in a collection named Table, and these results replace the results from any previous table command. The collect suite of commands can be used to change the layout, style, and formatting of tables created from results in collection; see [TABLES] Intro to learn about collections of results and creating customized tables. For our table, we will use the collect layout command, which specifies how items from a collection should be arranged. Below, we arrange the percentages in a table with rows defined by the categories of agegrp and columns defined by the categories of sex and category 1 of highbp.
```

. collect layout (agegrp) (highbp[1]\#sex)
Collection: Table
Rows: agegrp
Columns: highbp[1]\#sex
Table 1: 7 x 2

```
\begin{tabular}{c|rr}
\hline & \multicolumn{2}{|c}{\begin{tabular}{c} 
High blood pressure \\
Yes \\
Sex
\end{tabular}} \\
& \multicolumn{2}{|c}{ Male }
\end{tabular} Female

\section*{Appending tables}

When we specify multiple variables for the row or column specification, the levels of one variable are nested within the levels of another. When you simply wish to join rows or columns from multiple tables, this can be easily done with the append option.

For example, we first create a table with the percentage of males and females in each age group with diabetes.
```

. table (sex agegrp) (diabetes), nototals statistic(percent, across(diabetes))

```
\begin{tabular}{c|rr}
\hline & \multicolumn{2}{|c}{\begin{tabular}{c} 
Diabetes \\
No
\end{tabular}} \\
\hline & Yes \\
Sex & & \\
Male & & \\
Age group & & \\
\(20-29\) & 99.64 & 0.36 \\
\(30-39\) & 99.61 & 0.39 \\
\(40-49\) & 97.38 & 2.62 \\
\(50-59\) & 94.68 & 5.32 \\
\(60-69\) & 91.96 & 8.04 \\
\(70+\) & 88.39 & 11.61 \\
Female & & \\
Age group & & \\
\(20-29\) & 99.09 & 0.91 \\
\(30-39\) & 97.88 & 2.12 \\
\(40-49\) & 96.07 & 3.93 \\
\(50-59\) & 94.19 & 5.81 \\
\(60-69\) & 91.42 & 8.58 \\
\(70+\) & 89.03 & 10.97 \\
& & \\
\hline
\end{tabular}

We want to include the same information for highbp and heartatk in the same table, which indicates whether someone has had a heart attack. To do this, we will run the table command three times, once specifying each of these variables as the column variable. We will use the name (table1) option to specify that the results be stored in a collection named table1. To the second and third table commands, we will add the append option so that all the results are stored in the same collection rather than overriding the results from the previous command.
```

. quietly: table (sex agegrp) (diabetes), nototals
> statistic(percent, across(diabetes)) name(table1)
. quietly: table (sex agegrp) (highbp), nototals
> statistic(percent, across(highbp)) name(table1) append
. quietly: table (sex agegrp) (heartatk), nototals
> statistic(percent, across(heartatk)) name(table1) append

```

With the results from all our table commands stored in one collection, we can again take advantage of the collect layout command to arrange results into a table. We request that sex and agegroup define the rows. By including the \# between variable names, we specify that we want the levels of these variables to be interacted to form the rows. We request that levels of diabetes, highbp, and heartatk form the columns. Because we did not include \# between the variable names, their levels will not be interacted. Instead, they will be listed one after the other.
```

. collect layout (sex\#agegrp) (diabetes highbp heartatk)
Collection: table1
Rows: sex\#agegrp
Columns: diabetes highbp heartatk
Table 1: 17 x 6

```
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline \multirow[t]{2}{*}{} & \multicolumn{2}{|r|}{Diabetes} & \multicolumn{2}{|l|}{High blood pressure} & \multicolumn{2}{|l|}{Prior heart attack} \\
\hline & No & Yes & No & Yes & No & Yes \\
\hline \multicolumn{7}{|l|}{Sex} \\
\hline \multicolumn{7}{|l|}{Male} \\
\hline Age group & & & & & & \\
\hline 20-29 & 99.64 & 0.36 & 73.92 & 26.08 & 100.00 & \\
\hline 30-39 & 99.61 & 0.39 & 62.34 & 37.66 & 99.74 & 0.26 \\
\hline 40-49 & 97.38 & 2.62 & 55.08 & 44.92 & 98.03 & 1.97 \\
\hline 50-59 & 94.68 & 5.32 & 42.36 & 57.64 & 92.36 & 7.64 \\
\hline 60-69 & 91.96 & 8.04 & 41.49 & 58.51 & 86.56 & 13.44 \\
\hline 70+ & 88.39 & 11.61 & 32.81 & 67.19 & 83.48 & 16.52 \\
\hline \multicolumn{7}{|l|}{Female} \\
\hline Age group & & & & & & \\
\hline 20-29 & 99.09 & 0.91 & 91.61 & 8.39 & 99.92 & 0.08 \\
\hline 30-39 & 97.88 & 2.12 & 80.63 & 19.37 & 99.76 & 0.24 \\
\hline 40-49 & 96.07 & 3.93 & 65.56 & 34.44 & 98.79 & 1.21 \\
\hline 50-59 & 94.19 & 5.81 & 48.62 & 51.38 & 96.66 & 3.34 \\
\hline 60-69 & 91.42 & 8.58 & 41.92 & 58.08 & 94.57 & 5.43 \\
\hline 70+ & 89.03 & 10.97 & 33.46 & 66.54 & 92.01 & 7.99 \\
\hline
\end{tabular}

\section*{Multiple tables with specified totals}

There may be times when instead of creating one large table for multiple variables, you would prefer to create separate tables for each level of one or more variables or for different statistics. For example, we previously had levels of age group nested within categories of sex. Now, we would like to create tables that show how males and females in each age group rate their own health. The variable hlthstat records how individuals self-rate their health. Let's create a table that shows what percent of males in each age group selected each of the health categories and a separate table to list the percent of females.
```

. table (agegrp) (hlthstat) (sex), statistic(percent, across(hlthstat))
Sex = Male

```
\begin{tabular}{l|rrrrrr}
\hline & & \multicolumn{5}{c}{ Health status } \\
& Excellent & Very good & Good & Fair & Poor & Total \\
\hline Age group & & & & & & \\
\(20-29\) & 39.61 & 32.71 & 20.97 & 5.82 & 0.90 & 100.00 \\
\(30-39\) & 36.88 & 29.61 & 26.10 & 5.84 & 1.56 & 100.00 \\
\(40-49\) & 28.03 & 25.90 & 29.51 & 12.30 & 4.26 & 100.00 \\
\(50-59\) & 17.80 & 21.30 & 35.11 & 15.97 & 9.82 & 100.00 \\
\(60-69\) & 13.33 & 19.56 & 28.94 & 23.00 & 15.16 & 100.00 \\
\(70+\) & 14.77 & 14.99 & 26.62 & 28.41 & 15.21 & 100.00 \\
Total & 25.50 & 24.71 & 27.30 & 14.71 & 7.78 & 100.00 \\
\hline
\end{tabular}

Sex = Female
\begin{tabular}{l|rrrrrr}
\hline & & \multicolumn{6}{|c}{ Health status } \\
& Excellent & Very good & Good & Fair & Poor & Total \\
\hline Age group & & & & & & \\
\(20-29\) & 32.39 & 34.55 & 24.92 & 6.98 & 1.16 & 100.00 \\
\(30-39\) & 29.76 & 30.71 & 28.12 & 9.29 & 2.12 & 100.00 \\
\(40-49\) & 26.71 & 23.07 & 29.29 & 15.63 & 5.31 & 100.00 \\
\(50-59\) & 17.15 & 20.93 & 33.43 & 20.20 & 8.28 & 100.00 \\
\(60-69\) & 10.76 & 20.31 & 33.22 & 25.49 & 10.22 & 100.00 \\
\(70+\) & 10.78 & 19.14 & 26.39 & 30.48 & 13.20 & 100.00 \\
Total & 21.29 & 25.40 & 29.45 & 17.47 & 6.40 & 100.00 \\
\hline
\end{tabular}

Sex = Total
\begin{tabular}{l|rrrrrr}
\hline & & \multicolumn{6}{|c}{ Health status } \\
& Excellent & Very good & Good & Fair & Poor & Total \\
\hline Age group & & & & & & \\
\(20-29\) & 35.86 & 33.66 & 23.02 & 6.42 & 1.03 & 100.00 \\
\(30-39\) & 33.15 & 30.19 & 27.16 & 7.65 & 1.85 & 100.00 \\
\(40-49\) & 27.34 & 24.43 & 29.39 & 14.03 & 4.81 & 100.00 \\
\(50-59\) & 17.46 & 21.10 & 34.21 & 18.23 & 9.00 & 100.00 \\
\(60-69\) & 11.99 & 19.95 & 31.17 & 24.30 & 12.59 & 100.00 \\
\(70+\) & 12.59 & 17.26 & 26.50 & 29.54 & 14.11 & 100.00 \\
Total & 23.29 & 25.07 & 28.43 & 16.16 & 7.05 & 100.00 \\
\hline
\end{tabular}

We see that \(14.77 \%\) of males in their 70s and beyond rated their health as excellent and \(10.78 \%\) of females in their 70s and beyond rated their health as excellent.

Note that we actually created three tables, one for males, one for females, and one for everybody. We are mainly interested in the first two tables and would like to drop the third table. But if we use the nototals option, we will not get the row totals we see above. Instead, we can specify which totals we do want with the totals() option:
```

. table (agegrp) (hlthstat) (sex), statistic(percent, across(hlthstat))
> totals(sex\#agegrp sex\#hlthstat sex)
Sex = Male

```
\begin{tabular}{l|rrrrrr}
\hline & & \multicolumn{6}{|c}{ Health status } \\
& Excellent & Very good & Good & Fair & Poor & Total \\
\hline Age group & & & & & & \\
\(20-29\) & 39.61 & 32.71 & 20.97 & 5.82 & 0.90 & 100.00 \\
\(30-39\) & 36.88 & 29.61 & 26.10 & 5.84 & 1.56 & 100.00 \\
\(40-49\) & 28.03 & 25.90 & 29.51 & 12.30 & 4.26 & 100.00 \\
\(50-59\) & 17.80 & 21.30 & 35.11 & 15.97 & 9.82 & 100.00 \\
\(60-69\) & 13.33 & 19.56 & 28.94 & 23.00 & 15.16 & 100.00 \\
\(70+\) & 14.77 & 14.99 & 26.62 & 28.41 & 15.21 & 100.00 \\
Total & 25.50 & 24.71 & 27.30 & 14.71 & 7.78 & 100.00 \\
\hline
\end{tabular}
Sex = Female
\begin{tabular}{l|rrrrrr}
\hline & & \multicolumn{6}{|c}{ Health status } \\
& Excellent & Very good & Good & Fair & Poor & Total \\
\hline Age group & & & & & & \\
\(20-29\) & 32.39 & 34.55 & 24.92 & 6.98 & 1.16 & 100.00 \\
\(30-39\) & 29.76 & 30.71 & 28.12 & 9.29 & 2.12 & 100.00 \\
\(40-49\) & 26.71 & 23.07 & 29.29 & 15.63 & 5.31 & 100.00 \\
\(50-59\) & 17.15 & 20.93 & 33.43 & 20.20 & 8.28 & 100.00 \\
\(60-69\) & 10.76 & 20.31 & 33.22 & 25.49 & 10.22 & 100.00 \\
\(70+\) & 10.78 & 19.14 & 26.39 & 30.48 & 13.20 & 100.00 \\
Total & 21.29 & 25.40 & 29.45 & 17.47 & 6.40 & 100.00 \\
\hline
\end{tabular}
sex\#agegrp gives us the row totals, the total for each age group within each category of sex. sex\#hlthstat provides us with the column totals, one of which would be the total percent of females that rated their health as excellent. Finally, sex gives us the total in the rightmost cell in the bottom of each table.

In all our previous tables, we used variables to define the rows and columns, but you can also use results in the row, column, and table identifiers. For example, suppose that in addition to reporting percentages for males and females in each age group, we wanted to report frequencies. We can create separate tables for the percentages and frequencies by specifying result in the third set of parentheses. To include the totals for each age group and for each category of sex, we specify the interaction of sex and agegrp in the totals() option:
. table (sex agegrp) (hlthstat) (result), statistic(percent, across(hlthstat))
> statistic (frequency) totals(sex\#agegrp)
Percent
\begin{tabular}{c|rrrrrr}
\hline & & \multicolumn{5}{|c}{ Health status } \\
& Excellent & Very good & Good & Fair & Poor & Total \\
\hline Sex & & & & & & \\
Male & & & & & & \\
Age group & & & & & \\
\(20-29\) & 39.61 & 32.71 & 20.97 & 5.82 & 0.90 & 100.00 \\
\(30-39\) & 36.88 & 29.61 & 26.10 & 5.84 & 1.56 & 100.00 \\
\(40-49\) & 28.03 & 25.90 & 29.51 & 12.30 & 4.26 & 100.00 \\
\(50-59\) & 17.80 & 21.30 & 35.11 & 15.97 & 9.82 & 100.00 \\
60-69 & 13.33 & 19.56 & 28.94 & 23.00 & 15.16 & 100.00 \\
\(70+\) & 14.77 & 14.99 & 26.62 & 28.41 & 15.21 & 100.00 \\
Female & & & & & & \\
Age group & 32.39 & 34.55 & 24.92 & 6.98 & 1.16 & 100.00 \\
\(20-29\) & 29.76 & 30.71 & 28.12 & 9.29 & 2.12 & 100.00 \\
\(30-39\) & 26.71 & 23.07 & 29.29 & 15.63 & 5.31 & 100.00 \\
\(40-49\) & 17.15 & 20.93 & 33.43 & 20.20 & 8.28 & 100.00 \\
\(50-59\) & 10.76 & 20.31 & 33.22 & 25.49 & 10.22 & 100.00 \\
\(60-69\) & 10.78 & 19.14 & 26.39 & 30.48 & 13.20 & 100.00 \\
\(70+\) & & & & & & \\
\hline
\end{tabular}

Frequency
\begin{tabular}{c|rrrrrr}
\hline & & \multicolumn{6}{|c}{ Health status } \\
& Excellent & Very good & Good & Fair & Poor & Total \\
\hline Sex & & & & & & \\
Male & & & & & & \\
Age group & & & & & \\
\(20-29\) & 442 & 365 & 234 & 65 & 10 & 1,116 \\
\(30-39\) & 284 & 228 & 201 & 45 & 12 & 770 \\
\(40-49\) & 171 & 158 & 180 & 75 & 26 & 610 \\
\(50-59\) & 107 & 128 & 211 & 96 & 59 & 601 \\
\(60-69\) & 182 & 267 & 395 & 314 & 207 & 1,365 \\
\(70+\) & 66 & 67 & 119 & 127 & 68 & 447 \\
Female & & & & & & \\
Age group & 390 & 416 & 300 & 84 & 14 & 1,204 \\
\(20-29\) & 253 & 261 & 239 & 79 & 18 & 850 \\
\(30-39\) & 176 & 152 & 193 & 103 & 35 & 659 \\
\(40-49\) & 118 & 144 & 230 & 139 & 57 & 688 \\
\(50-59\) & 160 & 302 & 494 & 379 & 152 & 1,487 \\
\(60-69\) & 58 & 103 & 142 & 164 & 71 & 538 \\
\(70+\) & & & & & & \\
\hline
\end{tabular}

In the table of frequencies, we see that of the 1,204 females in their 20 s , only 390 rated their health as excellent. In the table of percentages, we see that this is equal to \(32.39 \%\) of females in their 20s.

If we wish to include one of these tables, for example, in a paper or on a webpage, we can easily export it in \(\mathrm{LAT}_{\mathrm{E}} \mathrm{X}\), Word, Excel, HTML, and a variety of other formats by using collect export.

\section*{Stored results}
table stores the following in \(r()\) :
Scalars \(r(N)\) number of observations

\section*{Reference}

McDowell, A., A. Engel, J. T. Massey, and K. Maurer. 1981. Plan and operation of the Second National Health and Nutrition Examination Survey, 1976-1980. Vital and Health Statistics 1(15): 1-144.

\section*{Also see}
[R] table - Table of frequencies, summaries, and command results
[R] table intro - Introduction to tables of frequencies, summaries, and command results
[R] table oneway - One-way tabulation
[R] table twoway - Two-way tabulation
[TABLES] Intro - Introduction

\section*{Title}
table summary - Table of summary statistics
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & References \\
Also see & & &
\end{tabular}

\section*{Description}

In this entry, we discuss how to use table to create a table of summary statistics.

\section*{Quick start}

Table with the mean of \(v 1, \mathrm{v} 2\), and v 3 for each category of a1 and a2; rows are defined by categories of a1 and variables v1, v2, and v3
table a1 a2, stat(mean v1 v2 v3)
Same as above, but also report standard deviations and suppress the totals; rows are defined by the results for each variable within each category of a1
```

table (a1 var result) (a2), stat(mean v1 v2 v3) ///
stat(sd v1 v2 v3) nototals

```

Table with number of observations in each category of a2 and a3, for each level of a1 table a1, stat(fvfrequency a2 a3)

Same as above, and report percentage of observations in each category
```

table a1, stat(fvfrequency a2 a3) ///
stat(fvpercent a2 a3)

```

Same as above, and report the percentages with a percent sign, using two decimal places, and enclose them in parentheses
```

table (a1) (var result), stat(fvfrequency a2 a3) ///
stat(fvpercent a2 a3) ///
nformat(%5.2f fvpercent) sformat("(%s%%)" fvpercent)

```

\section*{Menu}

Statistics \(>\) Summaries, tables, and tests \(>\) Tables of frequencies, summaries, and command results

\section*{Syntax}

Basic table of summary statistics
```

table [rowvar] [colvar] [if] [in] [weight], statistic(statspec)
[statistic(statspec) [...]][options]

```

Customized table of summary statistics
```

table [(rowspec)] [(colspec)] [(tabspec)] [if] [in] [weight], statistic(statspec)
[statistic(statspec) [...]] [options]

```
rowspec, colspec, and tabspec may be empty or may include variable names or any of the following keywords:
\begin{tabular}{ll} 
keyword & Description \\
\hline result & requested statistics \\
var & variables from statistic() option \\
across & index across() specifications \\
\hline
\end{tabular}
options
Main
totals(totals)
nototals

Statistics
statistic (statspec)

Description
report only the specified totals
suppress the marginal totals
statistic to be reported; default is statistic (frequency) when no weights are specified and statistic (sumw) otherwise

Formats
nformat (\% fmt [results] [, basestyle]) sformat (sfmt \([\) results \(]\) )

Options
listwise
missing
showcounts
zerocounts
name (cname)
append
replace
label (filename)
style(filename [, override])
markvar (newvar)
specify numeric format
specify string format
use listwise deletion to handle missing values treat numeric missing values of variables in rowspec, colspec, and tabspec like other values show sample size for all variables in statistic() option
report 0 for empty cell counts
collect results into a collection named cname append results to an existing collection replace results of an existing collection specify the collection labels
specify the collection style
create newvar that identifies observations used in the tabulation
fweights, aweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
strL variables are not allowed; see [U] 12.4.8 strL.
markvar() does not appear in the dialog box.

\section*{Options}
\(\qquad\) Main
totals (totals) and nototals control which totals are to be displayed in the table. By default, all totals are reported.
totals (totals) specifies which margin totals to display in the reported table. totals can contain variables in rowspec, colspec, tabspec, and their interaction. Interactions can be specified by using the \# operator.
nototals prevents table from displaying any totals.

Statistics
statistic (statspec) specifies the statistic to be displayed. statistic() may be repeated to request multiple statistics. Frequency statistics, summary statistics, and ratio statistics are available by specifying statistic (freqstat), statistic (sumstat varlist), and statistic (ratiostat [varlist] [, ratio_options]), respectively.
statistic() may be repeated to request multiple statistics.
statistic (freqstat) specifies that frequencies be computed.
\begin{tabular}{ll} 
freqstat & Definition \\
\hline \begin{tabular}{ll} 
frequency & frequency \\
sumw & sum of weights \\
\hline
\end{tabular} \\
\hline
\end{tabular}
statistic (sumstat varlist) specifies that summary statistic sumstat be computed for the variables in varlist.
\begin{tabular}{ll} 
sumstat & Definition \\
\hline mean & mean \\
semean & standard error of the mean \\
sebinomial & \begin{tabular}{l} 
standard error of the mean, binomial \\
sepoisson
\end{tabular} \\
standard error of the mean, Poisson
\end{tabular}
statistic (ratiostat \([\) varlist \(]\) [, ratio_options \(]\) ) specifies that ratio statistic ratiostat be computed. If varlist is specified, ratios are computed based on the totals of the specified variables. If varlist is not specified, ratios are computed based on frequencies.
\begin{tabular}{ll} 
ratiostat & Definition \\
\hline \begin{tabular}{l} 
proportion
\end{tabular} & proportion \\
percent & percentage \\
\hline rawproportion & proportion ignoring optionally specified weights \\
rawpercent & percentage ignoring optionally specified weights \\
\hline \hline
\end{tabular}
\begin{tabular}{ll} 
ratio_options & Definition \\
\hline across (cellspec) & \begin{tabular}{c} 
percentages or proportions across levels of \\
variables or interactions
\end{tabular} \\
total & \begin{tabular}{c} 
compute overall percentages or proportions
\end{tabular} \\
\hline
\end{tabular}
cellspec may contain any variables in rowspec, colspec, tabspec, or an interaction between any of these variables. Interactions can be specified by using the \# operator.

Formats
nformat ( \(\%\) fint [results] [, basestyle]) changes the numeric format, such as the number of decimal places, for specified results. If results are not specified, the numeric format is changed for all results.
results may be any statistic named in option statistic() (that is, any freqstat, sumstat, or ratiostat).

This option is repeatable, and when multiple formats apply to one result, the rightmost specification is applied.
This option does not affect the format of numeric layout variables (rowspec, colspec, and tabspec) or the format of factor variables specified in the statistic () option. The default format of these variables is taken from the dataset.
basestyle indicates that the format be applied to results that do not already have their own format instead of overriding the format for all results.
sformat (sfint [results]) changes the string format for specified results. You can, for instance, add symbols or text to the values reported in the table by modifying the string format.
sfint may contain a mix of text and \(\%\) s. Here \(\%\) s refers to the numeric value that is formatted as specified using nformat (). The text will be placed around the numeric values in your table as it is placed around \(\%\) s in this option. For instance, to place parentheses around the percent statistics, you can specify sformat (" (\%s)" percent).
results may be any statistic named in option statistic() (that is, any freqstat, sumstat, or ratiostat).
Two text characters must be specified using a special character sequence if you want them to be displayed in your table. To include \(\%\), type \(\% \%\). To include \(\backslash\), type \(\backslash \backslash\). For instance, to place a percent sign following percent statistics, you can specify sformat("\%s\%\%" percent).
This option is repeatable, and when multiple formats apply to one result, the rightmost specification is applied.
listwise handles missing values through listwise deletion, meaning that the entire observation is omitted from the sample if any variable specified in a statistic() option is missing for that observation. By default, table will omit an observation only if all variables specified in all statistic() options are missing for that observation.
missing specifies that numeric missing values of any variables specified in rowspec, colspec, or tabspec be treated as valid categories. By default, observations with a numeric missing value in any of these variables are omitted.

This option does not apply to factor variables specified with statistics fvfrequency, fvrawfrequency, fvproportion, fvrawproportion, fvpercent, or fvrawpercent.
showcounts specifies that table report the sample size for each variable specified in option statistic().
zerocounts specifies that table report a 0 in empty cells for results count, frequency, fvfrequency, and fvrawfrequency.
name (cname) specifies that a collection named cname be associated with the collected statistics and results. The default is name (Table).
append specifies that table append its collection information into the collection named in name(). replace permits table to overwrite an existing collection. This option is implied for name (Table) when append is not specified.
label (filename) specifies the filename containing the collection labels to use for your table. Labels in filename will be loaded for the table, and any labels not specified in filename will be taken from the labels defined in \(c\) (collect_label). The default is to use only the collection labels set in c(collect_label); see [TABLES] set collect_label.
style(filename [, override]) specifies the filename containing the collection styles to use for your table. The default collection styles will be discarded, and only the collection styles in filename will be applied.

If you prefer the default collection styles but also want to apply any styles in filename, specify override. If there are conflicts between the default collection styles and those in filename, the ones in filename will take precedence.
The default is to use only the collection styles set in c (table_style); see [TABLES] set table_style.
The following option is available with table but is not shown in the dialog box:
markvar (newvar) generates an indicator variable that identifies the observations used in the tabulation.

\section*{Remarks and examples}

Remarks are presented under the following headings:

\section*{Basic summary statistic tables}

The table command can be used to compute a variety of summary statistics and display them in a table. Summary statistics can be computed for the full dataset or across levels of one or more categorical variables.

To demonstrate, we use data from the Second National Health and Nutrition Examination Survey (NHANES II) (McDowell et al. 1981) and create a table reporting the mean body mass index (BMI) of individuals across four regions of the USA. We use the statistic() option to request that means be computed, and we specify region as our row variable for the table. Thus, means are computed for each region separately and for all the regions combined (Total).
```

. use https://www.stata-press.com/data/r18/nhanes2l
(Second National Health and Nutrition Examination Survey)
. table region, statistic(mean bmi)

|  | Mean |
| :--- | ---: |
| Region |  |
| NE | 25.57535 |
| MW | 25.51936 |
| S | 25.63317 |
| W | 25.42299 |
| Total | 25.5376 |

```

The mean BMI is very similar across regions. We might want to look at some additional statistics. We can add the minimums and maximums in our table by repeating our statistic() option for each statistic; we will use the stat() abbreviation.
```

. table region, stat(mean bmi) stat(min bmi) stat(max bmi)

```
\begin{tabular}{l|rrr}
\hline & Mean & Minimum value & Maximum value \\
\hline Region & & & \\
NE & 25.57535 & 15.36715 & 57.10803 \\
MW & 25.51936 & 14.1351 & 61.1297 \\
S & 25.63317 & 12.3856 & 55.43552 \\
W & 25.42299 & 15.69046 & 54.05056 \\
Total & 25.5376 & 12.3856 & 61.1297 \\
\hline
\end{tabular}

If we want to include even more statistics, the table will become very wide. We can move the statistics to the rows of our table by specifying the keyword result in the first set of parentheses. We place region on the columns by specifying this variable in the second set of parentheses.
```

. table (result) (region),
> stat(mean bmi) stat(median bmi) stat(sd bmi)
> stat(min bmi) stat(max bmi)

```
\begin{tabular}{l|rrrrr}
\hline & NE & MW & Region & W & Total \\
& & & & \\
\hline Mean & 25.57535 & 25.51936 & 25.63317 & 25.42299 & 25.5376 \\
Median & 25.00623 & 24.71567 & 24.98451 & 24.66734 & 24.81812 \\
Standard deviation & 4.72798 & 4.905965 & 5.084678 & 4.883534 & 4.914969 \\
Minimum value & 15.36715 & 14.1351 & 12.3856 & 15.69046 & 12.3856 \\
Maximum value & 57.10803 & 61.1297 & 55.43552 & 54.05056 & 61.1297 \\
\hline
\end{tabular}

Instead of computing many statistics for one variable, we might want to compute one statistic for multiple variables. To do this, we can include a list of variables within a single statistic() option. Let's compute the means of age, BMI, and systolic blood pressure (bpsystol).
```

. table (result) (region), stat(mean age bmi bpsystol)

```
\begin{tabular}{l|rrrrr}
\hline & NE & MW & Region & S & W \\
& & & & Total \\
\hline Age (years) & 47.81584 & 46.52776 & 48.19068 & 47.83828 & 47.57965 \\
Body mass index (BMI) & 25.57535 & 25.51936 & 25.63317 & 25.42299 & 25.5376 \\
Systolic blood pressure & 131.3836 & 130.4863 & 131.1626 & 130.5936 & 130.8817 \\
\hline
\end{tabular}

\section*{Classic Table 1}

In many reports, the first discussion of the data is accompanied by a "Table 1", a reporting of summary statistics for all variables of interest. Often the table includes a mixture of continuous and categorical variables. We may also want to specify these in a particular order based on importance. In many cases, the table has multiple columns with the summary statistics reported for each level of a categorical variable of interest.

Here we will demonstrate how to create one variety of such a table. We have two factor variables of interest, diabetes and hlthstat, for which we would like to compute the percentage of individuals in each category. We will use the fvpercent statistic to obtain these percentages. We also have three continuous variables, age, bmi, and bpsystol, for which we would like to compute means. We specify the statistic() options in the order we wish to see the results in the table. To specify that the variables in the statistic() options appear on the rows, we include the keyword var in the first set of parentheses. We place region on the columns by listing it in the second set of parentheses.
```

. table (var) (region),
> statistic(fvpercent diabetes)
> statistic(mean age bmi)
> statistic(fvpercent hlthstat)
> statistic(mean bpsystol)

```
\begin{tabular}{|c|c|c|c|c|c|}
\hline \multirow[t]{2}{*}{} & \multicolumn{5}{|c|}{Region} \\
\hline & NE & MW & S & W & Total \\
\hline Diabetes status=Not diabetic Factor-variable percent & 95.32 & 95.49 & 94.36 & 95.62 & 95.18 \\
\hline \begin{tabular}{l}
Diabetes status=Diabetic \\
Factor-variable percent
\end{tabular} & 4.68 & 4.51 & 5.64 & 4.38 & 4.82 \\
\hline Age (years) Mean & 47.81584 & 46.52776 & 48.19068 & 47.83828 & 47.57965 \\
\hline Body mass index (BMI) Mean & 25.57535 & 25.51936 & 25.63317 & 25.42299 & 25.5376 \\
\hline Health status=Excellent Factor-variable percent & 26.95 & 26.33 & 19.14 & 21.68 & 23.29 \\
\hline Health status=Very good Factor-variable percent & 26.76 & 26.01 & 22.82 & 25.18 & 25.07 \\
\hline Health status=Good Factor-variable percent & 30.26 & 26.52 & 28.29 & 29.14 & 28.43 \\
\hline \begin{tabular}{l}
Health status=Fair \\
Factor-variable percent
\end{tabular} & 12.33 & 15.12 & 18.65 & 17.60 & 16.16 \\
\hline Health status=Poor Factor-variable percent & 3.69 & 6.02 & 11.11 & 6.40 & 7.05 \\
\hline Systolic blood pressure Mean & 131.3836 & 130.4863 & 131.1626 & 130.5936 & 130.8817 \\
\hline
\end{tabular}

We have the statistics we want, but clearly our table could be improved. Let's start by applying one of the predefined styles, table-1, by adding the style() option.
```

. table (var) (region),
> statistic(fvpercent diabetes)
> statistic(mean age bmi)
> statistic(fvpercent hlthstat)
> statistic(mean bpsystol) style(table-1)

```
\begin{tabular}{|c|c|c|c|c|c|}
\hline \multirow[t]{2}{*}{} & \multicolumn{5}{|c|}{Region} \\
\hline & NE & MW & S & W & Total \\
\hline Diabetes status & & & & & \\
\hline Not diabetic & 95.32 & 95.49 & 94.36 & 95.62 & 95.18 \\
\hline Diabetic & 4.68 & 4.51 & 5.64 & 4.38 & 4.82 \\
\hline Age (years) & 47.81584 & 46.52776 & 48.19068 & 47.83828 & 47.57965 \\
\hline Body mass index (BMI) & 25.57535 & 25.51936 & 25.63317 & 25.42299 & 25.5376 \\
\hline Health status Excellent & 26.95 & 26.33 & 19.14 & 21.68 & 23.29 \\
\hline Very good & 26.76 & 26.01 & 22.82 & 25.18 & 25.07 \\
\hline Good & 30.26 & 26.52 & 28.29 & 29.14 & 28.43 \\
\hline Fair & 12.33 & 15.12 & 18.65 & 17.60 & 16.16 \\
\hline Poor & 3.69 & 6.02 & 11.11 & 6.40 & 7.05 \\
\hline Systolic blood pressure & 131.3836 & 130.4863 & 131.1626 & 130.5936 & 130.8817 \\
\hline
\end{tabular}

This style removes the labels for the type of statistic being reported, cleans up the reporting of factor variables in the row headers, and right-aligns the content in the row headers. In addition, we may want to specify that the means be reported to two decimal places using the nformat (\% \(6.2 f\) mean) option.
```

. table (var) (region),
> statistic(fvpercent diabetes)
> statistic(mean age bmi)
> statistic(fvpercent hlthstat )
> statistic(mean bpsystol) style(table-1) nformat(%6.2f mean)

```
\begin{tabular}{|c|c|c|c|c|c|}
\hline \multirow[t]{2}{*}{} & \multicolumn{5}{|c|}{Region} \\
\hline & NE & MW & S & W & Total \\
\hline Diabetes status & & & & & \\
\hline Not diabetic & 95.32 & 95.49 & 94.36 & 95.62 & 95.18 \\
\hline Diabetic & 4.68 & 4.51 & 5.64 & 4.38 & 4.82 \\
\hline Age (years) & 47.82 & 46.53 & 48.19 & 47.84 & 47.58 \\
\hline Body mass index (BMI) & 25.58 & 25.52 & 25.63 & 25.42 & 25.54 \\
\hline Health status Excellent & 26.95 & 26.33 & 19.14 & 21.68 & 23.29 \\
\hline Very good & 26.76 & 26.01 & 22.82 & 25.18 & 25.07 \\
\hline Good & 30.26 & 26.52 & 28.29 & 29.14 & 28.43 \\
\hline Fair & 12.33 & 15.12 & 18.65 & 17.60 & 16.16 \\
\hline Poor & 3.69 & 6.02 & 11.11 & 6.40 & 7.05 \\
\hline Systolic blood pressure & 131.38 & 130.49 & 131.16 & 130.59 & 130.88 \\
\hline
\end{tabular}

Let's go one step further. Perhaps we want the mean and standard deviation of each continuous variable, and we want the frequency and percent for each factor variable. We need to specify a few more statistic() options.
\begin{tabular}{|c|c|c|c|c|c|}
\hline \multicolumn{6}{|l|}{```
. table (var) (region),
> stat(fvfreq diabetes) statistic(fvpercent diabetes)
> statistic(mean age bmi) statistic(sd age bmi)
> statistic(fvfreq hlthstat) statistic(fvpercent hlthstat)
> statistic(mean bpsystol) statistic(sd bpsystol)
> style(table-1) nformat(%6.2f mean sd)
```} \\
\hline & \multicolumn{5}{|c|}{Region} \\
\hline \begin{tabular}{l}
Diabetes status \\
Not diabetic \\
Diabetic
\end{tabular} & \[
\begin{array}{r}
1,997 \\
95.32 \\
98 \\
4.68
\end{array}
\] & \[
\begin{array}{r}
2,648 \\
95.49 \\
125 \\
4.51
\end{array}
\] & \[
\begin{array}{r}
2,692 \\
94.36 \\
161 \\
5.64
\end{array}
\] & \[
\begin{array}{r}
2,513 \\
95.62 \\
115 \\
4.38
\end{array}
\] & \[
\begin{array}{r}
9,850 \\
95.18 \\
499 \\
4.82
\end{array}
\] \\
\hline Age (years) & \[
\begin{aligned}
& 47.82 \\
& 17.02
\end{aligned}
\] & \[
\begin{aligned}
& 46.53 \\
& 17.38
\end{aligned}
\] & \[
\begin{aligned}
& 48.19 \\
& 16.86
\end{aligned}
\] & \[
\begin{aligned}
& 47.84 \\
& 17.53
\end{aligned}
\] & \[
\begin{aligned}
& 47.58 \\
& 17.21
\end{aligned}
\] \\
\hline Body mass index (BMI) & \[
\begin{array}{r}
25.58 \\
4.73
\end{array}
\] & \[
\begin{array}{r}
25.52 \\
4.91
\end{array}
\] & \[
\begin{array}{r}
25.63 \\
5.08
\end{array}
\] & \[
\begin{array}{r}
25.42 \\
4.88
\end{array}
\] & \[
\begin{array}{r}
25.54 \\
4.91
\end{array}
\] \\
\hline Health status Excellent & 562
26.95 & 730
26.33 & 546
19.14 & 569
21.68 & 2,407 \\
\hline \multirow[t]{2}{*}{Very good} & 558 & 721 & 651 & 661 & 2,591 \\
\hline & 26.76 & 26.01 & 22.82 & 25.18 & 25.07 \\
\hline \multirow[t]{2}{*}{Good} & 631 & 735 & 807 & 765 & 2,938 \\
\hline & 30.26 & 26.52 & 28.29 & 29.14 & 28.43 \\
\hline \multirow[t]{2}{*}{Fair} & 257 & 419 & 532 & 462 & 1,670 \\
\hline & 12.33 & 15.12 & 18.65 & 17.60 & 16.16 \\
\hline \multirow[t]{2}{*}{Poor} & 77 & 167 & 317 & 168 & 729 \\
\hline & 3.69 & 6.02 & 11.11 & 6.40 & 7.05 \\
\hline \multirow[t]{2}{*}{Systolic blood pressure} & 131.38 & 130.49 & 131.16 & 130.59 & 130.88 \\
\hline & 24.31 & 22.50 & 24.21 & 22.42 & 23.33 \\
\hline
\end{tabular}

Finally, to distinguish among our statistics, we can use the sformat () option to add parentheses around our standard deviations and percent signs to our percentages.
```

. table (var) (region),
> stat(fvfreq diabetes) statistic(fvpercent diabetes)
> statistic(mean age bmi) statistic(sd age bmi)
> statistic(fvfreq hlthstat) statistic(fvpercent hlthstat)
> statistic(mean bpsystol) statistic(sd bpsystol)
> style(table-1) nformat(%6.2f mean sd)
> sformat("(%s)" sd) sformat("%s%%" fvpercent)

```
\begin{tabular}{|c|c|c|c|c|c|}
\hline \multirow[t]{2}{*}{} & \multicolumn{5}{|c|}{Region} \\
\hline & NE & MW & S & W & Total \\
\hline \multicolumn{6}{|l|}{Diabetes status} \\
\hline \multirow[t]{2}{*}{Not diabetic} & 1,997 & 2,648 & 2,692 & 2,513 & 9,850 \\
\hline & 95.32\% & 95.49\% & 94.36\% & 95.62\% & 95.18\% \\
\hline \multirow[t]{2}{*}{Diabetic} & 98 & 125 & 161 & 115 & 499 \\
\hline & 4.68\% & 4.51\% & 5.64\% & 4.38\% & 4.82\% \\
\hline \multirow[t]{2}{*}{Age (years)} & 47.82 & 46.53 & 48.19 & 47.84 & 47.58 \\
\hline & (17.02) & (17.38) & (16.86) & (17.53) & (17.21) \\
\hline \multirow[t]{2}{*}{Body mass index (BMI)} & 25.58 & 25.52 & 25.63 & 25.42 & 25.54 \\
\hline & (4.73) & (4.91) & (5.08) & (4.88) & (4.91) \\
\hline \multicolumn{6}{|l|}{Health status} \\
\hline Excellent & 26.95\% & 26.33\% & 19.14\% & 21.68\% & 2, \(23.29 \%\) \\
\hline \multirow[t]{2}{*}{Very good} & 558 & 721 & 651 & 661 & 2,591 \\
\hline & 26.76\% & 26.01\% & 22.82\% & 25.18\% & 25.07\% \\
\hline \multirow[t]{2}{*}{Good} & 631 & 735 & 807 & 765 & 2,938 \\
\hline & 30.26\% & 26.52\% & 28.29\% & 29.14\% & 28.43\% \\
\hline \multirow[t]{2}{*}{Fair} & 257 & 419 & 532 & 462 & 1,670 \\
\hline & 12.33\% & 15.12\% & 18.65\% & 17.60\% & 16.16\% \\
\hline \multirow[t]{2}{*}{Poor} & 77 & 167 & 317 & 168 & 729 \\
\hline & 3.69\% & 6.02\% & 11.11\% & 6.40\% & 7.05\% \\
\hline \multirow[t]{2}{*}{Systolic blood pressure} & 131.38 & 130.49 & 131.16 & 130.59 & 130.88 \\
\hline & (24.31) & (22.50) & (24.21) & (22.42) & (23.33) \\
\hline
\end{tabular}

We have added many customizations to our table. However, you may prefer a different look. For another style, you can select from the predefined styles described in [TABLES] Predefined styles. If none of these provide the exact style you want for your table, you can further customize the results by using the collect suite of commands. To learn more, see [TABLES] Intro.

If you wish to include this table in a paper, on a webpage, or in another format, you can easily export it in \(\mathrm{LAT}_{\mathrm{E}} \mathrm{X}\), Word, Excel, HTML, and a variety of other formats by using collect export.

\section*{Stored results}
table stores the following in \(r()\) :
Scalars
\(r(N)\) number of observations

\section*{References}

Huber, C. 2021a. Customizable tables in Stata 17, part 3: The classic table 1. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2021/06/24/customizable-tables-in-stata-17-part-3-the-classic-table-1/.
——. 2021b. Customizable tables in Stata 17, part 4: Table of statistical tests. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2021/08/24/customizable-tables-in-stata-17-part-4-table-of-statistical-tests/.
McDowell, A., A. Engel, J. T. Massey, and K. Maurer. 1981. Plan and operation of the Second National Health and Nutrition Examination Survey, 1976-1980. Vital and Health Statistics 1(15): 1-144.

\section*{Also see}
[R] dtable - Create a table of descriptive statistics
\([R]\) table intro - Introduction to tables of frequencies, summaries, and command results
[R] tabstat - Compact table of summary statistics
[TABLES] Intro - Introduction

\section*{Title}
table hypothesis tests - Table of hypothesis tests
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Reference \\
Also see & & &
\end{tabular}

\section*{Description}

In this entry, we discuss how to use table to create tables with results of hypothesis tests.

\section*{Quick start}

Table with pairwise correlations stored in matrix \(r(C)\)
table (rowname) (colname), command ( \(\mathrm{r}(\mathrm{C})\) : pwcorr v1 v2 v3)
Table with all the numeric scalars returned by ttest; rows correspond to the different results table (result) (command), command(ttest v1, by (catvar))

Table with means and two-sided \(p\)-values; columns correspond to the different results
table (command) (result), ///
command (r(mu_1) r(mu_2) \(r(p):\) ttest v1, by (catvar))
Same as above, but with statistics for v1 and v2
table (command) (result), /// command(r(mu_1) r(mu_2) r(p): ttest v1, by(catvar)) /// command(r(mu_1) r(mu_2) r(p): ttest v2, by(catvar))

\section*{Menu}

Statistics \(>\) Summaries, tables, and tests \(>\) Tables of frequencies, summaries, and command results

\section*{Syntax}
table ([rowspec]) ([colspec]) [(tabspec)] [if] [in] [weight], command (cmdspec) [command(cmdspec) ...] [options]
rowspec, colspec, and tabspec may be empty or may include variable names or any of the following keywords:
\begin{tabular}{ll} 
keyword & Description \\
\hline result & requested statistics \\
stars & stars denoting statistical significance \\
command & index option command () \\
colname & column names for matrix statistics \\
rowname & row names for matrix statistics \\
\hline
\end{tabular}
options
Commands
command (cmdspec)
Formats
nformat (\% fmt [results][, basestyle]) sformat (sfint [results])

Stars
stars(starspec)
Options
missing
name (cname)
append
replace
label(filename)
style(filename [, override])
noisily

Description
collect results from the specified Stata command
specify numeric format
specify string format
add stars to denote statistical significance
treat numeric missing values like other values
collect results into a collection named cname
append results to an existing collection
replace results of an existing collection
specify the collection labels
specify the collection style
display output from each command
fweights, aweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
strL variables are not allowed; see [U] 12.4.8 strL.
noisily does not appear in the dialog box.

\section*{Options}

Commands
command (cmdspec) specifies the Stata commands from which to collect results. command () may be repeated to collect results from multiple commands.
cmdspec is [explist: ] command [arguments] [, cmdoptions]
explist specifies which results to collect and report in the table. explist may include result identifiers and named expressions.
result identifiers are results stored in r() and e() by the command. For instance, result identifiers could be \(\mathrm{r}(\) mean \(), \mathrm{r}(\mathrm{C})\), or e (chi2). After estimation commands, result identifiers also include the following:
\begin{tabular}{|c|c|}
\hline Identifier & Result \\
\hline _r_b & coefficients or transformed coefficients reported by command \\
\hline _r_se & standard errors of _r_b \\
\hline -r_z & test statistics for _r_b \\
\hline _r_z_abs & absolute value of _r_z \\
\hline -r_p & \(p\)-values for _r_b \\
\hline _r_lb & lower bounds of confidence intervals for \(\_\)r_b \\
\hline _r_ub & upper bounds of confidence intervals for _r_b \\
\hline _r_ci & confidence intervals for \(\quad\) r_b \\
\hline _r_crlb & lower bounds of credible intervals for _r_b \\
\hline _r_crub & upper bounds of credible intervals for \(\quad\) r_b \\
\hline _r_cri & credible intervals for _r_b \\
\hline _r_df & degrees of freedom for \(-\mathrm{r} \_\)b \\
\hline
\end{tabular}
named expressions are specified as name \(=\) exp, where name may be any valid Stata name and exp is an expression, typically an expression that involves one or more result identifiers. An example of a named expression is sd \(=\operatorname{sqrt}(r\) (variance)).
For r-class commands, the default is to include all numeric scalars posted to \(r()\) in the table results. For e-class commands, the default is to include \(\_r \_b\) in the table results.
command is any command that follows standard Stata syntax.
arguments may be anything so long as they do not include an if clause, in range, or weight specification.

Any if or in qualifier and weights should be specified directly with table, not within the command() option. Weights are passed to command only if they are specified.
cmdoptions may be anything supported by command.

Formats
nformat ( \(\%\) fint [results][, basestyle]) changes the numeric format, such as the number of decimal places, for specified results. If results are not specified, the numeric format is changed for all results.
results may be any name in the e() or r() results produced by commands specified in option command ().

This option is repeatable, and when multiple formats apply to one result, the rightmost specification is applied.

This option does not affect the format of numeric layout variables (rowspec, colspec, and tabspec). The default format of these variables is taken from the dataset.
basestyle indicates that the format be applied to results that do not already have their own format instead of overriding the format for all results.
sformat (sfmt [results]) changes the string format for specified results. You can, for instance, add symbols or text to the values reported in the table by modifying the string format.
sfint may contain a mix of text and \(\%\) s. Here \(\%\) s refers to the numeric value that is formatted as specified using nformat(). The text will be placed around the numeric values in your table as it is placed around \(\% \mathrm{~s}\) in this option. For instance, to place parentheses around the percent statistics, you can specify sformat(" (\%s)" percent).
results may be any name in the e() or r() results produced by commands specified in option command ().
Two text characters must be specified using a special character sequence if you want them to be displayed in your table. To include \(\%\), type \(\% \%\). To include \(\backslash\), type \(\backslash \backslash\). For instance, to place a percent sign following percent statistics, you can specify sformat("\%s\%\%" percent).

This option is repeatable, and when multiple formats apply to one result, the rightmost specification is applied.

\section*{Stars}
stars (starspec) specifies that stars representing statistical significance be included in the table. starspec identifies the result whose values determine significance, which characters should represent each significance level, and where these characters should be displayed in the table. starspec is
starres [\#1 "label1" [\#2 "label2" [\#3 "label3" [\#4 "label4" [\#5 "label5" ] ]] ]]
[, attach(attachres) result dimension starsnoteopts]
starres is the name of the result whose values determine which characters, typically which number of stars, are to be displayed.
labell specifies the characters to be displayed when starres \(<\# 1\).
label2 specifies the characters to be displayed when starres \(<\# 2\).
label3 specifies the characters to be displayed when starres \(<\# 3\).
label4 specifies the characters to be displayed when starres \(<\# 4\).
label5 specifies the characters to be displayed when starres \(<\# 5\).
attach(attachres) specifies the name of the result to which the characters defined by labell, \(\ldots\). label5 are to be attached. If attach() is not specified, a new result named stars is created and is automatically added to the table.
result and dimension control how collect stars adds items when labeling significant results. These options are mutually exclusive.
result specifies the default behavior, and this option is necessary only if the following dimension behavior is in effect and you want to change back to the result behavior. dimension specifies that dimension stars be added to the collection. Items will be tagged with stars [value], and the labels will be tagged with stars [label]. Use this option for layouts where results are to be stacked within columns, and use new dimension stars in the column specification of the layout.
starsnoteopts control the display and composition of the stars note.
noshownote and shownote control whether to display the stars note. increasing and decreasing control the order of \(p\)-values in the stars note. pvname (string) specifies a name for the \(p\)-value in the stars note. The default is pvname ( p ). delimiter (string) specifies the delimiter between labels in the stars note. The default is delimiter (", ").
nformat ( \(\%\) fint ) specifies the numeric format for the cutoff values in the stars note. The default is nformat ( \(\% 9.0 \mathrm{~g}\) ).
prefix (string) specifies the prefix for the stars note. The prefix is empty by default. suffix (string) specifies the suffix for the stars note. The suffix is empty by default.
For example, stars(_r_p 0.01 "***" 0.05 "**" 0.1 "*", attach(_r_b)) could be added to a table of regression results to specify that stars be defined based on the \(p\)-values in _r_p and be attached to the reported coefficients (_r_b).

Options
missing specifies that numeric missing values of any variables specified in rowspec, colspec, or tabspec be treated as valid categories. By default, observations with a numeric missing value in any of these variables are omitted.
name (cname) specifies that a collection named cname be associated with the collected statistics and results. The default is name (Table).
append specifies that table append its collection information into the collection named in name ().
replace permits table to overwrite an existing collection. This option is implied for name (Table) when append is not specified.
label (filename) specifies the filename containing the collection labels to use for your table. Labels in filename will be loaded for the table, and any labels not specified in filename will be taken from the labels defined in \(c\) (collect_label). The default is to use only the collection labels set in c(collect_label); see [TABLES] set collect_label.
style(filename [, override]) specifies the filename containing the collection styles to use for your table. The default collection styles will be discarded, and only the collection styles in filename will be applied.
If you prefer the default collection styles but also want to apply any styles in filename, specify override. If there are conflicts between the default collection styles and those in filename, the ones in filename will take precedence.

The default is to use only the collection styles set in c (table_style); see [TABLES] set table_style.
The following option is available with table but is not shown in the dialog box:
noisily specifies that output from the commands specified in command() options be displayed. By default, output from commands is suppressed.

\section*{Remarks and examples}

Remarks are presented under the following headings:

\author{
Introduction \\ Creating tables from scalars \\ Creating tables from matrices
}

\section*{Introduction}

The table command can be used to create tables with results of hypothesis tests. For example, you can create a table with results from a mean-comparison test, a test of proportions, or a test of normality.
table does not perform hypothesis tests directly. Rather, table will run any Stata command that you include in its command () option and place results from that command into the table. You determine which results you would like to see in the table. You can select any of the results stored by the command.

\section*{Creating tables from scalars}

We have data from the Second National Health and Nutrition Examination Survey (NHANES II) (McDowell et al. 1981). The data contain some demographic information, such as the age, sex, and race of participants. The data also contain some measures of health, including whether the individual has high blood pressure (highbp), has diabetes, or has had a heart attack previously (heartatk).

Suppose we want to examine the proportion of males and females that have high blood pressure, that have diabetes, and that have had a heart attack previously. With prtest, we can test whether the proportions are equal between males and females. For example, let's perform a test of proportions for diabetes:


We would like to create a table that includes the proportion of men who have diabetes, the proportion of women who have diabetes, the difference in these proportions, and the \(p\)-value for a two-sided test. First, we need to determine how to refer to these statistics.
```

. return list
scalars:

| r(N1) | 4915 |
| :---: | :---: |
| r(N2) | 5434 |
| r(P1) | . 0441505595116989 |
| r(P2) | . 0518954729481045 |
| r(P_diff) | -. 0077449134364056 |
| $r$ (se1) | . 0029302258134317 |
| r(se2) | . 003009075122777 |
| r(se_diff0) | . 0042169418903878 |
| r(se_diff) | . 0042000900481081 |
| r(lb1) | . 0384074224508032 |
| r (ub1) | . 0498936965725946 |
| r(lb2) | . 0459977940806861 |
| $r$ (ub2) | . 0577931518155229 |
| r(lb_diff) | -. 0159769386625226 |
| r(ub_diff) | . 0004871117897114 |
| r (z) | -1.836618487454034 |
| $r\left(p \_l\right)$ | . 0331331180748532 |
| $r$ (p) | . 0662662361497065 |
| $r\left(p \_u\right)$ | . 9668668819251468 |
| $r$ (level) | 95 |

```

The statistics we want to see are stored as \(r(P 1), r(P 2), r\left(P \_d i f f\right)\), and \(r(p)\). We can specify this in the command () option by typing
```

. table ..., command(r(P1) r(P2) r(P3) r(p): prtest diabetes, by(sex))

```

This will get the results we want into our table. Furthermore, because we know what these values represent, we can give them names that will appear in the table headers. We can, for instance, type
```

. table ..., command(Males=r(P1) Females=r(P2) Difference=r(P3) ///
r(p): prtest diabetes, by(sex))

```

We can specify similar command() options for heartatk and highbp as well.
In addition, we need to specify how our results will be laid out in the table. Below, we type command in the first set of parentheses so that the rows correspond to the different commands. We type result in the second set of parentheses to specify that statistics appear in the columns.

Finally, we add two options to customize the results. We specify a numeric format so that the statistics be displayed only with three digits after the decimal. We also choose the predefined style table-right so that our row headers will be right-aligned. See [TABLES] Predefined styles for information on this and other styles.
```

. table (command) (result),
> command(Males=r(P1) Females=r(P2) Difference=r(P_diff) r(p):
> prtest diabetes, by(sex))
> command(Males=r(P1) Females=r(P2) Difference=r(P_diff) r(p):
> prtest heartatk, by(sex))
> command(Males=r(P1) Females=r(P2) Difference=r(P_diff) r(p):
> prtest highbp, by(sex))
> nformat(%5.3f) style(table-right)

```
\begin{tabular}{r|rrrr}
\hline & Males & Females & Difference & Two-sided p-value \\
\hline prtest diabetes, by (sex) & 0.044 & 0.052 & -0.008 & 0.066 \\
prtest heartatk, by (sex) & 0.065 & 0.029 & 0.036 & 0.000 \\
prtest highbp, by (sex) & 0.469 & 0.381 & 0.088 & 0.000 \\
\hline
\end{tabular}

Our table now includes all the statistics we want. Yet we might want to make some modifications. Table customization can go beyond the predefined styles and options available to you in the table command. table creates a collection of results that can be used in combination with the collect suite of commands to produce highly customized tables and to export those tables to presentation-ready formats, such as HTML, Word, LATEX, PDF, Excel, and more.

For this table, we want to modify the labels in our row headers. Instead of showing the full command that was run, row headers will identify the variable we are testing. In addition, we will modify the label for our \(p\)-value. We want to use the label p-value. Because this is not a valid Stata name, we could not specify it in the table command as we did with Males. However, we can use collect label levels to modify the label on our \(p\)-values.

After applying the label updates, we use collect preview to see our updated table.
```

. collect label levels command 1 "Diabetes" 2 "Heart attack" 3 "High BP", modify
. collect label levels result p "p-value", modify
. collect preview

```
\begin{tabular}{r|rrrr}
\hline & Males & Females & Difference & p-value \\
\hline Diabetes & 0.044 & 0.052 & -0.008 & 0.066 \\
Heart attack & 0.065 & 0.029 & 0.036 & 0.000 \\
High BP & 0.469 & 0.381 & 0.088 & 0.000 \\
\hline
\end{tabular}

\section*{Creating tables from matrices}

You may find that the results you want to include in your table are stored in a matrix; these results can also be easily included in a table.

To demonstrate, we create a table with \(p\)-values for tests of normality for height, weight, and diastolic blood pressure (bpdiast). The command sktest performs tests based on skewness, kurtosis, and a combined test statistic.
```

. sktest height weight bpdiast
Skewness and kurtosis tests for normality

```
\begin{tabular}{r|rrrrr} 
Variable & Obs & \(\operatorname{Pr}\) (skewness) & \(\operatorname{Pr}\) (kurtosis) & Adj chi2(2) & Prob>chi2 \\
\hline height & 10,351 & 0.0000 & 0.0000 & 147.47 & 0.0000 \\
weight & 10,351 & 0.0000 & 0.0000 & 801.40 & 0.0000 \\
bpdiast & 10,351 & 0.0000 & 0.0000 & 362.54 & 0.0000
\end{tabular}

Let's look at the returned results.
```

. return list
scalars:
r(N) = 10351
r(p_skew) = 1.58706287446e-72
r(p_kurt) = 6.22330331716e-26
r(chi2) = 362.5385838320567
r(p_chi2) = 1.88689086684e-79
matrices:
r(table): 3 x 5

```

The statistics we want plus a few others are stored in \(r\) (table).

Now, let's place these values in a table. We specify that our table be arranged with the row names (rowname) of the matrix defining the rows of the table. Similarly, the column names (colname) of the matrix define the columns of the table. Then, we specify that we want to collect the results from the matrix \(r(t a b l e)\) from the sktest command.
```

. table (rowname) (colname),
> command(r(table): sktest height weight bpdiast)

```
\begin{tabular}{l|rrrrr}
\hline & N & p_skew & p_kurt & chi2 & p_chi2 \\
\hline Height (cm) & 10351 & .0000179 & \(1.87 \mathrm{e}-35\) & 147.4712 & \(9.48 \mathrm{e}-33\) \\
Weight (kg) & 10351 & \(1.6 \mathrm{e}-166\) & \(1.63 \mathrm{e}-49\) & 801.3958 & \(9.5 \mathrm{e}-175\) \\
Diastolic blood pressure & 10351 & \(1.59 \mathrm{e}-72\) & \(6.22 \mathrm{e}-26\) & 362.5386 & \(1.89 \mathrm{e}-79\) \\
\hline
\end{tabular}

Because the row names in \(r\) (table) corresponded to variables, our table automatically put the variable labels in the row headers. However, the column headers are not nicely labeled.

We can create better labels and modify our table in many other ways. table creates a collection of results that can be used in combination with the collect suite of commands to further customize tables.

To clean up our table, let's use collect label levels to modify the labels for the \(p\)-values for the skewness, kurtosis, and joint tests; these are the statistics we will include in our table below. To use collect label levels, we need to know just a little about the collect system. In collections, values are organized according to dimensions and levels within those dimensions. In fact, we use these dimensions in table. The keywords that we can use to define our rows and columns are dimensions. Here colname is our dimension that defines the columns, and its levels are \(\mathrm{N}, \mathrm{p}\) _skew, .... To modify labels, we need to tell collect label levels which dimension (colname) we would like to change and then specify labels for levels of that dimension.

We specify the dimension and then the label for each level:
```

. collect label levels colname p_skew "Skewness p-value"
> p_kurt "Kurtosis p-value" p_chi2 "Joint p-value", modify

```

To learn more about modifying labels, see [TABLES] collect label.
Let's also change the numeric format of our \(p\)-values. With collect style cell, we can modify all cells in the table, all cells in a particular dimension, or particular cells of a particular dimension. Below, we specify the numeric formatting for only three levels of colname.
```

. collect style cell colname[p_skew p_kurt p_chi2], nformat(%7.3f)

```

Finally, we want to show only the three \(p\)-values in our table. We can use collect layout to specify the statistics we want to include in our final table.
```

. collect layout (rowname) (colname[p_skew p_kurt p_chi2])
Collection: Table
Rows: rowname
Columns: colname[p_skew p_kurt p_chi2]
Table 1: 3 x 3

```
\begin{tabular}{l|rrr}
\hline & Skewness & p-value & Kurtosis \\
& p-value & Joint p-value \\
\hline Height (cm) & 0.000 & 0.000 & 0.000 \\
Weight (kg) & 0.000 & 0.000 & 0.000 \\
Diastolic blood pressure & 0.000 & 0.000 & 0.000 \\
\hline
\end{tabular}

Notably, all \(p\)-values for all tests are very small, so this is not a particularly exciting table. However, our table customizations made it easy to quickly see the results of tests of normality for all our variables.

\section*{Stored results}
table stores the following in \(r()\) :
Scalars
\(r(N)\) number of observations

\section*{Reference}

McDowell, A., A. Engel, J. T. Massey, and K. Maurer. 1981. Plan and operation of the Second National Health and Nutrition Examination Survey, 1976-1980. Vital and Health Statistics 1(15): 1-144.

\section*{Also see}
[R] table - Table of frequencies, summaries, and command results
\([R]\) table intro - Introduction to tables of frequencies, summaries, and command results
[R] table regression - Table of regression results
[TABLES] Intro - Introduction

\section*{Title}
table regression - Table of regression results
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & References \\
Also see & & &
\end{tabular}

\section*{Description}

In this entry, we discuss how to create tables of regression results. These tables can include coefficients, standard errors, confidence intervals, and many more results stored by estimation commands.

\section*{Quick start}

Table of regression coefficients; rows correspond to covariates (colname)
table colname, command (regress y x1 x2 x3)
Table of coefficients and confidence intervals; columns correspond to the statistics (result)
table (colname) (result), command (_r_b _r_ci: regress y x1 x2 x3)
Same as above, but use the labels defined in mylabels.stjson and the styles in mystyle.stjson
```

table (colname) (result), ///
command(_r_b _r_ci: regress y x1 x2 x3) ///
label(mylabels) style(mystyle)

```

\section*{Menu}

Statistics \(>\) Summaries, tables, and tests \(>\) Tables of frequencies, summaries, and command results

\section*{Syntax}
table ([rowspec]) ([colspec]) [(tabspec)][if][in][weight], command (cmdspec) [command(cmdspec) ...] [options]
rowspec, colspec, and tabspec may be empty or may include variable names or any of the following keywords:
\begin{tabular}{ll} 
keyword & Description \\
\hline result & requested statistics \\
stars & stars denoting statistical significance \\
command & index option command () \\
colname & column names for matrix statistics \\
rowname & row names for matrix statistics \\
coleq & column equation names for matrix statistics \\
roweq & row equation names for matrix statistics \\
\hline
\end{tabular}
options
Commands
command (cmdspec)
Formats
nformat (\% fint [results][, basestyle])
sformat (sfint [results])
cidelimiter (char)
cridelimiter (char)

\section*{Stars}
stars(starspec)
Options
missing
name (cname)
append
replace
label(filename)
style(filename [, override])
noisily

Description
collect results from the specified Stata command
specify numeric format
specify string format
use character as delimiter for confidence interval limits use character as delimiter for credible interval limits
add stars to denote statistical significance
treat numeric missing values like other values
collect results into a collection named cname append results to an existing collection
replace results of an existing collection
specify the collection labels
specify the collection style
display output from each command
fweights, aweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
strL variables are not allowed; see [U] 12.4.8 strL.
noisily does not appear in the dialog box.

\section*{Options}

Commands
command (cmdspec) specifies the Stata commands from which to collect results. command() may be repeated to collect results from multiple commands.
cmdspec is [explist: ] command [arguments] [, cmdoptions]
explist specifies which results to collect and report in the table. explist may include result identifiers and named expressions.
result identifiers are results stored in \(r()\) and \(e()\) by the command. For instance, result identifiers could be \(r\) (mean), \(r(C)\), or e (chi2). After estimation commands, result identifiers also include the following:
\begin{tabular}{|c|c|}
\hline Identifier & Result \\
\hline -r_b & coefficients or transformed coefficients reported by command \\
\hline -r_se & standard errors of _r_b \\
\hline -r_z & test statistics for \(\quad\) r_b \\
\hline _r_z_abs & absolute value of \(\_\)r_z \\
\hline -r_p & \(p\)-values for _r_b \\
\hline _r_lb & lower bounds of confidence intervals for _r_b \\
\hline _r_ub & upper bounds of confidence intervals for \(\_\)r_b \\
\hline _r_ci & confidence intervals for _r_b \\
\hline _r_crlb & lower bounds of credible intervals for _r_b \\
\hline _r_crub & upper bounds of credible intervals for _r_b \\
\hline _r_cri & credible intervals for _r_b \\
\hline _r_df & degrees of freedom for \(\mathrm{r}^{\text {r_b }}\) \\
\hline
\end{tabular}
named expressions are specified as name \(=\) exp, where name may be any valid Stata name and exp is an expression, typically an expression that involves one or more result identifiers. An example of a named expression is sd \(=\operatorname{sqrt}(r\) (variance)).
For r-class commands, the default is to include all numeric scalars posted to \(r()\) in the table results. For e-class commands, the default is to include \(\_r \_b\) in the table results.
command is any command that follows standard Stata syntax.
arguments may be anything so long as they do not include an if clause, in range, or weight specification.

Any if or in qualifier and weights should be specified directly with table, not within the command() option. Weights are passed to command only if they are specified.
cmdoptions may be anything supported by command.

Formats
nformat ( \(\%\) fint [results][, basestyle]) changes the numeric format, such as the number of decimal places, for specified results. If results are not specified, the numeric format is changed for all results.
results may be any name in the e() or r() results produced by commands specified in option command ().
This option is repeatable, and when multiple formats apply to one result, the rightmost specification is applied.

This option does not affect the format of numeric layout variables (rowspec, colspec, and tabspec). The default format of these variables is taken from the dataset.
basestyle indicates that the format be applied to results that do not already have their own format instead of overriding the format for all results.
sformat (sfint \([\) results \(]\) ) changes the string format for specified results. You can, for instance, add symbols or text to the values reported in the table by modifying the string format.
sfint may contain a mix of text and \(\%\) s. Here \(\%\) s refers to the numeric value that is formatted as specified using nformat(). The text will be placed around the numeric values in your table as it is placed around \(\% \mathrm{~s}\) in this option. For instance, to place parentheses around the percent statistics, you can specify sformat(" (\%s)" percent).
results may be any name in the e() or r() results produced by commands specified in option command ().

Two text characters must be specified using a special character sequence if you want them to be displayed in your table. To include \(\%\), type \(\% \%\). To include \(\backslash\), type \(\backslash \backslash\). For instance, to place a percent sign following percent statistics, you can specify sformat( \(" \%\) s\%\%" percent).
This option is repeatable, and when multiple formats apply to one result, the rightmost specification is applied.
cidelimiter (char) changes the delimiter between confidence interval limits to char. The default is cidelimiter(" "), that is, two spaces.
cridelimiter (char) changes the delimiter between credible interval limits to char. The default is cridelimiter(" "), that is, two spaces.

\section*{Stars}
stars (starspec) specifies that stars representing statistical significance be included in the table. starspec identifies the result whose values determine significance, which characters should represent each significance level, and where these characters should be displayed in the table. starspec is
starres [\#1 "label1"[\#2 "label2" [\#3 "label3" [\#4 "label4" [\#5 "label5" ] ]] ]]
[, attach(attachres) result dimension starsnoteopts]
starres is the name of the result whose values determine which characters, typically which number of stars, are to be displayed.
labell specifies the characters to be displayed when starres \(<\# 1\).
label2 specifies the characters to be displayed when starres \(<\# 2\).
label3 specifies the characters to be displayed when starres \(<\# 3\).
label4 specifies the characters to be displayed when starres \(<\# 4\).
label5 specifies the characters to be displayed when starres \(<\# 5\).
attach(attachres) specifies the name of the result to which the characters defined by labell, \(\ldots\), label5 are to be attached. If attach() is not specified, a new result named stars is created and is automatically added to the table.
result and dimension control how collect stars adds items when labeling significant results. These options are mutually exclusive.
result specifies the default behavior, and this option is necessary only if the following dimension behavior is in effect and you want to change back to the result behavior.
dimension specifies that dimension stars be added to the collection. Items will be tagged with stars [value], and the labels will be tagged with stars [label]. Use this option for layouts where results are to be stacked within columns, and use new dimension stars in the column specification of the layout.
starsnoteopts control the display and composition of the stars note.
noshownote and shownote control whether to display the stars note.
increasing and decreasing control the order of \(p\)-values in the stars note.
pvname (string) specifies a name for the \(p\)-value in the stars note. The default is pvname ( p ). delimiter (string) specifies the delimiter between labels in the stars note. The default is delimiter (", ").
nformat (\%fmt) specifies the numeric format for the cutoff values in the stars note. The default is nformat ( \(\% 9.0 \mathrm{~g}\) ).
prefix (string) specifies the prefix for the stars note. The prefix is empty by default.
suffix (string) specifies the suffix for the stars note. The suffix is empty by default.
For example, stars(_r_p 0.01 "***" 0.05 "**" 0.1 "*", attach(_r_b)) could be added to a table of regression results to specify that stars be defined based on the \(p\)-values in _r_p and be attached to the reported coefficients (_r_b).

\section*{Options}
missing specifies that numeric missing values of any variables specified in rowspec, colspec, or tabspec be treated as valid categories. By default, observations with a numeric missing value in any of these variables are omitted.
name (cname) specifies that a collection named cname be associated with the collected statistics and results. The default is name (Table).
append specifies that table append its collection information into the collection named in name(). replace permits table to overwrite an existing collection. This option is implied for name (Table) when append is not specified.
label (filename) specifies the filename containing the collection labels to use for your table. Labels in filename will be loaded for the table, and any labels not specified in filename will be taken from the labels defined in \(c\) (collect_label). The default is to use only the collection labels set in c(collect_label); see [TABLES] set collect_label.
style(filename [, override]) specifies the filename containing the collection styles to use for your table. The default collection styles will be discarded, and only the collection styles in filename will be applied.

If you prefer the default collection styles but also want to apply any styles in filename, specify override. If there are conflicts between the default collection styles and those in filename, the ones in filename will take precedence.
The default is to use only the collection styles set in c (table_style); see [TABLES] set table_style.
The following option is available with table but is not shown in the dialog box:
noisily specifies that output from the commands specified in command() options be displayed. By default, output from commands is suppressed.

\section*{Remarks and examples}

Remarks are presented under the following headings:

\section*{Introduction}

Tables with results from a single command
Tables with results from multiple estimation commands
Regression results with factor variables

\section*{Introduction}

The table command allows us to create tables of regression results. You can create a table that reports coefficients, standard errors, test statistics, confidence intervals, and other statistics from a single model or a table that compares results from multiple models.
table does not fit models directly. Rather, table will run any Stata command that you include in its command() option and place results from that command into the table. You determine which results you would like to see in the table. You can select any of the results stored by the command.

You can also create a table of regression results with etable. However, etable will create tables only with active estimation results, results from margins, or results stored with estimates store. If you are working with any of these results, you can use etable to create and export a table of regression results. However, if you want to include results from other commands, you should use the table command.

\section*{Tables with results from a single command}

We have data from the Second National Health and Nutrition Examination Survey (NHANES II) (McDowell et al. 1981). The data contain some demographic information, such as the participants' age. The data also contain some measures of health, including the individual's weight, systolic blood pressure (bpsystol), and whether the individual has diabetes.

Here we will create a table with results from a linear regression model for systolic blood pressure as a function of age and weight. We type the command to fit the model in the command() option. In the first set of parentheses following table, we specify that we want the rows to correspond to the levels of colname-this is how we refer to the list of covariates in our regression model. In the second set of parentheses, we specify that we want the columns to correspond to the statistics (result).
```

. use https://www.stata-press.com/data/r18/nhanes2l
(Second National Health and Nutrition Examination Survey)
. table (colname) (result), command(regress bpsystol age weight)

```
\begin{tabular}{l|r}
\hline & Coefficient \\
\hline Age (years) & .6379892 \\
Weight (kg) & .4069041 \\
Intercept & 71.27096 \\
\hline
\end{tabular}

Our table is fairly simple. By default, table includes only the reported coefficients when an estimation command is specified in the command() option.

The table command can easily be used to compare results across groups in our data. For instance, if we want to fit the same model for males and females, we can add sex to our column specification.
. table (colname) (sex result), command(regress bpsystol age weight)
\begin{tabular}{l|rrr}
\hline & Male & \begin{tabular}{c} 
Sex \\
Female
\end{tabular} & Total \\
\hline Age (years) & .4789361 & .7735499 & .6379892 \\
Weight (kg) & .3346106 & .4586108 & .4069041 \\
Intercept & 84.08037 & 61.70456 & 71.27096 \\
\hline
\end{tabular}

We can now easily compare results for males, females, and both together.
We may want to see additional statistics reported. Let's extend our table to include both coefficients and standard errors. We can refer to the reported coefficients using the keyword _r_b and to the reported standard errors as _r_se, and we can list these in the command () option before our regression command. Here we also move result to the first set of parentheses so that coefficients and standard errors will be displayed on separate rows.
\begin{tabular}{l}
. table (colname result) (sex), \\
> command(_r_b _r_se: regress bpsystol age weight) \\
\hline
\end{tabular}

We now have the statistics we want in this table, but we may want to modify the look a bit. table allows us to customize the results in our table in a number of ways. We can use the nformat() option to report all results to two decimal places, and we can use the sformat() option to place parentheses around our standard errors.
```

. table (colname result) (sex),
> command(_r_b _r_se: regress bpsystol age weight)
> nformat(%6.2f) sformat("(%s)" _r_se)

```
\begin{tabular}{l|rrr}
\hline & Male & \begin{tabular}{c} 
Sex \\
Female
\end{tabular} & Total \\
\hline \begin{tabular}{c} 
Age (years) \\
Coefficient
\end{tabular} & 0.48 & 0.77 & 0.64 \\
\begin{tabular}{c} 
Std. error
\end{tabular} & \((0.02)\) & \((0.02)\) & \((0.01)\) \\
\begin{tabular}{l} 
Weight (kg) \\
Coefficient
\end{tabular} & 0.33 & 0.46 & 0.41 \\
\begin{tabular}{c} 
Std. error
\end{tabular} & \((0.02)\) & \((0.02)\) & \((0.01)\) \\
\begin{tabular}{l} 
Intercept \\
Coefficient \\
Std. error
\end{tabular} & 84.08 & 61.70 & 71.27 \\
\hline
\end{tabular}

Now that we have the parentheses to distinguish standard errors from coefficients, we may not want to see those labels in the row header. We add the style (table-reg3) option to use the predefined style table-reg3, which hides the names of these statistics, right-aligns the names of the variables in the row headers, center aligns the statistics horizontally within each column, and adds vertical space between variables.
```

. table (colname result) (sex),
> command(_r_b _r_se: regress bpsystol age weight)
> nformat(%05.3f) sformat("(%s)" _r_se) style(table-reg3)

```
\begin{tabular}{c|ccc}
\hline & Male & \begin{tabular}{c} 
Sex \\
Female
\end{tabular} & Total \\
\hline Age (years) & 0.479 & 0.774 & 0.638 \\
& \((0.016)\) & \((0.016)\) & \((0.011)\) \\
Weight (kg) & 0.335 & 0.459 & 0.407 \\
& \((0.020)\) & \((0.018)\) & \((0.012)\) \\
Intercept & 84.080 & 61.705 & 71.271 \\
& \((1.749)\) & \((1.376)\) & \((1.042)\) \\
\hline
\end{tabular}

\section*{Tables with results from multiple estimation commands}

Above, we fit the same model to the full dataset and then to groups of observations within that dataset. We may alternatively want to fit different models and display their results in a single table. To do this, we specify multiple command() options.
```

. table (colname result) (command),
> command(_r_b _r_se: regress bpsystol age weight)
> command(_r_b _r_se: regress bpsystol age weight iron vitaminc zinc)
> nformat(%6.2f) sformat("(%s)" _r_se) style(table-reg3)

```
\begin{tabular}{r|rr}
\hline & \multicolumn{1}{|c}{1} & 2 \\
\hline Age (years) & 0.64 & 0.64 \\
& \((0.01)\) & \((0.01)\) \\
Weight (kg) & 0.41 & 0.40 \\
& \((0.01)\) & \((0.01)\) \\
Serum iron (mcg/dL) & & -0.01 \\
& & \((0.01)\) \\
Serum vitamin C (mg/dL) & & -0.79 \\
& & \((0.36)\) \\
Serum zinc (mcg/dL) & & -0.05 \\
& & \((0.01)\) \\
Intercept & 71.27 & 77.50 \\
& \((1.04)\) & \((1.75)\) \\
\hline
\end{tabular}

We may want to modify this table a bit further. Customization of tables can go beyond the predefined styles and options available to you in the table command. table creates a collection of results that can be used in combination with the collect suite of commands to produce highly customized tables.

If we want to add more descriptive labels for the two models, we can use the collect label levels command to define our new labels. After a change using collect, we can type collect preview to see the results.
```

. collect label levels command 1 "Model 1" 2 "Model 2", modify
. collect style header command, level(label)
. collect preview

```
\begin{tabular}{r|cc}
\hline & Model 1 & Model 2 \\
\hline Age (years) & 0.64 & 0.64 \\
& \((0.01)\) & \((0.01)\) \\
Weight (kg) & 0.41 & 0.40 \\
& \((0.01)\) & \((0.01)\) \\
Serum iron (mcg/dL) & & -0.01 \\
& & \((0.01)\) \\
Serum vitamin C (mg/dL) & & -0.79 \\
& & \((0.36)\) \\
Serum zinc (mcg/dL) & & \((0.05\) \\
& & 71.27 \\
Intercept & \((1.04)\) & \((1.75)\) \\
& & \\
\hline
\end{tabular}

\section*{Regression results with factor variables}

The examples above included only continuous covariates in the models. When we include factor variables, there are a variety of ways that they can be displayed in the headers of the tables. In [TABLES] Predefined styles, you will find a number of styles that you can choose from. We demonstrate a few here.

We will start with the table-reg1 style. This style is our default table style, except that it identifies the commands in the headers using values \(1,2, \ldots\) rather than labeling them with the full command we typed in the command() option.
```

. table (colname) (command result),
> command(regress bpsystol i.agegrp i.sex weight)
> command(regress bpsystol i.agegrp\#\#i.sex weight)
> style(table-reg1)

```
\begin{tabular}{l|rr}
\hline & 1 & 2 \\
\hline Age group=20-29 & 0 & 0 \\
Age group=30-39 & 1.195226 & -.7808968 \\
Age group=40-49 & 7.251555 & 2.749774 \\
Age group=50-59 & 15.94216 & 10.43724 \\
Age group=60-69 & 22.83932 & 16.53001 \\
Age group=70+ & 30.46609 & 23.3076 \\
Sex=Male & 0 & 0 \\
Sex=Female & 1.040833 & -6.777535 \\
Weight (kg) & .4359741 & .4242392 \\
Age group=20-29 \# Sex=Male & 0 \\
Age group=20-29 \# Sex=Female & & 0 \\
Age group=30-39 \# Sex=Male & & 0 \\
Age group=30-39 \# Sex=Female & & 3.942553 \\
Age group=40-49 \# Sex=Male & 0 \\
Age group=40-49 \# Sex=Female & & 8.79336 \\
Age group=50-59 \# Sex=Male & & 0 \\
Age group=50-59 \# Sex=Female & & 10.6501 \\
Age group=60-69 \# Sex=Male & & 0 \\
Age group=60-69 \# Sex=Female & & 12.20669 \\
Age group=70+ \# Sex=Male & & 0 \\
Age group=70+ \# Sex=Female & & 13.51823 \\
Intercept & 86.71019 & 91.57774 \\
\hline
\end{tabular}

In some cases, for clarity, it is helpful to see both the factor variables and their levels. The table-reg1 style provides this in the output.

When we have nice value labels on our factor variables, we may want to see only those. The table-reg1-fv1 style removes the extra labels. Our table above also reports zero-valued coefficients for base categories in both the main effects of the factor variables and in their interactions. The table-reg1-fv1 style omits the rows for the base categories in the interactions.
```

. table (colname) (command result),
> command(regress bpsystol i.agegrp i.sex weight)
> command(regress bpsystol i.agegrp\#\#i.sex weight)
> style(table-reg1-fv1)

```
\begin{tabular}{l|rr}
\hline & 1 & 2 \\
\hline \(20-29\) & 0 & 0 \\
\(30-39\) & 1.195226 & -.7808968 \\
\(40-49\) & 7.251555 & 2.749774 \\
\(50-59\) & 15.94216 & 10.43724 \\
\(60-69\) & 22.83932 & 16.53001 \\
\(70+\) & 30.46609 & 23.3076 \\
Male & 0 & 0 \\
Female (kg) & 1.040833 & -6.777535 \\
Weight (kg) & .4359741 & .4242392 \\
30-39 \# Female & & 3.942553 \\
40-49 \# Female & & 8.79336 \\
50-59 \# Female & & 10.6501 \\
60-69 \# Female & & 12.20669 \\
70+ \# Female & 86.71019 & 13.51823 \\
Intercept & & 91.57774 \\
\hline
\end{tabular}

Sometimes, the tables are more readable when the row headers are right aligned. We can use the table-reg2-fv1 style in this case. Let's also change the numeric format of all the results so that they report only two decimal places.
```

. table (colname) (command result),
> command(regress bpsystol i.agegrp i.sex weight)
> command(regress bpsystol i.agegrp\#\#i.sex weight)
> style(table-reg2-fv1) nformat(%6.2f)

```
\begin{tabular}{r|rr}
\hline & 1 & 2 \\
\hline \(20-29\) & 0.00 & 0.00 \\
\(30-39\) & 1.20 & -0.78 \\
\(40-49\) & 7.25 & 2.75 \\
\(50-59\) & 15.94 & 10.44 \\
\(60-69\) & 22.84 & 16.53 \\
\(70+\) & 30.47 & 23.31 \\
Male & 0.00 & 0.00 \\
Female & 1.04 & -6.78 \\
Weight (kg) & 0.44 & 0.42 \\
30-39 \# Female & & 3.94 \\
40-49 \# Female & & 8.79 \\
50-59 \# Female & & 10.65 \\
60-69 \# Female & & 12.21 \\
\(70+\) \# Female & & 13.52 \\
Intercept & 86.71 & 91.58 \\
\hline
\end{tabular}

There are many ways that we can further customize our table using the collect suite of commands. We can add column titles for our models as we did above. In addition, we can use collect style row to specify a character to be used between terms in an interaction.
```

. collect label levels command 1 "Model 1" 2 "Model 2", modify
. collect style header command, level(label)
. collect style row stack, delimiter(" X ")
. collect preview

```
\begin{tabular}{r|rr}
\hline & Model 1 & Model 2 \\
\hline \(20-29\) & 0.00 & 0.00 \\
\(30-39\) & 1.20 & -0.78 \\
\(40-49\) & 7.25 & 2.75 \\
\(50-59\) & 15.94 & 10.44 \\
\(60-69\) & 22.84 & 16.53 \\
\(70+\) & 30.47 & 23.31 \\
Male & 0.00 & 0.00 \\
Female & 1.04 & -6.78 \\
Weight (kg) & 0.44 & 0.42 \\
30-39 X Female & & 3.94 \\
40-49 X Female & & 8.79 \\
50-59 X Female & & 10.65 \\
60-69 X Female & & 12.21 \\
\(70+\) X Female & & 13.52 \\
Intercept & 86.71 & 91.58 \\
\hline
\end{tabular}

If one of the predefined styles in [TABLES] Predefined styles does not suit your needs for factorvariable results (or for any other table customization), you can create your own style. To do this, you will use series of collect style commands, and then you can save the style to use later; see [TABLES] collect style save.

If you wish to include your table in a paper, on a webpage, or in another format, you can easily export it in \(\mathrm{LAT}_{\mathrm{E}} \mathrm{X}\), Word, Excel, HTML, and a variety of other formats by using collect export.

\section*{Stored results}
table stores the following in \(r()\) :
Scalars \(r(N)\) number of observations

\section*{References}

Huber, C. 2021a. Customizable tables in Stata 17, part 5: Tables for one regression model. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2021/08/26/customizable-tables-in-stata-17-part-5-tables-for-one-regression-model/.

2021b. Customizable tables in Stata 17, part 6: Tables for multiple regression models. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2021/09/02/customizable-tables-in-stata-17-part-6-tables-for-multiple-regression-models/.

McDowell, A., A. Engel, J. T. Massey, and K. Maurer. 1981. Plan and operation of the Second National Health and Nutrition Examination Survey, 1976-1980. Vital and Health Statistics 1(15): 1-144.

\section*{Also see}
[R] table - Table of frequencies, summaries, and command results
[R] table hypothesis tests - Table of hypothesis tests
\([R]\) table intro - Introduction to tables of frequencies, summaries, and command results
[R] etable - Create a table of estimation results
[TABLES] Intro - Introduction

\section*{Title}
table - Table of frequencies, summaries, and command results
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
Also see & & &
\end{tabular}

\section*{Description}
table is a flexible command for creating tables of many types-tabulations, tables of summary statistics, tables of regression results, and more. table can calculate summary statistics to display in the table. table can also include results from other Stata commands.

\section*{Quick start}

Two-way tabulation of a1 and a2
```

table a1 a2

```

Table of means for v 1 and v 2 across the levels of a1
table a1, statistic(mean v1 v2)
Two-way table with a1 defining rows and a2 defining columns, with frequencies and pairwise correlation coefficients between v3 and v4 computed for every cell
table a1 a2, command(pwcorr v3 v4)
Table of regression coefficients with means of the covariates; rows correspond to covariates and columns correspond to the statistics
```

table (colname) (statcmd result),
command(regress y x1 x2)
statistic(mean x1 x2)

```

Same as above, and include standard deviations for the covariates
```

table (colname) (statcmd result),
command(regress y x1 x2)
statistic(mean x1 x2)
statistic(sd x1 x2)

```

\section*{Menu}

Statistics \(>\) Summaries, tables, and tests \(>\) Tables of frequencies, summaries, and command results

\section*{Syntax}

Basic syntax for a one-way table
table rowvar
table () colvar

Basic syntax for a two-way table
table rowvar colvar

Basic syntax for an \(n\)-way table
table rowvars colvar
table rowvar (colvars)
table (rowvars) (colvars)

Basic syntax for multiple \(n\)-way tables
table (rowvars) (colvars) (tabvars)

Full syntax
table (rowspec) (colspec) [(tabspec)] [if] [in] [weight] [, options ]
rowspec, colspec, and tabspec may be empty or may include variable names or any of the following keywords:
\begin{tabular}{ll} 
keyword & Description \\
\hline result & requested statistics \\
stars & stars denoting statistical significance \\
var & variables from statistic() option \\
across & index across() specifications \\
colname & column names for matrix statistics \\
rowname & row names for matrix statistics \\
coleq & column equation names for matrix statistics \\
roweq & row equation names for matrix statistics \\
command & index option command() \\
statcmd & index options statistic() and command() \\
\hline
\end{tabular}
options
Main
totals (totals)
nototals
Statistics
statistic (statspec)

\section*{Commands}
command (cmdspec)

\section*{Formats}
nformat (\% fmt \([\) results \(][\), basestyle \(]\) )
sformat (sfmt [results])
cidelimiter (char)
cridelimiter (char)

\section*{Stars}
stars (starspec)
Options
listwise
missing
showcounts
zerocounts
name (cname)
append
replace
label (filename)
style(filename [, override])
markvar (newvar)
noisily

Description
report only the specified totals
suppress the marginal totals
statistic to be reported; default is
statistic (frequency) when no weights
are specified and statistic (sumw) otherwise
collect results from the specified Stata command
specify numeric format
specify string format
use character as delimiter for confidence interval limits use character as delimiter for credible interval limits
add stars to denote statistical significance
use listwise deletion to handle missing values treat numeric missing values like other values show sample size for all variables in statistic () option
report 0 for empty cell counts
collect results into a collection named cname append results to an existing collection
replace results of an existing collection
specify the collection labels
specify the collection style
create newvar that identifies observations used in the tabulation
display output from each command
fweights, aweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
strL variables are not allowed; see [U] 12.4.8 strL.
markvar() and noisily do not appear in the dialog box.

\section*{Options}
totals (totals) and nototals control which totals are to be displayed in the table. By default, all totals are reported.
totals (totals) specifies which margin totals to display in the reported table totals can contain variables in rowspec, colspec, tabspec, and their interaction. Interactions can be specified by using the \# operator.
nototals prevents table from displaying any totals.

\section*{Statistics}
statistic (statspec) specifies the statistic to be displayed. Frequency statistics, summary statistics, and ratio statistics are available by specifying statistic (freqstat), statistic (sumstat varlist), and statistic (ratiostat [varlist] [, ratio_options]), respectively.
statistic() may be repeated to request multiple statistics.
statistic (freqstat) specifies that frequencies be computed.
\begin{tabular}{ll} 
freqstat & Definition \\
\hline \begin{tabular}{ll} 
frequency & frequency \\
sumw & sum of weights \\
\hline
\end{tabular} \\
\hline
\end{tabular}
statistic (sumstat varlist) specifies that summary statistic sumstat be computed for the variables in varlist.
\begin{tabular}{ll} 
sumstat & Definition \\
\hline mean & mean \\
semean & \begin{tabular}{l} 
standard error of the mean \\
sebinomial \\
sepoisson
\end{tabular} \\
\begin{tabular}{l} 
standard error of the mean, binomial \\
standard error of the mean, Poisson
\end{tabular} \\
sd & \begin{tabular}{l} 
variance \\
skewness \\
surtosis
\end{tabular} \\
\begin{tabular}{l} 
standard deviation \\
skewness \\
kurtosis \\
svycv
\end{tabular} & \begin{tabular}{l} 
coefficient of variation \\
coefficient of variation (svy)
\end{tabular} \\
geomean & \begin{tabular}{l} 
geometric mean \\
geosd
\end{tabular} \\
geometric standard deviation
\end{tabular}
statistic (ratiostat [varlist] [, ratio_options]) specifies that ratio statistic ratiostat be computed. If varlist is specified, ratios are computed based on the totals of the specified variables. If varlist is not specified, ratios are computed based on frequencies.
\(\left.\begin{array}{ll}\hline \text { ratiostat } & \text { Definition } \\
\hline \begin{array}{l}\text { proportion } \\
\text { percent }\end{array} & \begin{array}{l}\text { proportion } \\
\text { percentage } \\
\text { rawproportion } \\
\text { rawpercent }\end{array}\end{array} \begin{array}{l}\text { proportion ignoring optionally specified weights } \\
\text { percentage ignoring optionally specified weights }\end{array}\right]\)\begin{tabular}{ll} 
ratio_options & Definition \\
\hline across (cellspec) & \begin{tabular}{l} 
percentages or proportions across levels of \\
variables or interactions \\
compute overall percentages or proportions
\end{tabular} \\
\hline
\end{tabular}
cellspec may contain rowvars, colvars, tabvars, or an interaction between any of these variables. Interactions can be specified by using the \# operator.

\section*{Commands}
command (cmdspec) specifies the Stata commands from which to collect results. command () may be repeated to collect results from multiple commands.
cmdspec is [explist: ] command [arguments] [, cmdoptions]
explist specifies which results to collect and report in the table. explist may include result identifiers and named expressions.
result identifiers are results stored in r() and e() by the command. For instance, result identifiers could be r (mean), \(\mathrm{r}(\mathrm{C})\), or e (chi2). After estimation commands, result identifiers also include the following:
\begin{tabular}{|c|c|}
\hline Identifier & Result \\
\hline -r_b & coefficients or transformed coefficients reported by command \\
\hline -r_se & standard errors of \(\_\)r_b \\
\hline -r_z & test statistics for \(\quad\) r_b \\
\hline -r_z_abs & absolute value of _r_z \\
\hline -r_p & \(p\)-values for _r_b \\
\hline _r_lb & lower bounds of confidence intervals for _r_b \\
\hline _r_ub & upper bounds of confidence intervals for _r_b \\
\hline _r_ci & confidence intervals for _r_b \\
\hline _r_crlb & lower bounds of credible intervals for _r_b \\
\hline _r_crub & upper bounds of credible intervals for _r_b \\
\hline _r_cri & credible intervals for _r_b \\
\hline -r_df & degrees of freedom for \(\mathrm{r}^{\mathrm{r}}\)-b \\
\hline
\end{tabular}
named expressions are specified as name \(=\exp\), where name may be any valid Stata name and exp is an expression, typically an expression that involves one or more result identifiers. An example of a named expression is sd \(=\operatorname{sqrt}(r\) (variance)).

For r-class commands, the default is to include all numeric scalars posted to \(r()\) in the table results. For e-class commands, the default is to include \(\_r \_b\) in the table results.
command is any command that follows standard Stata syntax.
arguments may be anything so long as they do not include an if clause, in range, or weight specification.

Any if or in qualifier and weights should be specified directly with table, not within the command() option.
cmdoptions may be anything supported by command.

Formats
nformat ( \(\%\) fint [results][, basestyle]) changes the numeric format, such as the number of decimal places, for specified results. If results are not specified, the numeric format is changed for all results.
results may be any statistic named in option statistic() (that is, any freqstat, sumstat, or ratiostat) or may be any name in the e() or \(\mathrm{r}(\) ) results produced by commands specified in option command ().
This option is repeatable, and when multiple formats apply to one result, the rightmost specification is applied.

This option does not affect the format of numeric layout variables (rowspec, colspec, and tabspec) or the format of factor variables specified in the statistic () option. The default format of these variables is taken from the dataset.
basestyle indicates that the format be applied to results that do not already have their own format instead of overriding the format for all results.
sformat (sfmt [results]) changes the string format for specified results. You can, for instance, add symbols or text to the values reported in the table by modifying the string format.
sfint may contain a mix of text and \(\%\) s. Here \(\%\) s refers to the numeric value that is formatted as specified using nformat(). The text will be placed around the numeric values in your table as it is placed around \(\%\) s in this option. For instance, to place parentheses around the percent statistics, you can specify sformat(" (\%s)" percent).
results may be any statistic named in option statistic() (that is, any freqstat, sumstat, or ratiostat) or may be any name in the e() or \(\mathrm{r}(\) ) results produced by commands specified in option command ().

Two text characters must be specified using a special character sequence if you want them to be displayed in your table. To include \(\%\), type \(\% \%\). To include \(\backslash\), type \(\backslash \backslash\). For instance, to place a percent sign following percent statistics, you can specify sformat("\%s\%\%" percent).
This option is repeatable, and when multiple formats apply to one result, the rightmost specification is applied.
cidelimiter (char) changes the delimiter between confidence interval limits to char. The default is cidelimiter (" "), that is, two spaces.
cridelimiter (char) changes the delimiter between credible interval limits to char. The default is cridelimiter (" "), that is, two spaces.
stars (starspec) specifies that stars representing statistical significance be included in the table. starspec identifies the result whose values determine significance, which characters should represent each significance level, and where these characters should be displayed in the table. starspec is

\section*{starres [\#1 "label1" [\#2 "label2" [\#3 "label3"[\#4 "label4"[\#5 "label5" ] ] ]] ]}
[, attach(attachres) result dimension starsnoteopts]
starres is the name of the result whose values determine which characters, typically which number of stars, are to be displayed.
labell specifies the characters to be displayed when starres \(<\# 1\).
label2 specifies the characters to be displayed when starres \(<\# 2\).
label3 specifies the characters to be displayed when starres \(<\# 3\).
label4 specifies the characters to be displayed when starres \(<\# 4\).
label5 specifies the characters to be displayed when starres \(<\# 5\).
attach(attachres) specifies the name of the result to which the characters defined by labell, \(\ldots\). label5 are to be attached. If attach() is not specified, a new result named stars is created and is automatically added to the table.
result and dimension control how collect stars adds items when labeling significant results. These options are mutually exclusive.
result specifies the default behavior, and this option is necessary only if the following dimension behavior is in effect and you want to change back to the result behavior.
dimension specifies that dimension stars be added to the collection. Items will be tagged with stars [value], and the labels will be tagged with stars [label]. Use this option for layouts where results are to be stacked within columns, and use new dimension stars in the column specification of the layout.
starsnoteopts control the display and composition of the stars note.
noshownote and shownote control whether to display the stars note.
increasing and decreasing control the order of \(p\)-values in the stars note.
pvname (string) specifies a name for the \(p\)-value in the stars note. The default is pvname ( p ). delimiter(string) specifies the delimiter between labels in the stars note. The default is delimiter(", ").
nformat ( \(\%\) fint) specifies the numeric format for the cutoff values in the stars note. The default is nformat \((\% 9.0 \mathrm{~g})\).
prefix(string) specifies the prefix for the stars note. The prefix is empty by default.
suffix(string) specifies the suffix for the stars note. The suffix is empty by default.
For example, stars(_r_p 0.01 "***" 0.05 "**" 0.1 "*", attach(_r_b)) could be added to a table of regression results to specify that stars be defined based on the \(p\)-values in \(\_r \_p\) and be attached to the reported coefficients (_r_b).
listwise handles missing values through listwise deletion, meaning that the entire observation is omitted from the sample if any variable specified in a statistic() option is missing for that observation. By default, table will omit an observation only if all variables specified in all statistic() options are missing for that observation.
missing specifies that numeric missing values of any variables specified in rowspec, colspec, or tabspec be treated as valid categories. By default, observations with a numeric missing value in any of these variables are omitted.

This option does not apply to factor variables specified with statistics fvfrequency, fvrawfrequency, fvproportion, fvrawproportion, fvpercent, or fvrawpercent.
showcounts specifies that table report the sample size for each variable specified in option statistic().
zerocounts specifies that table report a 0 in empty cells for results count, frequency, fvfrequency, and fvrawfrequency.
name (cname) specifies that a collection named cname be associated with the collected statistics and results. The default is name (Table).
append specifies that table append its collection information into the collection named in name().
replace permits table to overwrite an existing collection. This option is implied for name (Table) when append is not specified.
label (filename) specifies the filename containing the collection labels to use for your table. Labels in filename will be loaded for the table, and any labels not specified in filename will be taken from the labels defined in \(c\) (collect_label). The default is to use only the collection labels set in c(collect_label); see [TABLES] set collect_label.
style(filename [, override]) specifies the filename containing the collection styles to use for your table. The default collection styles will be discarded, and only the collection styles in filename will be applied.
If you prefer the default collection styles but also want to apply any styles in filename, specify override. If there are conflicts between the default collection styles and those in filename, the ones in filename will take precedence.

The default is to use only the collection styles set in c (table_style); see [TABLES] set table_style.
The following options are available with table but are not shown in the dialog box:
markvar (newvar) generates an indicator variable that identifies the observations used in the tabulation. noisily specifies that output from the commands specified in command() options be displayed. By default, output from commands is suppressed.

\section*{Remarks and examples}

Remarks are presented under the following headings:

\section*{Introduction}

Specifying the table layout
Advanced table customization

\section*{Introduction}

The table command can create many customized tables, ranging from simple one-way tabulations to multiple \(n\)-way tables with summary statistics and estimation results. table can compute and report frequencies, proportions, percentiles, and other summary statistics. It can also run other Stata commands and include their results in the table. This means you can combine the summary statistics computed by table with test statistics, correlations, regression coefficients, and other results collected from Stata commands. In addition to building tables with the desired statistics, you can customize them by formatting the values in the table and applying predefined styles and labels that affect how the row headers, column headers, and values are displayed in the table.
table can accommodate a variety of layouts. You can define the rows, columns, and even separate tables by levels of categorical variables, statistics, or Stata commands.

If your goal is simply to create a table of estimation results or descriptive statistics, you can use the etable and dtable commands, respectively. These commands allow you to create these tables and export them to a variety of file types in a single step. However, unlike table, these commands create tables with a predefined layout. Therefore, if you want control over the layout or you wish to include a combination of summary statistics, estimation results, and results from other Stata commands, you should use the table command.

In the following entries, we provide simplified syntax, examples, and discussion for specialized types of tables that can be created using table. If you are interested in creating one of these types of tables, we suggest reading the corresponding entry.
[R] table oneway
\([R]\) table twoway
[R] table multiway
[R] table summary
[R] table hypothesis tests
[R] table regression

One-way tabulation
Two-way tabulation
Multiway tables
Table of summary statistics
Table of hypothesis tests
Table of regression results

All the concepts demonstrated in the entries above can be combined to create tables including combinations of tabulations, summary statistics, hypothesis tests, and regression results.

In this entry, we provide additional information on specifying the table layout and which portions of the layout table will automate for you. In addition, we provide resources for customizing the table and exporting the results to your preferred format.

\section*{Specifying the table layout}

A table's layout is determined by our row, column, and table dimension specifications. For example, we specify variable names to define the rows and place statistics in the columns, or vice versa. Because we can include so many different statistics, we can specify keywords that we use to identify the results we have collected from commands and the statistics that table has calculated.

The syntax for specifying the table layout is
table ([rowspec ]) ([ colspec \(]\) ) ([tabspec \(]\) )
We refer to rowspec, colspec, and tabspec collectively as the "layout". For some tables, keywords are required in the layout to uniquely identify the values that we want to include in our table. If you omit a necessary keyword from the layout, table will fill one in for you.

The rules determining whether a keyword is necessary to uniquely identify values in the table are as follows:
1. If more than one statistic is specified, then result is needed in the layout.
2. If more than one variable is specified in option statistic() and option command() is not specified, then var is needed in the layout.
3. If more than one across() specification is used for ratio statistics, then across is needed in the layout.
4. If option command() is specified, then colname is needed in the layout. If, in addition, more than one variable is specified in option statistic(), then colname is needed instead of var, which was required in 2 .
5. If multiple command() options are specified and option statistic() is not specified, then command is needed in the layout.
6. If both options command() and statistic() are specified, then statcmd is needed in the layout.

If we do not directly specify a necessary keyword in one of rowspec, colspec, or tabspec, the missing keywords will be automatically added to the layout as follows:
1. If the row specification is empty, then put the missing keywords in rowspec.
2. If the row specification is not empty but the column specification is empty, then put the missing keywords in colspec.
3. If the row and column specifications are not empty but the table specification is empty and if result is the only missing keyword and there is only one statistic (result), then put result in tabspec.
4. Otherwise, append the missing keywords to rowvars.

Below, we demonstrate how missing keywords are added to the layout.
Using auto.dta, we create a table with the minimum and maximum mpg for each level of rep78. The keyword result identifies the statistics we computed. By listing an empty set of parentheses followed by rep78, we request that the levels of rep78 be placed on the columns.
```

. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. table () rep78, statistic(min mpg) statistic(max mpg)

```
\begin{tabular}{l|rrrrrr}
\hline & \multicolumn{6}{|c}{ Repair record 1978} \\
& 1 & 2 & 3 & 4 & 5 & Total \\
\hline Minimum value & 18 & 14 & 12 & 14 & 17 & 12 \\
Maximum value & 24 & 24 & 29 & 30 & 41 & 41 \\
\hline
\end{tabular}

Based on rule 1, if we request more than one statistic, result must be in the layout. Based on situation 1, if the row specification is empty, then the missing keyword will be placed in the row specification. We could have created the same table by typing
```

. table (result) (rep78), statistic(min mpg) statistic(max mpg)

```

Now, let's include multiple variables in our statistic() option. We also type rep78 immediately after table to specify that the levels of rep78 be placed on the rows.
. table rep78, statistic(mean mpg price)
\begin{tabular}{l|rr}
\hline & Mileage (mpg) & Price \\
\hline Repair record 1978 & & \\
1 & 21 & 4564.5 \\
2 & 19.125 & 5967.625 \\
3 & 19.43333 & 6429.233 \\
4 & 21.66667 & 6071.5 \\
5 & 27.36364 & 5913 \\
Total & 21.28986 & 6146.043 \\
\hline
\end{tabular}

Because we have more than one variable in the statistic() option, then keyword var must be in the layout (rule 2). If we include a row specification but leave the column specification empty, table will treat var as the column identifier. We could have equivalently typed
```

. table (rep78) (var), statistic(mean mpg price)

```

Next, let's include both a command() option and a statistic() option with multiple variables in the same table. We want a table with coefficients and means of the independent variables. We use the command() option to fit the regression and obtain the means with the statistic() option. Now, we need both colname and statcmd to uniquely identify the values in the table. Let's omit statcmd from our command.
```

. table (colname) (result[_r_b mean]),
> command(regress mpg turn trunk) statistic(mean turn trunk)

```
\begin{tabular}{l|cr}
\hline & Coefficient & Mean \\
\hline \begin{tabular}{c} 
Turn circle (ft.) \\
regress mpg turn trunk \\
Mean
\end{tabular} & -.7610113 & \\
\begin{tabular}{c} 
Trunk space (cu. ft.) \\
regress mpg turn trunk \\
Mean
\end{tabular} & -.3161825 & 39.64865 \\
\begin{tabular}{l} 
Intercept \\
regress mpg turn trunk
\end{tabular} & 55.82001 & 13.75676 \\
\hline
\end{tabular}

But based on situation 4, table will add statcmd to the row specification if we leave it out. So we could have also typed the following to create the same table:
```

table (colname statcmd) (result[_r_b mean]),
command(regress mpg turn trunk) statistic(mean turn trunk)

```

This table displays each of the statistics that we requested. If we simply wanted to compute some statistics quickly, it has served its purpose. However, if we wish to share these results with others or include a table in a report, we will want to make some modifications.

\section*{Advanced table customization}
table allows you to customize the results of your table using the stars(), nformat(), sformat(), cidelimiter(), label(), and style() options. With these, you can add significance stars, change the numeric format, and attach characters such as percent signs or parentheses to values in the table, use a stored set of labels, or use a predefined style. See [TABLES] Predefined styles for more information on selecting a style that adjusts elements of the table such as row header alignment, alignment of values within the cells, and which labels are included in the headers.

Customization can also go beyond the predefined styles and options available to you in the table command. table stores all of its results in a collection named Table. This means that you can use the specialized tools available in the collect suite of commands to further customize your table. With collect, you can modify specific labels, add borders, change the style of the headers, and the like. Once you have a publication-ready table, you can use collect export to export your table to HTML, Word, \(\mathrm{LAT}_{\mathrm{E}} \mathrm{X}\), PDF, Excel, or another format appropriate for your report.

\section*{Stored results}
table stores the following in \(r()\) :
Scalars
\(r(N)\) number of observations

\section*{Methods and formulas}

Variables specified in rowspec, colspec, and tabspec identify groups of observations within the dataset. These groups are represented in the table by cells and cell margins (totals). For a given cell or cell margin, let \(n\) denote the number of observations (frequency). Let \(x\) denote the variable on which we want to calculate summary statistics, and let \(x_{i}, i=1, \ldots, n\), denote an individual observation on \(x\). count is the number of nonmissing values of \(x\). first is \(x_{1}\) and last is \(x_{n}\). Let \(a\) be the smallest \(i\) such that \(x_{i}\) is not missing, and then firstnm is \(x_{a}\). Let \(b\) be the largest \(i\) such that \(x_{i}\) is not missing, and then lastnm is \(x_{b}\).

Let \(v_{i}\) be the weight, and if no weight is specified, define \(v_{i}=1\) for all \(i\). Let \(v\). denote the sum of the weights (sumw):
\[
v .=\sum_{i=1}^{n} v_{i}
\]

When aweights or pweights are specified, the normalized weights are given by \(w_{i}=v_{i}(n / v\). with \(w .=n\); otherwise, \(w_{i}=v_{i}\) and \(w .=v_{\text {. }}\).

The remaining summary statistics are computed according to the following formulas:
total
\[
x .= \begin{cases}\sum_{i=1}^{n} v_{i} x_{i} & \text { if pweights } \\ \sum_{i=1}^{n} w_{i} x_{i} & \text { otherwise }\end{cases}
\]
rawtotal
\[
\sum_{i=1}^{n} x_{i}
\]
mean
\[
\bar{x}=\frac{1}{w} \sum_{i=1}^{n} w_{i} x_{i}
\]

Define \(m_{r}\) as the \(r\) th moment about the mean:
\[
m_{r}=\frac{1}{w .} \sum_{i=1}^{n} w_{i}\left(x_{i}-\bar{x}\right)^{r}
\]
variance
\[
s^{2}=\frac{w .}{w .-1} m_{2}=\frac{1}{w .-1} \sum_{i=1}^{n} w_{i}\left(x_{i}-\bar{x}\right)^{2}
\]
sd (standard deviation)
\[
s=\sqrt{s^{2}}
\]
semean (standard error of the mean)
\[
\operatorname{se}(\bar{x})=\frac{s}{\sqrt{w}}
\]
sebinomial (standard error of the mean, binomial distribution)
\[
\sqrt{\frac{\bar{x}(1-\bar{x})}{w}}
\]
sepoisson (standard error of the mean, Poisson distribution)
\[
\sqrt{\frac{\bar{x}}{w}}
\]

When pweights are specified, semean, sebinomial, and sepoisson are all computed as
\[
\operatorname{se}_{\mathrm{pw}}(\bar{x})=\sqrt{\frac{n}{n-1} \sum_{i=1}^{n}\left\{\frac{v_{i}}{v}\left(x_{i}-\bar{x}\right)\right\}^{2}}
\]
skewness
\[
m_{3} m_{2}^{-3 / 2}
\]
kurtosis
\[
m_{4} m_{2}^{-2}
\]
cv (coefficient of variation)
\[
\frac{s}{\bar{x}}
\]
svycv (coefficient of variation, survey literature)
\[
100 \frac{\operatorname{se}(\bar{x})}{|\bar{x}|}
\]
svycv with pweights
\[
100 \frac{\mathrm{se}_{\mathrm{pw}}(\bar{x})}{|\bar{x}|}
\]
geomean (geometric mean)
\[
\bar{x}_{g}=\exp \left(\frac{1}{w} \sum_{i=1}^{n} w_{i} \ln x_{i}\right)
\]
geosd (geometric standard deviation)
\[
\exp \left(\sqrt{\frac{1}{w .-1} \sum_{i=1}^{n} w_{i}\left(\ln x_{i}-\ln \bar{x}_{g}\right)^{2}}\right)
\]

Let \(x_{(i)}\) refer to the \(x\) in ascending order, and let \(w_{(i)}\) refer to the corresponding weights of \(x_{(i)}\). minimum
\[
x_{(1)}
\]
maximum
\[
x_{(n)}
\]
range
\[
x_{(n)}-x_{(1)}
\]

To obtain the \(p\) th percentile, which we will denote as \(x_{[p]}\), let \(P=n p / 100\) and
\[
W_{(i)}=\frac{n}{w} \sum_{j=1}^{i} w_{(j)}
\]

Find the first index \(i\) such that \(W_{(i)}>P\). The \(p\) th percentile is then
\[
x_{[p]}= \begin{cases}\frac{x_{(i-1)}+x_{(i)}}{2} & \text { if } W_{(i-1)}=P \\ x_{(i)} & \text { otherwise }\end{cases}
\]
q1 (first quartile)
\[
x_{[25]}
\]
q2 (second quartile)
\[
x_{[50]}
\]
q3 (third quartile)
iqr (interquartile range)
\[
x_{[75]}-x_{[25]}
\]

Let \(f\) be an indicator for a specific level of a factor variable and \(f_{i}\) denote an individual observation on \(f\).
fvfrequency (frequency of the factor variable's level)
\[
\sum_{i=1}^{n} w_{i} f_{i}
\]
fvrawfrequency (unweighted frequency of the factor variable's level)
\[
\sum_{i=1}^{n} f_{i}
\]
fvproportion (proportion of the factor variable's level)
\[
\frac{1}{w} \sum_{i-1}^{n} w_{i} f_{i}
\]
fvrawproportion (unweighted proportion of the factor variable's level)
\[
\frac{1}{n} \sum_{i=1}^{n} f_{i}
\]
fvpercent (percentage of the factor variable's level)
\[
\frac{100}{w .} \sum_{i-1}^{n} w_{i} f_{i}
\]
fvrawpercent (unweighted percentage of the factor variable's level)
\[
\frac{100}{n} \sum_{i-1}^{n} f_{i}
\]
proportion is computed from ratios of totals. The numerator is taken from the total for the given cell or cell margin, and the denominator is taken from the total for a cell margin that contains the given cell or cell margin. percent is proportion multiplied by 100.
rawproportion and rawpercent are similarly computed using unweighted totals.

\section*{Also see}
[R] table intro - Introduction to tables of frequencies, summaries, and command results
[R] table hypothesis tests - Table of hypothesis tests
[R] table multiway - Multiway tables
[R] table oneway - One-way tabulation
[R] table regression - Table of regression results
[R] table summary - Table of summary statistics
[R] table twoway - Two-way tabulation
[TABLES] Intro - Introduction

\section*{Title}
tabstat - Compact table of summary statistics
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Acknowledgments & Reference \\
Also see & & &
\end{tabular}

\section*{Description}
tabstat displays summary statistics for a series of numeric variables in one table. It allows you to specify the list of statistics to be displayed. Statistics can be calculated (conditioned on) another variable. tabstat allows substantial flexibility in terms of the statistics presented and the format of the table.

\section*{Quick start}

Mean of v1 displayed using v1's display format
tabstat v1, format
Same as above, but use format with 2 significant digits and a comma
tabstat v1, format (\%9.2fc)
Nonmissing observations, mean, standard error, and coefficient of variation for v 1
tabstat v1, statistics( n mean semean cv )
Quartiles and interquartile range of v 1 and v 2
tabstat v1 v2, statistics(q iqr)
Same as above, but report statistics separately for each level of catvar
tabstat v 1 v 2 , by(catvar) statistics(q iqr)
Same as above, but display a separate column for each statistic
tabstat v1 v2, by(catvar) statistics(q iqr) columns(statistics)

\section*{Menu}

Statistics \(>\) Summaries, tables, and tests \(>\) Other tables \(>\) Compact table of summary statistics

\section*{Syntax}
```

tabstat varlist $[$ if $][$ in $][$ weight $][$, options $]$

```
options
Description
Main
by (varname)
statistics (statname[...])
group statistics by variable report specified statistics

Options
labelwidth(\#)
varwidth(\#)
columns (variables)
columns(statistics)
format [ \(\%\) f fmt \()\) ]
casewise
nototal
missing
noseparator
longstub
save
width for by () variable labels; default is labelwidth (16)
variable width; default is varwidth(12)
display variables in table columns; the default
display statistics in table columns
display format for statistics; default format is \(\% 9.0 \mathrm{~g}\)
perform casewise deletion of observations
do not report overall statistics; use with by ()
report statistics for missing values of by () variable
do not use separator line between by () categories
make left table stub wider
store summary statistics in \(r()\)
by is allowed; see [D] by.
aweights and fweights are allowed; see [U] 11.1.6 weight.

\section*{Options}
\(\underset{\text { by (varname) specifies that the statistics be displayed separately for each unique value of varname ; }}{ }\) varname may be numeric or string. For instance, tabstat height would present the overall mean of height. tabstat height, by (sex) would present the mean height of males, and of females, and the overall mean height. Do not confuse the by () option with the by prefix (see [D] by); both may be specified.
statistics (statname [...]) specifies the statistics to be displayed; the default is equivalent to specifying statistics(mean). (stats() is a synonym for statistics().) Multiple statistics may be specified and are separated by white space, such as statistics(mean sd). Available statistics are
\begin{tabular}{llll} 
statname & Definition & statname & Definition \\
\hline\(\underline{\text { mean }}\) & mean & p 1 & 1 st percentile \\
\(\underline{\text { count }}\) & count of nonmissing observations & p 5 & 5th percentile \\
n & same as count & p 10 & 10 th percentile \\
\(\underline{\text { sum }}\) & sum & p 25 & 25th percentile \\
\(\underline{\max }\) & maximum & \(\underline{\text { median }}\) & median (same as p50) \\
\(\underline{\min }\) & minimum & p 50 & 50th percentile (same as median) \\
\(\underline{\text { range }}\) & range \(=\max -\min\) & p 75 & 75th percentile \\
sd & standard deviation & p 90 & 90th percentile \\
\(\underline{\text { variance }}\) & variance & p 95 & 95th percentile \\
cv & coefficient of variation \((\mathrm{sd} / \mathrm{mean})\) & p 99 & 99th percentile \\
\(\underline{\text { semean }}\) & standard error of mean \((\mathrm{sd} / \sqrt{\mathrm{n}})\) & iqr & interquartile range \(=\mathrm{p} 75-\mathrm{p} 25\) \\
\(\underline{\text { skewness }}\) & skewness & q & equivalent to specifying p25 p50 p75 \\
\(\underline{\text { kurtosis }}\) & kurtosis & &
\end{tabular}

Options
labelwidth(\#) specifies the maximum width to be used within the stub to display the labels of the by () variable. The default is labelwidth(16). \(8 \leq \# \leq 32\).
varwidth (\#) specifies the maximum width to be used within the stub to display the names of the variables. The default is varwidth(12). varwidth() is effective only with columns (statistics). Setting varwidth() implies longstub. \(8 \leq \# \leq 32\).
columns (variables | statistics) specifies whether to display variables or statistics in the columns of the table. columns (variables) is the default when more than one variable is specified.
format and format (\%fmt) specify how the statistics are to be formatted. The default is to use a \(\% 9.0 \mathrm{~g}\) format.
format specifies that each variable's statistics be formatted with the variable's display format; see [D] format.
format (\%fmt) specifies the format to be used for all statistics.
The column width is the maximum width of these formats. The minimum column width is nine display characters.
casewise specifies casewise deletion of observations. Statistics are to be computed for the sample that is not missing for any of the variables in varlist. The default is to use all the nonmissing values for each variable.
nototal is for use with by (); it specifies that the overall statistics not be reported.
missing specifies that missing values of the by () variable be treated just like any other value and that statistics should be displayed for them. The default is not to report the statistics for the by ()== missing group. If the by () variable is a string variable, by ( \()==="\) is considered to mean missing.
noseparator specifies that a separator line between the by () categories not be displayed.
longstub specifies that the left stub of the table be made wider so that it can include names of the statistics or variables in addition to the categories of by (varname). The default is to describe the statistics or variables in a header. longstub is ignored if by (varname) is not specified.
save specifies that the summary statistics be returned in \(r()\). The overall (unconditional) statistics are returned in matrix \(r\) (StatTotal) (rows are statistics, columns are variables). The conditional statistics are returned in the matrices \(r\) (Stat1), \(r\) (Stat2), .., and the names of the corresponding variables are returned in the macros \(r\) (name1), \(r\) (name2), \(\ldots\).

\section*{Remarks and examples}

This command is probably most easily understood by going through a series of examples.

\section*{> Example 1}

We have data on the price, weight, mileage rating, and repair record of 22 foreign and 52 domestic 1978 automobiles. We want to summarize these variables for the different origins of the automobiles.


More summary statistics can be requested via the statistics() option. The group totals can be suppressed with the nototal option.
```

. tabstat price weight mpg rep78, by(foreign) stat(mean sd min max) nototal
Summary statistics: Mean, SD, Min, Max
Group variable: foreign (Car origin)

| foreign | price | weight | mpg | rep78 |
| :--- | ---: | ---: | ---: | ---: |
| Domestic | 6072.423 | 3317.115 | 19.82692 | 3.020833 |
|  | 3097.104 | 695.3637 | 4.743297 | .837666 |
|  | 3291 | 1800 | 12 | 1 |
|  | 15906 | 4840 | 34 | 5 |
| Foreign | 6384.682 | 2315.909 | 24.77273 | 4.285714 |
|  | 2621.915 | 433.0035 | 6.611187 | .7171372 |
|  | 3748 | 1760 | 14 | 3 |
|  | 12990 | 3420 | 41 | 5 |

```

Although the header of the table describes the statistics running vertically in the "cells", the table may become hard to read, especially with many variables or statistics. The longstub option specifies that a column be added describing the contents of the cells. The format option can be issued to specify that tabstat display the statistics by using the display format of the variables rather than the overall default \(\% 9.0 \mathrm{~g}\).
. tabstat price weight mpg rep78, by(foreign) stat(mean sd min max) long format
\begin{tabular}{lr|rrrr} 
foreign & Stats & price & weight & mpg & rep78 \\
\hline Domestic & Mean & \(6,072.4\) & \(3,317.1\) & 19.8269 & 3.02083 \\
& SD & \(3,097.1\) & 695.364 & 4.7433 & .837666 \\
& Min & 3,291 & 1,800 & 12 & 1 \\
& Max & 15,906 & 4,840 & 34 & 5 \\
\hline Foreign & Mean & \(6,384.7\) & \(2,315.9\) & 24.7727 & 4.28571 \\
& SD & \(2,621.9\) & 433.003 & 6.61119 & .717137 \\
& Min & 3,748 & 1,760 & 14 & 3 \\
& Max & 12,990 & 3,420 & 41 & 5 \\
& & & & & \\
& Total & Mean & \(6,165.3\) & \(3,019.5\) & 21.2973 \\
& SD & \(2,949.5\) & 777.194 & 5.7855 & .989932 \\
& Min & 3,291 & 1,760 & 12 & 1 \\
& Max & 15,906 & 4,840 & 41 & 5 \\
& & & & &
\end{tabular}

We can specify a layout of the table in which the statistics run horizontally and the variables run vertically by specifying the col(statistics) option.
\begin{tabular}{|c|c|c|c|c|}
\hline foreign & Variable & Min & Mean & Max \\
\hline \multirow[t]{4}{*}{Domestic} & price & 3291 & 6072.423 & 15906 \\
\hline & weight & 1800 & 3317.115 & 4840 \\
\hline & mpg & 12 & 19.82692 & 34 \\
\hline & rep78 & 1 & 3.020833 & 5 \\
\hline \multirow[t]{4}{*}{Foreign} & price & 3748 & 6384.682 & 12990 \\
\hline & weight & 1760 & 2315.909 & 3420 \\
\hline & mpg & 14 & 24.77273 & 41 \\
\hline & rep78 & 3 & 4.285714 & 5 \\
\hline \multirow[t]{4}{*}{Total} & price & 3291 & 6165.257 & 15906 \\
\hline & weight & 1760 & 3019.459 & 4840 \\
\hline & mpg & 12 & 21.2973 & 41 \\
\hline & rep78 & 1 & 3.405797 & 5 \\
\hline
\end{tabular}

Finally, tabstat can also be used to enhance summarize so we can specify the statistics to be displayed. For instance, we can display the number of observations, the mean, the coefficient of variation, and the \(25 \%, 50 \%\), and \(75 \%\) quantiles for a list of variables.
. tabstat price weight mpg rep78, stat(n mean cv q) col(stat)
\begin{tabular}{r|rrrrrr} 
variable & N & mean & cv & p25 & p50 & p75 \\
\hline price & 74 & 6165.257 & .478406 & 4195 & 5006.5 & 6342 \\
weight & 74 & 3019.459 & .2573949 & 2240 & 3190 & 3600 \\
mpg & 74 & 21.2973 & .2716543 & 18 & 20 & 25 \\
rep78 & 69 & 3.405797 & .290661 & 3 & 3 & 4
\end{tabular}

Because we did not specify the by () option, these statistics were not displayed for the subgroups of the data formed by the categories of the by () variable.

\section*{Video example}

Descriptive statistics in Stata

\section*{Acknowledgments}

The tabstat command was written by Jeroen Weesie and Vincent Buskens both of the Department of Sociology at Utrecht University, The Netherlands.

\section*{Reference}

Donath, S. 2018. baselinetable: A command for creating one- and two-way tables of summary statistics. Stata Journal 18: 327-344.

\section*{Also see}
[R] summarize - Summary statistics
[R] table - Table of frequencies, summaries, and command results
[R] table summary - Table of summary statistics
\([R]\) tabulate, summarize() - One- and two-way tables of summary statistics
[D] collapse - Make dataset of summary statistics

\section*{Title}
tabulate oneway - One-way table of frequencies
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & References \\
Also see & & &
\end{tabular}

\section*{Description}
tabulate produces a one-way table of frequency counts.
For information on a two-way table of frequency counts along with measures of association, including the common Pearson \(\chi^{2}\), the likelihood-ratio \(\chi^{2}\), Cramér's \(V\), Fisher's exact test, Goodman and Kruskal's gamma, and Kendall's \(\tau_{b}\), see [R] tabulate twoway.
tab1 produces a one-way tabulation for each variable specified in varlist.
Also see [R] table and [R] tabstat if you want one-, two-, or \(n\)-way table of frequencies and a wide variety of statistics. See [R] tabulate, summarize() for a description of tabulate with the summarize() option; it produces a table (breakdowns) of means and standard deviations. table is better than tabulate, summarize(), but tabulate, summarize() is faster. See [R] Epitab for a \(2 \times 2\) table with statistics of interest to epidemiologists.

\section*{Quick start}

One-way table of frequencies for v1
tabulate v1
Sort table in descending order of frequency
tabulate v1, sort
Generate indicator variables v1_1, v1_2, ...representing the levels of v1 tabulate v1, generate(v1_)

Treat missing values like other values of v 1
tabulate v1, missing
Display numeric values of v1 rather than value labels tabulate v1, nolabel

Create one-way tables for v1, v2, and v3
tab1 v1 v2 v3

\section*{Menu}

\section*{tabulate oneway}

Statistics \(>\) Summaries, tables, and tests \(>\) Frequency tables \(>\) One-way table
tabulate ..., generate()
Data \(>\) Create or change data \(>\) Other variable-creation commands \(>\) Create indicator variables

\section*{tab1}

Statistics \(>\) Summaries, tables, and tests \(>\) Frequency tables \(>\) Multiple one-way tables

\section*{Syntax}
```

One-way table
tabulate varname [if] [in] [weight] [, tabulatel_options]

```

One-way table for each variable-a convenience tool
tab1 varlist [if] [in] [weight] [, tabl_options]
tabulatel_options Description

Main
subpop (varname)
missing
nofreq
nolabel
plot
sort
Advanced
generate (stubname)
matcell (matname)
matrow (matname)
exclude observations for which varname \(=0\) treat missing values like other values
do not display frequencies
display numeric codes rather than value labels produce a bar chart of the relative frequencies display the table in descending order of frequency
create indicator variables for stubname
save frequencies in matname; programmer's option
save unique values of varname in matname; programmer's option
tabl_options
Description
Main
subpop (varname)
exclude observations for which varname \(=0\)
missing
treat missing values like other values
nofreq
nolabel
plot
sort
display numeric codes rather than value labels
produce a bar chart of the relative frequencies
display the table in descending order of frequency
by is allowed with tabulate and tab1, and collect is allowed with tabulate; see [U] 11.1.10 Prefix commands. fweights, aweights, and iweights are allowed; see [U] 11.1.6 weight.

\section*{Options}
\(\qquad\) Main
subpop(varname) excludes observations for which varname \(=0\) in tabulating frequencies. The mathematical results of tabulate ..., subpop(myvar) are the same as tabulate ... if myvar \(!=0\), but the table may be presented differently. The identities of the rows and columns will be determined from all the data, including the myvar \(=0\) group, so there may be entries in the table with frequency 0 .

Consider tabulating answer, a variable that takes on values 1,2 , and 3 , but consider tabulating it just for the male==1 subpopulation. Assume that answer is never 2 in this group. tabulate answer if male==1 produces a table with two rows: one for answer 1 and one for answer 3. There will be no row for answer 2 because answer 2 was never observed. tabulate answer, subpop(male) produces a table with three rows. The row for answer 2 will be shown as having 0 frequency.
missing requests that missing values be treated like other values in calculations of counts, percentages, and other statistics.
nofreq suppresses the printing of the frequencies.
nolabel causes the numeric codes to be displayed rather than the value labels.
plot produces a bar chart of the relative frequencies in a one-way table. (Also see [R] histogram.) sort puts the table in descending order of frequency (and ascending order of the variable within equal values of frequency).

Advanced
generate (stubname) creates a set of indicator variables (stubname1, stubname \(2, \ldots\) ) reflecting the observed values of the tabulated variable. The generate() option may not be used with the by prefix.
matcell (matname) saves the reported frequencies in matname. This option is for use by programmers. matrow (matname) saves the numeric values of the \(r \times 1\) row stub in matname. This option is for use by programmers. matrow() may not be specified if the row variable is a string.

\section*{Limits}

A one-way table may have a maximum of 12,000 rows (Stata/MP and Stata/SE) or 3,000 rows (Stata/BE).

\section*{Remarks and examples}

Remarks are presented under the following headings:
```

tabulate
tab1
Video example

```

For each value of a specified variable, tabulate reports the number of observations with that value. The number of times a value occurs is called its frequency.

\section*{tabulate}
> Example 1
We have data summarizing the speed limit and the accident rate per million vehicle miles along various Minnesota highways in 1973. The variable containing the speed limit is called spdlimit. If we summarize the variable, we obtain its mean and standard deviation:
```

. use https://www.stata-press.com/data/r18/hiway
(Minnesota highway data, 1973)

```
. summarize spdlimit
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline spdlimit & 39 & 55 & 5.848977 & 40 & 70
\end{tabular}

The average speed limit is 55 miles per hour. We can learn more about this variable by tabulating it:
. tabulate spdlimit
\begin{tabular}{r|rrr} 
Speed limit & Freq. & Percent & Cum. \\
\hline 40 & 1 & 2.56 & 2.56 \\
45 & 3 & 7.69 & 10.26 \\
50 & 7 & 17.95 & 28.21 \\
55 & 15 & 38.46 & 66.67 \\
60 & 11 & 28.21 & 94.87 \\
65 & 1 & 2.56 & 97.44 \\
70 & 1 & 2.56 & 100.00 \\
\hline Total & 39 & 100.00 &
\end{tabular}

We see that one highway has a speed limit of 40 miles per hour, three have speed limits of 45,7 of 50, and so on. The column labeled Percent shows the percentage of highways in the dataset that have the indicated speed limit. For instance, \(38.46 \%\) of highways in our dataset have a speed limit of 55 miles per hour. The final column shows the cumulative percentage. We see that \(66.67 \%\) of highways in our dataset have a speed limit of 55 miles per hour or less.

\section*{\(>\) Example 2}

The plot option places a sideways histogram alongside the table:


Of course, graph can produce better-looking histograms; see [R] histogram.

\section*{\(>\) Example 3}
tabulate labels tables using variable and value labels if they exist. To demonstrate how this works, let's add a new variable to our dataset that categorizes spdlimit into three categories. We will call this new variable spdcat:
```

. generate spdcat=recode(spdlimit,50,60,70)

```

The recode () function divides spdlimit into 50 miles per hour or below, \(51-60\), and above 60; see [FN] Programming functions. We specified the breakpoints in the arguments (spdlimit, 50, 60, 70 ). The first argument is the variable to be recoded. The second argument is the first breakpoint, the third argument is the second breakpoint, and so on. We can specify as many breakpoints as we wish.
recode() used our arguments not only as the breakpoints but also to label the results. If spdlimit is less than or equal to 50 , spdcat is set to 50 ; if spdlimit is between 51 and 60 , spdcat is 60 ; otherwise, spdcat is arbitrarily set to 70. (See [U] 26 Working with categorical data and factor variables.)

Because we just created the variable spdcat, it is not yet labeled. When we make a table using this variable, tabulate uses the variable's name to label it:
```

. tabulate spdcat

```
\begin{tabular}{r|rrr} 
spdcat & Freq. & Percent & Cum. \\
\hline 50 & 11 & 28.21 & 28.21 \\
60 & 26 & 66.67 & 94.87 \\
70 & 2 & 5.13 & 100.00 \\
\hline Total & 39 & 100.00 &
\end{tabular}

Even through the table is not well labeled, recode()'s coding scheme provides us with clues as to the table's meaning. The first line of the table corresponds to 50 miles per hour and below, the next to 51 through 60 miles per hour, and the last to above 60 miles per hour.

We can improve this table by labeling the values and variables:
```

. label define scat 50 "40 to 50" 60 "55 to 60" 70 "Above 60"
. label values spdcat scat
. label variable spdcat "Speed Limit Category"

```

We define a value label called scat that attaches labels to the numbers 50, 60, and 70 using the label define command; see [U] 12.6.3 Value labels. We label the value 50 as " 40 to 50 ", because we looked back at our original tabulation in the first example and saw that the speed limit was never less than 40. Similarly, we could have labeled the last category " 65 to 70 " because the speed limit is never greater than 70 miles per hour.

Next, we requested that Stata label the values of the new variable spdcat using the value label scat. Finally, we labeled our variable Speed Limit Category. We are now ready to tabulate the result:
\begin{tabular}{l|rrr}
\begin{tabular}{l} 
tabulate spdcat \\
\begin{tabular}{c} 
Speed Limit \\
Category
\end{tabular} \\
Freq.
\end{tabular} & Percent & Cum. \\
\hline 40 to 50 & 11 & 28.21 & 28.21 \\
55 to 60 & 26 & 66.67 & 94.87 \\
Above 60 & 2 & 5.13 & 100.00 \\
\hline Total & 39 & 100.00 &
\end{tabular}

\section*{> Example 4}

If we have missing values in our dataset, tabulate ignores them unless we explicitly indicate otherwise. We have no missing data in our example, so let's add some:
```

. replace spdcat=. in 39
(1 real change made, 1 to missing)

```

We changed the first observation on spdcat to missing. Let's now tabulate the result:
\begin{tabular}{r|rrr}
. tabulate spdcat & & & \\
\begin{tabular}{r} 
Speed Limit \\
Category
\end{tabular} & Freq. & Percent & Cum. \\
\hline 40 to 50 & 11 & 28.95 & 28.95 \\
55 to 60 & 26 & 68.42 & 97.37 \\
Above 60 & 1 & 2.63 & 100.00 \\
\hline Total & 38 & 100.00 &
\end{tabular}

Comparing this output with that in the previous example, we see that the total frequency count is now one less than it was - 38 rather than 39 . Also, the 'Above 60 ' category now has only one observation where it used to have two, so we evidently changed a road with a high speed limit.

We want tabulate to treat missing values just as it treats numbers, so we specify the missing option:
\begin{tabular}{r|rrr}
\begin{tabular}{c} 
tabulate spdcat, missing \\
Speed Limit \\
Category
\end{tabular} & Freq. & Percent & Cum. \\
\hline 40 to 50 & 11 & 28.21 & 28.21 \\
55 to 60 & 26 & 66.67 & 94.87 \\
Above 60 & 1 & 2.56 & 97.44 \\
. & 1 & 2.56 & 100.00 \\
\hline Total & 39 & 100.00 &
\end{tabular}

We now see our missing value-the last category, labeled '. ', shows a frequency count of 1 . The table sum is once again 39 .

Let's put our dataset back as it was originally:
```

. replace spdcat=70 in 39
(1 real change made)

```

\section*{- Technical note}
tabulate also can automatically create indicator variables from categorical variables. We will briefly review that capability here, but see [U] 26 Working with categorical data and factor variables for a complete description. Let's begin by describing our highway dataset:
\begin{tabular}{|c|c|c|c|c|}
\hline \multicolumn{5}{|l|}{Contains data from https://www.stata-press.com/data/r18/hiway.dta} \\
\hline \begin{tabular}{l}
Observations: \\
Variables
\end{tabular} & & \[
39
\] & & Minnesota highway data, 1973 \\
\hline Variable name & Storage type & Display format & \begin{tabular}{l}
Value \\
label
\end{tabular} & Variable label \\
\hline spdlimit & byte & \%8.0g & & Speed limit \\
\hline rate & byte & \%9.0g & rcat & ```
Accident rate per million vehicle
    miles
``` \\
\hline spdcat & float & \%9.0g & scat & Speed Limit Category \\
\hline
\end{tabular}

Sorted by:
Note: Dataset has changed since last saved.

Our dataset contains three variables. We will type tabulate spdcat, generate(spd), describe our data, and then explain what happened.


Sorted by:
Note: Dataset has changed since last saved.
When we typed tabulate with the generate() option, Stata responded by producing a one-way frequency table, so it appeared that the option did nothing. Yet when we describe our dataset, we find that we now have six variables instead of the original three. The new variables are named spd1, spd2, and spd3.

When we specify the generate() option, we are telling Stata to not only produce the table but also create a set of indicator variables that correspond to that table. Stata adds a numeric suffix to the name we specify in the parentheses. spd1 refers to the first line of the table, spd2 to the second line, and so on. Also, Stata labels the variables so that we know what they mean. spd1 is an indicator variable that is true (takes on the value 1) when spdcat is between 40 and 50 ; otherwise, it is zero. (There is an exception: if spdcat is missing, so are the \(\operatorname{spd} 1\), \(s p d 2\), and \(\operatorname{spd} 3\) variables. This did not happen in our dataset.)

We want to prove our claim. Because we have not yet introduced two-way tabulations, we will use the summarize statement:
. summarize spdlimit if spd1==1
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline spdlimit & 11 & 47.72727 & 3.437758 & 40 & 50
\end{tabular}
. summarize spdlimit if spd2==1
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline spdlimit & 26 & 57.11538 & 2.519157 & 55 & 60
\end{tabular}
\begin{tabular}{c|ccccr}
\multicolumn{7}{c}{ summarize spdlimit if spd3==1 } \\
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline spdlimit & 2 & 67.5 & 3.535534 & 65 & 70
\end{tabular}

Notice the indicated minimum and maximum in each of the tables above. When we restrict the sample to spd1, spdlimit is between 40 and 50 ; when we restrict the sample to spd2, spdlimit is between 55 and 60 ; when we restrict the sample to spd3, spdlimit is between 65 and 70 .

Thus tabulate provides an easy way to create indicator (sometimes called dummy) variables. For an overview of indicator and categorical variables, see [U] 26 Working with categorical data and factor variables.

\section*{tab1}
tab1 is a convenience tool. Typing
. tab1 myvar thisvar thatvar, plot
is equivalent to typing
. tabulate myvar, plot
. tabulate thisvar, plot
. tabulate thatvar, plot

\section*{Video example}

Tables and cross-tabulations in Stata

\section*{Stored results}
tabulate and tab1 store the following in \(r()\) :
Scalars
\(r(N)\) number of observations \(r(r) \quad\) number of rows

\section*{References}

Cox, N. J. 2009. Speaking Stata: I. J. Good and quasi-Bayes smoothing of categorical frequencies. Stata Journal 9: 306-314.

Donath, S. 2018. baselinetable: A command for creating one- and two-way tables of summary statistics. Stata Journal 18: 327-344.
Harrison, D. A. 2006. Stata tip 34: Tabulation by listing. Stata Journal 6: 425-427.

\section*{Also see}
[R] Epitab - Tables for epidemiologists
[R] table - Table of frequencies, summaries, and command results
[R] table oneway - One-way tabulation
[R] tabstat - Compact table of summary statistics
[R] tabulate twoway - Two-way table of frequencies
[R] tabulate, summarize() - One- and two-way tables of summary statistics
[D] collapse - Make dataset of summary statistics
[SVY] svy: tabulate oneway - One-way tables for survey data
[SVY] svy: tabulate twoway - Two-way tables for survey data
[XT] xttab - Tabulate xt data
[U] 12.6.3 Value labels
[U] 26 Working with categorical data and factor variables

\section*{Title}
tabulate twoway - Two-way table of frequencies
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
tabulate produces a two-way table of frequency counts, along with various measures of association, including the common Pearson's \(\chi^{2}\), the likelihood-ratio \(\chi^{2}\), Cramér's \(V\), Fisher's exact test, Goodman and Kruskal's gamma, and Kendall's \(\tau_{b}\).

Line size is respected. That is, if you resize the Results window before running tabulate, the resulting two-way tabulation will take advantage of the available horizontal space. Stata for Unix(console) users can instead use the set linesize command to take advantage of this feature.
tab2 produces all possible two-way tabulations of the variables specified in varlist.
tabi displays the \(r \times c\) table, using the values specified; rows are separated by ' \(\backslash\) '. If no options are specified, it is as if exact were specified for a \(2 \times 2\) table and chi2 were specified otherwise. See [U] 19 Immediate commands for a general description of immediate commands. See Tables with immediate data below for examples using tabi.

See [R] tabulate oneway if you want a one-way table of frequencies. See [R] table and [R] tabstat if you want one-, two-, or \(n\)-way table of frequencies and a wide variety of summary statistics. See \([R]\) tabulate, summarize() for a description of tabulate with the summarize() option; it produces a table (breakdowns) of means and standard deviations. table is better than tabulate, summarize(), but tabulate, summarize() is faster. See [R] Epitab for a \(2 \times 2\) table with statistics of interest to epidemiologists.

\section*{Quick start}

Two-way table of frequencies for v1 and v2
```

tabulate v1 v2

```

Add row percentages
```

tabulate v1 v2, row

```

Frequencies only for observations where v3 \(=1\)
```

tabulate v1 v2 if v3==1

```

Weighted cell counts using frequency weights defined by wvar tabulate v1 v2 [fweight=wvar]

Pearson's \(\chi^{2}\) test and each cell's contribution
```

tabulate v1 v2, chi2 cchi2

```

All available measures of association
```

tabulate v1 v2, all

```

All possible two-way tables for v1, v2, and v3
tab2 v1 v2 v3
Input cell frequencies and perform \(\chi^{2}\) test
tabi 301838 \ 13722 , chi2

\section*{Menu}

\section*{tabulate}

Statistics \(>\) Summaries, tables, and tests \(>\) Frequency tables \(>\) Two-way table with measures of association tab2

Statistics \(>\) Summaries, tables, and tests \(>\) Frequency tables \(>\) All possible two-way tables

\section*{tabi}

Statistics \(>\) Summaries, tables, and tests \(>\) Frequency tables \(>\) Table calculator

\section*{Syntax}

Two-way table
tabulate varname varname \(_{2}\) [if] [in] [weight] [, options]

Two-way table for all possible combinations-a convenience tool
tab2 varlist [if] [in] [weight] [, options]

Immediate form of two-way tabulations
```

tabi $\#_{11} \#_{12}[\ldots] \backslash \#_{21} \#_{22}[\ldots][\backslash \ldots][$, options $]$

```
options Description

Main
chi2
exact [(\#)]
gamma
lrchi2
taub
V
cchi2
column
row
clrchi2
cell
expected
nofreq
rowsort
colsort
missing
wrap
[no]key
nolabel
nolog
* firstonly

\section*{Advanced}
matcell (matname)
matrow (matname)
matcol(matname)
\(\dagger\) replace
all
report Pearson's \(\chi^{2}\)
report Fisher's exact test
report Goodman and Kruskal's gamma
report likelihood-ratio \(\chi^{2}\)
report Kendall's \(\tau_{b}\)
report Cramér's \(V\)
report Pearson's \(\chi^{2}\) in each cell
report relative frequency within its column of each cell
report relative frequency within its row of each cell
report likelihood-ratio \(\chi^{2}\) in each cell
report the relative frequency of each cell
report expected frequency in each cell
do not display frequencies
list rows in order of observed frequency
list columns in order of observed frequency
treat missing values like other values
do not wrap wide tables
report/suppress cell contents key
display numeric codes rather than value labels
do not display enumeration \(\log\) for Fisher's exact test show only tables that include the first variable in varlist
save frequencies in matname; programmer's option
save unique values of varname \(_{1}\) in matname; programmer's option save unique values of varname \(_{2}\) in matname; programmer's option replace current data with given cell frequencies
equivalent to specifying chi2 lrchi2 V gamma taub
*firstonly is available only for tab2.
\(\dagger_{\text {replace }}\) is available only for tabi.
by is allowed with tabulate and tab2, and collect is allowed with tabulate and tabi; see [U] 11.1.10 Prefix commands.
fweights, aweights, and iweights are allowed by tabulate. fweights are allowed by tab2. See [U] 11.1.6 weight. all does not appear in the dialog box.

\section*{Options}

\section*{Main}
chi2 calculates and displays Pearson's \(\chi^{2}\) for the hypothesis that the rows and columns in a two-way table are independent. chi2 may not be specified if aweights or iweights are specified.
exact [(\#)] displays the significance calculated by Fisher's exact test and may be applied to \(r \times c\) as well as to \(2 \times 2\) tables. For \(2 \times 2\) tables, both one- and two-sided probabilities are displayed. For \(r \times c\) tables, two-sided probabilities are displayed. The optional positive integer \# is a multiplier on the amount of memory that the command is permitted to consume. The default is 1 . This option should not be necessary for reasonable \(r \times c\) tables. If the command terminates with error 910, try exact (2). The maximum row or column dimension allowed when computing Fisher's exact test is the maximum row or column dimension for tabulate (see [R] Limits).
gamma displays Goodman and Kruskal's gamma along with its asymptotic standard error. gamma is appropriate only when both variables are ordinal. gamma may not be specified if aweights or iweights are specified.
Irchi2 displays the likelihood-ratio \(\chi^{2}\) statistic. Irchi2 may not be specified if aweights or iweights are specified.
taub displays Kendall's \(\tau_{b}\) along with its asymptotic standard error. taub is appropriate only when both variables are ordinal. taub may not be specified if aweights or iweights are specified.
V (note capitalization) displays Cramér's \(V\). V may not be specified if aweights or iweights are specified.
cchi2 displays each cell's contribution to Pearson's \(\chi^{2}\) in a two-way table.
column displays the relative frequency of each cell within its column in a two-way table.
row displays the relative frequency of each cell within its row in a two-way table.
clrchi2 displays each cell's contribution to the likelihood-ratio \(\chi^{2}\) in a two-way table.
cell displays the relative frequency of each cell in a two-way table.
expected displays the expected frequency of each cell in a two-way table.
nofreq suppresses the printing of the frequencies.
rowsort and colsort specify that the rows and columns, respectively, be presented in order of observed frequency.

By default, rows and columns are presented in ascending order of the row and column variable. For instance, if you type tabulate a b and a takes on the values 2,3 , and 5 , then the first row of the table will correspond to \(\mathrm{a}=2\); the second row will correspond to \(\mathrm{a}=3\); and the third row will correspond to \(\mathrm{a}=5\).
rowsort specifies that the rows instead be presented in descending order of observed frequency of the values. If you type twoway a b, rowsort, the most frequently observed value of a will be listed in the first row, the second most frequently observed value of a in the second row, and
so on. If there are rows with equal frequencies, they will be presented in ascending order of the values of a . If \(\mathrm{a}=5\) occurs with frequency 1,000 and values \(\mathrm{a}=2\) and \(\mathrm{a}=3\) each occur with frequency 500 , the rows will be presented in the order \(\mathrm{a}=5\), \(\mathrm{a}=2\), and \(\mathrm{a}=3\).
colsort does the same as rowsort, except with the columns and the column variable.
rowsort and colsort may be specified together.
missing requests that missing values be treated like other values in calculations of counts, percentages, and other statistics.
wrap requests that Stata take no action on wide, two-way tables to make them readable. Unless wrap is specified, wide tables are broken into pieces to enhance readability.
[no] key suppresses or forces the display of a key above two-way tables. The default is to display the key if more than one cell statistic is requested, and otherwise to omit it. key forces the display of the key. nokey suppresses its display.
nolabel causes the numeric codes to be displayed rather than the value labels.
nolog suppresses the display of the log for Fisher's exact test. Using Fisher's exact test requires counting all tables that have a probability exceeding that of the observed table given the observed row and column totals. The log counts down each stage of the network computations, starting from the number of columns and counting down to 1 , displaying the number of nodes in the network at each stage. A log is not displayed for \(2 \times 2\) tables.
firstonly, available only with tab2, restricts the output to only those tables that include the first variable in varlist. Use this option to interact one variable with a set of others.

\section*{Advanced}
matcell (matname) saves the reported frequencies in matname. This option is for use by programmers. matrow (matname) saves the numeric values of the \(r \times 1\) row stub in matname. This option is for use by programmers. matrow () may not be specified if the row variable is a string.
matcol (matname) saves the numeric values of the \(1 \times c\) column stub in matname. This option is for use by programmers. matcol() may not be specified if the column variable is a string.
replace indicates that the immediate data specified as arguments to the tabi command be left as the current data in place of whatever data were there.

The following option is available with tabulate but is not shown in the dialog box:
all is equivalent to specifying chi2 lrchi2 V gamma taub. Note the omission of exact. When all is specified, no may be placed in front of the other options. all noV requests all association measures except Cramér's \(V\) (and Fisher's exact). all exact requests all association measures, including Fisher's exact test. all may not be specified if aweights or iweights are specified.

\section*{Limits}

Two-way tables may have a maximum of 1,200 rows and 80 columns (Stata/MP and Stata/SE) or 300 rows and 20 columns (Stata/BE). If larger tables are needed, see [R] table.

\section*{Remarks and examples}

Remarks are presented under the following headings:
```

tabulate
Measures of association
N-way tables
Weighted data
Tables with immediate data
tab2
Video examples

```

For each value of a specified variable (or a set of values for a pair of variables), tabulate reports the number of observations with that value. The number of times a value occurs is called its frequency.

\section*{tabulate}
> Example 1
tabulate will make two-way tables if we specify two variables following the word tabulate. In our highway dataset, we have a variable called rate that divides the accident rate into three categories: below 4, 4-7, and above 7 per million vehicle miles. Let's make a table of the speed limit category and the accident-rate category:
\begin{tabular}{|c|c|c|c|c|}
\hline \[
\begin{array}{r}
\text { Speed } \\
\text { limit } \\
\text { category }
\end{array}
\] &  & \[
\begin{aligned}
& \text { per } \\
& \text { mile } \\
& 4-7
\end{aligned}
\] & \begin{tabular}{l}
lion \\
Above 7
\end{tabular} & Total \\
\hline 40 to 50 & 3 & 5 & 3 & 11 \\
\hline 55 to 50 & 19 & 6 & 1 & 26 \\
\hline Above 60 & 2 & 0 & 0 & 2 \\
\hline Total & 24 & 11 & 4 & 39 \\
\hline
\end{tabular}

The table indicates that three stretches of highway have an accident rate below 4 and a speed limit of 40 to 50 miles per hour. The table also shows the row and column sums (called the marginals). The number of highways with a speed limit of 40 to 50 miles per hour is 11 , which is the same result we obtained in our previous one-way tabulations.

Stata can present this basic table in several ways - 16, to be precise - and we will show just a few below. It might be easier to read the table if we included the row percentages. For instance, of 11 highways in the lowest speed limit category, three are also in the lowest accident-rate category. Three-elevenths amounts to some \(27.3 \%\). We can ask Stata to fill in this information for us by using the row option:
. tabulate spdcat rate, row

\begin{tabular}{r|rrr|r}
\begin{tabular}{r} 
Speed \\
limit \\
category
\end{tabular} & \multicolumn{3}{|c}{\begin{tabular}{r} 
Accident rate per million \\
vehicle miles \\
Below 4
\end{tabular}} & 4-7
\end{tabular} Above 7 \begin{tabular}{r} 
Total \\
\hline 40 to 50
\end{tabular}

The number listed below each frequency is the percentage of cases that each cell represents out of its row. That is easy to remember because we see \(100 \%\) listed in the "Total" column. The bottom row is also informative. We see that \(61.54 \%\) of all the highways in our dataset fall into the lowest accident-rate category, that \(28.21 \%\) are in the middle category, and that \(10.26 \%\) are in the highest.
tabulate can calculate column percentages and cell percentages, as well. It does so when we specify the column or cell options, respectively. We can even specify them together. Below is a table that includes everything:
. tabulate spdcat rate, row column cell

\begin{tabular}{|c|c|c|c|c|}
\hline \begin{tabular}{l}
Speed \\
limit
\end{tabular} & \multicolumn{3}{|l|}{Accident rate per million vehicle miles} & \multirow[b]{2}{*}{Total} \\
\hline category & Below 4 & 4-7 & Above 7 & \\
\hline \multirow[t]{4}{*}{40 to 50} & 3 & 5 & 3 & 11 \\
\hline & 27.27 & 45.45 & 27.27 & 100.00 \\
\hline & 12.50 & 45.45 & 75.00 & 28.21 \\
\hline & 7.69 & 12.82 & 7.69 & 28.21 \\
\hline \multirow[t]{4}{*}{55 to 50} & 19 & 6 & 1 & 26 \\
\hline & 73.08 & 23.08 & 3.85 & 100.00 \\
\hline & 79.17 & 54.55 & 25.00 & 66.67 \\
\hline & 48.72 & 15.38 & 2.56 & 66.67 \\
\hline \multirow[t]{4}{*}{Above 60} & 2 & 0 & 0 & 2 \\
\hline & 100.00 & 0.00 & 0.00 & 100.00 \\
\hline & 8.33 & 0.00 & 0.00 & 5.13 \\
\hline & 5.13 & 0.00 & 0.00 & 5.13 \\
\hline \multirow[t]{4}{*}{Total} & 24 & 11 & 4 & 39 \\
\hline & 61.54 & 28.21 & 10.26 & 100.00 \\
\hline & 100.00 & 100.00 & 100.00 & 100.00 \\
\hline & 61.54 & 28.21 & 10.26 & 100.00 \\
\hline
\end{tabular}

The number at the top of each cell is the frequency count. The second number is the row percentage - they sum to \(100 \%\) going across the table. The third number is the column percentage - they sum to \(100 \%\) going down the table. The bottom number is the cell percentage - they sum to \(100 \%\) going down all the columns and across all the rows. For instance, highways with a speed limit above 60 miles per hour and in the lowest accident rate category account for \(100 \%\) of highways with a speed limit above 60 miles per hour; \(8.33 \%\) of highways in the lowest accident-rate category; and \(5.13 \%\) of all our data.

A fourth option, nofreq, tells Stata not to print the frequency counts. To construct a table consisting of only row percentages, we type
\begin{tabular}{r|rrr|r}
. tabulate spdcat rate, row nofreq \\
\begin{tabular}{r} 
Speed \\
limit \\
category
\end{tabular} & \begin{tabular}{r} 
Accident rate per million \\
vehicle miles
\end{tabular} & \\
\hline 40 to 50 & 27.27 & 45.45 & 27.27 & 100.00 \\
55 to 50 & 73.08 & 23.08 & 3.85 & 100.00 \\
Above 60 & 100.00 & 0.00 & 0.00 & 100.00 \\
\hline Total & 61.54 & 28.21 & 10.26 & 100.00
\end{tabular}

\section*{Measures of association}

\section*{Example 2}
tabulate will calculate the Pearson \(\chi^{2}\) test for the independence of the rows and columns if we specify the chi2 option. Suppose that we have 1980 census data on 956 cities in the United States and wish to compare the age distribution across regions of the country. Assume that agecat is the median age in each city and that region denotes the region of the country in which the city is located.


We obtain the standard two-way table and, at the bottom, a summary of the \(\chi^{2}\) test. Stata informs us that the \(\chi^{2}\) associated with this table has 6 degrees of freedom and is 61.29 . The observed differences are significant.

The table is, perhaps, easier to understand if we suppress the frequencies and print just the row percentages:
\begin{tabular}{l}
. tabulate region agecat, row nofreq chi2 \\
\begin{tabular}{r} 
Census \\
region
\end{tabular} \\
\hline NE \\
Age category \\
N Cntrl
\end{tabular}

\section*{Example 3}

We have data on dose level and outcome for a set of patients and wish to evaluate the association between the two variables. We can obtain all the association measures by specifying the all and exact options:
```

. use https://www.stata-press.com/data/r18/dose
. tabulate dose function, all exact
Enumerating sample-space combinations:
stage 3: enumerations = 1
stage 2: enumerations = 9
stage 1: enumerations = 0

|  | Function |  |  |  |
| ---: | ---: | :---: | ---: | ---: |
| Dosage | $<1 \mathrm{hr}$ | 1 to 4 | $4+$ | Total |
| 1/day | 20 | 10 | 2 | 32 |
| 2/day | 16 | 12 | 4 | 32 |
| 3/day | 10 | 16 | 6 | 32 |
| Total | 46 | 38 | 12 | 96 |
| Pearson chi2(4) | $=6.7780$ | $\operatorname{Pr}=0.148$ |  |  |
| Likelihood-ratio chi2(4) | $=6.9844$ | $\operatorname{Pr}=0.137$ |  |  |
| Cramér's $V$ | $=0.1879$ |  |  |  |
| gamma | $=0.3689$ | ASE $=0.129$ |  |  |
| Kendall's tau-b | $=0.2378$ | ASE $=0.086$ |  |  |
| Fisher's exact | $=$ |  | 0.145 |  |

```

We find evidence of association but not enough to be truly convincing.
If we had not also specified the exact option, we would not have obtained Fisher's exact test. Stata can calculate this statistic both for \(2 \times 2\) tables and for \(r \times c\). For \(2 \times 2\) tables, the calculation is almost instant. On more general tables, however, the calculation can take longer.

We carefully constructed our example so that all would be meaningful. Kendall's \(\tau_{b}\) and Goodman and Kruskal's gamma are relevant only when both dimensions of the table can be ordered, say, from low to high or from worst to best. The other statistics, however, are always applicable.

\section*{- Technical note}

Be careful when attempting to compute the \(p\)-value for Fisher's exact test because the number of tables that contribute to the \(p\)-value can be extremely large and a solution may not be feasible. The errors that are indicative of this situation are errors 910, exceeded memory limitations, and 1401, integer overflow due to large row-margin frequencies. If execution terminates because of memory limitations, use exact (2) to permit the algorithm to consume twice the memory, exact (3) for three times the memory, etc. The default memory usage should be sufficient for reasonable tables.

\section*{N -way tables}

If you need more than two-way tables, your best alternative to is use table, not tabulate; see [R] table.

The technical note below shows you how to use tabulate to create a sequence of two-way tables that together form, in effect, a three-way table, but using table is easy and produces prettier results:
```

. use https://www.stata-press.com/data/r18/birthcat
(City data)

```
. table (agecat birthcat) (region), nototals
\begin{tabular}{|c|c|c|c|c|}
\hline & NE & \begin{tabular}{l}
Census \\
N Cntrl
\end{tabular} & region South & West \\
\hline \multicolumn{5}{|l|}{\multirow[t]{2}{*}{Age category}} \\
\hline & & & \multicolumn{2}{|c|}{19-29} \\
\hline \multicolumn{5}{|l|}{Birth-rate category} \\
\hline 29-136 & 11 & 23 & 11 & 11 \\
\hline 137-195 & 31 & 97 & 65 & 46 \\
\hline 196-529 & 4 & 38 & 59 & 91 \\
\hline \multicolumn{5}{|l|}{30-34} \\
\hline \multicolumn{5}{|l|}{Birth-rate category} \\
\hline 29-136 & 34 & 27 & 10 & 8 \\
\hline 137-195 & 48 & 58 & 45 & 42 \\
\hline 196-529 & 1 & 3 & 12 & 21 \\
\hline \multicolumn{5}{|l|}{35+} \\
\hline \multicolumn{5}{|l|}{Birth-rate category} \\
\hline 29-136 & 34 & 26 & 27 & 18 \\
\hline 137-195 & 3 & 4 & 7 & 4 \\
\hline 196-529 & & & 4 & \\
\hline
\end{tabular}

\section*{- Technical note}

We can make \(n\)-way tables by combining the by varlist: prefix with tabulate. Continuing with the dataset of 956 cities, say that we want to make a table of age category by birth-rate category by region of the country. The birth-rate category variable is named birthcat in our dataset. To make separate tables for each age category, we would type
. by agecat, sort: tabulate birthcat region
\begin{tabular}{l|rrrr|r}
\(\rightarrow\) agecat \(=19-29\) \\
\begin{tabular}{c} 
Birth-rate \\
category
\end{tabular} & NE & \begin{tabular}{c} 
Census \\
N Cntrl
\end{tabular} & Sogion \\
South & West & Total \\
\hline \(29-136\) & 11 & 23 & 11 & 11 & 56 \\
\(137-195\) & 31 & 97 & 65 & 46 & 239 \\
\(196-529\) & 4 & 38 & 59 & 91 & 192 \\
\hline Total & 46 & 158 & 135 & 148 & 487
\end{tabular}
\begin{tabular}{|c|c|c|c|c|c|}
\hline Birth-rate category & \multicolumn{4}{|c|}{Census region} & Total \\
\hline 29-136 & 34 & 27 & 10 & 8 & 79 \\
\hline 137-195 & 48 & 58 & 45 & 42 & 193 \\
\hline 196-529 & 1 & 3 & 12 & 21 & 37 \\
\hline Total & 83 & 88 & 67 & 71 & 309 \\
\hline
\end{tabular}
\begin{tabular}{l|rrrr|r}
\(->\) agecat \(=35+\) \\
\begin{tabular}{r} 
Birth-rate \\
category
\end{tabular} & NE & N Cntrl & Census region \\
South & West & Total \\
\hline \(29-136\) & 34 & 26 & 27 & 18 & 105 \\
\(137-195\) & 3 & 4 & 7 & 4 & 18 \\
\(196-529\) & 0 & 0 & 4 & 0 & 4 \\
\hline Total & 37 & 30 & 38 & 22 & 127
\end{tabular}

\section*{Weighted data}

\section*{> Example 4}
tabulate can process weighted as well as unweighted data. As with all Stata commands, we indicate the weight by specifying the [weight] modifier; see [U] 11.1.6 weight.

Continuing with our dataset of 956 cities, we also have a variable called pop, the population of each city. We can make a table of region by age category, weighted by population, by typing
. tabulate region agecat [fweight=pop]
\begin{tabular}{r|rrr|r}
\begin{tabular}{c} 
Census \\
region
\end{tabular} & \(19-29\) & \begin{tabular}{r} 
Age category \\
\(30-34\)
\end{tabular} & \(35+\) & Total \\
\hline NE & 4721387 & 10421387 & 5323610 & 20466384 \\
N Cntrl & 16901550 & 8964756 & 4015593 & 29881899 \\
South & 13894254 & 7686531 & 4141863 & 25722648 \\
West & 16698276 & 7755255 & 2375118 & 26828649 \\
\hline Total & 52215467 & 34827929 & 15856184 & 102899580
\end{tabular}

If we specify the cell, column, or row options, they will also be appropriately weighted. Below, we repeat the table, suppressing the counts and substituting row percentages:
\begin{tabular}{r|rrr|r}
. tabulate region agecat [fweight=pop], nofreq row \\
\begin{tabular}{r} 
Census \\
region
\end{tabular} & \(19-29\) & \begin{tabular}{c} 
Age category \\
\(30-34\)
\end{tabular} & \(35+\) & Total \\
\hline NE & 23.07 & 50.92 & 26.01 & 100.00 \\
N Cntrl & 56.56 & 30.00 & 13.44 & 100.00 \\
South & 54.02 & 29.88 & 16.10 & 100.00 \\
West & 62.24 & 28.91 & 8.85 & 100.00 \\
\hline Total & 50.74 & 33.85 & 15.41 & 100.00
\end{tabular}

\section*{Tables with immediate data}

\section*{\(>\) Example 5}
tabi ignores the dataset in memory and uses as the table the values that we specify on the command line:


We may specify any of the options of tabulate and are not limited to \(2 \times 2\) tables:
```

. tabi 30 18 38 \ 13 7 22, chi2 exact
Enumerating sample-space combinations:
stage 3: enumerations = 1
stage 2: enumerations = 3
stage 1: enumerations = 0

```
\begin{tabular}{r|rrr|r} 
row & 1 & col & 3 & Total \\
\hline 1 & 30 & 18 & 38 & 86 \\
2 & 13 & 7 & 22 & 42 \\
\hline Total & 43 & 25 & 60 & 128 \\
Pearson chi2(2) \(=\) & 0.7967 & \(\operatorname{Pr}=0.671\) & \\
Fisher's exact \(=\) & & 0.707
\end{tabular}
. tabi 3013 \ 187 \ 38 22, all exact col
\begin{tabular}{|l|}
\hline Key \\
\hline frequency \\
column percentage \\
\hline
\end{tabular}
\begin{tabular}{|c|c|c|c|}
\hline \multicolumn{4}{|l|}{```
Enumerating sample-space combinations:
stage 3: enumerations = 1
stage 2: enumerations = 3
stage 1: enumerations = 0
```} \\
\hline row & \[
1
\] & 2 & Total \\
\hline 1 & \[
\begin{array}{r}
30 \\
34.88
\end{array}
\] & \[
\begin{array}{r}
13 \\
30.95
\end{array}
\] & \[
\begin{array}{r}
43 \\
33.59
\end{array}
\] \\
\hline 2 & \[
\begin{array}{r}
18 \\
20.93
\end{array}
\] & \[
\begin{array}{r}
7 \\
16.67
\end{array}
\] & \[
\begin{array}{r}
25 \\
19.53
\end{array}
\] \\
\hline 3 & \[
\begin{array}{r}
38 \\
44.19
\end{array}
\] & \[
\begin{array}{r}
22 \\
52.38
\end{array}
\] & \[
\begin{array}{r}
60 \\
46.88
\end{array}
\] \\
\hline Total & \[
\begin{array}{r}
86 \\
100.00
\end{array}
\] & \[
\begin{array}{r}
42 \\
100.00
\end{array}
\] & \[
\begin{array}{r}
128 \\
100.00
\end{array}
\] \\
\hline
\end{tabular}
```

    Pearson chi2(2) = 0.7967 Pr = 0.671
    Likelihood-ratio chi2(2) = 0.7985 Pr = 0.671
Cramér's V = 0.0789
gamma = 0.1204 ASE = 0.160
Kendall's tau-b = 0.0630 ASE = 0.084
Fisher's exact = 0.707

```

For \(2 \times 2\) tables, both one- and two-sided Fisher's exact probabilities are displayed; this is true of both tabulate and tabi. See Cumulative incidence data and Case-control data in [R] Epitab for more discussion on the relationship between one- and two-sided probabilities.

\section*{Technical note}
tabi, as with all immediate commands, leaves any data in memory undisturbed. With the replace option, however, the data in memory are replaced by the data from the table:
. tabi 3018 \} 3 8 \text { 14, replace } \(\\{\text { row }} \end{array}\)
\begin{tabular}{l|rrr|}
\cline { 2 - 3 } & row & col & pop \\
\cline { 2 - 3 } 1. & 1 & 1 & 30 \\
2. & 1 & 2 & 18 \\
3. & 2 & 1 & 38 \\
4. & 2 & 2 & 14 \\
\cline { 2 - 3 } & & &
\end{tabular}

With this dataset, you could re-create the above table by typing
• tabulate row col [fweight=pop], exact
row

\section*{tab2}
tab2 is a convenience tool. Typing
. tab2 myvar thisvar thatvar, chi2
is equivalent to typing
```

. tabulate myvar thisvar, chi2

```
. tabulate myvar thatvar, chi2
. tabulate thisvar thatvar, chi2

\section*{Video examples}

Pearson's chi-squared and Fisher's exact test in Stata
Tables and cross-tabulations in Stata
Cross-tabulations and chi-squared tests calculator

\section*{Stored results}
tabulate, tab2, and tabi store the following in \(r()\) :
Scalars
\begin{tabular}{llll}
\(r(N)\) & number of observations & \(r\left(p \_e x a c t\right)\) & Fisher's exact \(p\) \\
\(r(r)\) & number of rows & \(r\left(c h i 2 \_l r\right)\) & likelihood-ratio \(\chi^{2}\) \\
\(r(c)\) & number of columns & \(r\left(p \_l r\right)\) & \(p\)-value for likelihood-ratio test \\
\(r(c h i 2)\) & Pearson's \(\chi^{2}\) test & \(r(\) CramersV) & Cramér's V \\
\(r(p)\) & \(p\)-value for of Pearson's \(\chi^{2}\) test & \(r(\) ase_gam \()\) & ASE of gamma \\
\(r(\) gamma \()\) & gamma & \(r(\) ase_taub \()\) & ASE of \(\tau_{\mathrm{b}}\) \\
\(r\left(\mathrm{p} 1 \_\right.\)exact \()\) & one-sided Fisher's exact \(p\) & \(r(\) taub \()\) & \(\tau_{\mathrm{b}}\)
\end{tabular}
\(r\) (p1_exact) is defined only for \(2 \times 2\) tables. Also, the matrow(), matcol(), and matcell() options allow you to obtain the row values, column values, and frequencies, respectively.

\section*{Methods and formulas}

Let \(n_{i j}, i=1, \ldots, I\) and \(j=1, \ldots, J\), be the number of observations in the \(i\) th row and \(j\) th column. If the data are not weighted, \(n_{i j}\) is just a count. If the data are weighted, \(n_{i j}\) is the sum of the weights of all data corresponding to the \((i, j)\) cell.

Define the row and column marginals as
\[
n_{i}=\sum_{j=1}^{J} n_{i j} \quad n_{\cdot j}=\sum_{i=1}^{I} n_{i j}
\]
and let \(n=\sum_{i} \sum_{j} n_{i j}\) be the overall sum. Also, define the concordance and discordance as
\[
A_{i j}=\sum_{k>i} \sum_{l>j} n_{k l}+\sum_{k<i} \sum_{l<j} n_{k l} \quad D_{i j}=\sum_{k>i} \sum_{l<j} n_{k l}+\sum_{k<i} \sum_{l>j} n_{k l}
\]
along with twice the number of concordances \(P=\sum_{i} \sum_{j} n_{i j} A_{i j}\) and twice the number of discordances \(Q=\sum_{i} \sum_{j} n_{i j} D_{i j}\).

The Pearson \(\chi^{2}\) statistic with \((I-1)(J-1)\) degrees of freedom (so called because it is based on Pearson (1900); see Conover [1999, 240] and Fienberg [1980, 9]) is defined as
\[
X^{2}=\sum_{i} \sum_{j} \frac{\left(n_{i j}-m_{i j}\right)^{2}}{m_{i j}}
\]
where \(m_{i j}=n_{i \cdot n \cdot j} / n\).
The likelihood-ratio \(\chi^{2}\) statistic with \((I-1)(J-1)\) degrees of freedom (Fienberg 1980, 40) is defined as
\[
G^{2}=2 \sum_{i} \sum_{j} n_{i j} \ln \left(n_{i j} / m_{i j}\right)
\]

Cramér's \(V\) (Cramér 1946) is a measure of association designed so that the attainable upper bound is 1 . For \(2 \times 2\) tables, \(-1 \leq V \leq 1\), and otherwise, \(0 \leq V \leq 1\).
\[
V= \begin{cases}\left(n_{11} n_{22}-n_{12} n_{21}\right) /\left(n_{1} \cdot n_{2} \cdot n_{\cdot 1} n_{\cdot 2}\right)^{1 / 2} & \text { for } 2 \times 2 \\ \left\{\left(X^{2} / n\right) / \min (I-1, J-1)\right\}^{1 / 2} & \text { otherwise }\end{cases}
\]

Gamma (Goodman and Kruskal 1954, 1959, 1963, 1972; also see Agresti [2010,186-188]) ignores tied pairs and is based only on the number of concordant and discordant pairs of observations, \(-1 \leq \gamma \leq 1\),
\[
\gamma=(P-Q) /(P+Q)
\]
with asymptotic variance
\[
16 \sum_{i} \sum_{j} n_{i j}\left(Q A_{i j}-P D_{i j}\right)^{2} /(P+Q)^{4}
\]

Kendall's \(\tau_{b}\) (Kendall 1945; also see Agresti 2010, 188-189), \(-1 \leq \tau_{b} \leq 1\), is similar to gamma, except that it uses a correction for ties,
\[
\tau_{b}=(P-Q) /\left(w_{r} w_{c}\right)^{1 / 2}
\]
with asymptotic variance
\[
\frac{\sum_{i} \sum_{j} n_{i j}\left(2 w_{r} w_{c} d_{i j}+\tau_{b} v_{i j}\right)^{2}-n^{3} \tau_{b}^{2}\left(w_{r}+w_{c}\right)^{2}}{\left(w_{r} w_{c}\right)^{4}}
\]
where
\[
\begin{aligned}
w_{r} & =n^{2}-\sum_{i} n_{i \cdot}^{2} \\
w_{c} & =n^{2}-\sum_{j} n_{\cdot j}^{2} \\
d_{i j} & =A_{i j}-D_{i j} \\
v_{i j} & =n_{i} \cdot w_{c}+n_{\cdot j} w_{r}
\end{aligned}
\]

Fisher's exact test (Fisher 1935; Finney 1948; see Zelterman and Louis [1992, 293-301] for the \(2 \times 2\) case) yields the probability of observing a table that gives at least as much evidence of association as the one actually observed under the assumption of no association. Holding row and column marginals fixed, the hypergeometric probability \(P\) of every possible table \(A\) is computed, and the
\[
P=\sum_{T \in A} \operatorname{Pr}(T)
\]
where \(A\) is the set of all tables with the same marginals as the observed table, \(T^{*}\), such that \(\operatorname{Pr}(T) \leq \operatorname{Pr}\left(T^{*}\right)\). For \(2 \times 2\) tables, the one-sided probability is calculated by further restricting \(A\) to tables in the same tail as \(T^{*}\). The first algorithm extending this calculation to \(r \times c\) tables was Pagano and Halvorsen (1981); the one implemented here is the FEXACT algorithm by Mehta and Patel (1986). This is a search-tree clipping method originally published by Mehta and Patel (1983) with further refinements by Joe (1988) and Clarkson, Fan, and Joe (1993). Fisher's exact test is a permutation test. For more information on permutation tests, see Good (2005 and 2006) and Pesarin (2001).

\section*{References}

Agresti, A. 2010. Analysis of Ordinal Categorical Data. 2nd ed. Hoboken, NJ: Wiley.
Campbell, M. J., D. Machin, and S. J. Walters. 2007. Medical Statistics: A Textbook for the Health Sciences. 4th ed. Chichester, UK: Wiley.
Clarkson, D. B., Y.-A. Fan, and H. Joe. 1993. A remark on Algorithm 643: FEXACT: An algorithm for performing Fisher's exact test in \(r \times c\) contingency tables. ACM Transactions on Mathematical Software 19: 484-488. https://doi.org/10.1145/168173.168412.
Conover, W. J. 1999. Practical Nonparametric Statistics. 3rd ed. New York: Wiley.
Cox, N. J. 2009. Speaking Stata: I. J. Good and quasi-Bayes smoothing of categorical frequencies. Stata Journal 9: 306-314.

Cramér, H. 1946. Mathematical Methods of Statistics. Princeton, NJ: Princeton University Press.
Donath, S. 2018. baselinetable: A command for creating one- and two-way tables of summary statistics. Stata Journal 18: 327-344.
Fienberg, S. E. 1980. The Analysis of Cross-Classified Categorical Data. 2nd ed. Cambridge, MA: MIT Press.
Finney, D. J. 1948. The Fisher-Yates test of significance in \(2 \times 2\) contingency tables. Biometrika 35: 145-156. https://doi.org/10.2307/2332635.
Fisher, R. A. 1935. The logic of inductive inference. Journal of the Royal Statistical Society 98: 39-82. https://doi.org/10.2307/2342435.

Good, P. I. 2005. Permutation, Parametric, and Bootstrap Tests of Hypotheses: A Practical Guide to Resampling Methods for Testing Hypotheses. 3rd ed. New York: Springer.
——. 2006. Resampling Methods: A Practical Guide to Data Analysis. 3rd ed. Boston: Birkhäuser.
Goodman, L. A., and W. H. Kruskal. 1954. Measures of association for cross classifications. Journal of the American Statistical Association 49: 732-764. https://doi.org/10.1080/01621459.1954.10501231.
——. 1959. Measures of association for cross classifications II: Further discussion and references. Journal of the American Statistical Association 54: 123-163. https://doi.org/10.1080/01621459.1959.10501503.
- 1963. Measures of association for cross classifications III: Approximate sampling theory. Journal of the American Statistical Association 58: 310-364. https://doi.org/10.2307/2283271.
-_ 1972. Measures of association for cross classifications IV: Simplification of asymptotic variances. Journal of the American Statistical Association 67: 415-421. https://doi.org/10.2307/2284396.

Harrison, D. A. 2006. Stata tip 34: Tabulation by listing. Stata Journal 6: 425-427.
Jann, B. 2008. Multinomial goodness-of-fit: Large-sample tests with survey design correction and exact tests for small samples. Stata Journal 8: 147-169.
Joe, H. 1988. Extreme probabilities for contingency tables under row and column independence with application to Fisher's exact test. Communications in Statistics-Theory and Methods 17: 3677-3685. https://doi.org/10.1080/03610928808829827.
Kendall, M. G. 1945. The treatment of ties in rank problems. Biometrika 33: 239-251. https://doi.org/10.2307/2332303.
Longest, K. C. 2020. Using Stata for Quantitative Analysis. 3rd ed. Thousand Oaks, CA: Sage.
Mehta, C. R., and N. R. Patel. 1983. A network algorithm for performing Fisher's exact test in \(r \times c\) contingency tables. Journal of the American Statistical Association 78: 427-434. https://doi.org/10.1080/01621459.1983.10477989.
——. 1986. Algorithm 643 FEXACT: A FORTRAN subroutine for Fisher's exact test on unordered \(r \times c\) contingency tables. ACM Transactions on Mathematical Software 12: 154-161. https://doi.org/10.1145/6497.214326.
Newson, R. B. 2002. Parameters behind "nonparametric" statistics: Kendall's tau, Somers' D and median differences. Stata Journal 2: 45-64.
Pagano, M., and K. T. Halvorsen. 1981. An algorithm for finding the exact significance levels of \(r \times c\) contingency tables. Journal of the American Statistical Association 76: 931-934. https://doi.org/10.2307/2287590.

Pearson, K. 1900. On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science 50(302): 157-175. https://doi.org/10.1080/14786440009463897.
Pesarin, F. 2001. Multivariate Permutation Tests: With Applications in Biostatistics. Chichester, UK: Wiley.
Zelterman, D., and T. A. Louis. 1992. Contingency tables in medical studies. In Medical Uses of Statistics, 2nd ed, ed. J. C. Bailar, III and C. F. Mosteller, 293-310. Boston: Dekker.

\section*{Also see}
[R] Epitab - Tables for epidemiologists
[R] table - Table of frequencies, summaries, and command results
[R] table twoway - Two-way tabulation
[R] tabstat - Compact table of summary statistics
[R] tabulate oneway - One-way table of frequencies
[R] tabulate, summarize() - One- and two-way tables of summary statistics
[D] collapse - Make dataset of summary statistics
[SVY] svy: tabulate oneway - One-way tables for survey data
[SVY] svy: tabulate twoway - Two-way tables for survey data
[XT] xttab - Tabulate xt data
[U] 12.6.3 Value labels
[U] 26 Working with categorical data and factor variables

\section*{Title}
tabulate, summarize() - One- and two-way tables of summary statistics
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Also see &
\end{tabular}

\section*{Description}
tabulate, summarize() produces one- and two-way tables (breakdowns) of means and standard deviations. See \([R]\) tabulate oneway and \([R]\) tabulate twoway for one- and two-way frequency tables. See [R] table for a more flexible command that produces one-, two-, and \(n\)-way tables of frequencies and a wide variety of summary statistics. table is better, but tabulate, summarize() is faster. Also see \([R]\) tabstat for yet another alternative.

\section*{Quick start}

Tabulation of \(v 1\), reporting means and standard deviations of \(x\) and frequencies
tabulate v 1 , summarize( x )
Same as above, but report summary statistics for the two-way tabulation of v1 and v2
tabulate v 1 v 2 , summarize ( x )
Weighted summary statistics using frequency weight wvar
tabulate v1 v2 [fweight=wvar], summarize(x)
Report only the mean of \(x\) for each group
tabulate v 1 v 2 , summarize( x ) means
Do not report standard deviations
tabulate v 1 v 2 , summarize(x) nostandard
Show numeric values of v 1 and v 2 rather than value labels
tabulate v1 v2, summarize(x) nolabel

\section*{Menu}

Statistics \(>\) Summaries, tables, and tests \(>\) Other tables \(>\) Table of means, std. dev., and frequencies

\section*{Syntax}
\(\underline{\text { tabulate }}^{\text {varname }}{ }_{1}\left[\right.\) varname \(\left._{2}\right][\) if \(][\) in \(][\) weight \(][\), options \(]\)
options
Description
Main
summarize ( varname \(_{3}\) )
[no]means
[no] standard
[no] freq
[no] obs
nolabel
wrap
missing
report summary statistics for varname \(_{3}\)
include or suppress means
include or suppress standard deviations
include or suppress frequencies
include or suppress number of observations
show numeric codes, not labels
do not break wide tables
treat missing values of varname \(_{1}\) and varname \({ }_{2}\) as categories
by and collect are allowed; see [U] 11.1.10 Prefix commands.
aweights and fweights are allowed; see [U] 11.1.6 weight.

\section*{Options}
\(\qquad\) Main
summarize ( varname \(_{3}\) ) identifies the name of the variable for which summary statistics are to be reported. If you do not specify this option, a table of frequencies is produced; see [R] tabulate oneway and \([R]\) tabulate twoway. The description here concerns tabulate when this option is specified.
[no]means includes or suppresses only the means from the table.
The summarize() table normally includes the mean, standard deviation, frequency, and, if the data are weighted, number of observations. Individual elements of the table may be included or suppressed by the [no]means, [no]standard, [no]freq, and [no]obs options. For example, typing
```

. tabulate category, summarize(myvar) means standard

```
produces a summary table by category containing only the means and standard deviations of myvar. You could also achieve the same result by typing
```

. tabulate category, summarize(myvar) nofreq

```
[no]standard includes or suppresses only the standard deviations from the table; see [no]means option above.
[no]freq includes or suppresses only the frequencies from the table; see [no]means option above.
[no]obs includes or suppresses only the reported number of observations from the table. If the data are not weighted, the number of observations is identical to the frequency, and by default only the frequency is reported. If the data are weighted, the frequency refers to the sum of the weights. See [no]means option above.
nolabel causes the numeric codes to be displayed rather than the label values.
wrap requests that no action be taken on wide tables to make them readable. Unless wrap is specified, wide tables are broken into pieces to enhance readability.
missing requests that missing values of varname \({ }_{1}\) and varname \(_{2}\) be treated as categories rather than as observations to be omitted from the analysis.

\section*{Remarks and examples}
tabulate with the summarize() option produces one- and two-way tables of summary statistics. When combined with the by prefix, it can produce \(n\)-way tables as well.

Remarks are presented under the following headings:

> One-way tables
> Two-way tables

\section*{One-way tables}

\section*{> Example 1}

We have data on 74 automobiles. Included in our dataset are the variables foreign, which marks domestic and foreign cars, and mpg, the car's mileage rating. Typing tabulate foreign displays a breakdown of the number of observations we have by the values of the foreign variable.
. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. tabulate foreign
Car origin Freq. \(\quad\) Percent \(\quad\) Cum.

We discover that we have 52 domestic cars and 22 foreign cars in our dataset. If we add the summarize (varname) option, however, tabulate produces a table of summary statistics for varname:
\begin{tabular}{r|rrr}
. tabulate foreign, summarize(mpg) & \\
Car origin & \begin{tabular}{r} 
Summary of Mileage (mpg) \\
Mean
\end{tabular} & \begin{tabular}{r} 
Std. dev.
\end{tabular} & Freq. \\
\hline Domestic & 19.826923 & 4.7432972 & 52 \\
Foreign & 24.772727 & 6.6111869 & 22 \\
\hline Total & 21.297297 & 5.7855032 & 74
\end{tabular}

We also discover that the average gas mileage for domestic cars is about 20 mpg and the average foreign is almost 25 mpg . Overall, the average is 21 mpg in our dataset.

\section*{- Technical note}

We might now wonder if the difference in gas mileage between foreign and domestic cars is statistically significant. We can use the oneway command to find out; see [R] oneway. To obtain an analysis-of-variance table of mpg on foreign, we type
. oneway mpg foreign
\begin{tabular}{|c|c|c|c|c|c|}
\hline \multicolumn{6}{|l|}{Analysis of variance \(\quad \begin{aligned} & \text { Af } \\ & \text { SS }\end{aligned}\)} \\
\hline Between groups & 378.153515 & 1 & 378.153515 & 13.18 & 0.0005 \\
\hline Within groups & 2065.30594 & 72 & 28.6848048 & & \\
\hline Total & 2443.45946 & 73 & 33.4720474 & & \\
\hline Bartlett's equa & riances tes & i2 & \(=3.4818\) & Prob>c & i2 \(=0.062\) \\
\hline
\end{tabular}

The \(F\) statistic is 13.18 , and the difference between foreign and domestic cars' mileage ratings is significant at the \(0.05 \%\) level.

There are several ways that we could have statistically compared mileage ratings-see, for instance, \([R]\) anova, \([R]\) oneway, \([R]\) regress, and \([R]\) ttest-but oneway seemed the most convenient.

\section*{Two-way tables}

\section*{\(>\) Example 2}
tabulate, summarize can be used to obtain two-way as well as one-way breakdowns. For instance, we obtained summary statistics on mpg decomposed by foreign by typing tabulate foreign, summarize (mpg). We can specify up to two variables before the comma:
\begin{tabular}{|c|c|c|c|}
\hline Mea & \begin{tabular}{l}
, Standard \\
Car or \\
Domestic
\end{tabular} & \begin{tabular}{l}
Deviation igin \\
Foreign
\end{tabular} & \begin{tabular}{l}
and Freque \\
Total
\end{tabular} \\
\hline \multirow[t]{3}{*}{2530} & \multirow[t]{2}{*}{\[
\begin{aligned}
& 28.285714 \\
& 3.0937725
\end{aligned}
\]} & \multirow[t]{3}{*}{\[
\begin{array}{r}
27.0625 \\
5.9829619 \\
16
\end{array}
\]} & \multirow[t]{3}{*}{\[
\begin{array}{r}
27.434783 \\
5.2295149 \\
23
\end{array}
\]} \\
\hline & & & \\
\hline & 7 & & \\
\hline \multirow[t]{3}{*}{3300} & 21.75 & 19.6 & 21.238095 \\
\hline & 2.4083189 & 3.4351128 & 2.7550819 \\
\hline & 16 & 5 & 21 \\
\hline \multirow[t]{3}{*}{4070} & 17.26087 & 14 & 17.125 \\
\hline & 1.8639497 & 0 & 1.9406969 \\
\hline & 23 & 1 & 24 \\
\hline \multirow[t]{3}{*}{4840} & 14.666667 & . & 14.666667 \\
\hline & 3.32666 & . & 3.32666 \\
\hline & 6 & 0 & 6 \\
\hline \multirow[t]{3}{*}{Total} & 19.826923 & 24.772727 & 21.297297 \\
\hline & 4.7432972 & 6.6111869 & 5.7855032 \\
\hline & 52 & 22 & 74 \\
\hline
\end{tabular}

In addition to the means, standard deviations, and frequencies for each weight-mileage cell, also reported are the summary statistics by weight, by mileage, and overall. For instance, the last row of the table reveals that the average mileage of domestic cars is 19.83 and that of foreign cars is 24.77 -domestic cars yield poorer mileage than foreign cars. But we now see that domestic cars yield better gas mileage within weight class - the reason domestic cars yield poorer gas mileage is because they are, on average, heavier.

\section*{> Example 3}

If we do not specify the statistics to be included in a table, tabulate reports the mean, standard deviation, and frequency. We can specify the statistics that we want to see using the means, standard, and freq options:
. tabulate wgtcat foreign, summarize(mpg) means
Means of Mileage (mpg)
\begin{tabular}{r|rr|r}
\multirow{2}{*}{ wgtcat } & \multicolumn{3}{|c}{ Car origin } \\
& Domestic & Foreign & Total \\
\hline 2530 & 28.285714 & 27.0625 & 27.434783 \\
3300 & 21.75 & 19.6 & 21.238095 \\
4070 & 17.26087 & 14 & 17.125 \\
4840 & 14.666667 &. & 14.666667 \\
\hline Total & 19.826923 & 24.772727 & 21.297297
\end{tabular}

When we specify one or more of the means, standard, and freq options, only those statistics are displayed. Thus, we could obtain a table containing just the means and standard deviations by typing means standard after the summarize (mpg) option. We can also suppress selected statistics by placing no in front of the option name. Another way of obtaining only the means and standard deviations is to add the nofreq option:
\begin{tabular}{|c|c|c|c|}
\hline wgtcat & Means and Car or Domestic & \begin{tabular}{l}
Standard igin \\
Foreign
\end{tabular} & \begin{tabular}{l}
viations o \\
Total
\end{tabular} \\
\hline \multirow[t]{2}{*}{2530} & 28.285714 & 27.0625 & 27.434783 \\
\hline & 3.0937725 & 5.9829619 & 5.2295149 \\
\hline \multirow[t]{2}{*}{3300} & 21.75 & 19.6 & 21.238095 \\
\hline & 2.4083189 & 3.4351128 & 2.7550819 \\
\hline \multirow[t]{2}{*}{4070} & 17.26087 & 14 & 17.125 \\
\hline & 1.8639497 & 0 & 1.9406969 \\
\hline \multirow[t]{2}{*}{4840} & 14.666667 & . & 14.666667 \\
\hline & 3.32666 & . & 3.32666 \\
\hline \multirow[t]{2}{*}{Total} & 19.826923 & 24.772727 & 21.297297 \\
\hline & 4.7432972 & 6.6111869 & 5.7855032 \\
\hline
\end{tabular}

\section*{Also see}
[R] table - Table of frequencies, summaries, and command results
[R] table summary - Table of summary statistics
[R] tabstat - Compact table of summary statistics
[R] tabulate oneway - One-way table of frequencies
[R] tabulate twoway - Two-way table of frequencies
[D] collapse - Make dataset of summary statistics
[SVY] svy: tabulate oneway - One-way tables for survey data
[SVY] svy: tabulate twoway - Two-way tables for survey data
[U] 12.6 Dataset, variable, and value labels
[U] 26 Working with categorical data and factor variables

\section*{Title}
test - Test linear hypotheses after estimation
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options for testparm & Options for test & Remarks and examples & Stored results \\
Methods and formulas & Acknowledgment & References & Also see
\end{tabular}

\section*{Description}
test performs Wald tests of simple and composite linear hypotheses about the parameters of the most recently fit model.
test supports svy estimators (see [SVY] svy estimation), carrying out an adjusted Wald test by default in such cases. test can be used with svy estimation results, see [SVY] svy postestimation.
testparm provides a useful alternative to test that permits varlist rather than a list of coefficients (which is often nothing more than a list of variables), allowing the use of standard Stata notation, including ' - ' and ' \(*\) ', which are given the expression interpretation by test.
test and testparm perform Wald tests. For likelihood-ratio tests, see [R] Irtest. For Wald-type tests of nonlinear hypotheses, see \([R]\) testnl. To display estimates for one-dimensional linear or nonlinear expressions of coefficients, see \([R]\) lincom and \([R]\) nlcom.

See \([R]\) anova postestimation for additional test syntax allowed after anova.
See [MV] manova postestimation for additional test syntax allowed after manova.

\section*{Quick start}

Linear tests after single-equation models
Joint test that the coefficients on x 1 and x 2 are equal to 0
test x 1 x 2
Joint test that coefficients on factor indicators \(2 . a\) and 3 .a are equal to 0
test 2.a 3.a
Test that coefficients on indicators 2.a and 3.a are equal
```

test 2.a = 3.a

```

Joint test that coefficients on indicators 1.a, 2.a, and 3.a are all equal
test (1.a=2.a) (1.a=3.a)
Same as above
test 1. \(\mathrm{a}=2 . \mathrm{a}=3 . \mathrm{a}\)
Same as above, but add separate tests for each pairing
test 1. \(\mathrm{a}=2 . \mathrm{a}=3 . \mathrm{a}\), mtest
Same as above, but with \(p\)-values adjusted for multiple comparisons using Šidák's method
```

test (1.a=2.a) (1.a=3.a), mtest(sidak)

```

Test that the sum of the coefficients for x 1 and x 2 is equal to 4
test \(\mathrm{x} 1+\mathrm{x} 2=4\)
Test the equality of two linear expressions involving coefficients on \(x 1\) and x 2
test \(2 * x 1=3 * x 2\)

\section*{Shorthand varlist notation}

Joint test that all coefficients on the indicators for a are equal to 0
testparm i.a
Joint test that all coefficients on the indicators for a and b are equal to 0
testparm i.a i.b
Joint test that all coefficients associated with the interaction of factor variables a and b are equal to 0 testparm i.a\#i.b

Joint test that the coefficients on all variables \(x^{*}\) are equal to 0
testparm x*

\section*{Linear tests after multiple-equation models}

Joint test that the coefficient on x 1 is equal to 0 in all equations
```

test x1

```

Joint test that the coefficients for x 1 and x 2 are equal to 0 in equation y 3
test [y3]x1 [y3]x2
Test that the coefficients for x 1 are equal in equations y 1 and y 3
test [y1]x1=[y3]x1
Same as above
test [y1=y3]: x1
Joint test of the equality of coefficients for x 1 and x 2 across equations y 1 and y 3
test [y1=y3]: x1 x2
Add coefficients for x 1 and x 2 from equation y 4 to test
test [y1=y3=y4]: x1 x2
Test that all coefficients in the equation for y1 are equal to those in the equation for y2
test [y1=y2]
Same as above, but only for coefficients on variables common to both equations
test [y1=y2], common

Shorthand varlist notation
Joint test that all coefficients on the indicators for a are 0 in all equations testparm i.a

Joint test that all coefficients on the indicators for a are equal to each other in the first equation testparm i.a, equal

Same as above, but for the equation for y4
testparm i.a, equal equation(y4)
Joint test that the coefficients on the indicators for a and b are equal to 0 in all equations testparm i.a i.b

Joint test that all coefficients associated with the interaction of factors \(a\) and \(b\) are 0 testparm i.a\#i.b

\section*{Menu}

Statistics > Postestimation

\section*{Syntax}

Basic syntax
```

test coeflist
test $\exp =\exp [=\ldots]$
test [eqno] [: coeflist]
test $[$ eqno $=$ eqno $[=. .]].[:$ coeflist $]$
testparm varlist [, testparm_options]

```

Full syntax
test (spec) [(spec) ...] [, test_options]
testparm_options Description
equal hypothesize that the coefficients are equal to each other
equation(eqno) specify equation name or number for which the hypothesis is tested
nosvyadjust
compute unadjusted Wald tests for survey results
\(\mathrm{df}(\#) \quad\) use \(F\) distribution with \# denominator degrees of freedom for the reference distribution of the test statistic; for survey data, \# specifies the design degrees of freedom unless nosvyadjust is specified
\(\mathrm{df}(\#)\) does not appear in the dialog box.
test_options Description
Options
mtest [(opt)]
coef
accumulate
notest
common constant
nosvyadjust
minimum
test each condition separately
report estimated constrained coefficients
test hypothesis jointly with previously tested hypotheses
suppress the output
test only variables common to all the equations
include the constant in coefficients to be tested
compute unadjusted Wald tests for survey results
perform test with the constant, drop terms until the test becomes nonsingular, and test without the constant on the remaining terms; highly technical
matvlc (matname) save the variance-covariance matrix; programmer's option
df (\#)
use \(F\) distribution with \# denominator degrees of freedom for the reference distribution of the test statistic; for survey data, \# specifies the design degrees of freedom unless nosvyadjust is specified

\footnotetext{
matvlc(matname) and \(\mathrm{df}(\#)\) do not appear in the dialog box.
}
coeflist and varlist may contain factor variables and time-series operators; see [U] 11.4.3 Factor variables and
[U] 11.4.4 Time-series varlists.
collect is allowed with test; see [U] 11.1.10 Prefix commands.

Syntax 1 tests that coefficients are 0 .
Syntax 2 tests that linear expressions are equal.
Syntax 3 tests that coefficients in eqno are 0.
Syntax 4 tests equality of coefficients between equations.
spec is one of
\[
\begin{aligned}
& \text { coeflist } \\
& \exp =\exp [=\exp ] \\
& {[\text { eqno }][: \text { coeflist }]} \\
& {\left[\text { eqno }{ }_{1}=\text { eqno }_{2}[=\ldots]\right][: \text { coeflist }]}
\end{aligned}
\]
coeflist is
\[
\begin{aligned}
& \operatorname{coef}[\operatorname{coef} \ldots] \\
& {[\text { eqno }] \operatorname{coef}[[\text { eqno }] \text { coef } \ldots]} \\
& \left.[\text { eqno }] \_\mathrm{b}[\text { coef }][\text { eqno }] \ldots \mathrm{b}[\text { coef }] \ldots\right]
\end{aligned}
\]
\(\exp\) is a linear expression containing
\[
\begin{aligned}
& \text { coef } \\
& \mathrm{-b}[\text { coef }] \\
& \mathrm{b}[\text { eqno:coef }] \\
& {[\text { [eqno }] \text { coef }} \\
& {[\text { eqno _b }[\text { coef }]}
\end{aligned}
\]
eqno is
\#\#
name
coef identifies a coefficient in the model. coef is typically a variable name, a level indicator, an interaction indicator, or an interaction involving continuous variables. Level indicators identify one level of a factor variable and interaction indicators identify one combination of levels of an interaction; see [U] 11.4.3 Factor variables. coef may contain time-series operators; see [U] 11.4.4 Time-series varlists.
Distinguish between [], which are to be typed, and [], which indicate optional arguments.
Although not shown in the syntax diagram, parentheses around spec are required only with multiple specifications. Also, the diagram does not show that test may be called without arguments to redisplay the results from the last test.
anova and manova (see [R] anova and [MV] manova) allow the test syntax above plus more (see [R] anova postestimation for test after anova; see [MV] manova postestimation for test after manova).

\section*{Options for testparm}
equal tests that the variables appearing in varlist, which also appear in the previously fit model, are equal to each other rather than jointly equal to zero.
equation (eqno) is relevant only for multiple-equation models, such as mvreg, mlogit, and heckman. It specifies the equation for which the all-zero or all-equal hypothesis is tested. equation(\#1) specifies that the test be conducted regarding the first equation \#1. equation(price) specifies that the test concern the equation named price.
nosvyadjust is for use with svy estimation commands; see [SVY] svy estimation. It specifies that the Wald test be carried out without the default adjustment for the design degrees of freedom. That is, the test is carried out as \(W / k \sim F(k, d)\) rather than as \((d-k+1) W /(k d) \sim F(k, d-k+1)\), where \(k=\) the dimension of the test and \(d=\) the total number of sampled PSUs minus the total number of strata. When the df() option is used, it will override the default design degrees of freedom.

The following option is available with testparm but is not shown in the dialog box:
\(\mathrm{df}(\#)\) specifies that the \(F\) distribution with \# denominator degrees of freedom be used for the reference distribution of the test statistic. The default is to use e(df_r) degrees of freedom or the \(\chi^{2}\) distribution if \(e\left(d f \_r\right)\) is missing. With survey data, \# is the design degrees of freedom unless nosvyadjust is specified.

\section*{Options for test}
\(\underset{\operatorname{mtest}[(o p t)] \text { specifies that tests be performed for each condition separately. opt specifies the method }}{ }\) for adjusting \(p\)-values for multiple testing. Valid values for opt are
\begin{tabular}{cl} 
bonferroni & Bonferroni's method \\
holm & Holm's method \\
sidak & Šidák's method \\
noadjust & no adjustment is to be made
\end{tabular}

Specifying mtest without an argument is equivalent to mtest (noadjust).
coef specifies that the constrained coefficients be displayed.
accumulate allows a hypothesis to be tested jointly with the previously tested hypotheses.
notest suppresses the output. This option is useful when you are interested only in the joint test of several hypotheses, specified in a subsequent call of test, accumulate.
common specifies that when you use the \(\left[e_{\text {eqno }}^{1}=e q n o_{2}[=\ldots]\right]\) form of spec, the variables common to the equations \(e^{2 n n o} o_{1}\), eqno \(_{2}\), etc., be tested. The default action is to complain if the equations have variables not in common.
constant specifies that _cons be included in the list of coefficients to be tested when using the \(\left[e^{2 n o} o_{1}=e^{2} n o_{2}[=. .].\right]\) or [eqno] forms of spec. The default is not to include _cons.
nosvyadjust is for use with svy estimation commands; see [SVY] svy estimation. It specifies that the Wald test be carried out without the default adjustment for the design degrees of freedom. That is, the test is carried out as \(W / k \sim F(k, d)\) rather than as \((d-k+1) W /(k d) \sim F(k, d-k+1)\), where \(k=\) the dimension of the test and \(d=\) the total number of sampled PSUs minus the total number of strata. When the df() option is used, it will override the default design degrees of freedom.
minimum is a highly technical option. It first performs the test with the constant added. If this test is singular, coefficients are dropped until the test becomes nonsingular. Then the test without the constant is performed with the remaining terms.

The following options are available with test but are not shown in the dialog box:
matvlc(matname), a programmer's option, saves the variance-covariance matrix of the linear combinations involved in the suite of tests. For the test of the linear constraints \(L b=c\), matname contains \(L \mathbf{V} L^{\prime}\), where \(\mathbf{V}\) is the estimated variance-covariance matrix of \(b\).
\(\mathrm{df}(\#)\) specifies that the \(F\) distribution with \# denominator degrees of freedom be used for the reference distribution of the test statistic. The default is to use e(df_r) degrees of freedom or the \(\chi^{2}\) distribution if \(e\left(d f \_r\right)\) is missing. With survey data, \# is the design degrees of freedom unless nosvyadjust is specified.

\section*{Remarks and examples}

Remarks are presented under the following headings:

\section*{Introductory examples}
test performs \(F\) or \(\chi^{2}\) tests of linear restrictions applied to the most recently fit model (for example, regress or svy: regress in the linear regression case; logit, stcox, svy: logit, ... in the single-equation maximum-likelihood case; and mlogit, mvreg, streg, ... in the multipleequation maximum-likelihood case). test may be used after any estimation command, although for maximum likelihood techniques, test produces a Wald test that depends only on the estimate of the covariance matrix - you may prefer to use the more computationally expensive likelihood-ratio test; see [U] 20 Estimation and postestimation commands and [R] Irtest.

There are several variations on the syntax for test. The second syntax,
\[
\text { test } \exp =\exp [=\ldots]
\]
is allowed after any form of estimation. After fitting a model of depvar on x 1 , x 2 , and x 3 , typing test \(\mathrm{x} 1+\mathrm{x} 2=\mathrm{x} 3\) tests the restriction that the coefficients on x 1 and x 2 sum to the coefficient on x 3 . The expressions can be arbitrarily complicated; for instance, typing test \(x 1+2 *(x 2+x 3)=x 2+3 * x 3\) is the same as typing test \(\mathrm{x} 1+\mathrm{x} 2=\mathrm{x} 3\).

As a convenient shorthand, test also allows you to specify equality for multiple expressions; for example, test \(\mathrm{x} 1+\mathrm{x} 2=\mathrm{x} 3+\mathrm{x} 4=\mathrm{x} 5+\mathrm{x} 6\) tests that the three specified pairwise sums of coefficients are equal.
test understands that when you type x 1 , you are referring to the coefficient on x 1 . You could also more explicitly type test \(\quad \mathrm{b}[\mathrm{x} 1]+\_\mathrm{b}[\mathrm{x} 2]=\_\mathrm{b}[\mathrm{x} 3]\); or you could test _coef \([\mathrm{x} 1]+\) _coef \([\mathrm{x} 2]=\) _coef \([\mathrm{x} 3]\), or test \([\# 1] \mathrm{x} 1+[\# 1] \times 2=[\# 1] \times 3\), or many other things because there is more than one way to refer to an estimated coefficient; see [U] 13.5 Accessing coefficients and standard errors. The shorthand involves less typing. On the other hand, you must be more explicit after estimation of multiple-equation models because there may be more than one coefficient associated
with an independent variable. You might type, for instance, test [\#2]x1+[\#2]x2=[\#2]x3 to test the constraint in equation 2 or, more readably, test [ford] x1+[ford] x2=[ford] x3, meaning that Stata will test the constraint on the equation corresponding to ford, which might be equation 2 . ford would be an equation name after, say, sureg, or, after mlogit, ford would be one of the outcomes. For mlogit, you could also type test [2] x1+[2] x2=[2] x3-note the lack of the \#-meaning not equation 2, but the equation corresponding to the numeric outcome 2. You can even test constraints across equations: test [ford] \(\mathrm{x} 1+[\) ford \(\mathrm{x} 2=[\) buick \(] \mathrm{x} 3\).

The syntax

\section*{test coeflist}
is available after all estimation commands and is a convenient way to test that multiple coefficients are zero following estimation. A coeflist can simply be a list of variable names,

> test varname [varname ...]
and it is most often specified that way. After you have fit a model of depvar on x1, x2, and x3, typing test x 1 x 3 tests that the coefficients on x 1 and x 3 are jointly zero. After multiple-equation estimation, this would test that the coefficients on \(x 1\) and \(x 3\) are zero in all equations that contain them. You can also be more explicit and type, for instance, test [ford]x1 [ford]x3 to test that the coefficients on x 1 and x 3 are zero in the equation for ford.

In the multiple-equation case, there are more alternatives. You could also test that the coefficients on x 1 and x 3 are zero in the equation for ford by typing test [ford]: x1 x3. You could test that all coefficients except the coefficient on the constant are zero in the equation for ford by typing test [ford]. You could test that the coefficients on \(x 1\) and \(x 3\) in the equation for ford are equal to the corresponding coefficients in the equation corresponding to buick by typing test[ford=buick]: x 1 x 3 . You could test that all the corresponding coefficients except the constant in three equations are equal by typing test [ford=buick=volvo].
testparm is much like the first syntax of test. Its usefulness will be demonstrated below.
The examples below use regress, but what is said applies equally after any single-equation estimation command (such as logistic). It also applies after multiple-equation estimation commands as long as references to coefficients are qualified with an equation name or number in square brackets placed before them. The convenient syntaxes for dealing with tests of many coefficients in multipleequation models are demonstrated in Special syntaxes after multiple-equation estimation below.

\section*{> Example 1: Testing for a single coefficient against zero}

We have 1980 census data on the 50 states recording the birth rate in each state (brate), the median age (medage), and the region of the country in which each state is located.

The region variable is 1 if the state is in the Northeast, 2 if the state is in the North Central, 3 if the state is in the South, and 4 if the state is in the West. We estimate the following regression:
```

. use https://www.stata-press.com/data/r18/census3
(1980 Census data by state)
. regress brate medage c.medage\#c.medage i.region

```
\begin{tabular}{r|crc} 
Source & SS & df & MS \\
\hline Model & 38803.4208 & 5 & 7760.68416 \\
Residual & 3393.39921 & 44 & 77.1227094 \\
\hline Total & 42196.82 & 49 & 861.159592
\end{tabular}
\begin{tabular}{llr} 
Number of obs & \(=\) & 50 \\
\(\mathrm{~F}(5,44)\) & \(=\) & 100.63 \\
Prob \(>\) F & \(=\) & 0.0000 \\
R-squared & \(=\) & 0.9196 \\
Adj R-squared & \(=\) & 0.9104 \\
Root MSE & \(=\) & 8.782
\end{tabular}
\begin{tabular}{c|cccccc}
\hline brate & Coefficient & Std. err. & t & \(\mathrm{P}>|\mathrm{t}|\) & [95\% conf. interval] \\
\hline medage & -109.0958 & 13.52452 & -8.07 & 0.000 & -136.3527 & -81.83892 \\
c.medage\# & & & & & & \\
c.medage & 1.635209 & .2290536 & 7.14 & 0.000 & 1.173582 & 2.096836 \\
region & & & & & & \\
NCentral & 15.00283 & 4.252067 & 3.53 & 0.001 & 6.433353 & 23.57231 \\
South & 7.366445 & 3.953335 & 1.86 & 0.069 & -.6009775 & 15.33387 \\
West & 21.39679 & 4.650601 & 4.60 & 0.000 & 12.02412 & 30.76946 \\
_cons & 1947.611 & 199.8405 & 9.75 & 0.000 & 1544.859 & 2350.363 \\
\hline
\end{tabular}
test can now be used to perform a variety of statistical tests. Specify the coeflegend option with your estimation command to see a legend of the coefficients and how to specify them; see [R] Estimation options. We can test the hypothesis that the coefficient on 3.region is zero by typing
```

. test 3.region=0
( 1) 3.region = 0
F( 1, 44) = 3.47
Prob > F = 0.0691

```

The \(F\) statistic with 1 numerator and 44 denominator degrees of freedom is 3.47 . The significance level of the test is \(6.91 \%\)-we can reject the hypothesis at the \(10 \%\) level but not at the \(5 \%\) level.

This result from test is identical to one presented in the output from regress, which indicates that the \(t\) statistic on the 3 .region coefficient is 1.863 and that its significance level is 0.069 . The \(t\) statistic presented in the output can be used to test the hypothesis that the corresponding coefficient is zero, although it states the test in slightly different terms. The \(F\) distribution with 1 numerator degree of freedom is, however, identical to the \(t^{2}\) distribution. We note that \(1.863^{2} \approx 3.47\) and that the significance levels in each test agree, although one extra digit is presented by the test command.

\section*{Technical note}

After all estimation commands, including those that use the maximum likelihood method, the test that one variable is zero is identical to that reported by the command's output. The tests are performed in the same way-using the estimated covariance matrix-and are known as Wald tests. If the estimation command reports significance levels and confidence intervals using \(z\) rather than \(t\) statistics, test reports results using the \(\chi^{2}\) rather than the \(F\) statistic.
> Example 2: Testing the value of a single coefficient
If that were all test could do, it would be useless. We can use test, however, to perform other

```

. test 2.region=21
( 1) 2.region = 21
F( 1, 44) = 1.99
Prob > F = 0.1654

```

We find that we cannot reject that hypothesis, or at least we cannot reject it at any significance level below \(16.5 \%\).
> Example 3: Testing the equality of two coefficients
The previous test is useful, but we could almost as easily perform it by hand using the results presented in the regression output if we were well read on our statistics. We could type
```

. display Ftail(1,44,((_coef[2.region]-21)/4.252068)^2)
. }1654487

```

So, now let's test something a bit more difficult: whether the coefficient on 2. region is the same as the coefficient on 4.region:
```

. test 2.region=4.region
( 1) 2.region - 4.region = 0
F( 1, 44) = 2.84
Prob > F = 0.0989

```

We find that we cannot reject the equality hypothesis at the \(5 \%\) level, but we can at the \(10 \%\) level.

\section*{> Example 4}

When we tested the equality of the 2.region and 4.region coefficients, Stata rearranged our algebra. When Stata displayed its interpretation of the specified test, it indicated that we were testing whether 2.region minus 4.region is zero. The rearrangement is innocuous and, in fact, allows Stata to perform much more complicated algebra, for instance,
```

. test 2*(2.region-3*(3.region-4.region))=3.region+2.region+6*(4.region-3.region)
(1) 2.region - 3.region = 0
F( 1, 44) = 5.06
Prob > F = 0.0295

```

Although we requested what appeared to be a lengthy hypothesis, once Stata simplified the algebra, it realized that all we wanted to do was test whether the coefficient on 2 .region is the same as the coefficient on 3.region.
- Technical note

Stata's ability to simplify and test complex hypotheses is limited to linear hypotheses. If you attempt to test a nonlinear hypothesis, you will be told that it is not possible:
```

. test 2.region/3.region=2.region+3.region
not possible with test
r(131);

```

To test a nonlinear hypothesis, see \([\mathrm{R}]\) testnl.

\section*{- Example 5: Testing joint hypotheses}

The real power of test is demonstrated when we test joint hypotheses. Perhaps we wish to test whether the region variables, taken as a whole, are significant by testing whether the coefficients on 2.region, 3.region, and 4.region are simultaneously zero. test allows us to specify multiple conditions to be tested, each embedded within parentheses.
```

. test (2.region=0) (3.region=0) (4.region=0)
( 1) 2.region = 0
( 2) 3.region = 0
( 3) 4.region = 0
F( 3, 44) = 8.85
Prob > F = 0.0001

```
test displays the set of conditions and reports an \(F\) statistic of 8.85 . test also reports the degrees of freedom of the test to be 3 , the "dimension" of the hypothesis, and the residual degrees of freedom, 44. The significance level of the test is close to 0 , so we can strongly reject the hypothesis of no difference between the regions.

An alternative method to specify simultaneous hypotheses uses the convenient shorthand of conditions with multiple equality operators.
```

. test 2.region=3.region=4.region=0
( 1) 2.region - 3.region = 0
( 2) 2.region - 4.region = 0
( 3) 2.region = 0
F( 3, 44) = 8.85
Prob > F = 0.0001

```
- Technical note

Another method to test simultaneous hypotheses is to specify a test for each constraint and accumulate it with the previous constraints:
```

. test 2.region=0
(1) 2.region = 0
F( 1, 44) = 12.45
Prob > F = 0.0010
. test 3.region=0, accumulate
( 1) 2.region = 0
( 2) 3.region = 0
F( 2, 44) = 6.42
Prob > F = 0.0036
. test 4.region=0, accumulate
( 1) 2.region = 0
( 2) 3.region = 0
( 3) 4.region = 0
F( 3, 44) = 8.85
Prob > F = 0.0001

```

We tested the hypothesis that the coefficient on 2.region was zero by typing test 2. region \(=0\). We then tested whether the coefficient on 3.region was also zero by typing test 3.region=0, accumulate. The accumulate option told Stata that this was not the start of a new test but a continuation of a previous one. Stata responded by showing us the two equations and reporting an \(F\) statistic of 6.42 . The significance level associated with those two coefficients being zero is \(0.36 \%\).

When we added the last constraint test 4.region=0, accumulate, we discovered that the three region variables are significant. If all we wanted was the overall significance and we did not want to bother seeing the interim results, we could have used the notest option:
```

. test 2.region=0, notest
( 1) 2.region = 0
. test 3.region=0, accumulate notest
( 1) 2.region = 0
( 2) 3.region = 0
. test 4.region=0, accumulate
(1) 2.region = 0
(2) 3.region = 0
(3) 4.region = 0
F( 3, 44) = 8.85
Prob > F = 0.0001

```

\section*{Example 6: Quickly testing coefficients against zero}

Because tests that coefficients are zero are so common in applied statistics, the test command has a more convenient syntax to accommodate this case:
```

. test 2.region 3.region 4.region
(1) 2.region = 0
( 2) 3.region = 0
( 3) 4.region = 0
F( 3, 44) = 8.85
Prob > F = 0.0001

```

\section*{> Example 7: Specifying varlists}

We will now show how to use testparm. In its first syntax, test accepts a list of variable names but not a varlist.
```

. test i(2/4).region
i not found
r(111);

```

In the varlist, \(i(2 / 4)\).region means all the level variables from 2.region through 4.region, yet we received an error. test does not actually understand varlists, but testparm does. In fact, it understands only varlists.
```

. testparm i(2/4).region
( 1) 2.region = 0
( 2) 3.region = 0
( 3) 4.region = 0
F( 3, 44) = 8.85
Prob > F = 0.0001

```

Another way to test all the region variables is to type testparm i.region.
That testparm accepts varlists has other advantages that do not involve factor variables. Suppose that we have a dataset that has dummy variables reg2, reg3, and reg4, rather than the categorical variable region.
```

. use https://www.stata-press.com/data/r18/census4
(Census data on birthrate, median age)
. regress brate medage c.medage\#c.medage reg2 reg3 reg4
(output omitted)
. test reg2-reg4

- not found
r(111);

```

In a varlist, reg2-reg4 means variables reg2 and reg4 and all the variables between, yet we received an error. test is confused because the - has two meanings: it means subtraction in an expression and "through" in a varlist. Similarly, '*' means "any set of characters" in a varlist and multiplication in an expression. testparm avoids this confusion-it allows only a varlist.
```

. testparm reg2-reg4
(1) reg2 = 0
( 2) reg3 = 0
( 3) reg4 = 0
F( 3, 44) = 8.85
Prob > F = 0.0001

```
testparm has another advantage. We have five variables in our dataset that start with the characters reg: region, reg1, reg2, reg3, and reg4. reg* thus means those five variables:
. describe reg*
\begin{tabular}{lccll}
\begin{tabular}{r} 
Variable \\
name
\end{tabular} & \begin{tabular}{r} 
Storage \\
type
\end{tabular} & \begin{tabular}{r} 
Display \\
format
\end{tabular} & \begin{tabular}{c} 
Value \\
label
\end{tabular} & Variable label \\
\hline region & byte & \(\% 8.0 \mathrm{~g}\) & region & Census region \\
reg1 & byte & \(\% 9.0 \mathrm{~g}\) & & Region: NE \\
reg2 & byte & \(\% 9.0 \mathrm{~g}\) & & Region: N Cntrl \\
reg3 & byte & \(\% 9.0 \mathrm{~g}\) & & Region: South \\
reg4 & byte & \(\% 9.0 \mathrm{~g}\) & & Region: West
\end{tabular}

We cannot type test reg* because, in an expression, '*' means multiplication, but here is what would happen if we attempted to test all the variables that begin with reg:
```

. test region reg1 reg2 reg3 reg4
region not found
r(111);

```

The variable region was not included in our model, so it was not found. However, with testparm,
```

. testparm reg*
( 1) reg2 = 0
( 2) reg3 = 0
( 3) reg4 = 0
F( 3, 44) = 8.85
Prob > F = 0.0001

```

That is, testparm took reg* to mean all variables that start with reg that were in our model.

\section*{- Technical note}

Actually, reg* means what it always does-all variables in our dataset that begin with reg-in this case, region reg1 reg2 reg3 reg4. testparm just ignores any variables you specify that are not in the model.

\section*{> Example 8: Replaying the previous test}

We just used test (testparm, actually, but it does not matter) to test the hypothesis that reg2, reg3, and reg4 are jointly zero. We can review the results of our last test by typing test without arguments:
```

. test
( 1) reg2 = 0
( 2) reg3 = 0
( 3) reg4 = 0
F( 3, 44) = 8.85
Prob > F = 0.0001

```

\section*{- Technical note}
test does not care how we build joint hypotheses; we may freely mix different forms of syntax. (We can even start with testparm, but we cannot use it thereafter because it does not have an accumulate option.)

Say that we type test reg2 reg3 reg4 to test that the coefficients on our region dummies are jointly zero. We could then add a fourth constraint, say, that medage \(=100\), by typing test medage \(=100\), accumulate. Or, if we had introduced the medage constraint first (our first test command had been test medage \(=100\) ), we could then add the region dummy test by typing test reg2 reg3 reg4, accumulate or test (reg2=0) (reg3=0) (reg4=0), accumulate.

Remember that all previous tests are cleared when we do not specify the accumulate option. No matter what tests we performed in the past, if we type test medage c.medage\#c.medage, omitting the accumulate option, we would test that medage and c.medage\#c.medage are jointly zero.
\(>\) Example 9: Testing the equality of multiple coefficients
Let's return to our census3.dta dataset and test the hypothesis that all the included regions have the same coefficient - that the Northeast is significantly different from the rest of the nation:
```

. use https://www.stata-press.com/data/r18/census3
(1980 Census data by state)
. regress brate medage c.medage\#c.medage i.region
(output omitted)
. test 2.region=3.region=4.region
(1) 2.region - 3.region = 0
( 2) 2.region - 4.region = 0
F( 2, 44) = 8.23
Prob > F = 0.0009

```
 \(=\) operators is just a convenient shorthand for typing that the first expression equals the second expression and that the first expression equals the third expression,
```

. test (2.region=3.region) (2.region=4.region)

```

We performed the test for equality of the three regions by imposing two constraints: region 2 has the same coefficient as region 3, and region 2 has the same coefficient as region 4. Alternatively, we could have tested that the coefficients on regions 2 and 3 are the same and that the coefficients on regions 3 and 4 are the same. We would obtain the same results in either case.

To test for equality of the three regions, we might, likely by mistake, type equality constraints for all pairs of regions:
```

. test (2.region=3.region) (2.region=4.region) (3.region=4.region)
( 1) 2.region - 3.region = 0
( 2) 2.region - 4.region = 0
( 3) 3.region - 4.region = 0
Constraint 3 dropped
F( 2, 44) = 8.23
Prob > F = 0.0009

```

Equality of regions 2 and 3 and of regions 2 and 4, however, implies equality of regions 3 and 4. test recognized that the last constraint is implied by the other constraints and hence dropped it.

\section*{- Technical note}

Generally, Stata uses \(=\) for assignment, as in gen newvar \(=\exp\), and \(==\) as the operator for testing equality in expressions. For your convenience, test allows both \(=\) and \(==\) to be used.

\section*{> Example 10}

The test for the equality of the regions is also possible with the testparm command. When we include the equal option, testparm tests that the coefficients of all the variables specified are equal:
```

. testparm i(2/4).region, equal
(1) - 2.region + 3.region = 0
(2) - 2.region + 4.region = 0
F(2, 44) = 8.23
Prob > F = 0.0009

```

We can also obtain the equality test by accumulating single equality tests.
```

. test 2.region=3.region, notest
(1) 2.region - 3.region = 0
. test 2.region=4.region, accum
( 1) 2.region - 3.region = 0
( 2) 2.region - 4.region = 0
F( 2, 44) = 8.23
Prob > F = 0.0009

```

\section*{- Technical note}

If we specify a set of inconsistent constraints, test will tell us by dropping the constraint or constraints that led to the inconsistency. For instance, let's test that the coefficients on region 2 and region 4 are the same, add the test that the coefficient on region 2 is 20 , and finally add the test that the coefficient on region 4 is 21 :
```

. test (2.region=4.region) (2.region=20) (4.region=21)
( 1) 2.region - 4.region = 0
( 2) 2.region = 20
( 3) 4.region = 21
Constraint 2 dropped
F( 2, 44) = 1.82
Prob > F = 0.1737

```
test informed us that it was dropping constraint 2 . All three equations cannot be simultaneously true, so test drops whatever it takes to get back to something that makes sense.

\section*{Special syntaxes after multiple-equation estimation}

Everything said above about tests after single-equation estimation applies to tests after multipleequation estimation, as long as you remember to specify the equation name. To demonstrate, let's estimate a seemingly unrelated regression by using sureg; see [R] sureg.
```

. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. sureg (price foreign mpg displ) (weight foreign length)
Seemingly unrelated regression

```
\begin{tabular}{lrrrrrr}
\hline Equation & Obs & Params & RMSE & "R-squared" & chi2 & P>chi2 \\
\hline price & 74 & 3 & 2165.321 & 0.4537 & 49.64 & 0.0000 \\
weight & 74 & 2 & 245.2916 & 0.8990 & 661.84 & 0.0000 \\
\hline
\end{tabular}
\begin{tabular}{r|rlrrrr}
\hline & Coefficient & Std. err. & \(z\) & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline price & & & & & & \\
foreign & 3058.25 & 685.7357 & 4.46 & 0.000 & 1714.233 & 4402.267 \\
mpg & -104.9591 & 58.47209 & -1.80 & 0.073 & -219.5623 & 9.644042 \\
displacement & 18.18098 & 4.286372 & 4.24 & 0.000 & 9.779842 & 26.58211 \\
_cons & 3904.336 & 1966.521 & 1.99 & 0.047 & 50.0263 & 7758.645 \\
\hline weight & & & & & & \\
foreign & -147.3481 & 75.44314 & -1.95 & 0.051 & -295.2139 & .517755 \\
length & 30.94905 & 1.539895 & 20.10 & 0.000 & 27.93091 & 33.96718 \\
_cons & -2753.064 & 303.9336 & -9.06 & 0.000 & -3348.763 & -2157.365 \\
\hline
\end{tabular}

To test the significance of foreign in the price equation, we could type
```

. test [price]foreign
( 1) [price]foreign = 0
chi2( 1) = 19.89
Prob > chi2 = 0.0000

```
which is the same result reported by sureg: \(4.460^{2} \approx 19.89\). To test foreign in both equations, we could type
```

. test [price]foreign [weight]foreign
( 1) [price]foreign = 0
( 2) [weight]foreign = 0
chi2( 2) = 31.61
Prob > chi2 = 0.0000

```
or
```

. test foreign
( 1) [price]foreign = 0
( 2) [weight]foreign = 0
chi2( 2) = 31.61
Prob > chi2 = 0.0000

```

This last syntax - typing the variable name by itself-tests the coefficients in all equations in which they appear. The variable length appears in only the weight equation, so typing
```

. test length
( 1) [weight]length = 0
chi2( 1) = 403.94
Prob > chi2 = 0.0000

```
yields the same result as typing test [weight]length. We may also specify a linear expression rather than a list of coefficients:
```

. test mpg=displ
( 1) [price]mpg - [price]displacement = 0
chi2( 1) = 4.85
Prob > chi2 = 0.0277

```
or
```

. test [price]mpg = [price]displ
( 1) [price]mpg - [price]displacement = 0
chi2( 1) = 4.85
Prob > chi2 = 0.0277

```

A variation on this syntax can be used to test cross-equation constraints:
```

. test [price]foreign = [weight]foreign
( 1) [price]foreign - [weight]foreign = 0
chi2( 1) = 23.07
Prob > chi2 = 0.0000

```

Typing an equation name in square brackets by itself tests all the coefficients except the intercept in that equation:
```

. test [price]
( 1) [price]foreign = 0
( 2) [price]mpg = 0
( 3) [price]displacement = 0
chi2( 3) = 49.64
Prob > chi2 = 0.0000

```

Typing an equation name in square brackets, a colon, and a list of variable names tests those variables in the specified equation:
```

. test [price]: foreign displ
( 1) [price]foreign = 0
( 2) [price]displacement = 0
chi2( 2) = 25.19
Prob > chi2 = 0.0000

```
test \(\left[\right.\) eqname \(_{1}=\) eqname \(_{2}\) ] tests that all the coefficients in the two equations are equal. We cannot use that syntax here because there are different variables in the model:
```

. test [price=weight]
variables differ between equations
(to test equality of coefficients in common, specify option common)
r(111);

```

The common option specifies a test of the equality coefficients common to the equations price and weight,
```

. test [price=weight], common
( 1) [price]foreign - [weight]foreign = 0
chi2( 1) = 23.07
Prob > chi2 = 0.0000

```

By default, test does not include the constant, the coefficient of the constant variable _cons, in the test. The cons option specifies that the constant be included.
```

. test [price=weight], common cons
( 1) [price]foreign - [weight]foreign = 0
( 2) [price]_cons - [weight]_cons = 0
chi2( 2) = 51.23
Prob > chi2 = 0.0000

```

We can also use a modification of this syntax with the model if we also type a colon and the names of the variables we want to test:
```

. test [price=weight]: foreign
( 1) [price]foreign - [weight]foreign = 0
chi2( 1) = 23.07
Prob > chi2 = 0.0000

```

We have only one variable in common between the two equations, but if there had been more, we could have listed them.

Finally, a simultaneous test of multiple constraints may be specified just as after single-equation estimation.
```

. test ([price]: foreign) ([weight]: foreign)
( 1) [price]foreign = 0
( 2) [weight]foreign = 0
chi2( 2) = 31.61
Prob > chi2 = 0.0000

```
test can also test for equality of coefficients across more than two equations. For instance, test [eq1=eq2=eq3] specifies a test that the coefficients in the three equations eq1, eq2, and eq3 are equal. This requires that the same variables be included in the three equations. If some variables are entered only in some of the equations, you can type test [eq1=eq2=eq3], common to test that the coefficients of the variables common to all three equations are equal. Alternatively, you can explicitly list the variables for which equality of coefficients across the equations is to be tested. For instance, test [eq1=eq2=eq3]: time money tests that the coefficients of the variables time and money do not differ between the equations.

\section*{- Technical note}
test [eq1=eq2=eq3], common tests the equality of the coefficients common to all equations, but it does not test the equality of all common coefficients. Consider the case where
\[
\begin{array}{ll}
\text { eq1 } & \text { contains the variables var1 var2 var3 } \\
\text { eq2 } & \text { contains the variables var1 var2 var4 } \\
\text { eq3 } & \text { contains the variables var1 var3 var4 }
\end{array}
\]

Obviously, only var1 is common to all three equations. Thus test [eq1=eq2=eq3], common tests that the coefficients of var1 do not vary across the equations, so it is equivalent to test [eq1=eq2=eq3]: var1. To perform a test of the coefficients of variables common to two equations, you could explicitly list the constraints to be tested,
```

. test ([eq1=eq2=eq3]:var1) ([eq1=eq2]:var2) ([eq1=eq3]:var3) ([eq2=eq3]:var4)

```
or use test with the accumulate option, and maybe also with the notest option, to form the appropriate joint hypothesis:
```

. test [eq1=eq2], common notest
. test [eq1=eq3], common accumulate notest
. test [eq2=eq3], common accumulate

```

\section*{Constrained coefficients}

If the test indicates that the data do not allow you to conclude that the constraints are not satisfied, you may want to inspect the constrained coefficients. The coef option specified that the constrained results, estimated by GLS, are shown.
```

. test [price=weight], common coef
( 1) [price]foreign - [weight]foreign = 0
chi2( 1) = 23.07
Prob > chi2 = 0.0000

```

Constrained coefficients
\begin{tabular}{r|rlllrr}
\hline & Coefficient & Std. err. & \(z\) & \(P>|z|\) & {\([95 \%\) conf. interval] } \\
\hline price & & & & & & \\
foreign & -216.4015 & 74.06083 & -2.92 & 0.003 & -361.558 & -71.2449 \\
mpg & -121.5717 & 58.36972 & -2.08 & 0.037 & -235.9742 & -7.169116 \\
displacement & 7.632566 & 3.681114 & 2.07 & 0.038 & .4177148 & 14.84742 \\
_cons & 7312.856 & 1834.034 & 3.99 & 0.000 & 3718.215 & 10907.5 \\
\hline weight & & & & & & \\
foreign & -216.4015 & 74.06083 & -2.92 & 0.003 & -361.558 & -71.2449 \\
length & 30.34875 & 1.534815 & 19.77 & 0.000 & 27.34057 & 33.35693 \\
_cons & -2619.719 & 302.6632 & -8.66 & 0.000 & -3212.928 & -2026.51 \\
\hline
\end{tabular}

The constrained coefficient of foreign is -216.40 with standard error 74.06 in equations price and weight. The other coefficients and their standard errors are affected by imposing the equality constraint of the two coefficients of foreign because the unconstrained estimates of these two coefficients were correlated with the estimates of the other coefficients.

\section*{Technical note}

The two-step constrained coefficients \(b_{c}\) displayed by test, coef are asymptotically equivalent to the one-stage constrained estimates that are computed by specifying the constraints during estimation using the constraint () option of estimation commands (Gouriéroux and Monfort 1995, chap. 10). Generally, one-step constrained estimates have better small-sample properties. For inspection and interpretation, however, two-step constrained estimates are a convenient alternative. Moreover, some estimation commands (for example, stcox, many xt estimators) do not have a constraint () option.

\section*{Multiple testing}

When performing the test of a joint hypothesis, you might want to inspect the underlying 1-degree-of-freedom hypotheses. Which constraint "is to blame"? test displays the univariate as well as the simultaneous test if the mtest option is specified. For example,
\begin{tabular}{l}
. test [price=weight], common cons mtest \\
\begin{tabular}{l} 
(1) \\
( 2 [price]foreign \\
[price]_cons - [weight]foreign
\end{tabular}\(=0\) \\
\hline
\end{tabular}
* Unadjusted \(p\)-values

Both coefficients seem to contribute to the highly significant result. The 1-degree-of-freedom test shown here is identical to those if test had been invoked to test just this simple hypotheses. There is, of course, a real risk in inspecting these simple hypotheses. Especially in high-dimensional hypotheses, you may easily find one hypothesis that happens to be significant. Multiple testing procedures are designed to provide some safeguard against this risk. \(p\)-values of the univariate hypotheses are modified so that the probability of falsely rejecting one of the null hypotheses is bounded. test provides the methods based on Bonferroni, Šidák, and Holm.
```

. test [price=weight], common cons mtest(b)
( 1) [price]foreign - [weight]foreign = 0
( 2) [price]_cons - [weight]_cons = 0

```
\begin{tabular}{c|ccc}
\hline & chi2 & df & \(\mathrm{p}>\) chi2 \\
\hline\((1)\) & 23.07 & 1 & \(0.0000 *\) \\
\((2)\) & 11.17 & 1 & \(0.0017 *\) \\
\hline All & 51.23 & 2 & 0.0000 \\
\hline
\end{tabular}
```

* Bonferroni-adjusted p-values

```

\section*{Stored results}
test and testparm store the following in \(r()\) :
Scalars
```

    r(p) two-sided p-value
    r(F) F statistic
    r(df) test constraints degrees of freedom
    r(df_r) residual degrees of freedom
    r(dropped_i) index of ith constraint dropped
    r(mtmethod) method of adjustment for multiple
                        testing
    r(mtest) multiple test results
    ```
Macros
Matrices
\(r\) (ss) and \(r\) (rss) are defined only when test is used for testing effects after anova.

\section*{Methods and formulas}
test and testparm perform Wald tests. Let the estimated coefficient vector be \(\mathbf{b}\) and the estimated variance-covariance matrix be \(\mathbf{V}\). Let \(\mathbf{R b}=\mathbf{r}\) denote the set of \(q\) linear hypotheses to be tested jointly.

The Wald test statistic is (Judge et al. 1985, 20-28)
\[
W=(\mathbf{R b}-\mathbf{r})^{\prime}\left(\mathbf{R V} \mathbf{R}^{\prime}\right)^{-1}(\mathbf{R b}-\mathbf{r})
\]

If the estimation command reports its significance levels using \(Z\) statistics, a \(\chi^{2}\) distribution with \(q\) degrees of freedom,
\[
W \sim \chi_{q}^{2}
\]
is used for computation of the significance level of the hypothesis test.
If the estimation command reports its significance levels using \(t\) statistics with \(d\) degrees of freedom, an \(F\) statistic,
\[
F=\frac{1}{q} W
\]
is computed, and an \(F\) distribution with \(q\) numerator degrees of freedom and \(d\) denominator degrees of freedom computes the significance level of the hypothesis test.

The two-step constrained estimates \(b_{c}\) displayed by test with the coef option are the GLS estimates of the unconstrained estimates \(b\) subject to the specified constraints \(R b=c\) (Gouriéroux and Monfort 1995, chap. 10),
\[
\mathbf{b}_{\mathbf{c}}=\mathbf{b}-\mathbf{V R}^{\prime}\left(\mathbf{R V R} \mathbf{R}^{\prime}\right)^{-1}(\mathbf{R} \mathbf{b}-\mathbf{r})
\]
with variance-covariance matrix
\[
\mathbf{V}_{\mathbf{c}}=\mathbf{V}-\mathbf{V R}^{\prime}\left(\mathbf{R V R}^{\prime}\right)^{-1} \mathbf{R V}
\]

If test displays a Wald test for joint (simultaneous) hypotheses, it can also display all 1-degree-offreedom tests, with \(p\)-values adjusted for multiple testing. Let \(p_{1}, p_{2}, \ldots, p_{k}\) be the unadjusted \(p\)-values of these 1-degree-of-freedom tests. The Bonferroni-adjusted \(p\)-values are defined as \(p_{i}^{b}=\min \left(1, k p_{i}\right)\). The Šidák-adjusted \(p\)-values are \(p_{i}^{s}=1-\left(1-p_{i}\right)^{k}\). Let \(p_{(1)}, p_{(2)}, \ldots, p_{(k)}\) be the ordered unadjusted \(p\)-values. The Holm's method adjusted \(p\)-values are then defined by
\[
p_{(i)}^{h}=\max _{j \leq i}\left[\min \left\{1,(k-j+1) p_{(j)}\right\}\right]
\]

If test is used after a svy command, it carries out an adjusted Wald test-this adjustment should not be confused with the adjustment for multiple testing. Both adjustments may actually be combined. Specifically, the survey adjustment uses an approximate \(F\) statistic \((d-k+1) W /(k d)\), where \(W\) is the Wald test statistic, \(k\) is the dimension of the hypothesis test, and \(d=\) the total number of sampled PSUs minus the total number of strata. Under the null hypothesis, \((d-k+1) F /(k d) \sim F(k, d-k+1)\), where \(F(k, d-k+1)\) is an \(F\) distribution with \(k\) numerator degrees of freedom and \(d-k+1\) denominator degrees of freedom. If nosvyadjust is specified, the \(p\)-value is computed using \(W / k \sim F(k, d)\).

See Korn and Graubard (1990) for a detailed description of the Bonferroni adjustment technique and for a discussion of the relative merits of it and of the adjusted and unadjusted Wald tests.

\section*{Acknowledgment}

The svy adjustment code was adopted from another command developed in collaboration with John L. Eltinge of the United States Census Bureau.

\section*{References}

Beale, E. M. L. 1960. Confidence regions in non-linear estimation. Journal of the Royal Statistical Society, Series B 22: 41-88. https://doi.org/10.1111/j.2517-6161.1960.tb00353.x.

Bormann, S.-K. 2022. A Stata implementation of second-generation p-values. Stata Journal 22: 496-520.
Canette, I. 2014. Using gsem to combine estimation results. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2014/08/18/using-gsem-to-combine-estimation-results/.

Clarke, D., J. P. Romano, and M. Wolf. 2020. The Romano-Wolf multiple-hypothesis correction in Stata. Stata Journal 20: 812-843.

Dietz, T., and L. Kalof. 2009. Introduction to Social Statistics: The Logic of Statistical Reasoning. Chichester, UK: Wiley.
Drukker, D. M. 2023. Simultaneous tests and confidence bands for Stata estimation commands. Stata Journal 23: 518-544.

Gouriéroux, C. S., and A. Monfort. 1995. Statistics and Econometric Models, Vol. 1: General Concepts, Estimation, Prediction, and Algorithms. Trans. Q. Vuong. Cambridge: Cambridge University Press.
Holm, S. 1979. A simple sequentially rejective multiple test procedure. Scandinavian Journal of Statistics 6: 65-70.
Judge, G. G., W. E. Griffiths, R. C. Hill, H. Lütkepohl, and T.-C. Lee. 1985. The Theory and Practice of Econometrics. 2nd ed. New York: Wiley.
Korn, E. L., and B. I. Graubard. 1990. Simultaneous testing of regression coefficients with complex survey data: Use of Bonferroni \(t\) statistics. American Statistician 44: 270-276. https://doi.org/10.2307/2684345.

Mehmetoglu, M., and T. G. Jakobsen. 2022. Applied Statistics Using Stata: A Guide for the Social Sciences. 2nd ed. Thousand Oaks, CA: Sage.
Ye, X., and Y. Sun. 2018. Heteroskedasticity- and autocorrelation-robust F and t tests in Stata. Stata Journal 18: 951-980.

\section*{Also see}
[R] anova - Analysis of variance and covariance
[R] anova postestimation - Postestimation tools for anova
[R] contrast - Contrasts and linear hypothesis tests after estimation
[R] lincom - Linear combinations of parameters
[R] Irtest - Likelihood-ratio test after estimation
[R] nestreg - Nested model statistics
[R] nlcom - Nonlinear combinations of parameters
[R] testnl - Test nonlinear hypotheses after estimation
[U] 13.5 Accessing coefficients and standard errors
[U] 20 Estimation and postestimation commands

\section*{Title}
testnl - Test nonlinear hypotheses after estimation
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
testnl tests (linear or nonlinear) hypotheses about the estimated parameters from the most recently fit model.
testnl produces Wald-type tests of smooth nonlinear (or linear) hypotheses about the estimated parameters from the most recently fit model. The \(p\)-values are based on the delta method, an approximation appropriate in large samples.
testnl can be used with svy estimation results; see [SVY] svy postestimation.
The format ( \(\exp _{1}=\exp _{2}=\exp _{3} \ldots\) ) for a simultaneous-equality hypothesis is just a convenient shorthand for a list \(\left(\exp _{1}=\exp _{2}\right)\left(\exp _{1}=\exp _{3}\right)\), etc.
testnl may also be used to test linear hypotheses. test is faster if you want to test only linear hypotheses; see [R] test. testnl is the only option for testing linear and nonlinear hypotheses simultaneously.

\section*{Quick start}

\section*{After single-equation models}

Test that the product of the coefficients for x 1 and x 2 is equal to 4
```

testnl _b[x1]*_b[x2] = 4

```

Test that the ratio of the indicators for the factor variable \(\mathrm{a}=2\) and \(\mathrm{a}=3\) is 1
```

testnl _b[2.a]/_b[3.a] = 1

```

Test that an expression involving continuous factor variable syntax is equal to 16
```

testnl -_b[x1]/(2*_b[c.x1\#c.x1]) = 16

```

Test the equality of two expressions
```

testnl _b[x1]*_b[x2] = _b[x1]*_b[x3]

```

Joint test that two products are both equal to 2
```

testnl (_b[x1]*_b[x2] = 2) (_b[x1]*_b[x3] = 2)

```

Same as above
testnl _b[x1]*_b[x2] = _b[x1]*_b[x3] = 2
Same as above, but add separate tests for each expression
```

testnl _b[x1]*_b[x2] = _b[x1]*_b[x3] = 2, mtest

```

Same as above, but adjust \(p\)-values for multiple comparisons using Holm's method
```

testnl _b[x1]*_b[x2] = _b[x1]*_b[x3] = 2, mtest(holm)

```

Test a linear hypothesis and a nonlinear hypothesis together
```

testnl (_b[x1] =_b[x2]) (_b[x2] ^ 2 =_b[x3])

```

After multiple-equation models
Test that the product of the coefficients for x 1 and x 2 in the equation for y 1 is equal to 1 testnl _b[y1:x1]*_b[y1:x2] = 1

Test that the product of the coefficients for x 1 and x 2 in the equation for y 2 is equal to 1 testnl _b[y2:x1]*_b[y2:x2] = 1

Test the equality of expressions involving coefficients from the equations for y1 and y4 testnl _b[y1:x1]*_b[y1:x2] = _b[y4:x1]*_b[y4:x2]

\section*{Menu}

Statistics \(>\) Postestimation

\section*{Syntax}
testnl \(\exp =\exp [=\exp \ldots][\), options \(]\)
testnl \((\exp =\exp [=\exp \ldots])[(\exp =\exp [=\exp \ldots]) \ldots][\), options \(]\)
options
mtest \([(o p t)] \quad\) test each condition separately
iterate(\#)
df (\#)
nosvyadjust

\section*{Description} reference distribution of the test statistic
use maximum \# of iterations to find the optimal step size
use \(F\) distribution with \# denominator degrees of freedom for the
carry out the Wald test as \(W / k \sim F(k, d)\); for use with svy estimation commands when the df() option is also specified
collect is allowed; see [U] 11.1.10 Prefix commands.
\(\mathrm{df}(\#)\) and nosvyadjust do not appear in the dialog box.
The second syntax means that if more than one expression is specified, each must be surrounded by parentheses.
\(\exp\) is a possibly nonlinear expression containing
\[
\begin{aligned}
& \text { _b }[\text { coef }] \\
& \text { _b }[e q n o: \text { coef }] \\
& {[e q n o] \text { coef }} \\
& {[e q n o] \_\mathrm{b}[\text { coef }]}
\end{aligned}
\]
eqno is
```


## 

    name
    ```
coef identifies a coefficient in the model. coef is typically a variable name, a level indicator, an interaction indicator, or an interaction involving continuous variables. Level indicators identify one level of a factor variable and interaction indicators identify one combination of levels of an interaction; see [U] 11.4.3 Factor variables. coef may contain time-series operators; see [U] 11.4.4 Time-series varlists.

Distinguish between [], which are to be typed, and [], which indicate optional arguments.

\section*{Options}
mtest \([(o p t)]\) specifies that tests be performed for each condition separately. opt specifies the method for adjusting \(p\)-values for multiple testing. Valid values for opt are
\begin{tabular}{rl}
\(\underline{\text { bonferroni }}\) & Bonferroni's method \\
\(\underline{\text { holm }}\) & Holm's method \\
\(\underline{\text { sidak }}\) & Šidák's method \\
noadjust & no adjustment is to be made
\end{tabular}

Specifying mtest without an argument is equivalent to specifying mtest(noadjust).
iterate(\#) specifies the maximum number of iterations used to find the optimal step size in the calculation of numerical derivatives of the test expressions. By default, the maximum number of iterations is 100 , but convergence is usually achieved after only a few iterations. You should rarely have to use this option.

The following options are available with testnl but are not shown in the dialog box:
\(\mathrm{df}(\#)\) specifies that the \(F\) distribution with \# denominator degrees of freedom be used for the reference distribution of the test statistic. With survey data, \# is the design degrees of freedom unless nosvyadjust is specified.
nosvyadjust is for use with svy estimation commands when the df() option is also specified; see [SVY] svy estimation. It specifies that the Wald test be carried out without the default adjustment for the design degrees of freedom. That is, the test is carried out as \(W / k \sim F(k, d)\) rather than as \((d-k+1) W /(k d) \sim F(k, d-k+1)\), where \(k=\) the dimension of the test and \(d=\) the design degrees of freedom specified in the df() option.

\section*{Remarks and examples}

Remarks are presented under the following headings:

\author{
Introduction \\ Using testnl to perform linear tests \\ Specifying constraints \\ Dropped constraints \\ Multiple constraints \\ Manipulability
}

\section*{Introduction}

\section*{\(>\) Example 1}

We have just estimated the parameters of an earnings model on cross-sectional time-series data using one of Stata's more sophisticated estimators:
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline \multicolumn{4}{|l|}{GEE population-averaged model} & \multicolumn{2}{|r|}{\multirow[t]{3}{*}{Number of obs Number of groups}} & \(=1,326\) \\
\hline \multicolumn{4}{|l|}{Group variable: idcode} & & & \(=269\) \\
\hline \multicolumn{4}{|l|}{Family: Gaussian} & & & Obs per group: \\
\hline \multicolumn{3}{|l|}{Link: Identity} & & \multicolumn{2}{|r|}{in} & \(=1\) \\
\hline \multicolumn{3}{|l|}{\multirow[t]{2}{*}{Correlation: exchangeable}} & & & avg & \(=4.9\) \\
\hline & & & & & max & \(=9\) \\
\hline & & & & & d chi2(3) & \(=327.33\) \\
\hline \multicolumn{3}{|l|}{Scale parameter = . 0976738} & & \multicolumn{2}{|r|}{Prob > chi2} & \(=0.0000\) \\
\hline ln_wage & Coefficient & Std. err. & z & \(P>|z|\) & [95\% conf. & interval] \\
\hline grade & . 0749686 & . 0066111 & 11.34 & 0.000 & . 062011 & . 0879261 \\
\hline age & . 1080806 & . 0235861 & 4.58 & 0.000 & . 0618526 & . 1543086 \\
\hline c.age\#c.age & -. 0016253 & . 0004739 & \(-3.43\) & 0.001 & -. 0025541 & -. 0006966 \\
\hline _cons & -. 8788933 & . 2830899 & -3.10 & 0.002 & -1.433739 & -. 3240473 \\
\hline
\end{tabular}

An implication of this model is that peak earnings occur at age -_b [age]/(2*_b[c.age\#c.age]), which here is equal to 33.2. Say that we have a theory that peak earnings should occur at age \(16+\) 1/_b [grade].
```

. testnl -_b[age]/(2*_b[c.age\#c.age]) = 16 + 1/_b[grade]
(1) __b[age]/(2*_b[c.age\#c.age]) = 16 + 1/_b[grade]
chi2(1) = 1.71
Prob > chi2 = 0.1914

```

These data do not reject our theory.

\section*{Using testnl to perform linear tests}
testnl may be used to test linear constraints, but test is faster; see [R] test. You could type
```

. testnl _b[x4] = _b[x1]

```
but it would take less computer time if you typed
```

. test _b[x4] = _b[x1]

```

\section*{Specifying constraints}

The constraints to be tested can be formulated in many different ways. You could type
```

. testnl _b[mpg]*_b[weight] = 1

```
or
```

. testnl _b[mpg] = 1/_b[weight]

```
or you could express the constraint any other way you wished. (To say that testnl allows constraints to be specified in different ways does not mean that the test itself does not depend on the formulation. This point is briefly discussed later.) In formulating the constraints, you must, however, exercise one caution: users of test often refer to the coefficient on a variable by specifying the variable name. For example,
```

. test mpg = 0

```

More formally, they should type
```

test _b[mpg] = 0

```
but test allows the _b [] surrounding the variable name to be omitted. testnl does not allow this shorthand. Typing
```

. testnl mpg=0

```
specifies the constraint that the value of variable mpg in the first observation is zero. If you make this mistake, sometimes testnl will catch it:
```

. testnl mpg=0
equation (1) contains reference to X rather than _b[X]
r(198);

```

In other cases, testnl may not catch the mistake; then, the constraint will be dropped because it does not make sense:
```

. testnl mpg=0
Constraint (1) dropped

```
(There are reasons other than this for constraints being dropped.) The worst case, however, is
```

. testnl _b[weight]*mpg = 1

```
when what you mean is not that _b [weight] equals the reciprocal of the value of mpg in the first observation, but rather that
```

. testnl _b[weight]*_b[mpg] = 1

```

Sometimes, this mistake will be caught by the "contains reference to X rather than \(\_\mathrm{b}[\mathrm{X}]\) " error, and sometimes it will not. Be careful.
testnl, like test, can be used after any Stata estimation command, including the survey estimators. When you use it after a multiple-equation command, such as mlogit or heckman, you refer to coefficients by using Stata's standard syntax: [eqname] _b [varname].

Stata's single-equation estimation output looks like this:
\begin{tabular}{r|rllll} 
& Coef & \(\ldots\) & & \\
\hline weight & 12.27 & \(\ldots\) & \(<-\) & coefficient is _b[weight] \\
mpg & 3.21 & \(\ldots\) & & & \\
\hline
\end{tabular}

Stata's multiple-equation output looks like this:
\begin{tabular}{|c|c|c|c|}
\hline & \multicolumn{3}{|l|}{Coef} \\
\hline \multicolumn{4}{|l|}{cat1} \\
\hline weight & 12.27 & <- coefficient is & [cat1]_b[weight] \\
\hline mpg & 3.21 & & \\
\hline \multicolumn{4}{|l|}{8} \\
\hline weight & 5.83 & <- coefficient is & [8]_b[weight] \\
\hline mpg & 7.43 & & \\
\hline
\end{tabular}

\section*{Dropped constraints}
testnl automatically drops constraints when
- They are nonbinding, for example, \(\_\mathrm{b}[\mathrm{mpg}]=\_\mathrm{b}[\mathrm{mpg}]\). More subtle cases include
```

_b[mpg]*_b[weight] = 4
_b[weight] = 2
_b[mpg] = 2

```

In this example, the third constraint is nonbinding because it is implied by the first two.
- They are contradictory, for example, \(\_\mathrm{b}[\mathrm{mpg}]=2\) and \(\_\mathrm{b}[\mathrm{mpg}]=3\). More subtle cases include
```

_b[mpg]*_b[weight] = 4
_b[weight] = 2
_b[mpg] = 3

```

The third constraint contradicts the first two.

\section*{Multiple constraints}

\section*{> Example 2}

We illustrate the simultaneous test of a series of constraints using simulated data on labor-market promotion in a given year. We fit a probit model with separate effects for education, experience, and experience-squared for men and women.
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline Probit regres & & & & & \begin{tabular}{l}
Number of obs LR chi2(7) \\
Prob > chi2
\end{tabular} & \[
\begin{aligned}
& =775 \\
& =424.42 \\
& =0.0000
\end{aligned}
\] \\
\hline \multicolumn{5}{|l|}{Log likelihood = -245.42768} & Pseudo R2 & \(=0.4637\) \\
\hline promo & Coefficient & Std. err. & z & \(P>|z|\) & [95\% conf. & interval] \\
\hline male & . 6489974 & . 203739 & 3.19 & 0.001 & . 2496763 & 1.048318 \\
\hline \multicolumn{7}{|l|}{male\#c.yedu} \\
\hline Female & . 9730237 & . 1056136 & 9.21 & 0.000 & . 7660248 & 1.180023 \\
\hline Male & 1.390517 & . 1527288 & 9.10 & 0.000 & 1.091174 & 1.68986 \\
\hline \multicolumn{7}{|l|}{male\#c.yexp} \\
\hline Female & . 4559544 & . 0901169 & 5.06 & 0.000 & . 2793285 & . 6325803 \\
\hline Male & 1.422539 & . 1544255 & 9.21 & 0.000 & 1.11987 & 1.725207 \\
\hline \multicolumn{7}{|l|}{male\#c. yexp2} \\
\hline Female & -. 1027149 & . 0573059 & -1.79 & 0.073 & -. 2150325 & . 0096026 \\
\hline Male & -. 3749457 & . 1160113 & -3.23 & 0.001 & -. 6023236 & -. 1475677 \\
\hline _cons & . 9872018 & . 1148215 & 8.60 & 0.000 & . 7621559 & 1.212248 \\
\hline
\end{tabular}

Note: 1 failure and 2 successes completely determined.
The effects of human capital seem to differ between men and women. A formal test confirms this.
```

. test (yedu\#O.male = yedu\#1.male) (yexp\#0.male = yexp\#1.male)
> (yexp2\#0.male = yexp2\#1.male)
( 1) [promo]0b.male\#c.yedu - [promo]1.male\#c.yedu = 0
( 2) [promo]Ob.male\#c.yexp - [promo]1.male\#c.yexp = 0
( 3) [promo]0b.male\#c.yexp2 - [promo]1.male\#c.yexp2 = 0
chi2( 3) = 35.43
Prob > chi2 = 0.0000

```

How do we interpret this gender difference? It has repeatedly been stressed (see, for example, Long [1997, 47-50]; Allison [1999]) that comparison of groups in binary response models, and similarly in other latent-variable models, is hampered by an identification problem: with \(\beta\) the regression coefficients for the latent variable and \(\sigma\) the standard deviation of the latent residual, only the \(\beta / \sigma\) are identified. In fact, in terms of the latent regression, the probit coefficients should be interpreted as \(\beta / \sigma\), not as the \(\beta\). If we cannot claim convincingly that the residual standard deviation \(\sigma\) does not vary between the sexes, equality of the regression coefficients \(\beta\) implies that the coefficients of the probit model for men and women are proportional but not necessarily equal. This is a nonlinear hypothesis in terms of the probit coefficients, not a linear one.
```

. testnl _b[yedu\#1.male]/_b[yedu\#0.male] = _b[yexp\#1.male]/_b[yexp\#0.male]
> = _b[yexp2\#1.male]/_b[yexp2\#0.male]

```
    (1) _b[yedu\#1.male]/_b[yedu\#0.male] = _b[yexp\#1.male]/_b[yexp\#0.male]
    (2) _b[yedu\#1.male]/_b[yedu\#0.male] = _b[yexp2\#1.male]/_b[yexp2\#0.male]
        \(\operatorname{chi2}(2)=9.21\)
    Prob > chi2 = 0.0100

We conclude that we find fairly strong evidence against the proportionality of the coefficients, and hence we have to conclude that success in the labor market is produced in different ways by men and women. (But remember, these were simulated data.)

\section*{> Example 3}

The syntax for specifying the equality of multiple expressions is just a convenient shorthand for specifying a series of constraints, namely, that the first expression equals the second expression, the first expression also equals the third expression, etc. The Wald test performed and the output of testnl are the same whether we use the shorthand or we specify the series of constraints.
```

. testnl (_b[yedu\#1.male]/_b[yedu\#0.male] =
> _b[yexp\#1.male]/_b[yexp\#0.male])
> (_b[yedu\#1.male]/_b[yedu\#0.male] =
> _b[yexp2\#1.male]/_b[yexp2\#0.male])
(1) _b[yedu\#1.male]/_b[yedu\#0.male] = _b[yexp\#1.male]/_b[yexp\#0.male]
(2) _b[yedu\#1.male]/_b[yedu\#0.male] = _b[yexp2\#1.male]/_b[yexp2\#0.male]
chi2(2) = 9.21
Prob > chi2 = 0.0100

```

Having established differences between men and women, we would like to do multiple testing between the ratios. Because we did not specify hypotheses in advance, we prefer to adjust the \(p\)-values of tests using, here, Bonferroni's method.
```

. testnl _b[yedu\#1.male]/_b[yedu\#0.male] =
> _b[yexp\#1.male]/_b[yexp\#0.male] =
> _b[yexp2\#1.male]/_b[yexp2\#0.male], mtest(b)
(1) _b[yedu\#1.male]/_b[yedu\#0.male] = _b[yexp\#1.male]/_b[yexp\#0.male]
(2) _b[yedu\#1.male]/_b[yedu\#0.male] = _b[yexp2\#1.male]/_b[yexp2\#0.male]

```
\begin{tabular}{c|ccc}
\hline & chi2 & df & \(\mathrm{p}>\mathrm{chi2}\) \\
\hline\((1)\) & 6.89 & 1 & \(0.0173 *\) \\
\((2)\) & 0.93 & 1 & \(0.6713^{*}\) \\
\hline All & 9.21 & 2 & 0.0100 \\
\hline
\end{tabular}

\footnotetext{
* Bonferroni-adjusted p-values
}

\section*{Manipulability}

Although testnl allows you to specify constraints in different ways that are mathematically equivalent, as noted above, this does not mean that the tests are the same. This difference is known as the manipulability of the Wald test for nonlinear hypotheses; also see [R] boxcox. The test might even be significant for one formulation but not significant for another formulation that is mathematically
equivalent. Trying out different specifications to find a formulation with the desired \(p\)-value is totally inappropriate, though it may actually be fun to try. There is no variance under representation because the nonlinear Wald test is actually a standard Wald test for a linearization of the constraint, which depends on the particular specification. We note that the likelihood-ratio test is not manipulable in this sense.

From a statistical point of view, it is best to choose a specification of the constraints that is as linear is possible. Doing so usually improves the accuracy of the approximation of the null-distribution of the test by a \(\chi^{2}\) or an \(F\) distribution. The example above used the nonlinear Wald test to test whether the coefficients of human capital variables for men were proportional to those of women. A specification of proportionality of coefficients in terms of ratios of coefficients is fairly nonlinear if the coefficients in the denominator are close to 0 . A more linear version of the test results from a bilinear formulation. Thus, instead of
```

. testnl _b[yedu\#1.male]/_b[yedu\#0.male] = _b[yexp\#1.male]/_b[yexp\#0.male]
(1) _b[yedu\#1.male]/_b[yedu\#0.male] = _b[yexp\#1.male]/_b[yexp\#0.male]
chi2(1) = 6.89
Prob > chi2 = 0.0087

```
perhaps
```

. testnl _b[yedu\#1.male]*_b[yexp\#0.male] = _b[yedu\#0.male]*_b[yexp\#1.male]
(1) _b[yedu\#1.male]*_b[yexp\#0.male] = _b[yedu\#0.male]*_b[yexp\#1.male]
chi2(1) = 13.95
Prob > chi2 = 0.0002

```
is better, and in fact it has been suggested that the latter version of the test is more reliable. This assertion is confirmed by performing simulations and is in line with theoretical results of Phillips and Park (1988). There is strong evidence against the proportionality of human capital effects between men and women, implying for this example that differences in the residual variances between the sexes can be ruled out as the explanation of the sex differences in the analysis of labor market participation.

\section*{Stored results}
testnl stores the following in \(r()\) :
Scalars
\(r(\mathrm{df}) \quad\) degrees of freedom
\(r\left(d f \_r\right) \quad\) residual degrees of freedom
\(r\) (chi2) \(\quad \chi^{2}\)
\(r(p) \quad p\)-value for Wald test
\(r(\mathrm{~F}) \quad F\) statistic
Macros
\(r\) (mtmethod) method specified in mtest()
Matrices
r(G) derivatives of \(R(\mathbf{b})\) with respect to \(\mathbf{b}\); see Methods and formulas below
\(\mathrm{r}(\mathrm{R}) \quad R(\mathbf{b})-\mathbf{q}\); see Methods and formulas below
\(r\) (mtest) multiple test results

\section*{Methods and formulas}

After fitting a model, define \(\mathbf{b}\) as the resulting \(1 \times k\) parameter vector and \(\mathbf{V}\) as the \(k \times k\) covariance matrix. The (linear or nonlinear) hypothesis is given by \(R(\mathbf{b})=\mathbf{q}\), where \(R\) is a function returning a \(j \times 1\) vector. The Wald test formula is (Greene 2018, 512-513)
\[
W=\{R(\mathbf{b})-\mathbf{q}\}^{\prime}\left(\mathbf{G} \mathbf{V G}{ }^{\prime}\right)^{-1}\{R(\mathbf{b})-\mathbf{q}\}
\]
where \(\mathbf{G}\) is the derivative matrix of \(R(\mathbf{b})\) with respect to \(\mathbf{b}\). \(W\) is distributed as \(\chi^{2}\) if \(\mathbf{V}\) is an asymptotic covariance matrix. \(F=W / j\) is distributed as \(F\) for linear regression.

The adjustment methods for multiple testing are described in \([R]\) test. The adjustment for survey design effects is described in [SVY] svy postestimation.

\section*{References}

Allison, P. D. 1999. Comparing logit and probit coefficients across groups. Sociological Methods and Research 28: 186-208. https://doi.org/10.1177/0049124199028002003.

Greene, W. H. 2018. Econometric Analysis. 8th ed. New York: Pearson.
Long, J. S. 1997. Regression Models for Categorical and Limited Dependent Variables. Thousand Oaks, CA: Sage.
Phillips, P. C. B., and J. Y. Park. 1988. On the formulation of Wald tests of nonlinear restrictions. Econometrica 56: 1065-1083. https://doi.org/10.2307/1911359.

\section*{Also see}
[R] contrast - Contrasts and linear hypothesis tests after estimation
[ \(R\) ] lincom - Linear combinations of parameters
[R] Irtest - Likelihood-ratio test after estimation
[R] nlcom - Nonlinear combinations of parameters
[R] test - Test linear hypotheses after estimation
[U] 13.5 Accessing coefficients and standard errors
[U] 20 Estimation and postestimation commands

\section*{Title}
tetrachoric - Tetrachoric correlations for binary variables
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
tetrachoric computes estimates of the tetrachoric correlation coefficients of the binary variables in varlist. All of these variables should be 0,1 , or missing values.

Tetrachoric correlations assume a latent bivariate normal distribution ( \(X_{1}, X_{2}\) ) for each pair of variables \(\left(v_{1}, v_{2}\right)\), with a threshold model for the manifest variables, \(v_{i}=1\) if and only if \(X_{i}>0\). The means and variances of the latent variables are not identified, but the correlation, \(r\), of \(X_{1}\) and \(X_{2}\) can be estimated from the joint distribution of \(v_{1}\) and \(v_{2}\) and is called the tetrachoric correlation coefficient.
tetrachoric computes pairwise estimates of the tetrachoric correlations by the (iterative) maximum likelihood estimator obtained from bivariate probit without explanatory variables (see [R] biprobit) by using the Edwards and Edwards (1984) noniterative estimator as the initial value.

The pairwise correlation matrix is returned as \(r\) (Rho) and can be used to perform a factor analysis or a principal component analysis of binary variables by using the factormat or pcamat commands; see [MV] factor and [MV] pca.

\section*{Quick start}

Tetrachoric correlation of v1 and v2 with standard error and test of independence tetrachoric v1 v2

Matrix of pairwise tetrachoric correlations for \(\mathrm{v} 1, \mathrm{v} 2\), and v 3
tetrachoric v1 v2 v3
Add standard errors and \(p\)-values
```

tetrachoric v1 v2 v3, stats(rho se p)

```

Same as above, but adjust \(p\)-values for multiple comparisons using Bonferroni's method tetrachoric v1 v2 v3, stats(rho se p) bonferroni

Add star to correlations significant at the 5\% level
tetrachoric v1 v2 v3, star(.05)
Use all available data for each pair of variables and report number of observations used
tetrachoric v1 v2 v3, pw stats(rho obs)
Adjust correlation matrix to be positive semidefinite
tetrachoric v1 v2 v3, posdef

\section*{Menu}

Statistics \(>\) Summaries, tables, and tests \(>\) Summary and descriptive statistics \(>\) Tetrachoric correlations

\section*{Syntax}
tetrachoric varlist [if] [in] [weight] [, options]
options Description

Main
stats (statlist) list of statistics; select up to 4 statistics; default is stats (rho)
edwards
print (\#) significance level for displaying coefficients
star (\#) significance level for displaying with a star
bonferroni
sidak
pw
zeroadjust
matrix
notable
posdef
use the noniterative Edwards and Edwards estimator; default is the maximum likelihood estimator
use Bonferroni-adjusted significance level
use Šidák-adjusted significance level data (pairwise deletion)
adjust frequencies when one cell has a zero count
display output in matrix form
suppress display of correlations
modify correlation matrix to be positive semidefinite
calculate all the pairwise correlation coefficients by using all available
statlist Description
rho tetrachoric correlation coefficient
se standard error of rho
obs number of observations
p
exact two-sided significance level
by and collect are allowed; see [U] 11.1.10 Prefix commands.
fweights are allowed; see [U] 11.1.6 weight.

\section*{Options}
\(\qquad\) Main
stats (statlist) specifies the statistics to be displayed in the matrix of output. stats(rho) is the default. Up to four statistics may be specified. stats (rho se p obs) would display the tetrachoric correlation coefficient, its standard error, the significance level, and the number of observations. If varlist contains only two variables, all statistics are shown in tabular form. stats(), print(), and star() have no effect unless the matrix option is also specified.
edwards specifies that the noniterative Edwards and Edwards estimator be used. The default is the maximum likelihood estimator. If you analyze many binary variables, you may want to use the fast noniterative estimator proposed by Edwards and Edwards (1984). However, if you have skewed variables, the approximation does not perform well.
print (\#) specifies the maximum significance level of correlation coefficients to be printed. Correlation coefficients with larger significance levels are left blank in the matrix. Typing tetrachoric ..., print (.10) would list only those correlation coefficients that are significant at the \(10 \%\) level or lower.
star(\#) specifies the maximum significance level of correlation coefficients to be marked with a star. Typing tetrachoric ..., star(.05) would "star" all correlation coefficients significant at the \(5 \%\) level or lower.
bonferroni makes the Bonferroni adjustment to calculated significance levels. This option affects printed significance levels and the print() and star() options. Thus, tetrachoric..., print(.05) bonferroni prints coefficients with Bonferroni-adjusted significance levels of 0.05 or less.
sidak makes the Šidák adjustment to calculated significance levels. This option affects printed significance levels and the print() and star() options. Thus, tetrachoric..., print (.05) sidak prints coefficients with Šidák-adjusted significance levels of 0.05 or less.
pw specifies that the tetrachoric correlation be calculated by using all available data. By default, tetrachoric uses casewise deletion, where observations are ignored if any of the specified variables in varlist are missing.
zeroadjust specifies that when one of the cells has a zero count, a frequency adjustment be applied in such a way as to increase the zero to one-half and maintain row and column totals.
matrix forces tetrachoric to display the statistics as a matrix, even if varlist contains only two variables. matrix is implied if more than two variables are specified.
notable suppresses the output.
posdef modifies the correlation matrix so that it is positive semidefinite, that is, a proper correlation matrix. The modified result is the correlation matrix associated with the least-squares approximation of the tetrachoric correlation matrix by a positive-semidefinite matrix. If the correlation matrix is modified, the standard errors and significance levels are not displayed and are returned in \(r()\).

\section*{Remarks and examples}

Remarks are presented under the following headings:
Association in 2-by-2 tables
Factor analysis of dichotomous variables
Tetrachoric correlations with simulated data

\section*{Association in 2-by-2 tables}

Although a wide variety of measures of association in cross tabulations have been proposed, such measures are essentially equivalent (monotonically related) in the special case of \(2 \times 2\) tables-there is only 1 degree of freedom for nonindependence. Still, some measures have more desirable properties than others. Here we compare two measures: the standard Pearson correlation coefficient and the tetrachoric correlation coefficient. Given asymmetric row or column margins, Pearson correlations are limited to a range smaller than -1 to 1 , although tetrachoric correlations can still span the range from -1 to 1 . To illustrate, consider the following set of tables for two binary variables, X and Z :
\begin{tabular}{l|cc|c} 
& \(\mathrm{Z}=0\) & \(\mathrm{Z}=1\) & \\
\hline \(\mathrm{X}=0\) & \(20-a\) & \(10+a\) & 30 \\
\(\mathrm{X}=1\) & \(a\) & \(10-a\) & 10 \\
\hline & 20 & 20 & 40
\end{tabular}

For \(a\) equal to \(0,1,2,5,8,9\), and 10 , the Pearson and tetrachoric correlations for the above table are
\begin{tabular}{r|ccccccc}
\(a\) & 0 & 1 & 2 & 5 & 8 & 9 & 10 \\
\hline Pearson & 0.577 & 0.462 & 0.346 & 0 & -0.346 & -0.462 & -0.577 \\
Tetrachoric & 1.000 & 0.792 & 0.607 & 0 & -0.607 & -0.792 & -1.000 \\
\hline
\end{tabular}

The restricted range for the Pearson correlation is especially unfortunate when you try to analyze the association between binary variables by using models developed for continuous data, such as factor analysis and principal component analysis.

The tetrachoric correlation of two variables \(\left(Y_{1}, Y_{2}\right)\) can be thought of as the Pearson correlation of two latent bivariate normal distributed variables \(\left(Y_{1}^{*}, Y_{2}^{*}\right)\) with threshold measurement models \(Y_{i}=\left(Y_{i}^{*}>c_{i}\right)\) for unknown cutpoints \(c_{i}\). Or equivalently, \(Y_{i}=\left(Y_{i}^{* *}>0\right)\) where the latent bivariate normal \(\left(Y_{1}^{* *}, Y_{2}^{* *}\right)\) are shifted versions of \(\left(Y_{1}^{*}, Y_{2}^{*}\right)\) so that the cutpoints are zero. Obviously, you must judge whether assuming underlying latent variables is meaningful for the data. If this assumption is justified, tetrachoric correlations have two advantages. First, you have an intuitive understanding of the size of correlations that are substantively interesting in your field of research, and this intuition is based on correlations that range from -1 to 1 . Second, because the tetrachoric correlation for binary variables estimates the Pearson correlation of the latent continuous variables (assumed multivariate normal distributed), you can use the tetrachoric correlations to analyze multivariate relationships between the dichotomous variables. When doing so, remember that you must interpret the model in terms of the underlying continuous variables.

\section*{> Example 1}

To illustrate tetrachoric correlations, we examine three binary variables from the familyvalues dataset (described in example 2).

\begin{tabular}{l|ccc}
\begin{tabular}{l} 
(obs=3,291) \\
(obstrachoric
\end{tabular} & RS074 RS075 & RS076 & \\
& RS074 & RS075 & RS076 \\
\hline RS074 & 1.0000 & & \\
RS075 & 0.0689 & 1.0000 & \\
RS076 & 0.2480 & 0.6427 & 1.0000
\end{tabular}

As usual, the tetrachoric correlation coefficients are larger (in absolute value) and more dispersed than the Pearson correlations.

\section*{Factor analysis of dichotomous variables}

\section*{> Example 2}

Factor analysis is a popular model for measuring latent continuous traits. The standard estimators are appropriate only for continuous unimodal data. Because of the skewness implied by Bernoullidistributed variables (especially when the probability is distributed unevenly), a factor analysis of a Pearson correlation matrix can be rather misleading when used in this context. A factor analysis of a matrix of tetrachoric correlations is more appropriate under these conditions (Uebersax 2000). We illustrate this with data on gender, relationship, and family attitudes of spouses using the Households in The Netherlands survey 1995 (Weesie et al. 1995). For attitude variables, it seems reasonable to assume that agreement or disagreement is just a coarse measurement of more nuanced underlying attitudes.

To demonstrate, we examine a few of the variables from the familyvalues dataset.
\begin{tabular}{|c|c|c|c|c|}
\hline Variable name & Storage type & Display format & \begin{tabular}{l}
Value \\
label
\end{tabular} & Variable label \\
\hline RS056 & byte & \% 9.0 g & & Fam att: should be together \\
\hline RS057 & byte & \% 9.0g & & Fam att: should fight for relat \\
\hline RS058 & byte & \%9.0g & & Fam att: should avoid conflict \\
\hline RS059 & byte & \(\% 9.0 \mathrm{~g}\) & & Fam att: woman better nurturer \\
\hline RS060 & byte & \%9.0g & & Fam att: both spouses money goo \\
\hline RS061 & byte & \%9.0g & & Fam att: woman techn school goo \\
\hline RS062 & byte & \%9.0g & & Fam att: man natural breadwinne \\
\hline RS063 & byte & \%9.0g & & Fam att: common leisure good \\
\hline
\end{tabular}
. summarize RS056-RS063
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline RS056 & 3,298 & .5630685 & .4960816 & 0 & 1 \\
RS057 & 3,296 & .5400485 & .4984692 & 0 & 1 \\
RS058 & 3,283 & .6387451 & .4804374 & 0 & 1 \\
RS059 & 3,308 & .654474 & .4756114 & 0 & 1 \\
RS060 & 3,302 & .3906723 & .487975 & 0 & 1 \\
\hline RS061 & 3,293 & .7102946 & .4536945 & 0 & 1 \\
RS062 & 3,307 & .5857272 & .4926705 & 0 & 1 \\
RS063 & 3,298 & .5379018 & .498637 & 0 & 1
\end{tabular}
. correlate RS056-RS063
(obs=3,221)
\begin{tabular}{r|rrrrrrr} 
& RS056 & RS057 & RS058 & RS059 & RS060 & RS061 & RS062 \\
\hline RS056 & 1.0000 & & & & & & \\
RS057 & 0.1350 & 1.0000 & & & & & \\
RS058 & 0.2377 & 0.0258 & 1.0000 & & & & \\
RS059 & 0.1816 & 0.0097 & 0.2550 & 1.0000 & & & \\
RS060 & -0.1020 & -0.0538 & -0.0424 & 0.0126 & 1.0000 & & \\
RS061 & -0.1137 & 0.0610 & -0.1375 & -0.2076 & 0.0706 & 1.0000 & \\
RS062 & 0.2014 & 0.0285 & 0.2273 & 0.4098 & -0.0793 & -0.2873 & 1.0000 \\
RS063 & 0.2057 & 0.1460 & 0.1049 & 0.0911 & 0.0179 & -0.0233 & 0.0975 \\
& RS063 & & & & & & \\
\hline RS063 & 1.0000 & & & & & & \\
& & & & & & &
\end{tabular}

Skewness in these data is relatively modest. For comparison, here are the tetrachoric correlations:
\begin{tabular}{l|rrrrrrr}
\begin{tabular}{c}
. tetrachoric RS056-RS063 \\
(obs=3,221)
\end{tabular} & RS056 & RS057 & RS058 & RS059 & RS060 & RS061 & RS062 \\
& RS056 & 1.0000 & & & & & \\
RS057 & 0.2114 & 1.0000 & & & & \\
RS058 & 0.3716 & 0.0416 & 1.0000 & & & \\
RS059 & 0.2887 & 0.0158 & 0.4007 & 1.0000 & & & \\
RS060 & -0.1620 & -0.0856 & -0.0688 & 0.0208 & 1.0000 & & \\
RS061 & -0.1905 & 0.1011 & -0.2382 & -0.3664 & 0.1200 & 1.0000 & \\
RS062 & 0.3135 & 0.0452 & 0.3563 & 0.6109 & -0.1267 & -0.4845 & 1.0000 \\
RS063 & 0.3187 & 0.2278 & 0.1677 & 0.1467 & 0.0286 & -0.0388 & 0.1538 \\
& RS063 & & & & & & \\
\hline RS063 & 1.0000 & & & & & & \\
\hline
\end{tabular}

Again, we see that the tetrachoric correlations are generally larger in absolute value than the Pearson correlations. The bivariate probit and Edwards and Edwards estimators (the edwards option) implemented in tetrachoric may return a correlation matrix that is not positive semidefinite-a mathematical property of any real correlation matrix. Positive definiteness is required by commands for analyses of correlation matrices, such as factormat and pcamat; see [MV] factor and [MV] pca. The posdef option of tetrachoric tests for positive definiteness and projects the estimated correlation matrix to a positive-semidefinite matrix if needed.
```

. tetrachoric RS056-RS063, notable posdef
. matrix C = r(Rho)

```

This time, we suppressed the display of the correlations with the notable option and requested that the correlation matrix be positive semidefinite with the posdef option. Had the correlation matrix not been positive definite, tetrachoric would have displayed a warning message and then adjusted the matrix to be positive semidefinite. We placed the resulting tetrachoric correlation matrix into a matrix, C, so that we can perform a factor analysis upon it.
tetrachoric with the posdef option asserted that C was positive definite because no warning message was displayed. We can verify this by using a familiar characterization of symmetric positivedefinite matrices: all eigenvalues are real and positive.
```

. matrix symeigen eigenvectors eigenvalues = C
. matrix list eigenvalues
eigenvalues[1,8]

```


We can proceed with a factor analysis on the matrix C. We use factormat and select iterated principal factors as the estimation method; see [MV] factor.
. factormat C, n (3221) ipf factor(2)
(obs=3,221)
Factor analysis/correlation \(\quad\) Number of obs \(=3,221\)
Method: iterated principal factors \(\quad\) Retained factors \(=\quad 2\)
Rotation: (unrotated)
Number of params \(=\quad 15\)
\begin{tabular}{c|rrrr}
\hline Factor & Eigenvalue & Difference & Proportion & Cumulative \\
\hline Factor1 & 2.06855 & 1.40178 & 0.7562 & 0.7562 \\
Factor2 & 0.66677 & 0.47180 & 0.2438 & 1.0000 \\
Factor3 & 0.19497 & 0.06432 & 0.0713 & 1.0713 \\
Factor4 & 0.13065 & 0.10967 & 0.0478 & 1.1191 \\
Factor5 & 0.02098 & 0.10085 & 0.0077 & 1.1267 \\
Factor6 & -0.07987 & 0.01037 & -0.0292 & 1.0975 \\
Factor7 & -0.09024 & 0.08626 & -0.0330 & 1.0645 \\
Factor8 & -0.17650 &. & -0.0645 & 1.0000 \\
\hline
\end{tabular}

LR test: independent vs. saturated: chi2(28) \(=4620.01\) Prob>chi2 \(=0.0000\)
Factor loadings (pattern matrix) and unique variances
\begin{tabular}{r|rr|r}
\hline Variable & Factor1 & Factor2 & Uniqueness \\
\hline RS056 & 0.5528 & 0.4120 & 0.5247 \\
RS057 & 0.1124 & 0.4214 & 0.8098 \\
RS058 & 0.5333 & 0.0718 & 0.7105 \\
RS059 & 0.6961 & -0.1704 & 0.4865 \\
RS060 & -0.1339 & -0.0596 & 0.9785 \\
RS061 & -0.5126 & 0.2851 & 0.6560 \\
RS062 & 0.7855 & -0.2165 & 0.3361 \\
RS063 & 0.2895 & 0.3919 & 0.7626 \\
\hline
\end{tabular}

\section*{Example 3}

We noted in example 2 that the matrix of estimates of the tetrachoric correlation coefficients need not be positive definite. Here is an example:
\begin{tabular}{|c|c|c|c|c|c|c|c|}
\hline \multicolumn{8}{|l|}{```
matrix with tetrachoric correlations is not positive semidefinite;
    it has 2 negative eigenvalues
    maxdiff(corr,adj-corr) = 0.2346
    (adj-corr: tetrachoric correlations adjusted to be positive semidefinite)
```} \\
\hline adj-corr & RS056 & RS057 & RS058 & RS059 & RS060 & RS061 & RS062 \\
\hline RS056 & 1.0000 & & & & & & \\
\hline RS057 & 0.5284 & 1.0000 & & & & & \\
\hline RS058 & 0.3012 & 0.2548 & 1.0000 & & & & \\
\hline RS059 & 0.3251 & 0.2791 & 0.0550 & 1.0000 & & & \\
\hline RS060 & -0.5197 & -0.4222 & -0.7163 & 0.0552 & 1.0000 & & \\
\hline RS061 & 0.3448 & 0.4815 & -0.0958 & -0.1857 & -0.0980 & 1.0000 & \\
\hline RS062 & 0.1066 & -0.0375 & 0.0072 & 0.3909 & -0.2333 & -0.7654 & 1.0000 \\
\hline RS063 & 0.3830 & 0.4939 & 0.4336 & 0.0075 & -0.8937 & -0.0337 & 0.4934 \\
\hline adj-corr & RS063 & & & & & & \\
\hline RS063 & 1.0000 & & & & & & \\
\hline
\end{tabular}
```

. mata:
_ mata (type end to exit) ___
: C2 = st_matrix("r(Rho)")
: eigenvecs = .
: eigenvals = .
: symeigensystem(C2, eigenvecs, eigenvals)
: eigenvals
1 2 3 4
1
4.156592567
1 . .4845368741 . 2131895139 -1.11022e-16 -2.27918e-16
: end

```

The estimated tetrachoric correlation matrix is rank-2 deficient. With this C2 matrix, we can only use models of correlation that allow for singular cases.

\section*{Tetrachoric correlations with simulated data}

\section*{> Example 4}

We use drawnorm (see [D] drawnorm) to generate a sample of 1,000 observations from a bivariate normal distribution with means -1 and 1 , unit variances, and correlation 0.4.
```

. clear
. set seed 11000
. matrix m = (1, -1)
. matrix V = (1, 0.4 \ 0.4, 1)
. drawnorm c1 c2, n(1000) means(m) cov(V)
(obs 1,000)

```

Now, consider the measurement model assumed by the tetrachoric correlations. We observe only whether c1 and c2 are greater than zero,


We want to estimate the correlation of c 1 and c 2 from the binary variables d 1 and d 2 . Pearson's correlation of the binary variables d 1 and d 2 is 0.129 -a seriously biased estimate of the underlying correlation \(\rho=0.4\).
```

. correlate d1 d2
(obs=1,000)

|  | d 1 | d 2 |
| ---: | ---: | ---: |
| d 1 | 1.0000 |  |
| d 2 | 0.1294 | 1.0000 |

```

The tetrachoric correlation coefficient of d 1 and d2 estimates the Pearson correlation of the latent continuous variables, c1 and c2.
```

. tetrachoric d1 d2
Number of obs = 1,000
Tetrachoric rho = 0.3875
Std error = 0.0787
Test of H0: d1 and d2 are independent
2-sided exact P = 0.0000

```

The estimate of the tetrachoric correlation of d 1 and \(\mathrm{d} 2,0.3875\), is much closer to the underlying correlation, 0.4 , between c 1 and c 2 .

\section*{Stored results}
```

tetrachoric stores the following in r():

```

Scalars
\(r\) (rho) tetrachoric correlation coefficient between variables 1 and 2
\(r(N) \quad\) number of observations
\(r\) (nneg) number of negative eigenvalues (posdef only)
\(r\) (se_rho) standard error of \(r\) (rho)
\(r(\mathrm{p}) \quad p\)-value for two-sided Fisher's exact test (for the first two variables)
Macros
\(r\) (method) estimator used
Matrices
\(r\) (Rho) tetrachoric correlation matrix
\(r\) (Se_Rho) standard errors of \(r\) (Rho)
\(r\) (Nobs) number of observations used in computing correlation
\(r(\mathrm{P}) \quad\) matrix of \(p\)-values for two-sided Fisher's exact test

\section*{Methods and formulas}
tetrachoric provides two estimators for the tetrachoric correlation \(\rho\) of two binary variables with the frequencies \(n_{i j}, i, j=0,1\). tetrachoric defaults to the slower (iterative) maximum likelihood estimator obtained from bivariate probit without explanatory variables (see [R] biprobit) by using the Edwards and Edwards noniterative estimator as the initial value. A fast (noniterative) estimator is also available by specifying the edwards option (Edwards and Edwards 1984; Digby 1983)
\[
\widehat{\rho}=\frac{\alpha-1}{\alpha+1}
\]
where
\[
\alpha=\left(\frac{n_{00} n_{11}}{n_{01} n_{10}}\right)^{\pi / 4}(\pi=3.14 \ldots)
\]
if all \(n_{i j}>0\). If \(n_{00}=0\) or \(n_{11}=0, \widehat{\rho}=-1\); if \(n_{01}=0\) or \(n_{10}=0, \widehat{\rho}=1\).
The asymptotic variance of the Edwards and Edwards estimator of the tetrachoric correlation is easily obtained by the delta method,
\[
\operatorname{avar}(\widehat{\rho})=\left\{\frac{\pi \alpha}{2(1+\alpha)^{2}}\right\}^{2}\left(\frac{1}{n_{00}}+\frac{1}{n_{01}}+\frac{1}{n_{10}}+\frac{1}{n_{11}}\right)
\]
provided all \(n_{i j}>0\), otherwise it is left undefined (missing). The Edwards and Edwards estimator is fast, but may be inaccurate if the margins are very skewed.
tetrachoric reports exact \(p\)-values for statistical independence, computed by the exact option of \([R]\) tabulate twoway.

\section*{References}

Brown, M. B. 1977. Algorithm AS 116: The tetrachoric correlation and its asymptotic standard error. Applied Statistics 26: 343-351. https://doi.org/10.2307/2346985.

Brown, M. B., and J. K. Benedetti. 1977. On the mean and variance of the tetrachoric correlation coefficient. Psychometrika 42: 347-355. https://doi.org/10.1007/BF02293655.
Digby, P. G. N. 1983. Approximating the tetrachoric correlation coefficient. Biometrics 39: 753-757. https://doi.org/10.2307/2531104.
Edwards, J. H., and A. W. F. Edwards. 1984. Approximating the tetrachoric correlation coefficient. Biometrics 40: 563.

Golub, G. H., and C. F. Van Loan. 2013. Matrix Computations. 4th ed. Baltimore: Johns Hopkins University Press.
Uebersax, J. S. 2000. Estimating a latent trait model by factor analysis of tetrachoric correlations. https://www.john-uebersax.com/stat/irt.htm.
Weesie, J., M. Kalmijn, W. Bernasco, and D. Giesen. 1995. Households in The Netherlands 1995. Utrecht, Netherlands: Datafile, ISCORE, University of Utrecht.

\section*{Also see}
[R] biprobit - Bivariate probit regression
[R] correlate - Correlations of variables
[R] spearman - Spearman's and Kendall's correlations
[R] tabulate twoway - Two-way table of frequencies
[MV] factor - Factor analysis
[MV] pca - Principal component analysis

\section*{Title}
tnbreg - Truncated negative binomial regression
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
Acknowledgment & References & Also see &
\end{tabular}

\section*{Description}
tnbreg estimates the parameters of a truncated negative binomial model by maximum likelihood. The dependent variable depvar is regressed on indepvars, where depvar is a positive count variable whose values are all above the truncation point.

\section*{Quick start}

Truncated negative binomial regression of y on x with truncation at 0
tnbreg \(\mathrm{y} x\)
Report incidence-rate ratios
tnbreg \(\mathrm{y} x\), irr
Add categorical variable a using factor variable syntax
tnbreg y x i.a
Same as above, but specify a constant truncation point of 2
tnbreg y x i.a, ll(2)
With exposure variable exp
tnbreg y x i.a, exposure (exp)
Same as above, but specifying a variable truncation point stored in variable min
tnbreg y x i.a, exposure (exp) ll(min)
With cluster-robust standard errors clustering by the levels of cvar
tnbreg y x i.a, exposure(exp) ll(min) vce(cluster cvar)

\section*{Menu}

Statistics \(>\) Count outcomes \(>\) Truncated negative binomial regression

\section*{Syntax}
tnbreg depvar [indepvars] [if][in] [weight] [, options]
options
Model
noconstant
ll (\#|varname)
dispersion(mean)
dispersion(constant)
exposure (varname \({ }_{e}\) )
offset ( varname \(_{o}\) )
constraints (constraints)
SE/Robust
vce (vcetype)

Reporting
level(\#)
nolrtest
irr
nocnsreport
display_options

Description
suppress constant term
truncation point; default value is \(11(0)\), zero truncation
parameterization of dispersion; the default
constant dispersion for all observations
include \(\ln \left(\right.\) varname \(\left._{e}\right)\) in model with coefficient constrained to 1
include varname \(_{o}\) in model with coefficient constrained to 1 apply specified linear constraints
vcetype may be oim, robust, cluster clustvar, opg, bootstrap, or jackknife
set confidence level; default is level (95)
suppress likelihood-ratio test
report incidence-rate ratios
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

Maximization
maximize_options
collinear
coeflegend
control the maximization process; seldom used
keep collinear variables
display legend instead of statistics
indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
depvar and indepvars may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bayes, bootstrap, by, collect, fp, jackknife, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: tnbreg.
Weights are not allowed with the bootstrap prefix; see \([R]\) bootstrap.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options}
\(\qquad\) Model
noconstant; see [R] Estimation options.
11 (\#|varname) specifies the truncation point, which is a nonnegative integer. The default is zero truncation, ll(0).
dispersion(mean |constant) specifies the parameterization of the model. dispersion(mean), the default, yields a model with dispersion equal to \(1+\alpha \exp \left(\mathbf{x}_{j} \boldsymbol{\beta}+\right.\) offset \(\left._{j}\right)\); that is, the dispersion is a function of the expected mean: \(\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}+\right.\) offset \(\left._{j}\right)\). dispersion (constant) has dispersion equal to \(1+\delta\); that is, it is a constant for all observations.
exposure ( varname \(_{e}\) ), offset ( varname \(_{o}\) ), constraints (constraints); see [R] Estimation options.

\section*{SE/Robust}
vce(vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.

\section*{Reporting}
level(\#); see [R] Estimation options.
nolrtest suppresses fitting the Poisson model. Without this option, a comparison Poisson model is fit, and the likelihood is used in a likelihood-ratio test of the null hypothesis that the dispersion parameter is zero.
irr reports estimated coefficients transformed to incidence-rate ratios, that is, \(e^{\beta_{i}}\) rather than \(\beta_{i}\). Standard errors and confidence intervals are similarly transformed. This option affects how results are displayed, not how they are estimated or stored. irr may be specified at estimation or when replaying previously estimated results.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fint), pformat (\%fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate (\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from (init_specs); see [R] Maximize. These options are seldom used.

Setting the optimization type to technique (bhhh) resets the default vcetype to vce (opg).
The following options are available with tnbreg but are not shown in the dialog box: collinear, coeflegend; see [R] Estimation options.

\section*{Remarks and examples}

Grogger and Carson (1991) showed that overdispersion causes inconsistent estimation of the mean in the truncated Poisson model. To solve this problem, they proposed using the truncated negative binomial model as an alternative. If data are truncated but do not exhibit overdispersion, the truncated Poisson model is more appropriate; see [R] tpoisson. For an introduction to negative binomial regression, see Cameron and Trivedi (2005, 2022) and Long and Freese (2014). For an introduction to truncated negative binomial models, see Cameron and Trivedi (2013) and Long (1997, chap. 8).
tnbreg fits the mean-dispersion and the constant-dispersion parameterizations of truncated negative binomial models. These parameterizations extend those implemented in nbreg; see [R] nbreg.

\section*{> Example 1}

We illustrate the truncated negative binomial model using the 1997 MedPar dataset (Hilbe 1999). The data are from 1,495 patients in Arizona who were assigned to a diagnostic-related group (DRG) of patients having a ventilator. Length of stay (los), the dependent variable, is a positive integer; it cannot have zero values. The data are truncated because there are no observations on individuals who stayed for zero days.

The objective of this example is to determine whether the length of stay was related to the binary variables: died, hmo, type1, type2, and type3.

The died variable was recorded as a 0 unless the patient died, in which case, it was recorded as a 1 . The other variables also adopted this encoding. The hmo variable was set to 1 if the patient belonged to a health maintenance organization (HMO).

The type1-type3 variables indicated the type of admission used for the patient. The type1 variable indicated an emergency admit. The type 2 variable indicated an urgent admit-that is, the first available bed. The type3 variable indicated an elective admission. Because type1-type3 were mutually exclusive, only two of the three could be used in the truncated negative binomial regression shown below.
```

. use https://www.stata-press.com/data/r18/medpar
(Arizona ventilator data)
. tnbreg los died hmo type2-type3, vce(cluster provnum) nolog
Truncated negative binomial regression Number of obs = 1,495
Truncation point = 0 Wald chi2(4) = 36.01
Dispersion: mean Prob > chi2 = 0.0000
Log pseudolikelihood = -4737.535 Pseudo R2 = 0.0139

```
(Std. err. adjusted for 54 clusters in provnum)
\begin{tabular}{r|rrrrrr}
\hline & & \multicolumn{3}{|c}{ Robust } \\
los & Coefficient & std. err. & \multicolumn{1}{c}{z} & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline died & -.2521884 & .061533 & -4.10 & 0.000 & -.3727908 & -.1315859 \\
hmo & -.0754173 & .0533132 & -1.41 & 0.157 & -.1799091 & .0290746 \\
type2 & .2685095 & .0666474 & 4.03 & 0.000 & .137883 & .3991359 \\
type3 & .7668101 & .2183505 & 3.51 & 0.000 & .338851 & 1.194769 \\
_cons & 2.224028 & .034727 & 64.04 & 0.000 & 2.155964 & 2.292091 \\
\hline /lnalpha & -.630108 & .0764019 & & & -.779853 & -.480363 \\
\hline alpha & .5325343 & .0406866 & & & .4584734 & .6185588 \\
\hline
\end{tabular}

Because observations within the same hospital (provnum) are likely to be correlated, we specified the vce(cluster provnum) option. The results show that whether the patient died in the hospital and the type of admission have significant effects on the patient's length of stay.

\section*{> Example 2}

To illustrate truncated negative binomial regression with more complex data than the previous example, similar data were created from 100 hospitals. Each hospital had its own way of tracking patient data. In particular, hospitals only recorded data from patients with a minimum length of stay, denoted by the variable minstay.

Definitions for minimum length of stay varied among hospitals, typically, from 5 to 18 days. The objective of this example is the same as before: to determine whether the length of stay, recorded in los, was related to the binary variables: died, hmo, type1, type2, and type3.

The binary variables encode the same information as in example 1 above. The minstay variable was used to allow for varying truncation points.


In this analysis, two variables have a statistically significant relationship with length of stay. On average, patients who died in the hospital had longer lengths of stay \((p=0.01)\). Because the coefficient for HMO is negative, that is, \(b_{\mathrm{HMO}}=-0.073\), on average, patients who were insured by an HMO had shorter lengths of stay ( \(p=0.047\) ). The type of admission was not statistically significant ( \(p>0.05\) ).

\section*{Stored results}
tnbreg stores the following in e():
Scalars
```

e(N)
e(k)
e(k_aux)
e(k_eq)
e(k_eq_model)
e(k_dv)
e(df_m)
e(r2_p)
e(ll)
e(ll_0)
e(ll_c)
e(alpha)
e(delta)
e(N_clust)
e(chi2)
e(chi2_c)
e(p)
e(rank)
e(rank0)
e(ic)
e(rc)
e(converged)

```

Macros
    e(cmd)
    e(cmdline)
    e(depvar)
    e(llopt)
    e(wtype)
    e(wexp)
    e(title)
    e(clustvar)
    e(offset)
    e(chi2type)
    e(chi2_ct)
    e(dispers)
    e(vce)
    e(vcetype)
    e(opt)
    e(which)
    e(ml_method)
    e(user)
    \(e(t e c h n i q u e)\)
    e(properties)
    e(predict)
    e(asbalanced)
    e(asobserved)
Matrices
e(b)
e(Cns)
e(ilog)
e(gradient)
e(V)
e(V_modelbased)

Functions
    e(sample)
number of observations
number of parameters
number of auxiliary parameters
number of equations in \(e(b)\)
number of equations in overall model test
number of dependent variables
model degrees of freedom
pseudo- \(R^{2}\)
log likelihood
log likelihood, constant-only model
log likelihood, comparison model
value of alpha
value of delta
number of clusters
\(\chi^{2}\)
\(\chi^{2}\) for comparison test
\(p\)-value for model test
rank of e(V)
rank of e(V) for constant-only model
number of iterations
return code
1 if converged, 0 otherwise
tnbreg
command as typed
name of dependent variable
contents of 11 (), or 0 if not specified
weight type
weight expression
title in estimation output
name of cluster variable
linear offset variable
Wald or LR; type of model \(\chi^{2}\) test
Wald or LR; type of model \(\chi^{2}\) test corresponding to e(chi2_c)
mean or constant
vcetype specified in vce()
title used to label Std. err.
type of optimization
max or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement predict
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
constraints matrix
iteration \(\log\) (up to 20 iterations)
gradient vector
variance-covariance matrix of the estimators
model-based variance
marks estimation sample

In addition to the above, the following is stored in r() :
Matrices
\(r(t a b l e) \quad\) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

\section*{Methods and formulas}

Methods and formulas are presented under the following headings:

\section*{Mean-dispersion model}

A negative binomial distribution can be regarded as a gamma mixture of Poisson random variables. The number of times an event occurs, \(y_{j}\), is distributed as Poisson \(\left(\nu_{j} \mu_{j}\right)\). That is, its conditional likelihood is
\[
f\left(y_{j} \mid \nu_{j}\right)=\frac{\left(\nu_{j} \mu_{j}\right)^{y_{j}} e^{-\nu_{j} \mu_{j}}}{\Gamma\left(y_{j}+1\right)}
\]
where \(\mu_{j}=\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}+\operatorname{offset}_{j}\right)\) and \(\nu_{j}\) is an unobserved parameter with a \(\operatorname{Gamma}(1 / \alpha, \alpha)\) density:
\[
g(\nu)=\frac{\nu^{(1-\alpha) / \alpha} e^{-\nu / \alpha}}{\alpha^{1 / \alpha} \Gamma(1 / \alpha)}
\]

This gamma distribution has a mean of 1 and a variance of \(\alpha\), where \(\alpha\) is our ancillary parameter.
The unconditional likelihood for the \(j\) th observation is therefore
\[
f\left(y_{j}\right)=\int_{0}^{\infty} f\left(y_{j} \mid \nu\right) g(\nu) d \nu=\frac{\Gamma\left(m+y_{j}\right)}{\Gamma\left(y_{j}+1\right) \Gamma(m)} p_{j}^{m}\left(1-p_{j}\right)^{y_{j}}
\]
where \(p_{j}=1 /\left(1+\alpha \mu_{j}\right)\) and \(m=1 / \alpha\). Solutions for \(\alpha\) are handled by searching for \(\ln \alpha\) because \(\alpha\) must be greater than zero. The conditional probability of observing \(y_{j}\) events given that \(y_{j}\) is greater than the truncation point \(\tau_{j}\) is
\[
\operatorname{Pr}\left(Y=y_{j} \mid y_{j}>\tau_{j}, \mathbf{x}_{j}\right)=\frac{f\left(y_{j}\right)}{\operatorname{Pr}\left(Y>\tau_{j} \mid \mathbf{x}_{j}\right)}
\]

The log likelihood (with weights \(w_{j}\) and offsets) is given by
\[
\begin{aligned}
& m=1 / \alpha \quad p_{j}=1 /\left(1+\alpha \mu_{j}\right) \quad \mu_{j}=\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}+\text { offset }_{j}\right) \\
& \ln L=\sum_{j=1}^{n} w_{j}\left[\ln \left\{\Gamma\left(m+y_{j}\right)\right\}-\ln \left\{\Gamma\left(y_{j}+1\right)\right\}\right. \\
& \left.\quad-\ln \{\Gamma(m)\}+m \ln \left(p_{j}\right)+y_{j} \ln \left(1-p_{j}\right)-\ln \left\{\operatorname{Pr}\left(Y>\tau_{j} \mid p_{j}, m\right)\right\}\right]
\end{aligned}
\]

\section*{Constant-dispersion model}

The constant-dispersion model assumes that \(y_{j}\) is conditionally distributed as Poisson \(\left(\mu_{j}^{*}\right)\), where \(\mu_{j}^{*} \sim \operatorname{Gamma}\left(\mu_{j} / \delta, \delta\right)\) for some dispersion parameter \(\delta\) [by contrast, the mean-dispersion model assumes that \(\left.\mu_{j}^{*} \sim \operatorname{Gamma}\left(1 / \alpha, \alpha \mu_{j}\right)\right]\). The log likelihood is given by
\[
\begin{gathered}
m_{j}=\mu_{j} / \delta \quad p=1 /(1+\delta) \\
\ln L=\sum_{j=1}^{n} w_{j}\left[\ln \left\{\Gamma\left(m_{j}+y_{j}\right)\right\}-\ln \left\{\Gamma\left(y_{j}+1\right)\right\}\right. \\
\left.\quad-\ln \left\{\Gamma\left(m_{j}\right)\right\}+m_{j} \ln (p)+y_{j} \ln (1-p)-\ln \left\{\operatorname{Pr}\left(Y>\tau_{j} \mid p, m_{j}\right)\right\}\right]
\end{gathered}
\]
with everything else defined as shown above in the calculations for the mean-dispersion model.
This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce (robust) and vce(cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.
tnbreg also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

\section*{Acknowledgment}

We gratefully acknowledge the previous work by Joseph Hilbe (1944-2017) (1999), coauthor of the Stata Press book Generalized Linear Models and Extensions.

\section*{References}

Cameron, A. C., and P. K. Trivedi. 2005. Microeconometrics: Methods and Applications. New York: Cambridge University Press.
——. 2013. Regression Analysis of Count Data. 2nd ed. New York: Cambridge University Press.
——. 2022. Microeconometrics Using Stata. 2nd ed. College Station, TX: Stata Press.
Grogger, J. T., and R. T. Carson. 1991. Models for truncated counts. Journal of Applied Econometrics 6: 225-238. https://doi.org/10.1002/jae. 3950060302.
Hardin, J. W., and J. M. Hilbe. 2015. Regression models for count data from truncated distributions. Stata Journal 15: 226-246.

Hilbe, J. M. 1999. sg102: Zero-truncated Poisson and negative binomial regression. Stata Technical Bulletin 47: 37-40. Reprinted in Stata Technical Bulletin Reprints, vol. 8, pp. 233-236. College Station, TX: Stata Press.
Long, J. S. 1997. Regression Models for Categorical and Limited Dependent Variables. Thousand Oaks, CA: Sage.
Long, J. S., and J. Freese. 2014. Regression Models for Categorical Dependent Variables Using Stata. 3rd ed. College Station, TX: Stata Press.
Simonoff, J. S. 2003. Analyzing Categorical Data. New York: Springer.

\section*{Also see}
[R] tnbreg postestimation - Postestimation tools for tnbreg
[R] nbreg - Negative binomial regression
[R] poisson - Poisson regression
[R] tpoisson - Truncated Poisson regression
\([\mathrm{R}]\) zinb - Zero-inflated negative binomial regression
[R] zip - Zero-inflated Poisson regression
[BAYES] bayes: tnbreg - Bayesian truncated negative binomial regression
[SVY] svy estimation - Estimation commands for survey data
[XT] xtnbreg - Fixed-effects, random-effects, \& population-averaged negative binomial models
[U] 20 Estimation and postestimation commands

\section*{Title}
tnbreg postestimation - Postestimation tools for tnbreg
Postestimation commands predict margins Methods and formulas
Also see

\section*{Postestimation commands}

The following postestimation commands are available after tnbreg:
\begin{tabular}{|c|c|}
\hline Command & Description \\
\hline contrast & contrasts and ANOVA-style joint tests of estimates \\
\hline estat ic & Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) \\
\hline estat summarize & summary statistics for the estimation sample \\
\hline estat vce & variance-covariance matrix of the estimators (VCE) \\
\hline estat (svy) & postestimation statistics for survey data \\
\hline estimates & cataloging estimation results \\
\hline etable & table of estimation results \\
\hline * forecast & dynamic forecasts and simulations \\
\hline * hausman & Hausman's specification test \\
\hline lincom & point estimates, standard errors, testing, and inference for linear combinations of coefficients \\
\hline * 1 rtest & likelihood-ratio test \\
\hline margins & marginal means, predictive margins, marginal effects, and average marginal effects \\
\hline marginsplot & graph the results from margins (profile plots, interaction plots, etc.) \\
\hline nlcom & point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients \\
\hline predict & number of events, incidence rates, probabilities, etc. \\
\hline predictnl & point estimates, standard errors, testing, and inference for generalized predictions \\
\hline pwcompare & pairwise comparisons of estimates \\
\hline suest & seemingly unrelated estimation \\
\hline test & Wald tests of simple and composite linear hypotheses \\
\hline testnl & Wald tests of nonlinear hypotheses \\
\hline
\end{tabular}

\footnotetext{
*forecast, hausman, and lrtest are not appropriate with svy estimation results.
}

\section*{predict}

\section*{Description for predict}
predict creates a new variable containing predictions such as numbers of events, incidence rates, conditional means, probabilities, conditional probabilities, linear predictions, and standard errors.

\section*{Menu for predict}

Statistics \(>\) Postestimation

\section*{Syntax for predict}
```

predict [type] newvar [if] [in] [, statistic nooffset]
predict [type] stub* [if] [in], scores

```
statistic Description
\begin{tabular}{ll} 
Main & \\
n & number of events; the default \\
ir & incidence rate \\
cm & conditional mean, \(E\left(y_{j} \mid y_{j}>\tau_{j}\right)\) \\
\(\operatorname{pr}(n)\) & probability \(\operatorname{Pr}\left(y_{j}=n\right)\) \\
\(\operatorname{pr}(a, b)\) & probability \(\operatorname{Pr}\left(a \leq y_{j} \leq b\right)\) \\
\(\mathrm{cpr}(n)\) & conditional probability \(\operatorname{Pr}\left(y_{j}=n \mid y_{j}>\tau_{j}\right)\) \\
\(\mathrm{cpr}(a, b)\) & conditional probability \(\operatorname{Pr}\left(a \leq y_{j} \leq b \mid y_{j}>\tau_{j}\right)\) \\
xb & linear prediction \\
stdp & standard error of the linear prediction
\end{tabular}

These statistics are available both in and out of sample; type predict . . . if e(sample) ... if wanted only for the estimation sample.

\section*{Options for predict}
\(\qquad\) Main
n , the default, calculates the predicted number of events, which is \(\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}\right)\) if neither offset () nor exposure() was specified when the model was fit; \(\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}+\right.\) offset \(\left._{j}\right)\) if offset() was specified; or \(\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}\right) \times\) exposure \(_{j}\) if exposure() was specified.
ir calculates the incidence rate \(\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}\right)\), which is the predicted number of events when exposure is 1 . This is equivalent to specifying both the n and the nooffset options.
cm calculates the conditional mean,
\[
E\left(y_{j} \mid y_{j}>\tau_{j}\right)=\frac{E\left(y_{j}, y_{j}>\tau_{j}\right)}{\operatorname{Pr}\left(y_{j}>\tau_{j}\right)}
\]
where \(\tau_{j}\) is the truncation point found in e (llopt).
\(\operatorname{pr}(n)\) calculates the probability \(\operatorname{Pr}\left(y_{j}=n\right)\), where \(n\) is a nonnegative integer that may be specified as a number or a variable.
\(\operatorname{pr}(a, b)\) calculates the probability \(\operatorname{Pr}\left(a \leq y_{j} \leq b\right)\), where \(a\) and \(b\) are nonnegative integers that may be specified as numbers or variables;
\(b\) missing ( \(b \geq\).) means \(+\infty\);
\(\operatorname{pr}(20,\).\() calculates \operatorname{Pr}\left(y_{j} \geq 20\right)\);
\(\operatorname{pr}(20, b)\) calculates \(\operatorname{Pr}\left(y_{j} \geq 20\right)\) in observations for which \(b \geq\). and calculates
\(\operatorname{Pr}\left(20 \leq y_{j} \leq b\right)\) elsewhere.
\(\operatorname{pr}(., b)\) produces a syntax error. A missing value in an observation of the variable \(a\) causes a missing value in that observation for \(\operatorname{pr}(a, b)\).
\(\operatorname{cpr}(n)\) calculates the conditional probability \(\operatorname{Pr}\left(y_{j}=n \mid y_{j}>\tau_{j}\right)\), where \(\tau_{j}\) is the truncation point found in e(llopt). \(n\) is an integer greater than the truncation point that may be specified as a number or a variable.
\(\operatorname{cpr}(a, b)\) calculates the conditional probability \(\operatorname{Pr}\left(a \leq y_{j} \leq b \mid y_{j}>\tau_{j}\right)\), where \(\tau_{j}\) is the truncation point found in e(llopt). The syntax for this option is analogous to that used for \(\mathrm{pr}(a, b)\) except that \(a\) must be greater than the truncation point.
xb calculates the linear prediction, which is \(\mathbf{x}_{j} \boldsymbol{\beta}\) if neither offset () nor exposure() was specified when the model was fit; \(\mathbf{x}_{j} \boldsymbol{\beta}+\) offset \(_{j}\) if offset () was specified; or \(\mathbf{x}_{j} \boldsymbol{\beta}+\ln \left(\right.\) exposure \(\left._{j}\right)\) if exposure() was specified; see nooffset below.
stdp calculates the standard error of the linear prediction.
nooffset is relevant only if you specified offset() or exposure() when you fit the model. It modifies the calculations made by predict so that they ignore the offset or exposure variable; the linear prediction is treated as \(\mathbf{x}_{j} \boldsymbol{\beta}\) rather than as \(\mathbf{x}_{j} \boldsymbol{\beta}+\) offset \(_{j}\) or \(\mathbf{x}_{j} \boldsymbol{\beta}+\ln \left(\right.\) exposure \(\left._{j}\right)\). Specifying predict..., nooffset is equivalent to specifying predict..., ir.
scores calculates equation-level score variables.
The first new variable will contain \(\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)\).
The second new variable will contain \(\partial \ln L / \partial(\ln \alpha)\) for dispersion(mean).
The second new variable will contain \(\partial \ln L / \partial(\ln \delta)\) for dispersion(constant).

\section*{margins}

\section*{Description for margins}
margins estimates margins of response for numbers of events, incidence rates, conditional means, probabilities, conditional probabilities, and linear predictions.

\section*{Menu for margins}

Statistics \(>\) Postestimation

\section*{Syntax for margins}
```

margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...)...] [options]

```
statistic
Description
n
ir
cm
\(\operatorname{pr}(n)\)
\(\operatorname{pr}(a, b)\)
\(\operatorname{cpr}(n)\)
\(\operatorname{cpr}(a, b)\)
xb
stdp
incidence rate
linear prediction
number of events; the default
conditional mean, \(E\left(y_{j} \mid y_{j}>\tau_{j}\right)\)
probability \(\operatorname{Pr}\left(y_{j}=n\right)\)
probability \(\operatorname{Pr}\left(a \leq y_{j} \leq b\right)\)
conditional probability \(\operatorname{Pr}\left(y_{j}=n \mid y_{j}>\tau_{j}\right)\)
conditional probability \(\operatorname{Pr}\left(a \leq y_{j} \leq b \mid y_{j}>\tau_{j}\right)\)
not allowed with margins

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

\section*{Methods and formulas}

In the following formulas, we use the same notation as in \([R]\) tnbreg.
Methods and formulas are presented under the following headings:

\section*{Mean-dispersion model}

The equation-level scores are given by
\[
\begin{aligned}
\operatorname{score}(\mathbf{x} \boldsymbol{\beta})_{j}= & p_{j}\left(y_{j}-\mu_{j}\right)-\frac{p_{j}^{(m+1)} \mu_{j}}{\operatorname{Pr}\left(Y>\tau_{j} \mid p_{j}, m\right)} \\
\operatorname{score}(\omega)_{j}= & -m\left\{\frac{\alpha\left(\mu_{j}-y_{j}\right)}{1+\alpha \mu_{j}}-\ln \left(1+\alpha \mu_{j}\right)+\psi\left(y_{j}+m\right)-\psi(m)\right\} \\
& -\frac{p_{j}^{m}}{\operatorname{Pr}\left(Y>\tau_{j} \mid p_{j}, m\right)}\left\{m \ln \left(p_{j}\right)+\mu_{j} p_{j}\right\}
\end{aligned}
\]
where \(\omega_{j}=\ln \alpha_{j}, \psi(z)\) is the digamma function, and \(\tau_{j}\) is the truncation point found in e(llopt).

\section*{Constant-dispersion model}

The equation-level scores are given by
\[
\begin{aligned}
\operatorname{score}(\mathbf{x} \boldsymbol{\beta})_{j} & =m_{j}\left\{\psi\left(y_{j}+m_{j}\right)-\psi\left(m_{j}\right)+\ln (p)+\frac{p^{m_{j}} \ln (p)}{\operatorname{Pr}\left(Y>\tau_{j} \mid p, m_{j}\right)}\right\} \\
\operatorname{score}(\omega)_{j} & =y_{j}-\left(y_{j}+m_{j}\right)(1-p)-\operatorname{score}(\mathbf{x} \boldsymbol{\beta})_{j}-\frac{\mu_{j} p}{\operatorname{Pr}\left(Y>\tau_{j} \mid p, m_{j}\right)}
\end{aligned}
\]
where \(\omega_{j}=\ln \delta_{j}\) and \(\tau_{j}\) is the truncation point found in e(llopt).

\section*{Also see}
[R] tnbreg - Truncated negative binomial regression
[U] 20 Estimation and postestimation commands

\section*{Title}
tobit - Tobit regression
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
tobit fits models for continuous responses where the outcome variable is censored. Censoring limits may be fixed for all observations or vary across observations.

\section*{Quick start}

Tobit regression of y on x 1 and x 2 , specifying that y is censored at the minimum of y tobit \(y\) x1 x2, ll

Same as above, but where the lower-censoring limit is zero
tobit y x1 x 2 , \(11(0)\)
Same as above, but specify the lower- and upper-censoring limits
```

tobit y x1 x2, ll(17) ul(34)

```

Same as above, but where lower and upper are variables containing the censoring limits
```

tobit y x1 x2, ll(lower) ul(upper)

```

\section*{Menu}

Statistics \(>\) Linear models and related \(>\) Censored regression \(>\) Tobit regression

\section*{Syntax}
tobit depvar [indepvars] [if] [in] [weight] [, options]
options
Model
noconstant
\(11[\) (varname \(\mid \#\) ) \(]\)
ul[(varname|\#)]
offset (varname)
constraints (constraints)
SE/Robust
vce (vcetype)

Reporting
level(\#)
nocnsreport
display_options

Maximization
maximize_options
collinear
coeflegend

Description
suppress constant term
left-censoring variable or limit
right-censoring variable or limit
include varname in model with coefficient constrained to 1
apply specified linear constraints
vcetype may be oim, opg, robust, cluster clustvar, bootstrap, or jackknife
set confidence level; default is level (95)
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
control the maximization process; seldom used
keep collinear variables
display legend instead of statistics
indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
depvar and indepvars may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bayes, bootstrap, by, collect, fmm, fp, jackknife, nestreg, rolling, statsby, stepwise, and svy are allowed; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: tobit and [FMM] fmm: tobit.
Weights are not allowed with the bootstrap prefix; see [R] bootstrap.
aweights are not allowed with the jackknife prefix; see \([R]\) jackknife.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
aweights, fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options}
\(\qquad\) Model

\section*{noconstant; see [R] Estimation options.}
\(11[(\) varname \(\mid \#)]\) and \(\mathrm{ul}[(\) varname \(\mid \#)]\) indicate the lower and upper limits for censoring, respectively. Observations with depvar \(\leq 11\) () are left-censored; observations with depvar \(\geq \mathrm{ul}\) () are right-censored; and remaining observations are not censored. You do not have to specify the censoring values. If you specify 11 , the lower limit is the minimum of depvar. If you specify ul, the upper limit is the maximum of depvar.
offset (varname), constraints (constraints); see [R] Estimation options.

\section*{SE/Robust}
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vee_option.

\section*{Reporting}
level(\#), nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique(algorithm_spec), iterate(\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance (\#), nonrtolerance, and from(init-specs); see [R] Maximize. These options are seldom used.

The following options are available with tobit but are not shown in the dialog box: collinear, coeflegend; see [R] Estimation options.

\section*{Remarks and examples}
tobit fits a linear regression model for a censored continuous outcome. Censoring occurs when the dependent variable is observed only within a certain range of values. When it is not, we know only that it is either above (right-censoring) or below (left-censoring) the censoring value. Censoring differs from truncation. When the data are truncated, we do not observe either the dependent variable or the covariates; see [R] truncreg.

Censoring may result from study design or may be a result of how the outcome is measured. Right-censoring of data may occur, for example, in income surveys that top code the highest income category. Any respondent that earns the censoring limit or more reports only the value at the limit, and we do not know the respondent's true income. Left-censoring arises naturally when measurements are obtained from an instrument or a laboratory procedure that has a limit of detection. If we observe a value at the measurement limit, we know the true value is at the limit or below it. tobit allows the censoring limits to be the same for all observations or to vary from observation to observation.

Tobin (1958) originally conceived the tobit model as one of consumption of consumer durables where purchases were left-censored at zero. Contemporary literature treats this and similar cases as a corner solution model. See Wooldridge (2020, sec. 17.2), Long (1997, 196-210), and Maddala and Lahiri (2006, 333-336) for an introduction to the tobit model. Wooldridge (2010, chap. 17 and 19) provides an advanced treatment of censored regression models. Cameron and Trivedi (2022, chap. 19) discuss the tobit model using Stata examples.

The tobit model can be written as the latent regression model \(\mathbf{y}=\mathbf{x} \boldsymbol{\beta}+\epsilon\) with a continuous outcome that is either observed or unobserved. Following Cong (2000), the observed outcome for observation \(i\) is defined as
\[
y_{i}^{*}= \begin{cases}y_{i} & \text { if } a<y_{i}<b \\ a & \text { if } y_{i} \leq a \\ b & \text { if } y_{i} \geq b\end{cases}
\]
where \(a\) is the lower-censoring limit and \(b\) is the upper-censoring limit. The tobit model assumes that the error term is normally distributed; \(\epsilon \sim N\left(\mathbf{0}, \sigma^{2} \mathbf{I}\right)\). Depending on the problem at hand, the quantity of interest in a tobit model may be the censored outcome, \(y_{i}^{*}\), or the uncensored outcome, \(y_{i}\). In the measurement instrument scenario above, we may wish to predict the values that fall below the measurement threshold. By contrast, in the consumption of consumer durables scenario above, the latent variable is an artificial construct and the variable of interest is the observed consumer expenditure.

\section*{> Example 1: Constant-censoring limit}

University administrators want to know the relationship between high school grade point average (GPA) and students' performance in college. gpa.dta contains fictional data on a cohort of 4,000 college students. College GPA (gpa2) and high school GPA (hsgpa) are measured on a continuous scale between zero and four. The outcome of interest is the student's college GPA. But, for reasons of confidentiality, GPAs below 2.0 are reported as 2.0 . In other words, the outcome is censored on the left.

We believe that GPA is also a function of the logarithm of income of the student's parents (pincome) and whether or not the student participated in a study-skills program while in college (program).
```

. use https://www.stata-press.com/data/r18/gpa
(High school GPA and performance in college)
. tobit gpa2 hsgpa pincome program, ll
Refining starting values:
Grid node 0: Log likelihood = -2551.3989
Fitting full model:
Iteration 0: Log likelihood = -2551.3989
Iteration 1: Log likelihood = -2065.4023
Iteration 2: Log likelihood = -2015.8135
Iteration 3: Log likelihood = -2015.1281
Iteration 4: Log likelihood = -2015.1258
Iteration 5: Log likelihood = -2015.1258
Tobit regression
Limits: Lower = 2
Upper = +inf

```
Log likelihood = -2015. 1258
\begin{tabular}{rrr} 
Number of obs & \(=4,000\) \\
Uncensored & \(=\) & 2,794 \\
Left-censored & \(=\) & 1,206 \\
Right-censored & \(=\) & 0 \\
LR chi2 \((3)\) & \(=4712.61\) \\
Prob \(>\) chi2 & \(=0.0000\) \\
Pseudo R2 & \(=0.5390\)
\end{tabular}
\begin{tabular}{r|rrrrrr}
\hline gpa2 & Coefficient & Std. err. & t & \(\mathrm{P}>\mid \mathrm{tl}\) & [95\% conf. interval] \\
\hline hsgpa & .6586311 & .0128699 & 51.18 & 0.000 & .633399 & .6838632 \\
pincome & .3159297 & .0074568 & 42.37 & 0.000 & .3013103 & .3305491 \\
program & .5554416 & .0147468 & 37.67 & 0.000 & .5265297 & .5843535 \\
_cons & -.8902578 & .0478484 & -18.61 & 0.000 & -.9840673 & -.7964482 \\
\hline \(\operatorname{var}(\mathrm{e} . \mathrm{gpa2})\) & .161703 & .0044004 & & & .1533019 & .1705645 \\
\hline
\end{tabular}
tobit reports the coefficients for the latent regression model. Thus, we can interpret the coefficients just as we would the coefficients from OLS. For example, participation in a study-skills program increases the expected uncensored GPA by 0.56 points.

\section*{> Example 2: Tobit model for a corner solution}

Suppose that we are interested in the number of hours married women spend working for wages, and we treat observations recording zero hours as observed, per the corner-solution approach discussed Wooldridge (2010, chap. 16). We use the labor supply data extracted by Mroz (1987) from the 1975 PSID for 753 married women. The variable whrs 75 records the annual number of hours worked. Forty-three percent of the surveyed women worked zero hours, and the remaining women worked on average 1,303 hours a year.

We regress hours worked on household income excluding wife's income (nwinc), years of schooling (wedyrs), years of labor market experience (wexper) and its square, age (wifeage), an indicator for the presence of children under 6 years of age at home (kl6), and an indicator for the presence of children from 6 to 18 years old at home ( k 618 ).
```

. use https://www.stata-press.com/data/r18/mroz87
(1975 PSID data from Mroz, 1987)
. tobit whrs75 nwinc wedyrs wexper c.wexper\#c.wexper wifeage kl6 k618, ll(0)
Refining starting values:
Grid node 0: Log likelihood = -3961.1577
Fitting full model:
Iteration 0: Log likelihood = -3961.1577
Iteration 1: Log likelihood = -3836.8928
Iteration 2: Log likelihood = -3819.2637
Iteration 3: Log likelihood = -3819.0948
Iteration 4: Log likelihood = -3819.0946
Tobit regression Number of obs = 753
Limits: Lower = 0
Upper = +inf
Log likelihood = -3819.0946

| Number of obs | $=$ | 753 |
| ---: | ---: | ---: |
| Uncensored | $=$ | 428 |
| Left-censored | $=$ | 325 |
| Right-censored | $=$ | 0 |
| LR chi2 $(7)$ | $=271.59$ |  |
| Prob $>$ chi2 | $=0.0000$ |  |
| Pseudo R2 | $=$ | 0.0343 |

```
\begin{tabular}{r|rrrrrr}
\hline whrs75 & Coefficient & Std. err. & t & \(\mathrm{P}>\mid \mathrm{tl}\) & [95\% conf. interval] \\
\hline nwinc & -8.814227 & 4.459089 & -1.98 & 0.048 & -17.56808 & -.0603708 \\
wedyrs & 80.64541 & 21.58318 & 3.74 & 0.000 & 38.27441 & 123.0164 \\
wexper & 131.564 & 17.27935 & 7.61 & 0.000 & 97.64211 & 165.486 \\
c.wexper\# & & & & & & \\
c.wexper & -1.864153 & .5376606 & -3.47 & 0.001 & -2.919661 & -.8086455 \\
wifeage & -54.40491 & 7.418483 & -7.33 & 0.000 & -68.9685 & -39.84133 \\
kl6 & -894.0202 & 111.8777 & -7.99 & 0.000 & -1113.653 & -674.3875 \\
k618 & -16.21805 & 38.6413 & -0.42 & 0.675 & -92.07668 & 59.64057 \\
_cons & 965.3068 & 446.4351 & 2.16 & 0.031 & 88.88827 & 1841.725 \\
\hline var(e.whrs75) & 1258927 & 93304.48 & & & 1088458 & 1456093 \\
\hline
\end{tabular}

Unlike in example 1, we are interested in the marginal effect of the covariates on the observed outcome. We can use margins to estimate, for example, the average marginal effect of years of education on the expected value of the actual hours worked.
```

. margins, dydx(wedyrs) predict(ystar(0,.))

```

Average marginal effects Number of obs \(=753\)
Model VCE: OIM
Expression: E(whrs75*|whrs75>0), predict(ystar(0,.))
dy/dx wrt: wedyrs
\begin{tabular}{l|rccccc}
\hline & \multicolumn{4}{|c}{ Delta-method } \\
& dy/dx & std. err. & z & P>|z| & [95\% conf. interval] \\
\hline wedyrs & 47.47306 & 12.6214 & 3.76 & 0.000 & 22.73558 & 72.21054 \\
\hline
\end{tabular}

The average marginal effect of years of education on the actual hours worked is 47.47. See [R] tobit postestimation for more examples using margins.

James Tobin (1918-2002) was an American economist who after education and research at Harvard moved to Yale, where he was on the faculty from 1950 to 1988 . He made many outstanding contributions to economics and was awarded the Nobel Prize in 1981 "for his analysis of financial markets and their relations to expenditure decisions, employment, production and prices". He trained in the US Navy with the writer, Herman Wouk, who later fashioned a character after Tobin in the novel The Caine Mutiny (1951): "A mandarin-like midshipman named Tobit, with a domed forehead, measured quiet speech, and a mind like a sponge, was ahead of the field by a spacious percentage."

\section*{Stored results}
tobit stores the following in e() :
Scalars
\begin{tabular}{|c|c|}
\hline e(N) & number of observations \\
\hline e(N_unc) & number of uncensored observations \\
\hline e(N_lc) & number of left-censored observations \\
\hline e(N_rc) & number of right-censored observations \\
\hline e(k) & number of parameters \\
\hline \(e\left(k \_e q\right)\) & number of equations in e(b) \\
\hline e(k_aux) & number of auxiliary parameters \\
\hline \(e\left(k \_d v\right)\) & number of dependent variables \\
\hline \(e\left(d f \_m\right)\) & model degrees of freedom \\
\hline e(df_r) & residual degrees of freedom \\
\hline e(r2_p) & pseudo- \(R^{2}\) \\
\hline e(11) & log likelihood \\
\hline e(ll_0) & log likelihood, constant-only model \\
\hline e(N_clust) & number of clusters \\
\hline e(chi2) & \(\chi^{2}\) \\
\hline e(F) & \(F\) statistic \\
\hline \(e(p)\) & \(p\)-value for model test \\
\hline \(e(r a n k)\) & rank of e(V) \\
\hline \(e(i c)\) & number of iterations \\
\hline \(e(r c)\) & return code \\
\hline e(converged) & 1 if converged, 0 otherwise \\
\hline
\end{tabular}

Macros
```

    e(cmd)
    e(cmdline)
    e(depvar)
    e(llopt)
    e(ulopt)
    e(wtype)
    ```
    e(wexp)
    e(covariates)
    e(title)
    e(clustvar)
    e(offset)
    e(chi2type)
    e(vce)
    e(vcetype)
    e(opt)
    e(which)
    e(method)
    e(ml_method)
    e(user)
    e(technique)
    e(properties)
    e(predict)
    e(marginsok)
    e(asbalanced)
    e(asobserved)
tobit
    command as typed
    name of dependent variable
    minimum of depvar or contents of 11()
    maximum of depvar or contents of ul()
    weight type
    weight expression
    list of covariates
    title in estimation output
    name of cluster variable
    linear offset variable
    type of model \(\chi^{2}\) test
    vcetype specified in vce()
    title used to label Std. err.
    type of optimization
    max or min; whether optimizer is to perform maximization or minimization
    estimation method: ml
    type of ml method
    name of likelihood-evaluator program
    maximization technique
    b V
    program used to implement predict
    predictions allowed by margins
    factor variables fvset as asbalanced
    factor variables fvset as asobserved
Matrices
    e(b)
    coefficient vector
    e(Cns)
    constraints matrix
    e(ilog)
    e(gradient)
    iteration \(\log\) (up to 20 iterations)
    gradient vector
    e(V)
    variance-covariance matrix of the estimators
    e(V_modelbased)
    model-based variance
Functions
    e(sample) marks estimation sample

In addition to the above, the following is stored in r() :

\section*{Matrices}
\(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

\section*{Methods and formulas}

See Methods and formulas in [ R\(]\) intreg.
This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce (robust) and vce(cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.
tobit also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

\section*{References}

Amemiya, T. 1973. Regression analysis when the dependent variable is truncated normal. Econometrica 41: 997-1016. https://doi.org/10.2307/1914031.
_—. 1984. Tobit models: A survey. Journal of Econometrics 24: 3-61. https://doi.org/10.1016/0304-4076(84)90074-5.
Belotti, F., P. Deb, W. G. Manning, and E. C. Norton. 2015. twopm: Two-part models. Stata Journal 15: 3-20.
Bertanha, M., A. H. McCallum, A. Payne, and N. Seegert. 2022. Bunching estimation of elasticities using Stata. Stata Journal 22: 597-624.

Burke, W. J. 2009. Fitting and interpreting Cragg's tobit alternative using Stata. Stata Journal 9: 584-592.
Cameron, A. C., and P. K. Trivedi. 2022. Microeconometrics Using Stata. 2nd ed. College Station, TX: Stata Press.
Canette, I. 2016. Understanding truncation and censoring. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2016/12/13/understanding-truncation-and-censoring/.

Chernozhukov, V., I. Fernández-Val, S. Han, and A. Kowalski. 2019. Censored quantile instrumental-variable estimation with Stata. Stata Journal 19: 768-781.

Cong, R. 2000. sg144: Marginal effects of the tobit model. Stata Technical Bulletin 56: 27-34. Reprinted in Stata Technical Bulletin Reprints, vol. 10, pp. 189-197. College Station, TX: Stata Press.
Deb, P., E. C. Norton, and W. G. Manning. 2017. Health Econometrics Using Stata. College Station, TX: Stata Press.
Drukker, D. M. 2002. Bootstrapping a conditional moments test for normality after tobit estimation. Stata Journal 2 : 125-139.

Goldberger, A. S. 1983. Abnormal selection bias. In Studies in Econometrics, Time Series, and Multivariate Statistics, ed. S. Karlin, T. Amemiya, and L. A. Goodman, 67-84. New York: Academic Press.
Hurd, M. 1979. Estimation in truncated samples when there is heteroscedasticity. Journal of Econometrics 11: 247-258. https://doi.org/10.1016/0304-4076(79)90039-3.

Long, J. S. 1997. Regression Models for Categorical and Limited Dependent Variables. Thousand Oaks, CA: Sage.
Maddala, G. S., and K. Lahiri. 2006. Introduction to Econometrics. 4th ed. New York: Wiley.
McDonald, J. F., and R. A. Moffitt. 1980. The use of tobit analysis. Review of Economics and Statistics 62: 318-321. https://doi.org/10.2307/1924766.
Mroz, T. A. 1987. The sensitivity of an empirical model of married women's hours of work to economic and statistical assumptions. Econometrica 55: 765-799. https://doi.org/10.2307/1911029.

Sánchez-Peñalver, A. 2019. Estimation methods in the presence of corner solutions. Stata Journal 19: 87-111.
Shiller, R. J. 1999. The ET interview: Professor James Tobin. Econometric Theory 15: 867-900. https://doi.org/10.1017/S0266466699156056.
Stewart, M. B. 1983. On least squares estimation when the dependent variable is grouped. Review of Economic Studies 50: 737-753. https://doi.org/10.2307/2297773.

Tobin, J. 1958. Estimation of relationships for limited dependent variables. Econometrica 26: 24-36. https://doi.org/10.2307/1907382.
Wooldridge, J. M. 2010. Econometric Analysis of Cross Section and Panel Data. 2nd ed. Cambridge, MA: MIT Press. -_. 2020. Introductory Econometrics: A Modern Approach. 7th ed. Boston: Cengage.

\section*{Also see}
[R] tobit postestimation - Postestimation tools for tobit
[R] heckman - Heckman selection model
[R] intreg - Interval regression
[R] ivtobit - Tobit model with continuous endogenous covariates
\([R]\) regress - Linear regression
[R] truncreg - Truncated regression
[BAYES] bayes: tobit - Bayesian tobit regression
[FMM] fmm: tobit - Finite mixtures of tobit regression models
[ERM] eintreg - Extended interval regression
[ME] metobit — Multilevel mixed-effects tobit regression
[SVY] svy estimation - Estimation commands for survey data
[XT] xtintreg - Random-effects interval-data regression models
[XT] xttobit - Random-effects tobit models
[U] 20 Estimation and postestimation commands

\section*{Title}
tobit postestimation - Postestimation tools for tobit
\begin{tabular}{lll} 
Postestimation commands & predict margins Remarks and examples \\
Reference & Also see &
\end{tabular}

\section*{Postestimation commands}

The following postestimation commands are available after tobit:
\begin{tabular}{|c|c|}
\hline Command & Description \\
\hline contrast & contrasts and ANOVA-style joint tests of estimates \\
\hline estat ic & Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) \\
\hline estat summarize & summary statistics for the estimation sample \\
\hline estat vce & variance-covariance matrix of the estimators (VCE) \\
\hline estat (svy) & postestimation statistics for survey data \\
\hline estimates & cataloging estimation results \\
\hline etable & table of estimation results \\
\hline * forecast & dynamic forecasts and simulations \\
\hline * hausman & Hausman's specification test \\
\hline lincom & point estimates, standard errors, testing, and inference for linear combinations of coefficients \\
\hline linktest & link test for model specification \\
\hline * Irtest & likelihood-ratio test \\
\hline margins & marginal means, predictive margins, marginal effects, and average marginal effects \\
\hline marginsplot & graph the results from margins (profile plots, interaction plots, etc.) \\
\hline nlcom & point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients \\
\hline predict & linear, censored, and truncated predictions \\
\hline predictnl & point estimates, standard errors, testing, and inference for generalized predictions \\
\hline pwcompare & pairwise comparisons of estimates \\
\hline suest & seemingly unrelated estimation \\
\hline test & Wald tests of simple and composite linear hypotheses \\
\hline testnl & Wald tests of nonlinear hypotheses \\
\hline
\end{tabular}

\footnotetext{
*forecast, hausman, and lrtest are not appropriate with svy estimation results.
}

\section*{predict}

\section*{Description for predict}
predict creates a new variable containing predictions such as linear predictions, standard errors, probabilities, and expected values.

\section*{Menu for predict}

Statistics \(>\) Postestimation

\section*{Syntax for predict}
```

predict [type] newvar [if] [in] [, statistic nooffset]
predict [type] stub* [if] [in], scores

```
statistic Description

Main
\(\mathrm{xb} \quad\) linear prediction; the default
stdp standard error of the linear prediction
stdf standard error of the forecast
\(\operatorname{pr}(a, b) \quad \operatorname{Pr}\left(a<y_{j}<b\right)\)
\(\mathrm{e}(a, b) \quad E\left(y_{j} \mid a<y_{j}<b\right)\)
ystar \((a, b) \quad E\left(y_{j}^{*}\right), y_{j}^{*}=\max \left\{a, \min \left(y_{j}, b\right)\right\}\)
These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.
stdf is not allowed with svy estimation results.
where \(a\) and \(b\) may be numbers or variables; \(a\) missing ( \(a \geq\).) means \(-\infty\), and \(b\) missing ( \(b \geq\).) means \(+\infty\); see [U] 12.2.1 Missing values.

\section*{Options for predict}
\(\qquad\) Main
xb , the default, calculates the linear prediction.
stdp calculates the standard error of the prediction, which can be thought of as the standard error of the predicted expected value or mean for the observation's covariate pattern. The standard error of the prediction is also referred to as the standard error of the fitted value.
stdf calculates the standard error of the forecast, which is the standard error of the point prediction for 1 observation. It is commonly referred to as the standard error of the future or forecast value. By construction, the standard errors produced by stdf are always larger than those produced by stdp; see Methods and formulas in [R] regress postestimation.
\(\operatorname{pr}(a, b)\) calculates \(\operatorname{Pr}\left(a<\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j}<b\right)\), the probability that \(y_{j} \mid \mathbf{x}_{j}\) would be observed in the interval \((a, b)\).
\(a\) and \(b\) may be specified as numbers or variable names; \(l b\) and \(u b\) are variable names;
\(\operatorname{pr}(20,30)\) calculates \(\operatorname{Pr}\left(20<\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j}<30\right)\);
\(\operatorname{pr}(l b, u b)\) calculates \(\operatorname{Pr}\left(l b<\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j}<u b\right)\); and
\(\operatorname{pr}(20, u b)\) calculates \(\operatorname{Pr}\left(20<\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j}<u b\right)\).
\(a\) missing \((a \geq\).\() means -\infty ; \operatorname{pr}(., 30)\) calculates \(\operatorname{Pr}\left(-\infty<\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j}<30\right)\);
\(\operatorname{pr}(l b, 30)\) calculates \(\operatorname{Pr}\left(-\infty<\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j}<30\right)\) in observations for which \(l b \geq\).
and calculates \(\operatorname{Pr}\left(l b<\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j}<30\right)\) elsewhere.
\(b\) missing \((b \geq\).\() means +\infty ; \operatorname{pr}(20,\).\() calculates \operatorname{Pr}\left(+\infty>\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j}>20\right)\);
\(\operatorname{pr}(20, u b)\) calculates \(\operatorname{Pr}\left(+\infty>\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j}>20\right)\) in observations for which \(u b \geq\).
and calculates \(\operatorname{Pr}\left(20<\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j}<u b\right)\) elsewhere.
\(\mathrm{e}(a, b)\) calculates \(E\left(\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j} \mid a<\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j}<b\right)\), the expected value of \(y_{j} \mid \mathbf{x}_{j}\) conditional on \(y_{j} \mid \mathbf{x}_{j}\) being in the interval \((a, b)\), meaning that \(y_{j} \mid \mathbf{x}_{j}\) is truncated.
\(a\) and \(b\) are specified as they are for \(\operatorname{pr}()\).
\(\operatorname{ystar}(a, b)\) calculates \(E\left(y_{j}^{*}\right)\), where \(y_{j}^{*}=a\) if \(\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j} \leq a, y_{j}^{*}=b\) if \(\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j} \geq b\), and \(y_{j}^{*}=\mathbf{x}_{j} \boldsymbol{\beta}+\epsilon_{j}\) otherwise, meaning that \(y_{j}^{*}\) is censored. \(a\) and \(b\) are specified as they are for \(\operatorname{pr}()\). nooffset is relevant only if you specified offset (varname). It modifies the calculations made by predict so that they ignore the offset variable; the linear prediction is treated as \(\mathbf{x}_{j} \boldsymbol{\beta}\) rather than as \(\mathbf{x}_{j} \boldsymbol{\beta}+\) offset \(_{j}\).
scores calculates equation-level score variables.
The first new variable will contain \(\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)\).
The second new variable will contain \(\partial \ln L / \partial \sigma\).

\section*{margins}

\section*{Description for margins}
margins estimates margins of response for linear predictions, probabilities, and expected values.

\section*{Menu for margins}

Statistics \(>\) Postestimation

\section*{Syntax for margins}
```

margins [marginlist] [, options]

```


\section*{statistic Description}
\begin{tabular}{ll}
\hline xb & linear prediction; the default \\
\(\mathrm{pr}(a, b)\) & \(\operatorname{Pr}\left(a<y_{j}<b\right)\) \\
\(\mathrm{e}(a, b)\) & \(E\left(y_{j} \mid a<y_{j}<b\right)\) \\
ystar \((a, b)\) & \(E\left(y_{j}^{*}\right), y_{j}^{*}=\max \left\{a, \min \left(y_{j}, b\right)\right\}\) \\
stdp & not allowed with margins \\
stdf & not allowed with margins
\end{tabular}

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

\section*{Remarks and examples}

\section*{> Example 1: Marginal predictions}

In example 2 of \([R]\) tobit, we fit a tobit model of annual number of hours that married women spend working and then obtained estimated average marginal effect of 47.47 for years of education on observed hours worked.
```

. use https://www.stata-press.com/data/r18/mroz87
(1975 PSID data from Mroz, 1987)
. tobit whrs75 nwinc wedyrs wexper c.wexper\#c.wexper wifeage kl6 k618, ll(0)
(output omitted)
. margins, dydx(wedyrs) predict(ystar(0,.))
(output omitted)

```

However, we may not want this overall effect. To obtain marginal effects for specific alternative scenarios, we use margins with the at () option. For example, continuing with example 2 of \([R]\) tobit to estimate the means of the marginal effects on the expected value of the censored outcome conditional on education ranging from 8 years to 17 years, we type
```

. margins, dydx(wedyrs) predict(ystar(0,.)) at(wedyrs=(8(1)17))

```
Average marginal effects Number of obs \(=753\)
Model VCE: OIM
Expression: E(whrs75*|whrs75>0), predict(ystar(0,.))
dy/dx wrt: wedyrs
1._at: wedyrs \(=8\)
2._at: wedyrs \(=9\)
3._at: wedyrs \(=10\)
4._at: wedyrs \(=11\)
5._at: wedyrs \(=12\)
6._at: wedyrs \(=13\)
7._at: wedyrs \(=14\)
8._at: wedyrs \(=15\)
9._at: wedyrs \(=16\)
10._at: wedyrs \(=17\)
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline & dy/dx & lta-method std. err. & z & \(P>|z|\) & [95\% con & interval] \\
\hline \multicolumn{7}{|l|}{wedyrs} \\
\hline _at & & & & & & \\
\hline 1 & 39.58775 & 8.432006 & 4.69 & 0.000 & 23.06132 & 56.11418 \\
\hline 2 & 41.4497 & 9.421414 & 4.40 & 0.000 & 22.98407 & 59.91533 \\
\hline 3 & 43.30531 & 10.41233 & 4.16 & 0.000 & 22.89752 & 63.71309 \\
\hline 4 & 45.14859 & 11.39804 & 3.96 & 0.000 & 22.80885 & 67.48833 \\
\hline 5 & 46.97371 & 12.37208 & 3.80 & 0.000 & 22.72489 & 71.22254 \\
\hline 6 & 48.77504 & 13.32825 & 3.66 & 0.000 & 22.65216 & 74.89793 \\
\hline 7 & 50.54717 & 14.26071 & 3.54 & 0.000 & 22.5967 & 78.49765 \\
\hline 8 & 52.28499 & 15.16403 & 3.45 & 0.001 & 22.56403 & 82.00594 \\
\hline 9 & 53.98369 & 16.03324 & 3.37 & 0.001 & 22.55912 & 85.40827 \\
\hline 10 & 55.63887 & 16.8639 & 3.30 & 0.001 & 22.58624 & 88.6915 \\
\hline
\end{tabular}

The estimated mean of the marginal effects is about 39.59 hours for 8 years of schooling, about 41.45 hours for 9 years of schooling, and so on.

\section*{Reference}

McDonald, J. F., and R. A. Moffitt. 1980. The use of tobit analysis. Review of Economics and Statistics 62: 318-321. https://doi.org/10.2307/1924766.

\section*{Also see}
[R] tobit - Tobit regression
[U] 20 Estimation and postestimation commands

\section*{Title}
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
total produces estimates of totals, along with standard errors.

\section*{Quick start}

Total of continuous variable v1
total v1
Same as above, but restrict estimation to observations where catvar \(=1\)
total v1 if catvar==1
Same as above, but using svyset data
svy, subpop(if catvar==1): total v1
Total of v1 for each level of catvar
total v1, over(catvar)
With jackknife standard errors
```

total v1, vce(jackknife)

```

\section*{Menu}

Statistics \(>\) Summaries, tables, and tests \(>\) Summary and descriptive statistics \(>\) Totals

\section*{Syntax}
total varlist \([\) if \(][\) in \(][\) weight \(][\), options \(]\)
options
Description
iffin/over
over (varlisto \()\)
group over subpopulations defined by varlisto
SE/Cluster
vce (vcetype)
vcetype may be analytic, cluster clustvar, bootstrap, or jackknife

Reporting
level(\#)
noheader
display_options
coeflegend
set confidence level; default is level (95)
suppress table header
control column formats, line width, display of omitted variables and base and empty cells, and factor-variable labeling
display legend instead of statistics
varlist may contain factor variables; see [U] 11.4.3 Factor variables.
bootstrap, collect, jackknife, mi estimate, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands.
vce(bootstrap) and vce(jackknife) are not allowed with the mi estimate prefix.
Weights are not allowed with the bootstrap prefix; see \([R]\) bootstrap.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
coeflegend does not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options}

ك if/in/over
over (varlist \(t_{o}\) ) specifies that estimates be computed for multiple subpopulations, which are identified by the different values of the variables in varlist \(t_{o}\). Only numeric, nonnegative, integer-valued variables are allowed in over \(\left(\right.\) varlist \(\left._{o}\right)\).

SE/Cluster
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (analytic), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
vce(analytic), the default, uses the analytically derived variance estimator associated with the sample total.

Reporting
level(\#); see [R] Estimation options.
noheader prevents the table header from being displayed.
display_options: noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap (\#), fvwrapon (style), cformat (\%fmt), and nolstretch; see [R] Estimation options.

The following option is available with total but is not shown in the dialog box:
coeflegend; see [R] Estimation options.

\section*{Remarks and examples}

\section*{> Example 1}

Suppose that we collected data on incidence of heart attacks. The variable heartatk indicates whether a person ever had a heart attack ( 1 means yes; 0 means no). We can then estimate the total number of persons who have had heart attacks for each sex in the population represented by the data we collected.
\begin{tabular}{|c|c|c|c|c|}
\hline \multicolumn{3}{|l|}{Total estimation} & \multicolumn{2}{|l|}{Number of obs \(=4,946\)} \\
\hline & Total & Std. err. & [95\% con & interval] \\
\hline \multicolumn{5}{|l|}{c.heartatk@sex} \\
\hline Male & 944559 & 104372.3 & 739943 & 1149175 \\
\hline Female & 581590 & 82855.59 & 419156.3 & 744023.7 \\
\hline
\end{tabular}

\section*{Stored results}
total stores the following in e():
Scalars
\(e(N) \quad\) number of observations
e(N_over) number of subpopulations
e(N_clust) number of clusters
\(e\left(k \_e q\right) \quad\) number of equations in \(e(b)\)
\(e\left(d f \_r\right)\) sample degrees of freedom
\(e\) (rank) rank of e(V)
Macros
e(cmd)
\(e(c m d l i n e)\)
e(varlist)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(over)
e(vce)
e(vcetype)
e(properties)
total
command as typed
varlist
weight type
weight expression
title in estimation output
name of cluster variable
varlist from over()
vcetype specified in vce()
e(estat_cmd) program used to implement estat
\(e\) (marginsnotok) predictions disallowed by margins
```

Matrices
e(b) vector of total estimates
e(V) (co)variance estimates
e(_N) vector of numbers of nonmissing observations
e(error) error code corresponding to e(b)
Functions
e(sample) marks estimation sample

```

In addition to the above, the following is stored in r() :

Matrices
r(table)
matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

\section*{Methods and formulas}

Methods and formulas are presented under the following headings:
The total estimator
Survey data
The survey total estimator
The poststratified total estimator
Subpopulation estimation

\section*{The total estimator}

Let \(y\) denote the variable on which to calculate the total and \(y_{j}, j=1, \ldots, n\), denote an individual observation on \(y\). Let \(w_{j}\) be the frequency weight (or iweight or pweight), and if no weight is specified, define \(w_{j}=1\) for all \(j\). The sum of the weights is an estimate of the population size:
\[
\widehat{N}=\sum_{j=1}^{n} w_{j}
\]

If the population values of \(y\) are denoted by \(Y_{j}, j=1, \ldots, N\), the associated population total is
\[
Y=\sum_{j=1}^{N} Y_{j}=N \bar{y}
\]
where \(\bar{y}\) is the population mean. The total is estimated as
\[
\widehat{Y}=\widehat{N} \bar{y}
\]

The variance estimator for the total is
\[
\widehat{V}(\widehat{Y})=\widehat{N}^{2} \widehat{V}(\bar{y})
\]
where \(\widehat{V}(\bar{y})\) is the variance estimator for the mean; see \([\mathrm{R}]\) mean. The standard error of the total is the square root of the variance.

If \(x, x_{j}, \bar{x}\), and \(\widehat{X}\) are similarly defined for another variable (observed jointly with \(y\) ), the covariance estimator between \(\widehat{X}\) and \(\widehat{Y}\) is
\[
\widehat{\operatorname{Cov}}(\widehat{X}, \widehat{Y})=\widehat{N}^{2} \widehat{\operatorname{Cov}}(\bar{x}, \bar{y})
\]
where \(\widehat{\operatorname{Cov}}(\bar{x}, \bar{y})\) is the covariance estimator between two means; see \([\mathrm{R}]\) mean.

\section*{Survey data}

See [SVY] Variance estimation and [SVY] Poststratification for discussions that provide background information for the following formulas.

\section*{The survey total estimator}

Let \(Y_{j}\) be a survey item for the \(j\) th individual in the population, where \(j=1, \ldots, M\) and \(M\) is the size of the population. The associated population total for the item of interest is
\[
Y=\sum_{j=1}^{M} Y_{j}
\]

Let \(y_{j}\) be the survey item for the \(j\) th sampled individual from the population, where \(j=1, \ldots, m\) and \(m\) is the number of observations in the sample.

The estimator \(\widehat{Y}\) for the population total \(Y\) is
\[
\widehat{Y}=\sum_{j=1}^{m} w_{j} y_{j}
\]
where \(w_{j}\) is a sampling weight. The estimator for the number of individuals in the population is
\[
\widehat{M}=\sum_{j=1}^{m} w_{j}
\]

The score variable for the total estimator is the variable itself,
\[
z_{j}(\widehat{Y})=y_{j}
\]

\section*{The poststratified total estimator}

Let \(P_{k}\) denote the set of sampled observations that belong to poststratum \(k\), and define \(I_{P_{k}}(j)\) to indicate if the \(j\) th observation is a member of poststratum \(k\), where \(k=1, \ldots, L_{P}\) and \(L_{P}\) is the number of poststrata. Also, let \(M_{k}\) denote the population size for poststratum \(k . P_{k}\) and \(M_{k}\) are identified by specifying the poststrata() and postweight () options on svyset; see [SVY] svyset.

The estimator for the poststratified total is
\[
\widehat{Y}^{P}=\sum_{k=1}^{L_{P}} \frac{M_{k}}{\widehat{M}_{k}} \widehat{Y}_{k}=\sum_{k=1}^{L_{P}} \frac{M_{k}}{\widehat{M}_{k}} \sum_{j=1}^{m} I_{P_{k}}(j) w_{j} y_{j}
\]
where
\[
\widehat{M}_{k}=\sum_{j=1}^{m} I_{P_{k}}(j) w_{j}
\]

The score variable for the poststratified total is
\[
z_{j}\left(\widehat{Y}^{P}\right)=\sum_{k=1}^{L_{P}} I_{P_{k}}(j) \frac{M_{k}}{\widehat{M}_{k}}\left(y_{j}-\frac{\widehat{Y}_{k}}{\widehat{M}_{k}}\right)
\]

\section*{Subpopulation estimation}

Let \(S\) denote the set of sampled observations that belong to the subpopulation of interest, and define \(I_{S}(j)\) to indicate if the \(j\) th observation falls within the subpopulation.

The estimator for the subpopulation total is
\[
\widehat{Y}^{S}=\sum_{j=1}^{m} I_{S}(j) w_{j} y_{j}
\]
and its score variable is
\[
z_{j}\left(\widehat{Y}^{S}\right)=I_{S}(j) y_{j}
\]

The estimator for the poststratified subpopulation total is
\[
\widehat{Y}^{P S}=\sum_{k=1}^{L_{P}} \frac{M_{k}}{\widehat{M}_{k}} \widehat{Y}_{k}^{S}=\sum_{k=1}^{L_{P}} \frac{M_{k}}{\widehat{M}_{k}} \sum_{j=1}^{m} I_{P_{k}}(j) I_{S}(j) w_{j} y_{j}
\]
and its score variable is
\[
z_{j}\left(\widehat{Y}^{P S}\right)=\sum_{k=1}^{L_{P}} I_{P_{k}}(j) \frac{M_{k}}{\widehat{M}_{k}}\left\{I_{S}(j) y_{j}-\frac{\widehat{Y}_{k}^{S}}{\widehat{M}_{k}}\right\}
\]

\section*{References}

Cochran, W. G. 1977. Sampling Techniques. 3rd ed. New York: Wiley.
Stuart, A., and J. K. Ord. 1994. Kendall's Advanced Theory of Statistics: Distribution Theory, Vol. 1. 6th ed. London: Arnold.

\section*{Also see}
[R] total postestimation - Postestimation tools for total
[R] mean - Estimate means
\([\mathrm{R}]\) proportion - Estimate proportions
[ R ] ratio - Estimate ratios
[MI] Estimation - Estimation commands for use with mi estimate
[SVY] Direct standardization - Direct standardization of means, proportions, and ratios
[SVY] Poststratification - Poststratification for survey data
[SVY] Subpopulation estimation - Subpopulation estimation for survey data
[SVY] svy estimation - Estimation commands for survey data
[SVY] Variance estimation - Variance estimation for survey data
[U] 20 Estimation and postestimation commands

\section*{Title}
total postestimation - Postestimation tools for total

Postestimation commands Remarks and examples Also see

\section*{Postestimation commands}

The following postestimation commands are available after total:
\begin{tabular}{ll} 
Command & Description \\
\hline contrast & contrasts and ANOVA-style joint tests of estimates \\
estat vce & variance-covariance matrix of the estimators (VCE) \\
estat (svy) & \begin{tabular}{l} 
postestimation statistics for survey data \\
cataloging estimation results
\end{tabular} \\
estimates & \begin{tabular}{l} 
table of estimation results \\
etable \\
lincom estimates, standard errors, testing, and inference for linear combinations \\
of coefficients
\end{tabular} \\
marginsplot & \begin{tabular}{l} 
graph the results from total \\
point estimates, standard errors, testing, and inference for nonlinear combinations \\
of coefficients
\end{tabular} \\
nlcom & \begin{tabular}{l} 
pairwise comparisons of estimates
\end{tabular} \\
pwcompare & Wald tests of simple and composite linear hypotheses \\
test & Wald tests of nonlinear hypotheses \\
testnl &
\end{tabular}

\section*{Remarks and examples}

\section*{> Example 1}

Continuing with our data on incidence of heart attacks from example 1 in [R] total, we want to test whether there are twice as many heart attacks among men than women in the population.
```

. use https://www.stata-press.com/data/r18/total
(Fictional incidence of heart-attack data)
. total heartatk [pw=swgt], over(sex)
(output omitted)
. test heartatk@1.sex = 2 * heartatk@2.sex
( 1) c.heartatk@1bn.sex - 2*c.heartatk@2.sex = 0
F( 1, 4945) = 1.25
Prob > F = 0.2643

```

Thus we do not reject our hypothesis that the total number of heart attacks for men is twice that for women in the population.

\section*{Also see}
[R] total - Estimate totals
[U] 20 Estimation and postestimation commands

\section*{Title}
tpoisson - Truncated Poisson regression
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
Acknowledgment & References & Also see &
\end{tabular}

\section*{Description}
tpoisson fits a truncated Poisson regression model when the number of occurrences of an event is restricted to be above a truncation point, below a truncation point, or between two truncation points. Truncated Poisson models are appropriate when neither the dependent variable nor the covariates are observed in the truncated part of the distribution. By default, tpoisson assumes left-truncation occurs at zero, but truncation may be specified at other fixed points or at values that vary across observations.

\section*{Quick start}

Truncated Poisson regression of y on x 1 and x 2 with left-truncation at 0 tpoisson y x1 x2

Add categorical variable a using factor-variable syntax
tpoisson y x1 x2 i.a
Same as above, but report incidence-rate ratios and use a constant truncation point of 4 tpoisson y x1 x2 i.a, irr ll(4)

With offset variable lnexp
tpoisson y x1 x2 i.a, offset (lnexp)
Same as above, but with a variable truncation point stored in variable min
tpoisson y x1 x2 i.a, offset (lnexp) ll(min)
With variable left- and right-truncation points
tpoisson \(y \mathrm{x} 1 \mathrm{x} 2\), \(\mathrm{ll}(\mathrm{min}) \mathrm{ul}(\max )\)
With variable right-truncation points
tpoisson \(y\) x1 x2, ul(max)
Constrain the coefficients for 2. a and 3 . a to equality
constraint define \(12 . a=3 . a\)
tpoisson y x1 x2 i.a, constraints(1)

\section*{Menu}

Statistics \(>\) Count outcomes \(>\) Truncated Poisson regression

\section*{Syntax}
tpoisson depvar \([\) indepvars \(][\) if \(][\) in \(][\) weight \(][\), options \(]\)
options
Model
noconstant
11(\#| varname)
ul (\# | varname)
exposure ( varname \(_{e}\) )
offset ( varname \(_{o}\) )
constraints (constraints)
SE/Robust
vce (vcetype)

Reporting
level(\#)
irr
nocnsreport
display_options

Description
suppress constant term
lower limit for truncation; default is \(11(0)\) when neither ll() nor ul() is specified
upper limit for truncation
include \(\ln \left(\right.\) varname \(\left._{e}\right)\) in model with coefficient constrained to 1
include varname \({ }_{o}\) in model with coefficient constrained to 1
apply specified linear constraints
vcetype may be oim, robust, cluster clustvar, opg, bootstrap, or jackknife
set confidence level; default is level (95)
report incidence-rate ratios
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

Maximization
maximize_options
collinear keep collinear variables
coeflegend
control the maximization process; seldom used
display legend instead of statistics
indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
depvar and indepvars may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bayes, bootstrap, by, collect, fmm, fp, jackknife, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: tpoisson and [FMM] fmm: tpoisson.
Weights are not allowed with the bootstrap prefix; see [R] bootstrap.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options}

\footnotetext{
Model
noconstant; see [R] Estimation options.
11 (\#|varname) and ul(\#|varname) specify the lower and upper limits for truncation, respectively. You may specify nonnegative integer values for one or both.

When neither 11 () nor \(u l()\) is specified, the default is zero truncation, \(11(0)\), equivalent to left-truncation at zero.
}
exposure ( varname \(_{e}\) ), offset ( varname \(_{o}\) ), constraints (constraints); see [R] Estimation options.

SE/Robust
vce(vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.

Reporting
level(\#); see [R] Estimation options.
irr reports estimated coefficients transformed to incidence-rate ratios, that is, \(e^{\beta_{i}}\) rather than \(\beta_{i}\). Standard errors and confidence intervals are similarly transformed. This option affects how results are displayed, not how they are estimated. irr may be specified at estimation or when replaying previously estimated results.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\% fmt), sformat (\% \(\% m t\) ), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate(\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from(init-specs); see [R] Maximize. These options are seldom used.

Setting the optimization type to technique (bhhh) resets the default vcetype to vce (opg).
The following options are available with tpoisson but are not shown in the dialog box:
collinear, coeflegend; see [R] Estimation options.

\section*{Remarks and examples}
tpoisson fits a truncated Poisson regression model by maximum likelihood estimation when the number of occurrences of an event is restricted to be above a truncation point, below a truncation point, or between two truncation points. If the dependent variable is not truncated, standard Poisson regression may be more appropriate; see [ \(R\) ] poisson.

When the data are truncated, we do not observe either the dependent variable or the covariates. For example, consider a study about the number of days that individuals with hyperglycemia are hospitalized after presenting to the hospital. If we select our sample only from admission records, then the sample is truncated at zero because we have data only on individuals who stayed at least one day. Now assume that we are relying on billing data and that hospitals may submit either a final bill when a patient is discharged or an interim bill every 30 days. In this case, we have no information about patients who are hospitalized fewer than 1 day or more than 30 days. Our data are left-truncated at 0 and right-truncated at 30 .

A related phenomenon is censoring. For censored observations, we observe complete covariate information but only a censored value of the dependent variable. Different research designs can give rise to censored data or truncated data. See \([R]\) cpoisson for information about censored Poisson regression.

Truncated Poisson regression was first proposed by Grogger and Carson (1991). For an introduction to Poisson regression, see Cameron and Trivedi \((2005,2022)\) and Long and Freese (2014). For an introduction to truncated Poisson models, see Cameron and Trivedi (2013) and Long (1997, chap. 8).
\(>\) Example 1: Left-truncation at zero
Consider the Simonoff (2003) dataset of running shoes for a sample of runners who registered an online running log. A running-shoe marketing executive is interested in knowing how the number of pairs of running shoes purchased relates to other factors such as gender, marital status, age, education, income, typical number of runs per week, average miles run per week, and the preferred type of running. These data are naturally truncated at zero. A truncated Poisson model is fit to the number of pairs of shoes owned on runs per week, miles run per week, gender, age, and marital status.

No options are needed because zero truncation is the default for tpoisson.
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline \multicolumn{7}{|l|}{\begin{tabular}{l}
(Running shoes) \\
. tpoisson shoes rpweek mpweek male age married
\end{tabular}} \\
\hline \multicolumn{7}{|l|}{Iteration 0: Log likelihood \(=-88.328151\)} \\
\hline \multicolumn{7}{|l|}{Iteration 1: Log likelihood \(=-86.272639\)} \\
\hline \multicolumn{7}{|l|}{Iteration 2: Log likelihood \(=-86.257999\)} \\
\hline \multicolumn{7}{|l|}{Iteration 3: Log likelihood \(=-86.257994\)} \\
\hline Truncated Poi & son regressi & & & Number & f obs & 60 \\
\hline \multicolumn{2}{|l|}{Limits: lower =} & & & LR chi2 & & 22.75 \\
\hline uppe & +in & & & Prob > & hi2 & 0.0004 \\
\hline \multicolumn{4}{|l|}{Log likelihood \(=-86.257994\)} & Pseudo & & 0.1165 \\
\hline shoes & Coefficient & Std. err. & z & \(P>|z|\) & [95\% & interval] \\
\hline rpweek & . 1575811 & . 1097893 & 1.44 & 0.151 & -. 05 & . 3727641 \\
\hline mpweek & . 0210673 & . 0091113 & 2.31 & 0.021 & . 003 & . 0389252 \\
\hline male & . 0446134 & . 2444626 & 0.18 & 0.855 & -. 434 & . 5237513 \\
\hline age & . 0185565 & . 0137786 & 1.35 & 0.178 & -. 00 & . 045562 \\
\hline married & -. 1283912 & . 2785044 & -0.46 & 0.645 & -. 674 & . 4174674 \\
\hline _cons & -1.205844 & . 6619774 & -1.82 & 0.069 & -2.50 & . 0916078 \\
\hline
\end{tabular}

Using the zero-truncated Poisson regression with these data, only the coefficient on average miles per week is statistically significant at the \(5 \%\) level.

\section*{> Example 2: Left-truncation with a fixed-truncation point}

Semiconductor manufacturing requires that silicon wafers be coated with a layer of metal oxide. The depth of this layer is strictly controlled. In this example, a critical oxide layer is designed for \(300 \pm 20\) angstroms ( \(\AA\) ).

After the oxide layer is coated onto a wafer, the wafer enters a photolithography step in which the lines representing the electrical connections are printed on the oxide and later etched and filled with metal. The widths of these lines are measured. In this example, they are controlled to \(90 \pm 5\) micrometers ( \(\mu \mathrm{m}\) ).

After these and other steps, each wafer is electrically tested at probe. If too many failures are discovered, the wafer is rejected and sent for engineering analysis. In this example, the maximum number of probe failures tolerated for this product is 10.

A major failure at probe has been encountered- 88 wafers had more than 10 failures each. The 88 wafers that failed were tested using 4 probe machines. The engineer suspects that the failures were a result of faulty probe machines, poor depth control, or poor line widths. The line widths and depths in these data are the actual measurement minus its specification target, \(300 \AA\) for the oxide depths and \(90 \mu \mathrm{~m}\) for the line widths.

The following table tabulates the average failure rate for each probe using Stata's mean command; see [ \(R\) ] mean.
\begin{tabular}{l}
. use https://www.stata-press.com/data/r18/probe \\
(Silicon wafers) \\
. mean failures, over(probe) \\
Mean estimation \\
M \\
\hline
\end{tabular}

The \(95 \%\) confidence intervals in this table suggest that there are about 5-11 additional failures per wafer on probe 4 . These are unadjusted for varying line widths and oxide depths. Possibly, probe 4 received the wafers with larger line widths or extreme oxide depths.

Truncated Poisson regression more clearly identifies the root causes for the increased failures by estimating the differences between probes adjusted for the line widths and oxide depths. It also allows us to determine whether the deviations from specifications in line widths or oxide depths might be contributing to the problem.
\begin{tabular}{|c|c|c|c|c|c|c|c|}
\hline \multicolumn{4}{|l|}{Truncated Poisson regression} & \multicolumn{4}{|l|}{Number of obs \(=88\)} \\
\hline \multicolumn{4}{|l|}{\multirow[t]{2}{*}{\(\begin{array}{rlr}\text { Limits: } & \text { lower }= & 10 \\ & \text { upper }= & +i n f\end{array}\)}} & \multicolumn{4}{|l|}{LR chi2(5) \(=73.70\)} \\
\hline & & & & Prob > & hi2 & & 0.0000 \\
\hline \multicolumn{4}{|l|}{Log likelihood \(=-239.35746\)} & \multicolumn{4}{|l|}{Pseudo R2 \(=0.1334\)} \\
\hline failures & Coefficient & Std. err. & z & \(P>|z|\) & [95\% & & interval] \\
\hline probe & & & & & & & \\
\hline 2 & -. 1113037 & . 1019786 & -1.09 & 0.275 & -. 311 & & . 0885707 \\
\hline 3 & . 0114339 & . 1036032 & 0.11 & 0.912 & -. 191 & & . 2144924 \\
\hline 4 & . 4254115 & . 0841277 & 5.06 & 0.000 & . 260 & & . 5902989 \\
\hline depth & -. 0005034 & . 0033375 & -0.15 & 0.880 & -. 007 & & . 006038 \\
\hline width & . 0330225 & . 015573 & 2.12 & 0.034 & . 002 & & . 063545 \\
\hline _cons & 2.714025 & . 0752617 & 36.06 & 0.000 & 2.56 & & 2.861536 \\
\hline
\end{tabular}

The coefficients listed for the probes are testing the null hypothesis: \(H_{0}\) : probe \(_{i}=\) probe \(_{1}\), where \(i\) equals 2,3 , and 4 . Because the only coefficient that is statistically significant is the one for testing for \(H_{0}:\) probe \(_{4}=\) probe \(_{1}, p<0.001\), and because the \(p\)-values for the other probes are not statistically significant, that is, \(p \geq 0.275\), the implication is that there is a difference between probe 4 and the other machines. Because the coefficient for this test is positive, 0.425 , the conclusion is that the average failure rate for probe 4 , after adjusting for line widths and oxide depths, is higher than the other probes. Possibly, probe 4 needs calibration or the head used with this machine is defective.

Line-width control is statistically significant, \(p=0.034\), but variation in oxide depths is not causing the increased failure rate. The engineer concluded that the sudden increase in failures is the result of
two problems. First, probe 4 is malfunctioning, and second, there is a possible lithography or etching problem.

\section*{Stored results}
tpoisson stores the following in e() :
Scalars
```

        e(N)
    e(k)
    e(k_eq)
    number of observations
number of parameters

```
    e(k_eq_model)
    e(k_dv)
    e(df_m)
    e(r2_p)
    e(11)
    e(ll_0)
    e(N_clust)
    e(chi2)
    e(p)
    e(rank)
    \(e(i c)\)
    e(rc)
    e(converged)
number of equations in \(e(b)\)
number of equations in overall model test
number of dependent variables
model degrees of freedom
pseudo- \(R^{2}\)
log likelihood
log likelihood, constant-only model
number of clusters
\(\chi^{2}\)
\(p\)-value for model test
rank of e(V)
number of iterations
return code
1 if converged, 0 otherwise
Macros
```

e(cmd)
e(cmdline)
e(depvar)

```
e(llopt)
e(ulopt)
\(e\) (wtype)
e(wexp)
e(title)
e(clustvar)
e(offset)
e(chi2type)
e(vce)
e(vcetype)
e(opt)
\(e\) (which)
e(ml_method)
e(user)
e(technique)
e(properties)
e(predict)
e(asbalanced)
e(asobserved)
tpoisson
command as typed
name of dependent variable
contents of 11() , or 0 if neither 11() nor \(u l()\) is specified
contents of ul(), if specified
weight type
weight expression
title in estimation output
name of cluster variable
linear offset variable
Wald or LR; type of model \(\chi^{2}\) test
vcetype specified in vce()
title used to label Std. err.
type of optimization
max or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement predict
factor variables fvset as asbalanced
factor variables fvset as asobserved
Matrices
```

e(b)
e(Cns)
e(ilog)
e(gradient)
e(V)
e(V_modelbased)

```

Functions
e(sample) marks estimation sample
coefficient vector
constraints matrix
iteration \(\log\) (up to 20 iterations)
gradient vector
variance-covariance matrix of the estimators
model-based variance

In addition to the above, the following is stored in r() :
Matrices
\(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

\section*{Methods and formulas}

For a nonnegative count outcome \(Y\) with left-truncation point \(l l_{j}\) and right-truncation point \(u l_{j}\), we can write the truncated Poisson model as
\[
f\left(y_{j}\right)=\frac{\exp \left(-\lambda_{j}\right) \lambda_{j}^{y_{j}}}{y_{j}!\operatorname{Pr}\left(l l_{j}<Y<u l_{j} \mid \xi_{j}\right)}
\]
where
\[
\begin{aligned}
\xi_{j} & =\mathbf{x}_{j} \boldsymbol{\beta}+\text { offset }_{j} \\
\lambda_{j} & =\exp \left(\xi_{j}\right)
\end{aligned}
\]
and \(\mathbf{x}_{j}\) is a vector of observed covariates. The conditional probability of observing \(y_{j}\) events, therefore given by \(l l_{j}<y_{j}<u l_{j}\), is
\[
\operatorname{Pr}\left(Y=y_{j} \mid l l_{j}<y_{j}<u l_{j}, \mathbf{x}_{j}\right)=\frac{\exp \left(-\lambda_{j}\right) \lambda_{j}^{y_{j}}}{y_{j}!\operatorname{Pr}\left(l l_{j}<Y<u l_{j} \mid \mathbf{x}_{j}\right)}
\]

The log likelihood is given by
\[
\ln L=\sum_{j=1}^{n} w_{j}\left[-\lambda_{j}+\xi_{j} y_{j}-\ln \left(y_{j}!\right)-\ln \left\{\operatorname{Pr}\left(l l_{j}<Y<u l_{j} \mid \xi_{j}\right)\right\}\right]
\]

If no weights are specified, \(w_{j}=1\).
This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce (robust) and vce (cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.
tpoisson also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

\section*{Acknowledgment}

We gratefully acknowledge the previous work by Joseph Hilbe (1944-2017) (1999), coauthor of the Stata Press book Generalized Linear Models and Extensions.

\section*{References}

Cameron, A. C., and P. K. Trivedi. 2005. Microeconometrics: Methods and Applications. New York: Cambridge University Press.
——. 2013. Regression Analysis of Count Data. 2nd ed. New York: Cambridge University Press.
——. 2022. Microeconometrics Using Stata. 2nd ed. College Station, TX: Stata Press.
Farbmacher, H. 2011. Estimation of hurdle models for overdispersed count data. Stata Journal 11: 82-94.
Grogger, J. T., and R. T. Carson. 1991. Models for truncated counts. Journal of Applied Econometrics 6: 225-238. https://doi.org/10.1002/jae. 3950060302.
Hardin, J. W., and J. M. Hilbe. 2015. Regression models for count data from truncated distributions. Stata Journal 15: 226-246.
Hilbe, J. M. 1999. sg102: Zero-truncated Poisson and negative binomial regression. Stata Technical Bulletin 47: 37-40. Reprinted in Stata Technical Bulletin Reprints, vol. 8, pp. 233-236. College Station, TX: Stata Press.

Long, J. S. 1997. Regression Models for Categorical and Limited Dependent Variables. Thousand Oaks, CA: Sage.
Long, J. S., and J. Freese. 2014. Regression Models for Categorical Dependent Variables Using Stata. 3rd ed. College Station, TX: Stata Press.

Simonoff, J. S. 2003. Analyzing Categorical Data. New York: Springer.

\section*{Also see}
\([R]\) tpoisson postestimation - Postestimation tools for tpoisson
[R] poisson - Poisson regression
[R] nbreg - Negative binomial regression
\([R]\) tnbreg - Truncated negative binomial regression
[R] zinb - Zero-inflated negative binomial regression
[R] zip — Zero-inflated Poisson regression
[BAYES] bayes: tpoisson - Bayesian truncated Poisson regression
[FMM] fmm: tpoisson - Finite mixtures of truncated Poisson regression models
[SVY] svy estimation - Estimation commands for survey data
[XT] xtpoisson - Fixed-effects, random-effects, and population-averaged Poisson models
[U] 20 Estimation and postestimation commands

\section*{Title}
tpoisson postestimation - Postestimation tools for tpoisson
\begin{tabular}{lll} 
Postestimation commands & predict & margins
\end{tabular} Remarks and examples

\section*{Postestimation commands}

The following postestimation commands are available after tpoisson:
\(\left.\begin{array}{ll}\text { Command } & \text { Description } \\
\hline \text { contrast } & \begin{array}{l}\text { contrasts and ANOVA-style joint tests of estimates } \\
\text { estat ic }\end{array} \\
\text { Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian in- } \\
\text { formation criteria (AIC, CAIC, AICc, and BIC) }\end{array}\right]\)\begin{tabular}{ll} 
estat vummarize & summary statistics for the estimation sample \\
estat (svy) & variance-covariance matrix of the estimators (VCE) \\
postestimation statistics for survey data \\
estable & cataloging estimation results \\
* table of estimation results
\end{tabular}

\footnotetext{
*forecast, hausman, and lrtest are not appropriate with svy estimation results.
}

\section*{predict}

\section*{Description for predict}
predict creates a new variable containing predictions such as numbers of events, incidence rates, conditional means, probabilities, conditional probabilities, linear predictions, standard errors, and the equation-level score.

\section*{Menu for predict}

Statistics \(>\) Postestimation

\section*{Syntax for predict}
```

predict [type] newvar [if] [in] [, statistic nooffset]

```
statistic Description
Main
n
ir
cm
\(\operatorname{pr}(n)\)
\(\operatorname{pr}(a, b)\)
\(\operatorname{cpr}(n)\)
\(\operatorname{cpr}(a, b)\)
xb
stdp
score
number of events; the default
incidence rate
conditional mean, \(E\left(y_{j} \mid l l_{j}<y_{j}<u l_{j}\right)\)
probability \(\operatorname{Pr}\left(y_{j}=n\right)\)
probability \(\operatorname{Pr}\left(a \leq y_{j} \leq b\right)\)
conditional probability \(\operatorname{Pr}\left(y_{j}=n \mid l l_{j}<y_{j}<u l_{j}\right)\)
conditional probability \(\operatorname{Pr}\left(a \leq y_{j} \leq b \mid l l_{j}<y_{j}<u l_{j}\right)\)
linear prediction
standard error of the linear prediction
first derivative of the log likelihood with respect to \(\mathbf{x}_{j} \boldsymbol{\beta}\)
These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

\section*{Options for predict}
\(\qquad\) Main
n , the default, calculates the predicted number of events, which is \(\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}\right)\) if neither offset () nor exposure() was specified when the model was fit; \(\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}+\right.\) offset \(\left._{j}\right)\) if offset () was specified; or \(\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}\right) \times\) exposure \(_{j}\) if exposure() was specified.
ir calculates the incidence rate \(\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}\right)\), which is the predicted number of events when exposure is 1 . This is equivalent to specifying both the n and the nooffset options.
cm calculates the conditional mean,
\[
E\left(y_{j} \mid l l_{j}<y_{j}<u l_{j}\right)=\frac{E\left(y_{j}, l l_{j}<y_{j}<u l_{j}\right)}{\operatorname{Pr}\left(l l_{j}<y_{j}<u l_{j}\right)}
\]
where \(l l_{j}\) is the left-truncation point specified at estimation and \(u l_{j}\) is the right-truncation point specified at estimation.
\(\operatorname{pr}(n)\) calculates the probability \(\operatorname{Pr}\left(y_{j}=n\right)\), where \(n\) is a nonnegative integer that may be specified as a number or a variable.
\(\operatorname{pr}(a, b)\) calculates the probability \(\operatorname{Pr}\left(a \leq y_{j} \leq b\right)\), where \(a\) and \(b\) are nonnegative integers that may be specified as numbers or variables;
\(b\) missing ( \(b \geq\).) means \(+\infty\);
\(\operatorname{pr}(20,\).\() calculates \operatorname{Pr}\left(y_{j} \geq 20\right)\);
\(\operatorname{pr}(20, b)\) calculates \(\operatorname{Pr}\left(y_{j} \geq 20\right)\) in observations for which \(b \geq\). and calculates
\(\operatorname{Pr}\left(20 \leq y_{j} \leq b\right)\) elsewhere.
\(\operatorname{pr}(., b)\) produces a syntax error. A missing value in an observation of the variable \(a\) causes a missing value in that observation for \(\operatorname{pr}(a, b)\).
\(\operatorname{cpr}(n)\) calculates the conditional probability \(\operatorname{Pr}\left(y_{j}=n \mid l l_{j}<y_{j}<u l_{j}\right)\), where \(n\) is a nonnegative integer that may be specified as a number or a variable. \(l l_{j}\) and \(u l_{j}\) are as defined in cm .
\(\operatorname{cpr}(a, b)\) calculates the conditional probability \(\operatorname{Pr}\left(a \leq y_{j} \leq b \mid l l_{j}<y_{j}<u l_{j}\right)\), where \(a\) and \(b\) are as defined in \(\operatorname{pr}(a, b)\) with the additional restrictions that \(a>l l_{j}\) and \(b<u l_{j} . l l_{j}\) and \(u l_{j}\) are as defined in cm .
xb calculates the linear prediction, which is \(\mathbf{x}_{j} \boldsymbol{\beta}\) if neither offset() nor exposure() was specified when the model was fit; \(\mathbf{x}_{j} \boldsymbol{\beta}+\) offset \(_{j}\) if offset() was specified; or \(\mathbf{x}_{j} \boldsymbol{\beta}+\ln \left(\right.\) exposure \(\left._{j}\right)\) if exposure() was specified; see nooffset below.
stdp calculates the standard error of the linear prediction.
score calculates the equation-level score, \(\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)\).
nooffset is relevant only if you specified offset() or exposure() when you fit the model. It modifies the calculations made by predict so that they ignore the offset or exposure variable; the linear prediction is treated as \(\mathbf{x}_{j} \boldsymbol{\beta}\) rather than as \(\mathbf{x}_{j} \boldsymbol{\beta}+\) offset \(_{j}\) or \(\mathbf{x}_{j} \boldsymbol{\beta}+\ln \left(\right.\) exposure \(\left._{j}\right)\). Specifying predict..., nooffset is equivalent to specifying predict..., ir.

\section*{margins}

\section*{Description for margins}
margins estimates margins of response for numbers of events, incidence rates, conditional means, probabilities, conditional probabilities, and linear predictions.

\section*{Menu for margins}

Statistics \(>\) Postestimation

\section*{Syntax for margins}
```

margins [marginlist] [, options]
margins [marginlist], predict(statistic ...) [自redict(statistic ...)...] [options]

```
statistic Description
n
ir
cm
\(\mathrm{pr}(n)\)
\(\operatorname{pr}(a, b)\)
\(\mathrm{cpr}(n)\)
\(\mathrm{cpr}(a, b)\)
xb
stdp
score
number of events; the default incidence rate
    conditional mean, \(E\left(y_{j} \mid l l_{j}<y_{j}<u l_{j}\right)\)
    probability \(\operatorname{Pr}\left(y_{j}=n\right)\)
    probability \(\operatorname{Pr}\left(a \leq y_{j} \leq b\right)\)
    conditional probability \(\operatorname{Pr}\left(y_{j}=n \mid l l_{j}<y_{j}<u l_{j}\right)\)
    conditional probability \(\operatorname{Pr}\left(a \leq y_{j} \leq b \mid l l_{j}<y_{j}<u l_{j}\right)\)
    linear prediction
    not allowed with margins
    not allowed with margins

Statistics not allowed with margins are functions of stochastic quantities other than e(b). For the full syntax, see [R] margins.

\section*{Remarks and examples}
- Example 1: Obtaining margins of the conditional mean

In example 1 of \([\mathrm{R}]\) tpoisson, a truncated Poisson model is fit to the number of pairs of shoes owned on runs per week, miles run per week, gender, age, and marital status. We continue that example to determine the effect of miles run per week on the average number of pairs of shoes owned.

After reading in the data, we use summarize to obtain the 25 th, 50 th, and 75 th percentiles for miles run per week.
```

. use https://www.stata-press.com/data/r18/runshoes
(Running shoes)
. summarize mpweek, detail
Miles per week

```
\begin{tabular}{rrrlr}
\hline & Percentiles & Smallest & & \\
\(1 \%\) & 5 & 5 & & \\
\(5 \%\) & 5 & 5 & & \\
\(10 \%\) & 5 & 5 & Obs & 60 \\
\(25 \%\) & 12.5 & 5 & Sum of wgt. & \\
\(50 \%\) & 27.5 & & Mean & 24.71167 \\
& & Largest & Std. dev. & 14.34934 \\
\(75 \%\) & 32.5 & 47.5 & & \\
\(90 \%\) & 47.5 & 47.5 & Variance & 205.9034 \\
\(95 \%\) & 47.5 & 47.5 & Skewness & .1948568 \\
\(99 \%\) & 57.5 & 57.5 & Kurtosis & 2.065304
\end{tabular}

We fit the model from example 1 of [R] tpoisson again. We next specify these values for the percentiles to margins to estimate the conditional mean of the number of pairs of shoes at different quantiles of miles run per week. To do this, we use the at () option of margins.
```

. quietly tpoisson shoes rpweek mpweek male age married
. margins, at(mpweek=(12.5 27.5 32.5)) predict(cm)
Predictive margins Number of obs = 60
Model VCE: OIM
Expression: Conditional mean of n > ll(0), predict(cm)
1._at: mpweek = 12.5
2._at: mpweek = 27.5
3._at: mpweek = 32.5

```
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline & \multicolumn{3}{|c|}{Delta-method} & \(P>|z|\) & \multicolumn{2}{|l|}{[95\% conf. interval]} \\
\hline _at & & & & & & \\
\hline 1 & 1.942149 & . 2111564 & 9.20 & 0.000 & 1.52829 & 2.356008 \\
\hline 2 & 2.376253 & . 1714522 & 13.86 & 0.000 & 2.040213 & 2.712293 \\
\hline 3 & 2.564339 & . 1948129 & 13.16 & 0.000 & 2.182513 & 2.946165 \\
\hline
\end{tabular}

We see that people who run 12.5 miles per week are expected to own 1.94 pairs of shoes. The expected number of pairs of shoes owned increases as the average miles per week increases. We expect people who run 27.5 miles per week have 2.38 pairs of shoes and those who run 32.5 miles per week have 2.56 pairs of shoes.

\section*{Methods and formulas}

Continuing from Methods and formulas in [R] tpoisson, the equation-level score is given by
\[
\operatorname{score}(\mathbf{x} \boldsymbol{\beta})_{j}=y_{j}-\lambda_{j}-\frac{\left\{\exp \left(-\lambda_{j}\right) \lambda_{j}^{l l_{j}} / l l_{j}!-\exp \left(-\lambda_{j}\right) \lambda_{j}^{\left(u l_{j}-1\right)} /\left(u l_{j}-1\right)!\right\} \lambda_{j}}{\operatorname{Pr}\left(l l_{j}<Y<u l_{j} \mid \xi_{j}\right)}
\]

\section*{Also see}
[R] tpoisson - Truncated Poisson regression
[U] 20 Estimation and postestimation commands

\section*{Title}
translate - Print and translate logs
\begin{tabular}{llll} 
Description & Quick start & Syntax & Options for print \\
Options for translate & Remarks and examples & Stored results & Also see
\end{tabular}

\section*{Description}
print prints log, SMCL, and text files. translate translates log and SMCL files from one format to another, the other typically being suitable for printing. translate can also translate SMCL logs to plain text.

\section*{Quick start}

Translate SMCL-format log mylog.smcl to a text log file mylog.log translate mylog.smcl mylog.log

Same as above, but translate mylog.smcl to PostScript file mylog.ps
translate mylog.smcl mylog.ps
Send output from the Results window to new mylog.txt when no log was started translate @Results mylog.txt

\section*{Syntax}

Print log and SMCL files
print filename [, like(ext) name(windowname) override_options]
Translate \(\log\) files to SMCL files and vice versa
```

translate filename in filename out [, translator(tname) name(windowname)
override_options replace]

```

View translator parameter settings
```

translator query [tname]

```

Change translator parameter settings translator set [tname setopt setval]

Return translator parameter settings to default values
translator reset tname

List current mappings from one extension to another transmap query [.ext]

Specify that files with one extension be treated the same as files with another extension
```

transmap define .ext new .ext old

```
filename in print, in addition to being a filename to be printed, may be specified as @Results to mean the Results window and @Viewer to mean the Viewer window.
filename \(_{\text {in }}\) in translate may be specified just as filename in print.
thame in translator specifies the name of a translator; see the translator () option under Options for translate.

\section*{Options for print}
like (ext) specifies how the file should be translated to a form suitable for printing. The default is to determine the translation method from the extension of filename. Thus, mylog.smcl is translated according to the rule for translating smcl files, myfile.txt is translated according to the rule for translating txt files, and so on. (These rules are, in fact, translate's smcl2prn and txt2prn translators, but put that aside for the moment.)
Rules for the following extensions are predefined:
\begin{tabular}{ll}
.txt & assume input file contains plain text \\
.log & assume input file contains Stata \(\log\) text \\
.smel & assume input file contains SMCL
\end{tabular}

To print a file that has an extension different from those listed above, you can define a new extension, but you do not have to do that. Assume that you wish to print the file read.me, which you know to contain plain text. If you were just to type print read.me, you would be told that Stata cannot translate .me files. (You would actually be told that the translator for me2prn was not found.) You could type print read.me, like(txt) to tell print to print read.me like a .txt file.

On the other hand, you could type
```

. transmap define .me .txt

```
to tell Stata that .me files are always to be treated like .txt files. If you did that, Stata would remember the new rule, even in future sessions.
When you specify the like() option, you override the recorded rules. So, if you were to type print mylog.smcl, like(txt), the file would be printed as plain text (meaning that all the SMCL commands would show).
name (windowname) specifies which window to print when printing a Viewer. The default is for Stata to print the topmost Viewer [Unix(GUI) users: See the third technical note in Printing files, Unix]. The name() option is ignored when printing the Results window.
The window name is located inside the parentheses in the window title. For example, if the title for a Viewer window is Viewer (\#1) [help print], the name for the window is \#1.
override_options refer to translate's options for overriding default values. print uses translate to translate the file into a format suitable for sending to the printer, and thus translate's override_options may also be used with print. The settings available vary between each translator (for example, smcl2ps will have different settings than smcl2txt) and may also differ across operating systems (for example, Windows may have different printing options than macOS). To find out what you can override when printing .smcl files, type
```

. translator query smcl2prn
(output omitted)

```

In the omitted output, you might learn that there is an rmargin \# tunable value, which specifies the right margin in inches. You could specify the override_option rmargin(\#) to temporarily override the default value, or you could type translator set smcl2prn rmargin \# beforehand to permanently reset the value.
Alternatively, on some computers with some translators, you might discover that nothing can be set.

\section*{Options for translate}
translator (tname) specifies the name of the translator to be used to translate the file. The available translators are
\begin{tabular}{lll} 
tname & Input & Output \\
\hline smc12ps & SMCL & PostScript \\
log2ps & Stata text log & PostScript \\
txt2ps & generic text file & PostScript \\
Viewer2ps & Viewer window & PostScript \\
Results2ps & Results window & PostScript \\
smcl2prn & SMCL & default printer format \\
log2prn & Stata text log & default printer format \\
txt2prn & generic text log & default printer format \\
Results2prn & Results window & default printer format \\
Viewer2prn & Viewer window & default printer format \\
smcl2txt & SMCL & generic text log \\
smcl2log & SMCL & Stata text log \\
Results2txt & Results window & generic text file \\
Viewer2txt & Viewer window & generic text file \\
smcl2pdf & SMCL & PDF \\
log2pdf & Stata text log & PDF \\
txt2pdf & generic text log & PDF \\
Results2pdf & Results window & PDF \\
Viewer2pdf & Viewer window & PDF \\
\hline
\end{tabular}

If translator () is not specified, translate determines which translator to use from extensions of the filenames specified. Typing translate myfile.smcl myfile.ps would use the smcl2ps translator. Typing translate myfile.smcl myfile.ps, translate(smcl2prn) would override the default and use the smcl2prn translator.
Actually, when you type translate \(a . b c . d\), translate looks up.\(b\) in the transmap extensionsynonym table. If.\(b\) is not found, the translator \(b 2 d\) is used. If.\(b\) is found in the table, the mapped extension is used (call it \(b^{\prime}\) ), and then the translator \(b^{\prime} 2 d\) is used. For example,
\begin{tabular}{ll} 
Command & Translator used \\
\hline . translate myfile.smcl myfile.ps & smcl2ps \\
. translate myfile.odd myfile.ps & odd2ps, which does not exist, so error \\
. transmap define .odd .txt & \\
. translate myfile.odd myfile.ps & txt2ps
\end{tabular}

You can list the mappings that translate uses by typing transmap query.
name (windowname) specifies which window to translate when translating a Viewer. The default is for Stata to translate the topmost Viewer. The name () option is ignored when translating the Results window.
The window name is located inside the parentheses in the window title. For example, if the title for a Viewer window is Viewer (\#1) [help print], the name for the window is \#1.
override_options override any of the default options of the specified or implied translator. To find out what you can override for, say, \(\log 2 \mathrm{ps}\), type
```

. translator query log2ps

```

In the omitted output, you might learn that there is an rmargin \# tunable value, which, for \(\log 2 \mathrm{ps}\), specifies the right margin in inches. You could specify the override_option rmargin(\#) to temporarily override the default value or type translator set log2ps rmargin \# beforehand to permanently reset the value.
replace specifies that filename \(_{\text {out }}\) be replaced if it already exists.

\section*{Remarks and examples}

Remarks are presented under the following headings:

\author{
Overview \\ Printing files \\ Printing files, Mac and Windows \\ Printing files, Unix \\ Translating files from one format to another
}

\section*{Overview}
print prints log, SMCL, and text files. Although there is considerable flexibility in how print (and translate, which print uses) can be set to work, they have already been set up and should just work:
. print mylog.smcl
. print mylog.log
Unix users may discover that they need to do a bit of setup before print works; see Printing files, Unix below. International Unix users may also wish to modify the default paper size. All users can tailor print and translate to their needs.
print may also be used to print the current contents of the Results window or the Viewer. For instance, the current contents of the Results window could be printed by typing
```

. print @Results

```
translate translates \(\log\) and SMCL files from one format to another, the other typically being suitable for printing. translate can also translate SMCL logs (logs created by typing, say, log using mylog) to plain text:
. translate mylog.smcl mylog.log
You can use translate to recover a log when you have forgotten to start one. You may type . translate @Results mylog.txt
to capture as plain text what is currently shown in the Results window.
This entry provides a general overview of print and translate and covers in detail the printing and translation of text (nongraphic) files.
translator query, translator set, and translator reset show, change, and restore the default values of the settings for each translator.
transmap define and transmap query create and show mappings from one file extension to another for use with print and translate.

For example, print myfile.txt knows to use a translator appropriate for printing text files because of the .txt extension. However, it does not know what to do with . xyz files. If you have .xyz files and always wish to treat them as .txt files, you can type transmap define .xyz .txt.

\section*{Printing files}

Printing should be easy; just type
. print mylog.smcl
. print mylog.log
You can use print to print SMCL files, plain text files, and even the contents of the Results and Viewer windows:
- print @Results
- print @Viewer
- print @Viewer, name(\#2)

For information about printing and translating graph files, see [G-2] graph print and see [G-2] graph export.

\section*{Printing files, Mac and Windows}

When you type print, you are using the same facility that you would be using if you had selected Print from the File menu. If you try to print a file that Stata does not know about, Stata will complain:
```

. print read.me
translator me2prn not found
r(111);

```

Then, you could type
```

. print read.me, like(txt)

```
to indicate that you wanted read.me sent to the printer in the same fashion as if the file were named readme.txt, or you could type
. transmap define .me .txt
. print read.me
Here you are telling Stata once and for all that you want files ending in .me to be treated in the same way as files ending in .txt. Stata will remember this mapping, even across sessions. To clear the .me mapping, type
. transmap define .me
To see all the mappings, type
. transmap query
To print to a file, use the translate command, not print:
. translate mylog.smcl mylog.prn
translate prints to a file by using the Windows print driver when the new filename ends in .prn. Under Mac, the prn translators are the same as the pdf translators. We suggest that you simply use the .pdf file extension when printing to a file.

\section*{Printing files, Unix}

Stata assumes that you have a PostScript printer attached to your Unix computer and that the Unix command \(\operatorname{lpr}(1)\) can be used to send PostScript files to it, but you can change this. On your Unix system, typing
mycomputer\$ lpr < filename
may not be sufficient to print PostScript files. For instance, perhaps on your system you would need to type
```

mycomputer\$ lpr -Plexmark < filename

```
or
mycomputer\$ lpr -Plexmark filename
or something else. To set the print command to be lpr -Plexmark filename and to state that the printer expects to receive PostScript files, type
```

. printer define prn ps "lpr -Plexmark @"

```

To set the print command to lpr -Plexmark < filename and to state that the printer expects to receive plain text files, type
```

. printer define prn txt "lpr -Plexmark < @"

```

That is, just type the command necessary to send files to your printer and include an @ sign where the filename should be substituted. Two file formats are available: ps and txt. The default setting, as shipped from the factory, is
```

. printer define prn ps "lpr < @"

```

We will return to the printer command in the technical note that follows because it has some other capabilities you should know about.

In any case, after you redefine the default printer, the following should just work:
. print mylog.smcl
. print mylog.log
If you try to print a file that Stata does not know about, it will complain:
. print read.me
translator me2prn not found
\(r(111)\);
Here you could type
. print read.me, like(txt)
to indicate that you wanted read.me sent to the printer in the same fashion as if the file were named readme.txt, or you could type
. transmap define .me .txt
. print read.me
Here you are telling Stata once and for all that you want files ending in .me to be treated in the same way as files ending in .txt. Stata will remember this setting for. me, even across sessions.

If you want to clear the .me setting, type
. transmap define .me
If you want to see all your settings, type
```

. transmap query

```

\section*{- Technical note}

If the text you wish to print contains Unicode characters, those characters may not appear correctly in PostScript files because the PostScript fonts do not support Unicode. Stata will map as many characters as possible to characters supported by Unicode but will print a question mark (?) for any unsupported character. We recommend that you export the text to a PDF file, which has fonts with better support for Unicode characters.

\section*{- Technical note}

The syntax of the printer command is
```

printer define printername [{ps|txt } "Unix command with @"]
printer query [printername]

```

You may define multiple printers. By default, print uses the printer named prn, but print has the syntax
print filename [, like(ext) printer (printername) override_options ]
so, if you define multiple printers, you may route your output to them.
For instance, if you have a second printer on your system, you might type
```

. printer define lexmark ps "lpr -Plexmark < @"

```

After doing that, you could type
. print myfile.smcl, printer(lexmark)
Any printers that you set will be remembered even across sessions. You can delete printers:
- printer define lexmark

You can list all the defined printers by typing printer query, and you can list the definition of a particular printer, say, prn, by typing printer query prn.

The default printer prn we have predefined for you is
. printer define prn ps "lpr < @"
meaning that we assume that it is a PostScript printer and that the Unix command \(\operatorname{lpr}(1)\), without options, is sufficient to cause files to print. Feel free to change the default definition. If you change it, the change will be remembered across sessions.
- Technical note

Unix(GUI) users should note that X-Windows does not have the concept of a window z-order, which prevents Stata from determining which window is the topmost window. Instead, Stata determines which window is topmost based on which window has the focus. However, some window managers will set the focus to a window without bringing the window to the top. What Stata considers the topmost window may not appear topmost visually. For this reason, you should always use the name () option to ensure that the correct window is printed.

\section*{- Technical note}

When you select the Results window to print from the Print menu or toolbar button, the result is the same as if you were to issue the print command. When you select a Viewer window to print from the Print menu or toolbar button, the result is the same as if you were to issue the print command with a name() option.

The translation to PostScript format is done by translate and, in particular, is performed by the translators smcl2ps, log2ps, and txt2ps. There are many tunable parameters in each of these translators. You can display the current values of these tunable parameters for, say, smcl2ps by typing
. translator query smcl2ps
(output omitted)
and you can set any of the tunable parameters (for instance, setting smcl2ps's rmargin value to 1 ) by typing
```

. translator set smcl2ps rmargin 1
(output omitted)

```

Any settings you make will be remembered across sessions. You can reset smcl2ps to be as it was when Stata was shipped by typing
```

. translator reset smcl2ps

```

\section*{Translating files from one format to another}

If you have a SMCL log, which you might have created by previously typing log using mylog, you can translate it to an text log by typing
. translate myfile.smcl myfile.log
and you can translate it to a PostScript file by typing
. translate myfile.smcl myfile.ps
translate translates files from one format to another, and, in fact, print uses translate to produce a file suitable for sending to the printer.

When you type
```

. translate a.b c.d

```
translate looks for the predefined translator \(b 2 d\) and uses that to perform the translation. If there is a transmap synonym for \(b\), however, the mapped value \(b^{\prime}\) is used: \(b^{\prime} 2 d\).

Only certain translators exist, and they are listed under the description of the translate() option in Options for translate above, or you can type
. translator query
for a complete (and perhaps more up-to-date) list.
Anyway, translate forms the name \(b 2 d\) or \(b^{\prime} 2 d\), and if the translator does not exist, translate issues an error message. With the translator() option, you can specify exactly which translator to use, and then it does not matter how your files are named.

The only other thing to know is that some translators have tunable parameters that affect how they perform their translation. You can type
. translator query translator_name
to find out what those parameters are. Some translators have no tunable parameters, and some have many:
. translator query smcl2ps


You can temporarily override any setting by specifying the setopt(setval) option on the translate (or print) command. For instance, you can type
```

. translate ..., ... cmdnumber(off)

```
or you can reset the value permanently by typing
. translator set smcl2ps setopt setval
For instance,
. translator set smcl2ps cmdnumber off
If you reset a value, Stata will remember the change, even in future sessions.

Mac and Windows users: The smcl2ps (and the other \(* 2\) ps translators) are not used by print, even when you have a PostScript printer attached to your computer. Instead, the Mac or Windows print driver is used. Resetting smcl2ps values will not affect printing; instead, you change the defaults in the Printers Control Panel in Windows and by selecting Page Setup... from the File menu in Mac. You can, however, translate files yourself using the smcl2ps translator and the other \(* 2\) ps translators.

\section*{- Technical note: Using PDF translators (Windows and Unix)}

When using translate to translate a log, SMCL, or text file or a Stata graph into a PDF file, some characters may not display correctly in the resulting PDF file. This happens when translating content that contains Unicode characters from different languages; one font may not be adequate to display every character used. If an appropriate font is not available for a character, that character will not display correctly in the resulting PDF file.

Specific fonts can be used to build a font chain. Those fonts are specified using fontname and are separated by commas. The most preferred font should be listed first, followed by the less preferred fonts. A maximum of 16 fonts can be specified. Stata will inspect each character in the original content, and if a glyph for the corresponding character is not available in the default font, then each font in the font chain will be checked until a font is found to display that character.

You can set fonts in the font chain by specifying addfonts for each PDF translator as follows:
```

. translator set tname addfonts fontname

```

Here thame is the name of a PDF translator, which can be smcl2pdf, log2pdf, txt2pdf, Results2pdf, Viewer2pdf, or Graph2pdf. See the translator () option under Options for translate.

Stata has the ability to select a font automatically if the default font and the fonts in the font chain do not contain a glyph for the character. Automatic font selection can be controlled using the following:
. translator set thame autofont on off
By default, autofont is on. Unix (console) users will need to set the font directories so the fonts can be found.

PDF files accept Base 14 fonts, Type 1 fonts (.pfa or .pfb), TrueType fonts (.ttf or .ttc), and OpenType fonts (.otf). If fontname refers to a TrueType font, an OpenType font, or a Type 1 font, the corresponding TrueType, OpenType, or Type 1 font file will be searched for within the default font directory. Unix (console) users can type
```

. translator set tname fontdir fontdirectory

```
to set the default font directory. fontdirectory can contain multiple font directories separated by semicolons. If the specified TrueType, OpenType, or Type 1 font cannot be found under the current font directory, an error message will be issued.

\section*{Stored results}
transmap query .ext stores in macro r(suffix) the mapped extension (without the leading period) or stores ext if the ext is not mapped.
translator query translatorname stores setval in macro \(r\) (setopt) for every setopt, setval pair.
printer query printername (Unix only) stores in macro \(r\) (suffix) the "filetype" of the input that the printer expects (currently "ps" or "txt") and, in macro r(command), the command to send output to the printer.

\section*{Also see}
[R] \(\log\) - Echo copy of session to file
[G-2] graph export - Export current graph
[G-2] graph print - Print a graph
[G-2] graph set - Set graphics options
[P] smcl - Stata Markup and Control Language
[U] 15 Saving and printing output-log files

\section*{Title}
truncreg - Truncated regression
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
truncreg fits a regression model of depvar on indepvars from a sample drawn from a restricted part of the population. Under the normality assumption for the whole population, the error terms in the truncated regression model have a truncated normal distribution, which is a normal distribution that has been scaled upward so that the distribution integrates to one over the restricted range.

\section*{Quick start}

Truncated regression of y on x 1 and x 2 truncated below 16
truncreg y x1 x2, ll(16)
Specify that y is truncated above 35
truncreg y x1 x2, ul(35)
With y truncated below 17 and above 35
truncreg y x1 x2, ll(17) ul(35)
Specify a lower truncation point that varies across observations using the variable trunc
truncreg y x1 x2, ll(trunc)
Same as above, but with bootstrap standard errors using 200 replications
truncreg y x1 x2, ll(trunc) vce(bootstrap, reps(200))
See last estimates with legend of coefficient names instead of statistics
truncreg, coeflegend

\section*{Menu}

Statistics \(>\) Linear models and related \(>\) Truncated regression

\section*{Syntax}
truncreg depvar [indepvars] [if] [in] [weight] [, options]

\section*{options}

Model

\section*{noconstant}
ll(varname|\#)
ul (varname|\#)
offset (varname)
constraints (constraints)
SE/Robust
vce (vcetype)

Reporting
level(\#)
lrmodel
nocnsreport
display_options

Description
suppress constant term
left-truncation variable or limit
right-truncation variable or limit
include varname in model with coefficient constrained to 1
apply specified linear constraints
vcetype may be oim, robust, cluster clustvar, opg, bootstrap, or jackknife
set confidence level; default is level (95)
perform the likelihood-ratio model test instead of the default Wald test do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
control the maximization process; seldom used
keep collinear variables
display legend instead of statistics
indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
depvar and indepvars may contain time-series operators; see [U] 11.4.4 Time-series varlists.
bayes, bootstrap, by, collect, fmm, fp, jackknife, mi estimate, rolling, statsby, and svy are allowed;
see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: truncreg and [FMM] fmm: truncreg.
vce(bootstrap) and vce(jackknife) are not allowed with the mi estimate prefix; see [MI] mi estimate.
Weights are not allowed with the bootstrap prefix; see \([R]\) bootstrap.
aweights are not allowed with the jackknife prefix; see \([R]\) jackknife.
vce(), lrmodel, and weights are not allowed with the svy prefix; see [SVY] svy.
aweights, fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options}

\section*{Model \\ noconstant; see [R] Estimation options.}

11 (varname |\#) and ul (varname |\#) indicate the lower and upper limits for truncation, respectively. You may specify one or both. Observations with depvar \(\leq 11\) () are left-truncated, observations with depvar \(\geq \mathrm{ul}\) () are right-truncated, and the remaining observations are not truncated. See \([\mathrm{R}]\) tobit for a more detailed description.
offset (varname), constraints (constraints); see [R] Estimation options.

\section*{SE/Robust}
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
\(\qquad\) Reporting
level(\#), lrmodel, nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap (\#), fvwrapon(style), cformat (\%fmt), pformat (\% fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.
\(\qquad\) Maximization
maximize_options: difficult, technique (algorithm_spec), iterate (\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from (init_specs); see [R] Maximize. These options are seldom used, but you may use the ltol(\#) option to relax the convergence criterion; the default is \(1 \mathrm{e}-6\) during specification searches.
Setting the optimization type to technique (bhhh) resets the default vcetype to vce (opg).
The following options are available with truncreg but are not shown in the dialog box: collinear, coeflegend; see [R] Estimation options.

\section*{Remarks and examples}

Truncated regression fits a model of a dependent variable on independent variables from a restricted part of a population. Truncation is essentially a characteristic of the distribution from which the sample data are drawn. If \(x\) has a normal distribution with mean \(\mu\) and standard deviation \(\sigma\), the density of the truncated normal distribution is
\[
\begin{aligned}
f(x \mid a<x<b) & =\frac{f(x)}{\Phi\left(\frac{b-\mu}{\sigma}\right)-\Phi\left(\frac{a-\mu}{\sigma}\right)} \\
& =\frac{\frac{1}{\sigma} \phi\left(\frac{x-\mu}{\sigma}\right)}{\Phi\left(\frac{b-\mu}{\sigma}\right)-\Phi\left(\frac{a-\mu}{\sigma}\right)}
\end{aligned}
\]
where \(\phi\) and \(\Phi\) are the density and distribution functions of the standard normal distribution.
Compared with the mean of the untruncated variable, the mean of the truncated variable is greater if the truncation is from below, and the mean of the truncated variable is smaller if the truncation is from above. Moreover, truncation reduces the variance compared with the variance in the untruncated distribution.

\section*{- Example 1}

We will demonstrate truncreg with part of the Mroz dataset distributed with Berndt (1996). This dataset contains 753 observations on women's labor supply. Our subsample is of 250 observations, with 150 market laborers and 100 nonmarket laborers.


Sorted by:
. summarize, sep(0)
\begin{tabular}{r|rrrrr} 
Variable & Obs & Mean & Std. dev. & Min & Max \\
\hline lfp & 250 & .6 & .4908807 & 0 & 1 \\
whrs & 250 & 799.84 & 915.6035 & 0 & 4950 \\
kl6 & 250 & .236 & .5112234 & 0 & 3 \\
k618 & 250 & 1.364 & 1.370774 & 0 & 8 \\
wa & 250 & 42.92 & 8.426483 & 30 & 60 \\
we & 250 & 12.352 & 2.164912 & 5 & 17
\end{tabular}

We first perform ordinary least-squares estimation on the market laborers.
. regress whrs kl6 k 618 wa we if whrs \(>0\)
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline Source & SS & df & MS & Number of obs & = & 150 \\
\hline & & & & \(\mathrm{F}(4,145)\) & = & 2.80 \\
\hline Model & 7326995.15 & 4 & 1831748.79 & Prob > F & = & 0.0281 \\
\hline Residual & 94793104.2 & 145 & 653745.546 & R -squared & = & 0.0717 \\
\hline & & & & Adj R-squared & = & 0.0461 \\
\hline Total & 102120099 & 149 & 685369.794 & Root MSE & = & 808.55 \\
\hline
\end{tabular}
\begin{tabular}{r|rlccrr}
\hline whrs & Coefficient & Std. err. & t & \(\mathrm{P}>|\mathrm{t}|\) & [95\% conf. interval] \\
\hline kl6 & -421.4822 & 167.9734 & -2.51 & 0.013 & -753.4748 & -89.48953 \\
k618 & -104.4571 & 54.18616 & -1.93 & 0.056 & -211.5538 & 2.639668 \\
wa & -4.784917 & 9.690502 & -0.49 & 0.622 & -23.9378 & 14.36797 \\
we & 9.353195 & 31.23793 & 0.30 & 0.765 & -52.38731 & 71.0937 \\
_cons & 1629.817 & 615.1301 & 2.65 & 0.009 & 414.0371 & 2845.597 \\
\hline
\end{tabular}

Now, we use truncreg to perform truncated regression with truncation from below zero.
```

. truncreg whrs kl6 k618 wa we, ll(0)
(100 obs truncated)
Fitting full model:
Iteration 0: Log likelihood = -1205.6992
Iteration 1: Log likelihood = -1200.9873
Iteration 2: Log likelihood = -1200.9159

```
\begin{tabular}{lll} 
Iteration 3: Log likelihood \(=-1200.9157\) & & \\
Iteration 4: Log likelihood \(=-1200.9157\) & & \\
Truncated regression & & \\
Limit: Lower \(=\) 0 & Number of obs & \(=\) \\
Upper \(=+i n f\) & Wald chi2 (4) & \(=\) \\
Log likelihood \(=-1200.9157\) & Prob > chi2 & \(=\)
\end{tabular}
\begin{tabular}{r|rrrrrr}
\hline whrs & Coefficient & Std. err. & \multicolumn{2}{c}{z} & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
kl6 & -803.0042 & 321.3614 & -2.50 & 0.012 & -1432.861 & -173.1474 \\
k 618 & -172.875 & 88.72898 & -1.95 & 0.051 & -346.7806 & 1.030578 \\
wa & -8.821123 & 14.36848 & -0.61 & 0.539 & -36.98283 & 19.34059 \\
we & 16.52873 & 46.50375 & 0.36 & 0.722 & -74.61695 & 107.6744 \\
_cons & 1586.26 & 912.355 & 1.74 & 0.082 & -201.9233 & 3374.442 \\
\hline /sigma & 983.7262 & 94.44303 & 10.42 & 0.000 & 798.6213 & 1168.831 \\
\hline
\end{tabular}

If we assume that our data were censored, the tobit model is
\begin{tabular}{|c|c|c|c|}
\hline \multirow[t]{2}{*}{Tobit regression} & Number of obs & & 250 \\
\hline & Uncensored & & 150 \\
\hline Limits: Lower = 0 & Left-censored & = & 100 \\
\hline Upper = +inf & Right-censored & & 0 \\
\hline & LR chi2(4) & & 23.03 \\
\hline & Prob > chi2 & & 0.0001 \\
\hline Log likelihood = -1367.0903 & Pseudo R2 & & 0.0084 \\
\hline
\end{tabular}
\begin{tabular}{r|rlcrrr}
\hline whrs & Coefficient & Std. err. & t & P>|t| & [95\% conf. interval] \\
\hline kl6 & -827.7655 & 214.7521 & -3.85 & 0.000 & -1250.753 & -404.7781 \\
k618 & -140.0191 & 74.22719 & -1.89 & 0.060 & -286.221 & 6.182766 \\
wa & -24.97918 & 13.25715 & -1.88 & 0.061 & -51.09118 & 1.13281 \\
we & 103.6896 & 41.82629 & 2.48 & 0.014 & 21.30625 & 186.0729 \\
_cons & 589.0002 & 841.5952 & 0.70 & 0.485 & -1068.651 & 2246.652 \\
\hline var (e.whrs) & 1715859 & 216775.7 & & & 1337864 & 2200650 \\
\hline
\end{tabular}

\section*{Technical note}

Whether truncated regression is more appropriate than the ordinary least-squares estimation depends on the purpose of that estimation. If we are interested in the mean of wife's working hours conditional on the subsample of market laborers, least-squares estimation is appropriate. However if we are interested in the mean of wife's working hours regardless of market or nonmarket labor status, least-squares estimates could be seriously misleading.

Truncation and censoring are different concepts. A sample has been censored if no observations have been systematically excluded but some of the information contained in them has been suppressed. In a truncated distribution, only the part of the distribution above (or below, or between) the truncation points is relevant to our computations. We need to scale it up by the probability that an observation falls in the range that interests us to make the distribution integrate to one. The censored distribution used by tobit, however, is a mixture of discrete and continuous distributions. Instead of rescaling over the observable range, we simply assign the full probability from the censored regions to the censoring points. The truncated regression model is sometimes less well behaved than the tobit model.

Davidson and MacKinnon (1993) provide an example where truncation results in more inconsistency than censoring.

\section*{Stored results}
truncreg stores the following in e() :
Scalars
e(N)
e(N_bf)
e(chi2)
e(k_eq)
e(k_eq_model)
e(k_aux)
e(df_m)
e(11)
e(11_0)
e(N_clust)
e(sigma)
\(e(p)\)
e(rank)
e(ic)
e(rc)
e(converged)
Macros
e(cmd)
e(cmdline)
e(llopt)
e(ulopt)
e(depvar)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(offset1)
e(chi2type)
e(vce)
e(vcetype)
e(opt)
\(e\) (which)
e(ml_method)
e(user)
e(technique)
\(e\) (properties)
e(predict)
e(marginsok)
e(asbalanced)
e(asobserved)
Matrices
e(b)
e(Cns)
e(ilog)
e(gradient)
e(V)
e(V_modelbased)
\(e\) (means)
e(dummy)
Functions
e(sample) marks estimation sample

In addition to the above, the following is stored in r() :
```

Matrices

```
    \(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values,
    and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

\section*{Methods and formulas}

Greene (2018, 918-924) and Davidson and MacKinnon (1993, 534-537) provide introductions to the truncated regression model.

Let \(\mathbf{y}=\mathbf{X} \boldsymbol{\beta}+\boldsymbol{\epsilon}\) be the model. \(\mathbf{y}\) represents continuous outcomes either observed or not observed. Our model assumes that \(\boldsymbol{\epsilon} \sim N\left(\mathbf{0}, \sigma^{2} \mathbf{I}\right)\).

Let \(a\) be the lower limit and \(b\) be the upper limit. The log likelihood is
\[
\ln L=-\frac{n}{2} \log \left(2 \pi \sigma^{2}\right)-\frac{1}{2 \sigma^{2}} \sum_{j=1}^{n}\left(y_{j}-\mathbf{x}_{j} \boldsymbol{\beta}\right)^{2}-\sum_{j=1}^{n} \log \left\{\Phi\left(\frac{b-\mathbf{x}_{j} \boldsymbol{\beta}}{\sigma}\right)-\Phi\left(\frac{a-\mathbf{x}_{j} \boldsymbol{\beta}}{\sigma}\right)\right\}
\]

This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce(robust) and vce(cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.
truncreg also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

\section*{References}

Berndt, E. R. 1996. The Practice of Econometrics: Classic and Contemporary. New York: Addison-Wesley.
Canette, I. 2016. Understanding truncation and censoring. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2016/12/13/understanding-truncation-and-censoring/.
Davidson, R., and J. G. MacKinnon. 1993. Estimation and Inference in Econometrics. New York: Oxford University Press.

Gray, L. A., and M. Hernández-Alava. 2018. A command for fitting mixture regression models for bounded dependent variables using the beta distribution. Stata Journal 18: 51-75.
Greene, W. H. 2018. Econometric Analysis. 8th ed. New York: Pearson.

\footnotetext{
Also see
[R] truncreg postestimation - Postestimation tools for truncreg
\([\mathrm{R}]\) regress - Linear regression
[R] tobit - Tobit regression
[BAYES] bayes: truncreg - Bayesian truncated regression
[FMM] fmm: truncreg - Finite mixtures of truncated linear regression models
[MI] Estimation - Estimation commands for use with mi estimate
[SVY] svy estimation - Estimation commands for survey data
[U] 20 Estimation and postestimation commands
}

\section*{Title}
truncreg postestimation - Postestimation tools for truncreg

Postestimation commands predict margins Also see

\section*{Postestimation commands}

The following postestimation commands are available after truncreg:
\begin{tabular}{|c|c|}
\hline Command & Description \\
\hline contrast & contrasts and ANOVA-style joint tests of estimates \\
\hline estat ic & Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) \\
\hline estat summarize & summary statistics for the estimation sample \\
\hline estat vce & variance-covariance matrix of the estimators (VCE) \\
\hline estat (svy) & postestimation statistics for survey data \\
\hline estimates & cataloging estimation results \\
\hline etable & table of estimation results \\
\hline * forecast & dynamic forecasts and simulations \\
\hline * hausman & Hausman's specification test \\
\hline lincom & point estimates, standard errors, testing, and inference for linear combinations of coefficients \\
\hline * 1 rtest & likelihood-ratio test \\
\hline margins & marginal means, predictive margins, marginal effects, and average marginal effects \\
\hline marginsplot & graph the results from margins (profile plots, interaction plots, etc.) \\
\hline nlcom & point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients \\
\hline predict & linear, censored, and truncated predictions \\
\hline predictnl & point estimates, standard errors, testing, and inference for generalized predictions \\
\hline pwcompare & pairwise comparisons of estimates \\
\hline suest & seemingly unrelated estimation \\
\hline test & Wald tests of simple and composite linear hypotheses \\
\hline testnl & Wald tests of nonlinear hypotheses \\
\hline
\end{tabular}

\footnotetext{
*forecast, hausman, and lrtest are not appropriate with svy estimation results. forecast is also not appropriate with mi estimation results.
}

\section*{predict}

\section*{Description for predict}
predict creates a new variable containing predictions such as linear predictions, standard errors, probabilities, and expected values.

\section*{Menu for predict}

Statistics \(>\) Postestimation

\section*{Syntax for predict}
```

predict [type] newvar [if] [in] [, statistic nooffset]
predict [type] stub* [if] [in], scores

```
statistic Description

\section*{Main}
\(\mathrm{xb} \quad\) linear prediction; the default
stdp standard error of the prediction
stdf standard error of the forecast
\(\operatorname{pr}(a, b) \quad \operatorname{Pr}\left(a<y_{j}<b\right)\)
\(\mathrm{e}(a, b) \quad E\left(y_{j} \mid a<y_{j}<b\right)\)
\(\underline{y s t a r}(a, b) \quad E\left(y_{j}^{*}\right), y_{j}^{*}=\max \left\{a, \min \left(y_{j}, b\right)\right\}\)

These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.
stdf is not allowed with svy estimation results.
where \(a\) and \(b\) may be numbers or variables; \(a\) missing ( \(a \geq\).) means \(-\infty\), and \(b\) missing ( \(b \geq\).) means \(+\infty\); see [U] 12.2.1 Missing values.

\section*{Options for predict}
\(\qquad\) Main
xb , the default, calculates the linear prediction.
stdp calculates the standard error of the prediction, which can be thought of as the standard error of the predicted expected value or mean for the observation's covariate pattern. The standard error of the prediction is also referred to as the standard error of the fitted value.
stdf calculates the standard error of the forecast, which is the standard error of the point prediction for 1 observation. It is commonly referred to as the standard error of the future or forecast value. By construction, the standard errors produced by stdf are always larger than those produced by stdp; see Methods and formulas in [R] regress postestimation.
\(\operatorname{pr}(a, b)\) calculates \(\operatorname{Pr}\left(a<\mathbf{x}_{j} \mathbf{b}+u_{j}<b\right)\), the probability that \(y_{j} \mid \mathbf{x}_{j}\) would be observed in the interval \((a, b)\).
\(a\) and \(b\) may be specified as numbers or variable names; \(l b\) and \(u b\) are variable names;
\(\operatorname{pr}(20,30)\) calculates \(\operatorname{Pr}\left(20<\mathbf{x}_{j} \mathbf{b}+u_{j}<30\right)\);
\(\operatorname{pr}(l b, u b)\) calculates \(\operatorname{Pr}\left(l b<\mathbf{x}_{j} \mathbf{b}+u_{j}<u b\right)\); and
\(\operatorname{pr}(20, u b)\) calculates \(\operatorname{Pr}\left(20<\mathbf{x}_{j} \mathbf{b}+u_{j}<u b\right)\).
\(a\) missing ( \(a \geq\). ) means \(-\infty\); \(\operatorname{pr}(., 30)\) calculates \(\operatorname{Pr}\left(-\infty<\mathbf{x}_{j} \mathbf{b}+u_{j}<30\right)\); \(\operatorname{pr}(l b, 30)\) calculates \(\operatorname{Pr}\left(-\infty<\mathbf{x}_{j} \mathbf{b}+u_{j}<30\right)\) in observations for which \(l b \geq\). and calculates \(\operatorname{Pr}\left(l b<\mathbf{x}_{j} \mathbf{b}+u_{j}<30\right)\) elsewhere.
\(b\) missing ( \(b \geq\).) means \(+\infty\); \(\operatorname{pr}(20,\).\() calculates \operatorname{Pr}\left(+\infty>\mathbf{x}_{j} \mathbf{b}+u_{j}>20\right)\); \(\operatorname{pr}(20, u b)\) calculates \(\operatorname{Pr}\left(+\infty>\mathbf{x}_{j} \mathbf{b}+u_{j}>20\right)\) in observations for which \(u b \geq\). and calculates \(\operatorname{Pr}\left(20<\mathbf{x}_{j} \mathbf{b}+u_{j}<u b\right)\) elsewhere.
\(\mathrm{e}(a, b)\) calculates \(E\left(\mathbf{x}_{j} \mathbf{b}+u_{j} \mid a<\mathbf{x}_{j} \mathbf{b}+u_{j}<b\right)\), the expected value of \(y_{j} \mid \mathbf{x}_{j}\) conditional on \(y_{j} \mid \mathbf{x}_{j}\) being in the interval \((a, b)\), meaning that \(y_{j} \mid \mathbf{x}_{j}\) is truncated.
\(a\) and \(b\) are specified as they are for \(\operatorname{pr}()\).
ystar \((a, b)\) calculates \(E\left(y_{j}^{*}\right)\), where \(y_{j}^{*}=a\) if \(\mathbf{x}_{j} \mathbf{b}+u_{j} \leq a\), \(y_{j}^{*}=b\) if \(\mathbf{x}_{j} \mathbf{b}+u_{j} \geq b\), and \(y_{j}^{*}=\mathbf{x}_{j} \mathbf{b}+u_{j}\) otherwise, meaning that \(y_{j}^{*}\) is censored. \(a\) and \(b\) are specified as they are for pr() .
nooffset is relevant only if you specified offset (varname). It modifies the calculations made by predict so that they ignore the offset variable; the linear prediction is treated as \(\mathbf{x}_{j} \mathbf{b}\) rather than as \(\mathbf{x}_{j} \mathbf{b}+\) offset \(_{j}\).
scores calculates equation-level score variables.
The first new variable will contain \(\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)\).
The second new variable will contain \(\partial \ln L / \partial \sigma\).

\section*{margins}

\section*{Description for margins}
margins estimates margins of response for linear predictions, probabilities, and expected values.

\section*{Menu for margins}

\author{
Statistics \(>\) Postestimation
}

\section*{Syntax for margins}
```

margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...)...] [options]

```
statistic Description
xb
linear prediction; the default
\(\operatorname{pr}(a, b) \quad \operatorname{Pr}\left(a<y_{j}<b\right)\)
\(\mathrm{e}(a, b)\)
\(E\left(y_{j} \mid a<y_{j}<b\right)\)
ystar (a, \(b\) )
stdp
stdf
\(E\left(y_{j}^{*}\right), y_{j}^{*}=\max \left\{a, \min \left(y_{j}, b\right)\right\}\)
not allowed with margins
not allowed with margins

Statistics not allowed with margins are functions of stochastic quantities other than e(b). For the full syntax, see [R] margins.

\section*{Also see}
\([\mathrm{R}]\) truncreg - Truncated regression
[U] 20 Estimation and postestimation commands

\section*{Title}
ttest \(-t\) tests (mean-comparison tests)
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
ttest performs \(t\) tests on the equality of means. The test can be performed for one sample against a hypothesized population mean. Two-sample tests can be conducted for paired and unpaired data. The assumption of equal variances can be optionally relaxed in the unpaired two-sample case.
ttesti is the immediate form of ttest; see [U] 19 Immediate commands.

\section*{Quick start}

Test that the mean of v 1 is equal between two groups defined by catvar
ttest v1, by(catvar)
Same as above, but assume unequal variances
ttest v1, by (catvar) unequal
Paired \(t\) test of v2 and v3
ttest v2 == v3
Same as above, but with unpaired data and conduct test separately for each level of catvar by catvar: ttest v2 == v3, unpaired

Test that the mean of v 4 is 3 at the \(90 \%\) confidence level
ttest v4 == 3, level (90)
Test \(\mu_{1}=\mu_{2}\) if \(\bar{x}_{1}=3.2, s d_{1}=0.1, \bar{x}_{2}=3.4\), and \(s d_{2}=0.15\) with \(n_{1}=n_{2}=12\)
ttesti 123.2 . 1123.4 . 15

\section*{Menu}

\section*{ttest}

Statistics \(>\) Summaries, tables, and tests \(>\) Classical tests of hypotheses \(>\mathrm{t}\) test (mean-comparison test)

\section*{ttesti}

Statistics \(>\) Summaries, tables, and tests \(>\) Classical tests of hypotheses \(>\mathrm{t}\) test calculator

\section*{Syntax}

One-sample \(t\) test
```

ttest varname == \# [if][in][, level(\#)]

```

Two-sample \(t\) test using groups
ttest varname [if][in], by (groupvar) [options \({ }_{1}\) ]

Two-sample \(t\) test using variables
```

ttest varname}\mp@subsup{1}{1}{==}\mp@subsup{\mathrm{ varname }}{2}{}[\mathrm{ [if] [in], unpaired [unequal welch level(\#)]

```

Paired \(t\) test
```

ttest varname

```

Immediate form of one-sample \(t\) test
```

ttesti \# obs \#mean \# \#d \# val [, level(\#)]

```

Immediate form of two-sample \(t\) test
```

ttesti \# \#obs1 \#mean1 \# \#s1 \# obs2 \#mean2 \#sd2 [, options m

```
options \(_{1}\)
Description
Main
* by (groupvar)
variable defining the groups
reverse
unequal unpaired data have unequal variances
welch use Welch's approximation
level(\#) set confidence level; default is level (95)
*by (groupvar) is required.
options \(_{2}\)
Description
Main
\begin{tabular}{ll} 
unequal & unpaired data have unequal variances \\
welch & use Welch's approximation \\
level(\#) & set confidence level; default is level(95)
\end{tabular}
by and collect are allowed with ttest and ttesti; see [U] 11.1.10 Prefix commands.

\section*{Options}

\section*{Main}
by (groupvar) specifies the groupvar that defines the two groups that ttest will use to test the hypothesis that their means are equal. Specifying by (groupvar) implies an unpaired (two sample) \(t\) test. Do not confuse the by () option with the by prefix; you can specify both.
reverse reverses the order of the mean difference between groups defined in by (). By default, the mean of the group corresponding to the largest value in the variable in by () is subtracted from the mean of the group with the smallest value in by (). reverse reverses this behavior and the order in which variables appear on the table.
unpaired specifies that the data be treated as unpaired. The unpaired option is used when the two sets of values to be compared are in different variables.
unequal specifies that the unpaired data not be assumed to have equal variances.
welch specifies that the approximate degrees of freedom for the test be obtained from Welch's formula (1947) rather than from Satterthwaite's approximation formula (1946), which is the default when unequal is specified. Specifying welch implies unequal.
level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals.

\section*{Remarks and examples}

Remarks are presented under the following headings:

\author{
One-sample \(t\) test \\ Two-sample \(t\) test \\ Paired t test \\ Two-sample \(t\) test compared with one-way ANOVA \\ Immediate form \\ Video examples
}

\section*{One-sample t test}
\(>\) Example 1
In the first form, ttest tests whether the mean of the sample is equal to a known constant under the assumption of unknown variance. Assume that we have a sample of 74 automobiles. We know each automobile's average mileage rating and wish to test whether the overall average for the sample is 20 miles per gallon.
```

. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. ttest mpg==20
One-sample t test

| Variable | Obs | Mean | Std. err. | Std. dev. | [95\% conf. interval] |  |
| ---: | ---: | ---: | ---: | ---: | ---: | ---: |
| mpg | 74 | 21.2973 | .6725511 | 5.785503 | 19.9569 | 22.63769 |


| mean $=$ mean (mpg) |  | $\mathrm{t}=1.9289$ |
| :---: | :---: | :---: |
| H0: mean $=20$ |  | Degrees of freedom $=\quad 73$ |
| Ha: mean < 20 | Ha: mean ! $=20$ | Ha: mean > 20 |
| $\operatorname{Pr}(\mathrm{T}<\mathrm{t})=0.9712$ | $\operatorname{Pr}(\|T\|>\|t\|)=0.0576$ | $6 \quad \operatorname{Pr}(\mathrm{~T}>\mathrm{t})=0.0288$ |

```

The test indicates that the underlying mean is not 20 with a significance level of \(5.8 \%\).

\section*{Two-sample t test}

\section*{> Example 2: Two-sample \(t\) test using groups}

We are testing the effectiveness of a new fuel additive. We run an experiment in which 12 cars are given the fuel treatment and 12 cars are not. The results of the experiment are as follows:
\begin{tabular}{cl} 
treated & mpg \\
\hline 0 & 20 \\
0 & 23 \\
0 & 21 \\
0 & 25 \\
0 & 18 \\
0 & 17 \\
0 & 18 \\
0 & 24 \\
0 & 20 \\
0 & 24 \\
0 & 23 \\
0 & 19 \\
1 & 24 \\
1 & 25 \\
1 & 21 \\
1 & 22 \\
1 & 23 \\
1 & 18 \\
1 & 17 \\
1 & 28 \\
1 & 24 \\
1 & 27 \\
1 & 21 \\
1 & 23
\end{tabular}

The treated variable is coded as 1 if the car received the fuel treatment and 0 otherwise.

We can test the equality of means of the treated and untreated group by typing
. use https://www.stata-press.com/data/r18/fuel3
. ttest mpg, by(treated)
Two-sample t test with equal variances
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline Group & Obs & Mean & Std. err. & Std. dev. & [95\% conf & interval] \\
\hline 0 & 12 & 21 & . 7881701 & 2.730301 & 19.26525 & 22.73475 \\
\hline 1 & 12 & 22.75 & . 9384465 & 3.250874 & 20.68449 & 24.81551 \\
\hline Combined & 24 & 21.875 & . 6264476 & 3.068954 & 20.57909 & 23.17091 \\
\hline diff & & -1.75 & 1.225518 & & -4. 291568 & . 7915684 \\
\hline \multicolumn{3}{|l|}{diff \(=\) mean(0) - mean(1)} & & \multicolumn{2}{|l|}{\multirow[b]{2}{*}{Degrees of freedom}} & -1.4280 \\
\hline \multicolumn{3}{|l|}{HO: diff = 0} & & & & 22 \\
\hline Ha: & & \multicolumn{3}{|c|}{Ha: diff != 0} & \multicolumn{2}{|l|}{\multirow[t]{2}{*}{Ha: diff \(>0\)
\(\operatorname{Pr}(\mathrm{~T}>\mathrm{t})=0.9163\)}} \\
\hline \(\operatorname{Pr}(\mathrm{T}<\mathrm{t})\) & 0837 & \multicolumn{3}{|r|}{\(\operatorname{Pr}(|T|>|t|)=0.1673\)} & & \\
\hline
\end{tabular}

We do not find a statistically significant difference in the means.
If we were not willing to assume that the variances were equal and wanted to use Welch's formula, we could type
```

. ttest mpg, by(treated) welch

```

Two-sample t test with unequal variances
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline Group & Obs & Mean & Std. err. & Std. dev. & [95\% con & interval] \\
\hline 0 & 12 & 21 & . 7881701 & 2.730301 & 19.26525 & 22.73475 \\
\hline 1 & 12 & 22.75 & . 9384465 & 3.250874 & 20.68449 & 24.81551 \\
\hline Combined & 24 & 21.875 & . 6264476 & 3.068954 & 20.57909 & 23.17091 \\
\hline diff & & -1.75 & 1.225518 & & -4.28369 & . 7836902 \\
\hline \multicolumn{3}{|l|}{\multirow[t]{2}{*}{```
    diff = mean(0) - mean(1)
HO: diff = O
```}} & & & & -1.4280 \\
\hline & & & \multicolumn{4}{|r|}{Welch's degrees of freedom \(=23.2465\)} \\
\hline \multicolumn{2}{|l|}{\multirow[t]{2}{*}{\[
\begin{gathered}
\text { Ha: diff }<0 \\
\operatorname{Pr}(T<t)=0.0833
\end{gathered}
\]}} & \multicolumn{3}{|c|}{Ha: diff ! = 0} & \multicolumn{2}{|r|}{Ha: diff > 0} \\
\hline & & \multicolumn{3}{|r|}{\(\operatorname{Pr}(|T|>|t|)=0.1666\)} & \multicolumn{2}{|l|}{\(\operatorname{Pr}(\mathrm{T}>\mathrm{t})=0.9167\)} \\
\hline
\end{tabular}

\section*{Technical note}

In two-sample randomized designs, subjects will sometimes refuse the assigned treatment but still be measured for an outcome. In this case, take care to specify the group properly. You might be tempted to let varname contain missing where the subject refused and thus let ttest drop such observations from the analysis. Zelen (1979) argues that it would be better to specify that the subject belongs to the group in which he or she was randomized, even though such inclusion will dilute the measured effect.
> Example 3: Two-sample \(t\) test using variables
There is a second, inferior way to organize the data in the preceding example. We ran a test on 24 cars, 12 without the additive and 12 with. We now create two new variables, mpg1 and mpg2.
\begin{tabular}{cl} 
mpg1 & mpg2 \\
\hline 20 & 24 \\
23 & 25 \\
21 & 21 \\
25 & 22 \\
18 & 23 \\
17 & 18 \\
18 & 17 \\
24 & 28 \\
20 & 24 \\
24 & 27 \\
23 & 21 \\
19 & 23
\end{tabular}

This method is inferior because it suggests a connection that is not there. There is no link between the car with 20 mpg and the car with 24 mpg in the first row of the data. Each column of data could be arranged in any order. Nevertheless, if our data are organized like this, ttest can accommodate us.
. use https://www.stata-press.com/data/r18/fuel
. ttest mpg1==mpg2, unpaired
Two-sample t test with equal variances
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline Variable & Obs & Mean & Std. err. & Std. dev. & [95\% conf & interval] \\
\hline mpg1 & 12 & 21 & . 7881701 & 2.730301 & 19.26525 & 22.73475 \\
\hline mpg2 & 12 & 22.75 & . 9384465 & 3.250874 & 20.68449 & 24.81551 \\
\hline Combined & 24 & 21.875 & . 6264476 & 3.068954 & 20.57909 & 23.17091 \\
\hline diff & & -1.75 & 1.225518 & & -4.291568 & . 7915684 \\
\hline \multicolumn{4}{|l|}{diff \(=\) mean (mpg1) - mean (mpg2)} & \multicolumn{3}{|r|}{1.4280} \\
\hline \multicolumn{2}{|l|}{HO: diff = 0} & & & Degree & of freedom & 22 \\
\hline Ha: & & \multicolumn{3}{|c|}{Ha: diff ! \(=0\)} & \multicolumn{2}{|r|}{Ha: diff > 0} \\
\hline \(\operatorname{Pr}(\mathrm{T}<\mathrm{t})\) & 0837 & \multicolumn{3}{|r|}{\(\operatorname{Pr}(|T|>|t|)=0.1673\)} & \multicolumn{2}{|l|}{\(\operatorname{Pr}(\mathrm{T}>\mathrm{t})=0.9163\)} \\
\hline
\end{tabular}

\section*{Paired t test}

\section*{> Example 4}

Suppose that the preceding data were actually collected by running a test on 12 cars. Each car was run once with the fuel additive and once without. Our data are stored in the same manner as in example 3, but this time, there is most certainly a connection between the mpg values that appear in the same row. These come from the same car. The variables mpg1 and mpg2 represent mileage without and with the treatment, respectively.
. use https://www.stata-press.com/data/r18/fuel
. ttest mpg1==mpg2
Paired t test
\begin{tabular}{r|rrrrrr}
\hline Variable & Obs & Mean & Std. err. & Std. dev. & [95\% conf. interval] \\
\hline mpg1 & 12 & 21 & .7881701 & 2.730301 & 19.26525 & 22.73475 \\
mpg2 & 12 & 22.75 & .9384465 & 3.250874 & 20.68449 & 24.81551 \\
\hline diff & 12 & -1.75 & .7797144 & 2.70101 & -3.46614 & -.0338602 \\
\hline
\end{tabular}


We find that the means are statistically different from each other at any level greater than \(4.6 \%\).

\section*{Two-sample t test compared with one-way ANOVA}

\section*{Example 5}

In example 2, we saw that ttest can be used to test the equality of a pair of means; see \([R]\) oneway for an extension that allows testing the equality of more than two means.

Suppose that we have data on the 50 states. The dataset contains the median age of the population (medage) and the region of the country (region) for each state. Region 1 refers to the Northeast, region 2 to the North Central, region 3 to the South, and region 4 to the West. Using oneway, we can test the equality of all four means.
```

. use https://www.stata-press.com/data/r18/census
(1980 Census data by state)
. oneway medage region

```

Analysis of variance
\begin{tabular}{cccccr} 
Source & SS & df & MS & F & Prob \(>\mathrm{F}\) \\
\hline Between groups & 46.3961903 & 3 & 15.4653968 & 7.56 & 0.0003 \\
Within groups & 94.1237947 & 46 & 2.04616945 & & \\
\hline Total & 140.519985 & 49 & 2.8677548 & & \\
Bartlett's equal-variances test: & chi2 (3) \(=10.5757\) & Prob>chi2 \(=0.014\)
\end{tabular}

We find that the means are different, but we are interested only in testing whether the means for the Northeast (region==1) and West (region==4) are different. We could use oneway:
\begin{tabular}{|c|c|c|c|c|c|}
\hline Source & Analysi
SS & df & ance
MS & F & Prob > F \\
\hline Between groups & 46.241247 & 1 & 46.241247 & 20.02 & 0.0002 \\
\hline Within groups & 46.1969169 & 20 & 2.30984584 & & \\
\hline Total & 92.4381638 & 21 & 4.40181733 & & \\
\hline Bartlett's equa & riances test & i2( & \(=2.4679\) & Prob> & i2 \(=0.116\) \\
\hline
\end{tabular}

We could also use ttest:
. ttest medage if region==1 | region==4, by(region)
Two-sample t test with equal variances
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline Group & Obs & Mean & Std. err. & Std. dev & [95\% con & interval] \\
\hline NE & 9 & 31.23333 & . 3411581 & 1.023474 & 30.44662 & 32.02005 \\
\hline West & 13 & 28.28462 & . 4923577 & 1.775221 & 27.21186 & 29.35737 \\
\hline Combined & 22 & 29.49091 & . 4473059 & 2.098051 & 28.56069 & 30.42113 \\
\hline diff & & 2.948718 & . 6590372 & & 1.57399 & 4.323445 \\
\hline \multicolumn{3}{|l|}{diff \(=\) mean(NE) - mean(West)} & & \multicolumn{2}{|l|}{\multirow[b]{2}{*}{Degrees of freedom}} & 4.4743 \\
\hline \multicolumn{2}{|l|}{HO: diff = 0} & & & & & 20 \\
\hline Ha: & & \multicolumn{3}{|c|}{Ha: diff != 0} & \multicolumn{2}{|r|}{Ha: diff > 0} \\
\hline \(\operatorname{Pr}(\mathrm{T}<\mathrm{t})\) & 9999 & \multicolumn{3}{|r|}{\(\operatorname{Pr}(|T|>|t|)=0.0002\)} & \multicolumn{2}{|l|}{\(\operatorname{Pr}(\mathrm{T}>\mathrm{t})=0.0001\)} \\
\hline
\end{tabular}

The significance levels of both tests are the same.

\section*{Immediate form}

\section*{> Example 6}
ttesti is like ttest, except that we specify summary statistics rather than variables as arguments. For instance, we are reading an article that reports the mean number of sunspots per month as 62.6 with a standard deviation of 15.8 . There are 24 months of data. We wish to test whether the mean is 75 :
. ttesti 2462.615 .875
One-sample t test


\section*{> Example 7}

There is no immediate form of ttest with paired data because the test is also a function of the covariance, a number unlikely to be reported in any published source. For unpaired data, however, we might type
. ttesti 2020532154
Two-sample t test with equal variances


If we had typed ttesti 2020532154 , unequal, the test would have assumed unequal variances.

\section*{Video examples}

One-sample \(t\) test in Stata
\(t\) test for two independent samples in Stata
\(t\) test for two paired samples in Stata
One-sample \(t\)-test calculator
Two-sample \(t\)-test calculator

\section*{Stored results}
ttest and ttesti store the following in \(r()\) :
Scalars
\begin{tabular}{|c|c|c|c|}
\hline \(r\left(N \_1\right)\) & sample size \(n_{1}\) & \(r\) (sd_1) & standard deviation for first variable \\
\hline r ( N _2) & sample size \(n_{2}\) & \(r\) (sd_2) & standard deviation for second variable \\
\hline \(r\) (p_l) & lower one-sided \(p\)-value & r (sd) & combined standard deviation \\
\hline \(r\) (p_u) & upper one-sided \(p\)-value & \(r\) (mu_1) & \(\bar{x}_{1}\) mean for population 1 \\
\hline r (p) & two-sided \(p\)-value & \(r\) (mu_2) & \(\bar{x}_{2}\) mean for population 2 \\
\hline r (se) & estimate of standard error & \(\mathrm{r}(\mathrm{df}\) _t) & degrees of freedom \\
\hline \(r\) ( t ) & \(t\) statistic & r (level) & confidence level \\
\hline
\end{tabular}

\section*{Methods and formulas}

See, for instance, Hoel (1984, 140-161) or Dixon and Massey (1983, 121-130) for an introduction and explanation of the calculation of these tests. Acock (2023, 165-179) and Hamilton (2013, 145-150) describe \(t\) tests using applications in Stata.

The test for \(\mu=\mu_{0}\) for unknown \(\sigma\) is given by
\[
t=\frac{\left(\bar{x}-\mu_{0}\right) \sqrt{n}}{s}
\]

The statistic is distributed as Student's \(t\) with \(n-1\) degrees of freedom (Gosset [Student, pseud.] 1908).
The test for \(\mu_{x}=\mu_{y}\) when \(\sigma_{x}\) and \(\sigma_{y}\) are unknown but \(\sigma_{x}=\sigma_{y}\) is given by
\[
t=\frac{\bar{x}-\bar{y}}{\left\{\frac{\left(n_{x}-1\right) s_{x}^{2}+\left(n_{y}-1\right) s_{y}^{2}}{n_{x}+n_{y}-2}\right\}^{1 / 2}\left(\frac{1}{n_{x}}+\frac{1}{n_{y}}\right)^{1 / 2}}
\]

The result is distributed as Student's \(t\) with \(n_{x}+n_{y}-2\) degrees of freedom.
You could perform ttest (without the unequal option) in a regression setting given that regression assumes a homoskedastic error model. To compare with the ttest command, denote the underlying observations on \(x\) and \(y\) by \(x_{j}, j=1, \ldots, n_{x}\), and \(y_{j}, j=1, \ldots, n_{y}\). In a regression framework, typing ttest without the unequal option is equivalent to
1. creating a new variable \(z_{j}\) that represents the stacked observations on \(x\) and \(y\) (so that \(z_{j}=x_{j}\) for \(j=1, \ldots, n_{x}\) and \(z_{n_{x}+j}=y_{j}\) for \(j=1, \ldots, n_{y}\) )
2. and then estimating the equation \(z_{j}=\beta_{0}+\beta_{1} d_{j}+\epsilon_{j}\), where \(d_{j}=0\) for \(j=1, \ldots, n_{x}\) and \(d_{j}=1\) for \(j=n_{x}+1, \ldots, n_{x}+n_{y}\) (that is, \(d_{j}=0\) when the \(z\) observations represent \(x\), and \(d_{j}=1\) when the \(z\) observations represent \(y\) ).
The estimated value of \(\beta_{1}, b_{1}\), will equal \(\bar{y}-\bar{x}\), and the reported \(t\) statistic will be the same \(t\) statistic as given by the formula above.

The test for \(\mu_{x}=\mu_{y}\) when \(\sigma_{x}\) and \(\sigma_{y}\) are unknown and \(\sigma_{x} \neq \sigma_{y}\) is given by
\[
t=\frac{\bar{x}-\bar{y}}{\left(s_{x}^{2} / n_{x}+s_{y}^{2} / n_{y}\right)^{1 / 2}}
\]

The result is distributed as Student's \(t\) with \(\nu\) degrees of freedom, where \(\nu\) is given by (with Satterthwaite's [1946] formula)
\[
\frac{\left(s_{x}^{2} / n_{x}+s_{y}^{2} / n_{y}\right)^{2}}{\frac{\left(s_{x}^{2} / n_{x}\right)^{2}}{n_{x}-1}+\frac{\left(s_{y}^{2} / n_{y}\right)^{2}}{n_{y}-1}}
\]

With Welch's formula (1947), the number of degrees of freedom is given by
\[
-2+\frac{\left(s_{x}^{2} / n_{x}+s_{y}^{2} / n_{y}\right)^{2}}{\frac{\left(s_{x}^{2} / n_{x}\right)^{2}}{n_{x}+1}+\frac{\left(s_{y}^{2} / n_{y}\right)^{2}}{n_{y}+1}}
\]

The test for \(\mu_{x}=\mu_{y}\) for matched observations (also known as paired observations, correlated pairs, or permanent components) is given by
\[
t=\frac{\bar{d} \sqrt{n}}{s_{d}}
\]
where \(\bar{d}\) represents the mean of \(x_{i}-y_{i}\) and \(s_{d}\) represents the standard deviation. The test statistic \(t\) is distributed as Student's \(t\) with \(n-1\) degrees of freedom.

You can also use ttest without the unpaired option in a regression setting because a paired comparison includes the assumption of constant variance. The ttest with an unequal variance assumption does not lend itself to an easy representation in regression settings and is not discussed here. \(\left(x_{j}-y_{j}\right)=\beta_{0}+\epsilon_{j}\).

William Sealy Gosset (1876-1937) was born in Canterbury, England. He studied chemistry and mathematics at Oxford and worked as a chemist with the brewers Guinness in Dublin. Gosset became interested in statistical problems, which he discussed with Karl Pearson and later with Fisher and Neyman. He published several important papers under the pseudonym "Student", and he lent that name to the \(t\) test he invented.

Stella Cunliffe (1917-2012) was an advocate for increased understanding of the role of human nature in experiments and methodological rigor in social statistics. She was born in Battersea, England. She was the first person from her local public girls' school to attend college, obtaining a bachelor of science from the London School of Economics. Her first job was with the Danish Bacon Company during World War II, where she was in charge of bacon rations for London. After the war, she moved to Germany and again helped to ration food, this time for refugees.
She then spent a long career in quality control at the Guinness Brewing Company. Cunliffe observed that the weights of rejected casks skewed lighter. Noting that workers had to roll casks that were too light or too heavy uphill to be remade, she had the scales moved to the top of the hill. With workers able to roll rejected casks downhill, the weight of these casks began to follow a normal distribution.

After 25 years at Guinness, Cunliffe joined the British Home Office, where she would go on to become the first woman to serve as director of statistics. During her tenure at the Home Office, she emphasized applying principles of experimental design she had learned at Guinness to the study of such topics as birthrates, recidivism, and criminology. In 1975, she became the first woman to serve as president of the Royal Statistical Society.

\section*{References}

Acock, A. C. 2023. A Gentle Introduction to Stata. Rev. 6th ed. College Station, TX: Stata Press.
Boland, P. J. 2000. William Sealy Gosset—alias 'Student' 1876-1937. In Creators of Mathematics: The Irish Connection, ed. K. Houston, 105-112. Dublin: University College Dublin Press.

Dixon, W. J., and F. J. Massey, Jr. 1983. Introduction to Statistical Analysis. 4th ed. New York: McGraw-Hill.
Earnest, A. 2017. Essentials of a Successful Biostatistical Collaboration. Boca Raton, FL: CRC Press.
Gosset, W. S. 1943. "Student's" Collected Papers. London: Biometrika Office, University College.

Gosset [Student, pseud.], W. S. 1908. The probable error of a mean. Biometrika 6: 1-25. https://doi.org/10.2307/2331554.

Hamilton, L. C. 2013. Statistics with Stata: Updated for Version 12. 8th ed. Boston: Brooks/Cole.
Hoel, P. G. 1984. Introduction to Mathematical Statistics. 5th ed. New York: Wiley.
Huber, C. 2013. Measures of effect size in Stata 13. The Stata Blog: Not Elsewhere Classified. https://blog.stata.com/2013/09/05/measures-of-effect-size-in-stata-13/.
Kaplan, D. M. 2019. distcomp: Comparing distributions. Stata Journal 19: 832-848.
Mehmetoglu, M., and T. G. Jakobsen. 2022. Applied Statistics Using Stata: A Guide for the Social Sciences. 2nd ed. Thousand Oaks, CA: Sage.

Pearson, E. S., R. L. Plackett, and G. A. Barnard. 1990. 'Student': A Statistical Biography of William Sealy Gosset. Oxford: Oxford University Press.
Preece, D. A. 1982. \(t\) is for trouble (and textbooks): A critique of some examples of the paired-samples \(t\)-test. Statistician 31: 169-195. https://doi.org/10.2307/2987888.

Satterthwaite, F. E. 1946. An approximate distribution of estimates of variance components. Biometrics Bulletin 2: 110-114. https://doi.org/10.2307/3002019.
Senn, S. J., and W. Richardson. 1994. The first t-test. Statistics in Medicine 13: 785-803. https://doi.org/10.1002/sim. 4780130802.

Welch, B. L. 1947. The generalization of 'student's' problem when several different population variances are involved. Biometrika 34: 28-35. https://doi.org/10.2307/2332510.
Zelen, M. 1979. A new design for randomized clinical trials. New England Journal of Medicine 300: 1242-1245. https://doi.org/10.1056/NEJM197905313002203.

\section*{Also see}
[R] bitest - Binomial probability test
\([R]\) ci - Confidence intervals for means, proportions, and variances
[R] esize - Effect size based on mean comparison
[R] mean - Estimate means
[R] oneway - One-way analysis of variance
[R] prtest - Tests of proportions
[R] sdtest - Variance-comparison tests
[R] ztest \(-z\) tests (mean-comparison tests, known variance)
[MV] hotelling - Hotelling's \(T^{2}\) generalized means test

\section*{Title}
update - Check for official updates
\begin{tabular}{llll} 
Description & Menu & Syntax & Options \\
Remarks and examples & Stored results & Also see &
\end{tabular}

\section*{Description}

The update command reports on the current update level and installs official updates. Official updates are updates to Stata as it was originally shipped from StataCorp, not the additions to Stata published in, for instance, the Stata Journal (SJ). Those additions are installed using the net command and updated using the ado update command; see \([R]\) net and \([R]\) ado update.
update without arguments reports on the update level of the currently installed Stata.
update from sets an update source, which may be a directory name or URL. If you are on the Internet, type update from https://www.stata.com.
update query compares the update level of the currently installed Stata with that available from the update source and displays a report.
update all updates all necessary files. This is what you should type to check for and install updates.
set update_query determines if update query is to be automatically performed when Stata is launched. Only Mac and Windows platforms can be set for automatic updating.
set update_interval \# sets the number of days to elapse before performing the next automatic update query. The default \# is 7 . The interval starts from the last time an update query was performed (automatically or manually). Only Mac and Windows platforms can be set for automatic updating.
set update_prompt determines whether a dialog is to be displayed before performing an automatic update query. The dialog allows you to perform an update query now, perform one the next time Stata is launched, perform one after the next interval has passed, or disable automatic update query. Only Mac and Windows platforms can be set for automatic updating.

\section*{Menu}

Help \(>\) Check for updates

\section*{Syntax}

Report on update level of currently installed Stata
update

Set update source
update from location

Compare update level of currently installed Stata with that of source
```

update query [, from(location)]

```

Perform update if necessary
```

update all [, from(location) detail force exit]

```

Set automatic updates (Mac and Windows only)
```

set update_query {on|off }
set update_interval \#
set update_prompt { on|off }

```

\section*{Options}
from(location) specifies the location of the update source. You can specify the from() option on the individual update commands or use the update from command. Which you do makes no difference. You typically do not need to use this option.
detail specifies to display verbose output during the update process.
force specifies to force downloading of all files even if, based on the date comparison, Stata does not think it is necessary. There is seldom a reason to specify this option.
exit instructs Stata to exit when the update has successfully completed. There is seldom a reason to specify this option.

\section*{Remarks and examples}
update updates the official components of Stata from the official source: https://www.stata.com. If you are connected to the Internet, the easy thing to do is to type
. update all
and follow the instructions. If Stata is up to date, update all will do nothing. Otherwise, it will download whatever is necessary and install the files. If you just want to know what updates are available, type
update query will check if any updates are available and report that information. If updates are available, it will recommend that you type update all.

If you want to report the current update level, type
. update
update will report the update level of the Stata installation. update will also show you the date that updates were last checked and if any updates were available at that time.

\section*{Stored results}
update without a subcommand, update from, and update query store the following in \(r()\) :
Scalars
```

r(inst_exe) date of executable installed (*)
r(avbl_exe) date of executable available over web (*) (**)
r(inst_ado) date of ado-files installed (*)
r(avbl_ado) date of ado-files available over web (*) (**)
r(inst_utilities) date of utilities installed (*)
r(avbl_utilities) date of utilities available over web (*) (**)
r(inst_docs) date of documentation installed (*)
r(avbl_docs) date of documentation available over web (*) (**)
r(name_exe) name of the Stata executable
r(dir_exe) directory in which executable is stored
r(dir_ado) directory in which ado-files are stored
r(dir_utilities) directory in which utilities are stored
r(dir_docs) directory in which PDF documentation is stored

```
Macros

Notes:
* Dates are stored as integers counting the number of days since January 1, 1960; see [D] Datetime.
** These dates are not stored by update without a subcommand because update by itself reports information solely about the local computer and does not check what is available on the web.

\section*{Also see \\ [ R\(]\) ado update - Update community-contributed packages \\ [R] net - Install and manage community-contributed additions from the Internet \\ [R] ssc - Install and uninstall packages from SSC \\ [P] sysdir - Query and set system directories \\ [U] 29 Using the Internet to keep up to date \\ [GSM] 19 Updating and extending Stata-Internet functionality \\ [GSU] 19 Updating and extending Stata-Internet functionality \\ [GSW] 19 Updating and extending Stata-Internet functionality}

\section*{Title}
\(\boldsymbol{v c e}\) _option - Variance estimators
\begin{tabular}{llll} 
Description & Syntax & Options & Remarks and examples \\
Methods and formulas & Also see & &
\end{tabular}

\section*{Description}

This entry describes the vce() option, which is common to most estimation commands. vce() specifies how to estimate the variance-covariance matrix (VCE) corresponding to the parameter estimates. The standard errors reported in the table of parameter estimates are the square root of the variances (diagonal elements) of the VCE.

\section*{Syntax}
estimation_cmd ... [, vce(vcetype) ...]
vcetype
Likelihood based
oim observed information matrix (OIM)
opg
Sandwich estimators
robust
cluster clustvar

Replication based
bootstrap [, bootstrap_options] bootstrap estimation
jackknife [, jackknife_options] jackknife estimation

\section*{Options}

\footnotetext{
SE/Robust
vce (oim) is usually the default for models fit using maximum likelihood. vce (oim) uses the observed information matrix (OIM); see [R] ml.
vce (opg) uses the sum of the outer product of the gradient (OPG) vectors; see [R] ml. This is the default VCE when the technique (bhhh) option is specified; see [R] Maximize.
vce (robust) uses the robust or sandwich estimator of variance. This estimator is robust to some types of misspecification so long as the observations are independent; see [U] 20.22 Obtaining robust variance estimates.
}

If the command allows pweights and you specify them, vce(robust) is implied; see [U] 20.24.3 Sampling weights.
vce (cluster clustvar) specifies that the standard errors allow for intragroup correlation, relaxing the usual requirement that the observations be independent. That is, the observations are independent across groups (clusters) but not necessarily within groups. clustvar specifies to which group each observation belongs, for example, vce (cluster personid) in data with repeated observations on individuals. vce (cluster clustvar) affects the standard errors and variance-covariance matrix of the estimators but not the estimated coefficients; see [U] 20.22 Obtaining robust variance estimates.
vce (bootstrap [, bootstrap_options]) uses a bootstrap; see [R] bootstrap. After estimation with vce(bootstrap), see [R] bootstrap postestimation to obtain percentile-based or bias-corrected confidence intervals.
vce(jackknife [, jackknife_options]) uses the delete-one jackknife; see [R] jackknife.

\section*{Remarks and examples}

Remarks are presented under the following headings:

\section*{Prefix commands}

Passing options in vce()

\section*{Prefix commands}

Specifying vce(bootstrap) or vce(jackknife) is often equivalent to using the corresponding prefix command. Here is an example using jackknife with regress.

\begin{tabular}{|c|c|c|c|c|c|c|}
\hline \multicolumn{7}{|l|}{Jackknife replications (74): > ..50.........60..........70.... done} \\
\hline Linear regres & ion & & & & \begin{tabular}{l}
Number of obs Replications \\
F (2, 73) \\
Prob > F \\
R-squared \\
Adj R-squared \\
Root MSE
\end{tabular} & \[
\begin{array}{rrr}
s & = & 74 \\
& = & 74 \\
& =66.26 \\
& =0.0000 \\
& =0.5521 \\
d & =0.5395 \\
& =3.9260
\end{array}
\] \\
\hline mpg & Coefficient & Jackknife std. err. & t & \(P>|t|\) & [95\% conf. & interval] \\
\hline turn & -. 7610113 & . 150726 & -5.05 & 0.000 & -1.061408 & -. 4606147 \\
\hline trunk & -. 3161825 & . 1282326 & -2.47 & 0.016 & -. 5717498 & -. 0606152 \\
\hline _cons & 55.82001 & 5.031107 & 11.09 & 0.000 & 45.79303 & 65.84699 \\
\hline
\end{tabular}

Here it does not matter whether we specify the vce (jackknife) option or instead use the jackknife prefix.

However, vce(jackknife) should be used in place of the jackknife prefix whenever available because they are not always equivalent. For example, to use the jackknife prefix with clogit properly, you must tell jackknife to omit whole groups rather than individual observations. Specifying vce(jackknife) does this automatically.
```

. use https://www.stata-press.com/data/r18/clogitid
. jackknife, cluster(id): clogit y x1 x2, group(id)
(output omitted)

```

This extra information is automatically communicated to jackknife by clogit when the vce() option is specified.


\section*{Passing options in vce()}

If you wish to specify more options to the bootstrap or jackknife estimation, you can include them within the vce() option. Below, we request 300 bootstrap replications and save the replications in bsreg.dta:
```

. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. regress mpg turn trunk, vce(bootstrap, nodots seed(123) rep(300) saving(bsreg))

```
Linear regression
Number of obs = 74
Replications \(=300\)
Wald chi2(2) \(=144.17\)
Prob > chi2 \(=0.0000\)
R-squared \(=0.5521\)
Adj R-squared \(=0.5395\)
Root MSE \(=3.9260\)
\begin{tabular}{r|rrrrrr}
\hline & \multicolumn{2}{c}{\begin{tabular}{c} 
Observed \\
mpg \\
coefficient
\end{tabular}} & \begin{tabular}{c} 
Bootstrap \\
std. err.
\end{tabular} & z & P>|z| & \multicolumn{2}{c}{\begin{tabular}{r} 
Normal-based \\
[95\% conf. interval]
\end{tabular}} \\
\hline turn & -.7610113 & .1497877 & -5.08 & 0.000 & -1.05459 & -.4674329 \\
trunk & -.3161825 & .1286802 & -2.46 & 0.014 & -.5683909 & -.063974 \\
_cons & 55.82001 & 4.9221 & 11.34 & 0.000 & 46.17287 & 65.46715 \\
\hline
\end{tabular}
. bstat using bsreg
Bootstrap results \(\quad\) Number of obs \(=74\)
Replications \(=300\)
Command: regress mpg turn trunk
\begin{tabular}{r|rrrrrr}
\hline & \begin{tabular}{c} 
Observed \\
coefficient
\end{tabular} & \begin{tabular}{c} 
Bootstrap \\
std. err.
\end{tabular} & \(\mathbf{z}\) & \(\mathrm{P}>|\mathrm{z}|\) & & \multicolumn{2}{c}{\begin{tabular}{c} 
Normal-based \\
[95\% conf. interval]
\end{tabular}} \\
\hline turn & -.7610113 & .1497877 & -5.08 & 0.000 & -1.05459 & -.4674329 \\
trunk & -.3161825 & .1286802 & -2.46 & 0.014 & -.5683909 & -.063974 \\
_cons & 55.82001 & 4.9221 & 11.34 & 0.000 & 46.17287 & 65.46715 \\
\hline
\end{tabular}

\section*{Methods and formulas}

By default, Stata's maximum likelihood estimators display standard errors based on variance estimates given by the inverse of the negative Hessian (second derivative) matrix. If vce (robust), vce (cluster clustvar), or pweights is specified, standard errors are based on the robust variance estimator (see [U] 20.22 Obtaining robust variance estimates); likelihood-ratio tests are not appropriate here (see [SVY] Survey), and the model \(\chi^{2}\) is from a Wald test. If vce \((\mathrm{opg})\) is specified, the standard errors are based on the outer product of the gradients; this option has no effect on likelihood-ratio tests, though it does affect Wald tests.

If vce(bootstrap) or vce(jackknife) is specified, the standard errors are based on the chosen replication method; here the model \(\chi^{2}\) or \(F\) statistic is from a Wald test using the respective replicationbased covariance matrix. The \(t\) distribution is used in the coefficient table when the vce (jackknife) option is specified. vce(bootstrap) and vce(jackknife) are also available with some commands that are not maximum likelihood estimators.

\section*{Also see}
\([\mathrm{R}]\) bootstrap - Bootstrap sampling and estimation
[R] jackknife - Jackknife estimation
[XT] vce_options - Variance estimators
[U] 20 Estimation and postestimation commands

\section*{Title}
view - View files and logs
\begin{tabular}{llll} 
Description & Menu & Syntax & Options \\
Remarks and examples & Also see & &
\end{tabular}

\section*{Description}
view displays file contents in the Viewer.
view file displays the specified file. file is optional, so if you had a SMCL session log created by typing log using mylog, you could view it by typing view mylog.smcl. view file can properly display .smcl files (logs and the like), .sthlp files, and text files. view file's asis option specifies that the file be displayed as plain text, regardless of the filename's extension.
view browse opens your browser pointed to url. Typing view browse https://www.stata.com would bring up your browser pointed to the website https://www.stata.com.
[view] help displays the specified topic in the Viewer. For example, to review the help for Stata's print command, you could type help print. See [R] help for more details.
[view] search displays the results of the search command in the Viewer. For instance, to search the system help for information on robust regression, you could type search robust regression. See [R] search for more details.
view net does the same as the net command—see [R] net—but displays the result in the Viewer. For instance, typing view net search hausman test would search the Internet for additions to Stata related to the Hausman test. Typing view net from https://www.stata.com would go to the Stata additions download site at https://www.stata.com.
view ado does the same as the ado command-see [R] net-but displays the result in the Viewer. For instance, typing view ado dir would show a list of files you have installed.
view update does the same as the update command-see [R] update-but displays the result in the Viewer. Typing view update would show the dates of what you have installed, and from there you could click to compare those dates with the latest updates available. Typing view update query would skip the first step and show the comparison.

\section*{Menu}

File \(>\) View...

\section*{Syntax}

Display file in Viewer
```

view [file] ["]filename["] [, asis adopath]

```

Bring up browser pointed to specified URL
```

view browse ["]url["]

```

Display help results in Viewer
```

[view] help [topic_or_command_name]

```

Display search results in Viewer [view] search keywords

Display net results in Viewer view net [netcmd]

Display ado-results in Viewer view ado [adocmd]

Display update results in Viewer view update [updatecmd]

\section*{Options}
asis, allowed with view file, specifies that the file be displayed as text, regardless of the filename's extension. view file's default action is to display files ending in .smcl and .sthlp as SMCL; see [P] smcl.
adopath, allowed with view file, specifies that Stata search the S_ADO path for filename and display it, if found.

\section*{Remarks and examples}

Most users access the Viewer by selecting File > View... and proceeding from there. Some commands allow you to skip that step. Some common interactive uses of commands that display their results in the Viewer are the following:
. view mysession.smcl
. view mysession.log
. help print
. help regress
. search hausman test
. view net
. view ado
. view update query

\section*{Also see}
[R] help - Display help in Stata
[R] net - Install and manage community-contributed additions from the Internet
[R] search - Search Stata documentation and other resources
[R] update - Check for official updates
[D] type - Display contents of a file
[GSM] 3 Using the Viewer
[GSU] 3 Using the Viewer
[GSW] 3 Using the Viewer

\section*{Title}
vwls - Variance-weighted least squares
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
vwls estimates a linear regression using variance-weighted least squares. It differs from ordinary least-squares (OLS) regression in that it does not assume homogeneity of variance, but requires that the conditional variance of depvar be estimated prior to the regression. The estimated variance need not be constant across observations. vwls treats the estimated variance as if it were the true variance when it computes standard errors of the coefficients.

You must supply an estimate of the conditional standard deviation of depvar to vwls by using the sd(varname) option, or you must have grouped data with the groups defined by the indepvars variables. In the latter case, vwls treats all indepvars as categorical variables, computes the mean and standard deviation of depvar separately for each subgroup, and computes the regression of the subgroup means on indepvars.
regress with analytic weights can be used to produce another kind of "variance-weighted least squares"; see Remarks and examples for an explanation of the difference.

\section*{Quick start}

Variance-weighted least-squares regression of y on x 1 and x 2 , with the estimated conditional std. dev. of y stored in sd
vwls y1 x1 x2, sd(sd)
Add categorical variable a using factor-variable syntax
vwls y1 x1 x2 i.a, sd(sd)
Same as above, but restrict the sample to cases where v is greater than 1
vwls y1 x1 x2 i.a if v>1, sd(sd)
Variance-weighted least-squares regression for grouped data with subgroups defined by a2 and a3
```

vwls y2 i.a2 i.a3

```

\section*{Menu}

Statistics \(>\) Linear models and related \(>\) Other \(>\) Variance-weighted least squares

\section*{Syntax}
```

    vwls depvar indepvars [if] [in] [weight] [, options ]
    ```
options Description

\section*{Model}
noconstant
sd(varname)
Reporting
level(\#)
display_options
coeflegend
suppress constant term
variable containing estimate of conditional standard deviation
set confidence level; default is level (95)
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
display legend instead of statistics
indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
bootstrap, by, collect, jackknife, rolling, and statsby are allowed; see [U] 11.1.10 Prefix commands.
Weights are not allowed with the bootstrap prefix; see [R] bootstrap.
fweights are allowed; see [U] 11.1.6 weight.
coeflegend does not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options}
\(\qquad\) Model
noconstant; see [R] Estimation options.
sd(varname) is an estimate of the conditional standard deviation of depvar (that is, it can vary observation by observation). All values of varname must be \(>0\). If you specify \(\operatorname{sd}(\) ), you cannot use fweights.

If sd() is not given, the data will be grouped by indepvars. Here indepvars are treated as categorical variables, and the means and standard deviations of depvar for each subgroup are calculated and used for the regression. Any subgroup for which the standard deviation is zero is dropped.
\(\qquad\) Reporting
level (\#); see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon (style), cformat (\%fmt), pformat (\%fmt), sformat (\%fmt), and nolstretch; see [R] Estimation options.

The following option is available with vwls but is not shown in the dialog box:
coeflegend; see [R] Estimation options.

\section*{Remarks and examples}

The vwls command is intended for use with two special—and different—types of data. The first contains data that consist of measurements from physical science experiments in which all error is due solely to measurement errors and the sizes of the measurement errors are known.

You can also use variance-weighted least-squares linear regression for certain problems in categorical data analysis, such as when all the independent variables are categorical and the outcome variable is either continuous or a quantity that can sensibly be averaged. If each of the subgroups defined by the categorical variables contains a reasonable number of subjects, then the variance of the outcome variable can be estimated independently within each subgroup. For the purposes of estimation, vwls treats each subgroup as one observation, with the dependent variable being the subgroup mean of the outcome variable.

The vwls command fits the model
\[
y_{i}=\mathbf{x}_{i} \boldsymbol{\beta}+\varepsilon_{i}
\]
where the errors \(\varepsilon_{i}\) are independent normal random variables with the distribution \(\varepsilon_{i} \sim N\left(0, \nu_{i}\right)\). The independent variables \(\mathbf{x}_{i}\) are assumed to be known without error.

As described above, vwls assumes that you already have estimates \(s_{i}^{2}\) for the variances \(\nu_{i}\). The error variance is not estimated in the regression. The estimates \(s_{i}^{2}\) are used to compute the standard errors of the coefficients; see Methods and formulas below.

In contrast, weighted OLS regression assumes that the errors have the distribution \(\varepsilon_{i} \sim N\left(0, \sigma^{2} / w_{i}\right)\), where the \(w_{i}\) are known weights and \(\sigma^{2}\) is an unknown parameter that is estimated in the regression. This is the difference from variance-weighted least squares: in weighted OLS, the magnitude of the error variance is estimated in the regression using all the data.
> Example 1
An artificial, but informative, example illustrates the difference between variance-weighted least squares and weighted OLS.

We measure the quantities \(x_{i}\) and \(y_{i}\) and estimate that the standard deviation of \(y_{i}\) is \(s_{i}\). We enter the data into Stata:
```

. use https://www.stata-press.com/data/r18/vwlsxmpl
. list

```
\begin{tabular}{|ccc|}
\hline\(x\) & \(y\) & \(s\) \\
\hline 1 & 1.2 & .5 \\
2 & 1.9 & .5 \\
3 & 3.2 & 1 \\
4 & 4.3 & 1 \\
5 & 4.9 & 1 \\
\hline 6 & 6.0 & 2 \\
7 & 7.2 & 2 \\
8 & 7.9 & 2 \\
\hline
\end{tabular}

Because we want observations with smaller variance to carry larger weight in the regression, we compute an OLS regression with analytic weights proportional to the inverse of the squared standard deviations:
```

. regress y x [aweight=s^(-2)]
(sum of wgt is 11.75)

```
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline Source & SS & df & MS & Number of obs & = & 8 \\
\hline & & & & F (1, 6) & = & 702.26 \\
\hline Model & 22.6310183 & 1 & 22.6310183 & Prob > F & = & 0.0000 \\
\hline Residual & . 193355117 & 6 & . 032225853 & R -squared & = & 0.9915 \\
\hline & & & & Adj R-squared & & 0.9901 \\
\hline Total & 22.8243734 & 7 & 3.26062477 & Root MSE & = & . 17952 \\
\hline
\end{tabular}
\begin{tabular}{r|rrrrrr}
\hline y & Coefficient & Std. err. & t & \(\mathrm{P}>|\mathrm{t}|\) & {\([95 \%\) conf. interval] } \\
\hline x & .9824683 & .0370739 & 26.50 & 0.000 & .8917517 & 1.073185 \\
_cons & .1138554 & .1120078 & 1.02 & 0.349 & -.1602179 & .3879288 \\
\hline
\end{tabular}

If we compute a variance-weighted least-squares regression by using vwls, we get the same results for the coefficient estimates but very different standard errors:
\begin{tabular}{|c|c|c|c|c|c|c|c|}
\hline \multicolumn{4}{|l|}{Variance-weighted least-squares regression} & \multicolumn{2}{|l|}{\multirow[t]{2}{*}{Number of obs Model chi2(1)}} & \multirow[t]{3}{*}{=} & 8 \\
\hline \multicolumn{4}{|l|}{Goodness-of-fit chi2(6) \(=0.28\)} & & & & 33.24 \\
\hline Prob > chi2 & & \(=0.9996\) & & Prob > & & & 0.0000 \\
\hline y & Coefficient & Std. err. & z & \(P>|z|\) & [95\% & nf & interval] \\
\hline x & . 9824683 & . 170409 & 5.77 & 0.000 & . 648 & & 1.316464 \\
\hline _cons & . 1138554 & . 51484 & 0.22 & 0.825 & -. 895 & & 1.122923 \\
\hline
\end{tabular}

Although the values of \(y_{i}\) were nicely linear with \(x_{i}\), the vwls regression used the large estimates for the standard deviations to compute large standard errors for the coefficients. For weighted OLS regression, however, the scale of the analytic weights has no effect on the standard errors of the coefficients-only the relative proportions of the analytic weights affect the regression.

If we are sure of the sizes of our error estimates for \(y_{i}\), using vwls is valid. However, if we can estimate only the relative proportions of error among the \(y_{i}\), then vwls is not appropriate.

\section*{\(>\) Example 2}

Let's now consider an example of the use of vwls with categorical data. Suppose that we have blood pressure data for \(n=400\) subjects, categorized by gender and race (black or white). Here is a description of the data:
. use https://www.stata-press.com/data/r18/bp
. table gender race, statistic(mean bp) statistic(sd bp) statistic(freq)
> nformat(\%8.1f)
\begin{tabular}{l|rrr}
\hline & Race \\
& White & \begin{tabular}{r} 
Black
\end{tabular} & Total \\
\hline Gender & & & \\
Female & 117.1 & 118.5 & 117.8 \\
Mean & 10.3 & 11.6 & 10.9 \\
\(\quad\) Standard deviation & 100.0 & 100.0 & 200.0 \\
Frequency & & & \\
Male & 122.1 & 125.8 & 124.0 \\
\(\quad\) Mean & 10.6 & 15.5 & 13.3 \\
\(\quad\) Standard deviation & 100.0 & 100.0 & 200.0 \\
\(\quad\) Frequency & & & \\
Total & 119.6 & 122.2 & 120.9 \\
\(\quad\) Mean & 10.7 & 14.1 & 12.6 \\
Standard deviation & 200.0 & 200.0 & 400.0 \\
\(\quad\) Frequency & & & \\
\hline
\end{tabular}

Performing a variance-weighted regression using vwls gives
. vwls bp gender race
Variance-weighted least-squares regression Number of obs \(=\quad 400\)
Goodness-of-fit chi2(1) \(=0.88 \quad\) Model chi2(2) \(=27.11\)
Prob > chi2 \(=0.3486 \quad\) Prob > chi2 \(=0.0000\)
\begin{tabular}{r|rlcccc}
\hline bp & Coefficient & Std. err. & \(z\) & \(\mathrm{P}>|\mathrm{z}|\) & {\([95 \%\) conf. interval] } \\
\hline gender & 5.876522 & 1.170241 & 5.02 & 0.000 & 3.582892 & 8.170151 \\
race & 2.372818 & 1.191683 & 1.99 & 0.046 & .0371631 & 4.708473 \\
_cons & 116.6486 & .9296297 & 125.48 & 0.000 & 114.8266 & 118.4707 \\
\hline
\end{tabular}

By comparison, an OLS regression gives the following result:
- regress bp gender race
\begin{tabular}{r|rrr} 
Source & SS & df & MS \\
\hline \begin{tabular}{r} 
Model \\
Residual
\end{tabular} & 4485.66639 & 2 & 2242.83319 \\
\hline Total & 629442.7305 & 397 & 147.210908 \\
\hline
\end{tabular}
\begin{tabular}{llr} 
Number of obs & \(=\) & 400 \\
\(\mathrm{~F}(2,397)\) & \(=\) & 15.24 \\
Prob > F & \(=\) & 0.0000 \\
R-squared & \(=\) & 0.0713 \\
Adj R-squared & \(=\) & 0.0666 \\
Root MSE & \(=\) & 12.133
\end{tabular}
\begin{tabular}{r|rlccrr}
\hline bp & Coefficient & Std. err. & t & \(\mathrm{P}>|\mathrm{t}|\) & [95\% conf. interval] \\
\hline gender & 6.1775 & 1.213305 & 5.09 & 0.000 & 3.792194 & 8.562806 \\
race & 2.5875 & 1.213305 & 2.13 & 0.034 & .2021938 & 4.972806 \\
_cons & 116.4862 & 1.050753 & 110.86 & 0.000 & 114.4205 & 118.552 \\
\hline
\end{tabular}

Note the larger value for the race coefficient (and smaller \(p\)-value) in the OLS regression. The assumption of homogeneity of variance in OLS means that the mean for black men pulls the regression line higher than in the vwls regression, which takes into account the larger variance for black men and reduces its effect on the regression.

\section*{Stored results}
vwls stores the following in e():
Scalars
e(N)
e(df_m)
e(chi2)
e(df_gf)
e(chi2_gf)
\(e\) (rank)
Macros
\(e(\mathrm{cmd})\)
e(cmdline)
e(depvar)
e(properties)
e(predict)
e(asbalanced)
e(asobserved)
Matrices
e(b)
e(V)
Functions
e(sample) marks estimation sample

In addition to the above, the following is stored in r() :

Matrices
\(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r -class command is run after the estimation command.

\section*{Methods and formulas}

Let \(\mathbf{y}=\left(y_{1}, y_{2}, \ldots, y_{n}\right)^{\prime}\) be the vector of observations of the dependent variable, where \(n\) is the number of observations. When \(\operatorname{sd}()\) is specified, let \(s_{1}, s_{2}, \ldots, s_{n}\) be the standard deviations supplied by sd(). For categorical data, when sd() is not given, the means and standard deviations of \(y\) for each subgroup are computed, and \(n\) becomes the number of subgroups, \(\mathbf{y}\) is the vector of subgroup means, and \(s_{i}\) are the standard deviations for the subgroups.

Let \(\mathbf{V}=\operatorname{diag}\left(s_{1}^{2}, s_{2}^{2}, \ldots, s_{n}^{2}\right)\) denote the estimate of the variance of \(\mathbf{y}\). Then the estimated regression coefficients are
\[
\mathbf{b}=\left(\mathbf{X}^{\prime} \mathbf{V}^{-1} \mathbf{X}\right)^{-1} \mathbf{X}^{\prime} \mathbf{V}^{-1} \mathbf{y}
\]
and their estimated covariance matrix is
\[
\widehat{\operatorname{Cov}}(\mathbf{b})=\left(\mathbf{X}^{\prime} \mathbf{V}^{-1} \mathbf{X}\right)^{-1}
\]

A statistic for the goodness of fit of the model is
\[
Q=(\mathbf{y}-\mathbf{X b})^{\prime} \mathbf{V}^{-1}(\mathbf{y}-\mathbf{X b})
\]
where \(Q\) has a \(\chi^{2}\) distribution with \(n-k\) degrees of freedom ( \(k\) is the number of independent variables plus the constant, if any).

\section*{References}

Gini, R., and J. Pasquini. 2006. Automatic generation of documents. Stata Journal 6: 22-39.
Grizzle, J. E., C. F. Starmer, and G. G. Koch. 1969. Analysis of categorical data by linear models. Biometrics 25: 489-504. https://doi.org/10.2307/2528901.

\section*{Also see}
[R] vwls postestimation - Postestimation tools for vwls
[ R ] regress - Linear regression

\section*{[U] 11.1.6 weight}
[U] 20 Estimation and postestimation commands

\section*{Title}
vwls postestimation - Postestimation tools for vwls

Postestimation commands predict margins Also see

\section*{Postestimation commands}

The following postestimation commands are available after vwls:
\begin{tabular}{ll} 
Command & Description \\
\hline contrast & contrasts and ANOVA-style joint tests of estimates \\
estat summarize & summary statistics for the estimation sample \\
estat vce & variance-covariance matrix of the estimators (VCE) \\
estimates & cataloging estimation results \\
etable & table of estimation results \\
forecast & dynamic forecasts and simulations \\
hausman & \begin{tabular}{l} 
Hausman's specification test \\
lincom
\end{tabular} \\
point estimates, standard errors, testing, and inference for linear combinations of \\
linktest & coefficients \\
margins & \begin{tabular}{l} 
marginal means, predictive margins, marginal effects, and average marginal effects \\
marginsplot
\end{tabular} \\
graph the results from margins (profile plots, interaction plots, etc.) \\
nlcom & point estimates, standard errors, testing, and inference for nonlinear combinations \\
predict & of coefficients \\
predictnl & \begin{tabular}{l} 
linear, censored, and truncated predictions \\
pwcompare
\end{tabular} \\
pairwise comparisons of estimates
\end{tabular}

\section*{predict}

\section*{Description for predict}
predict creates a new variable containing predictions such as linear predictions and standard errors.

\section*{Menu for predict}

Statistics \(>\) Postestimation

\section*{Syntax for predict}
predict [type] newvar [if] [in] [, xb stdp]
These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

\section*{Options for predict}

Main
xb , the default, calculates the linear prediction.
stdp calculates the standard error of the linear prediction.

\section*{margins}

\section*{Description for margins}
margins estimates margins of response for linear predictions.

\section*{Menu for margins}

Statistics > Postestimation

\section*{Syntax for margins}
```

margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [options]

```
statistic Description
\begin{tabular}{ll}
xb & linear prediction; the default \\
stdp & not allowed with margins
\end{tabular}

Statistics not allowed with margins are functions of stochastic quantities other than e(b). For the full syntax, see [R] margins.

\section*{Also see}
[R] vwls - Variance-weighted least squares
[U] 20 Estimation and postestimation commands

\section*{Title}
which - Display location of an ado-file

\section*{Description Syntax Option Remarks and examples Also see}

\section*{Description}
which looks for file fname .ftype along the S_ADO path. If Stata finds the file, which displays the full path and filename, along with, if the file is text, all lines in the file that begin with " \(*\) !" in the first column. If Stata cannot find the file, which issues the message "file not found along ado-path" and sets the return code to 111. ftype must be a file type for which Stata usually looks along the ado-path to find. Allowable ftypes are
```

.ado, .class,.dlg, .idlg, .sthlp,.ihlp, .hlp, .jar, .key, .maint, .mata, .mlib,
.mo, .mnu, .plugin, .png, .py, .scheme, .stbcal, and .style

```

If ftype is omitted, which assumes .ado. When searching for . ado files, if Stata cannot find the file, Stata then checks to see if fname is a built-in Stata command, allowing for valid abbreviations. If it is, the message "built-in command" is displayed; if not, the message "command not found as either built-in or ado-file" is displayed and the return code is set to 111 .

\section*{Syntax}
which fname[.ftype] [, all]

\section*{Option}
all forces which to report the location of all files matching the fname.ftype found along the search path. The default is to report just the first one found.

\section*{Remarks and examples}

If you write programs, you know that you make changes to the programs over time. If you are like us, you also end up with multiple versions of the program stored on your disk, perhaps in different directories. You may even have given copies of your programs to other Stata users, and you may not remember which version of a program you or your friends are using. The which command helps you solve this problem.
> Example 1
The which command displays the path for filename.ado and any lines in the code that begin with "*!". For example, we might want information about the test command, described in \([R]\) test, which is an ado-file written by StataCorp. Here is what happens when we type which test:
```

. which test
C:\Program Files\Stata18\ado\base\t\test.ado
*! version 2.4.1 28jan2021

```
which displays the path for the test.ado file and also a line beginning with " \(*\) !" that indicates the version of the file.

Do not confuse "version 2.4.1" above with the version command or the version of Stata. The "*!" code indicates notes. In this case, the \(*\) ! version 2.4.1 line is merely a note to the author to identify the iteration (version) of the test program. It is an unfortunate coincidence that the word version is overloaded in this manner.

We do not need to be so formal. which will display anything typed after lines that begin with ' \(*\) !'. For instance, we might write myprog. ado:
. which myprog
. \myprog.ado
*! first written 1/03/2023
*! bug fix on 1/05/2023 (no variance case)
*! updated \(1 / 24 / 2023\) to include noconstant option
*! still suspicious if variable takes on only two values
It does not matter where in the program the lines beginning with \(*\) ! are-which will list them (in particular, our "still suspicious" comment was buried about 50 lines down in the code). All that is important is that the \(*!\) marker appear in the first two columns of a line.
\(>\) Example 2
If we type which command, where command is a built-in command rather than an ado-file, Stata responds with
```

. which summarize
built-in command: summarize

```

If command was neither a built-in command nor an ado-file, Stata would respond with
```

. which junk
command junk not found as either built-in or ado-file
r(111);

```

\section*{Also see}
[P] findfile - Find file in path
[U] 17 Ado-files

\section*{Title}
wildbootstrap - Wild cluster bootstrap inference
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
Acknowledgments & References & Also see &
\end{tabular}

\section*{Description}
wildbootstrap performs wild cluster bootstrap (WCB) inference for linear hypotheses about parameters in linear regression models. These hypotheses can be simple or composite. When the assumptions required for the consistency of the cluster-robust variance estimator (CRVE) do not hold, the WCB is a good alternative.

\section*{Quick start}

Estimate the WCB \(p\)-value and confidence interval (CI) for the coefficient on x 1 in a linear regression of y on x 1 with clusters identified in cvar
wildbootstrap regress y x1, cluster (cvar)
Same as above, but test whether the coefficients on x 1 and x 2 are equal in a regression of y on x 1 and x 2
```

    wildbootstrap regress y x1 x2, cluster(cvar) test(x1 = x2)
    ```

\section*{Menu}

Statistics \(>\) Resampling \(>\) Wild cluster bootstrap

\section*{Syntax}
wildbootstrap estimator depvar [indepvars] [if] [in] [weight] [, options]
\begin{tabular}{ll} 
estimator & Description \\
\hline regress & linear regression \\
areg & linear regression with a large indicator-variable set \\
xtreg & fixed-effects linear models
\end{tabular}
options
Main
noconstant
hascons
absorb (varname)

Statistics
coefficients (coeflist)
test (testspec)
```

Bootstrap
cluster(clustvar)
ptype(ptype)
errorweight (wcbwtype)
reps(\#)
rseed (\#|statecode)
blocksize(\#)
cistop(largest|first)

```

Reporting
level(\#)
display_options

Description
suppress constant term; available only with estimator regress has user-supplied constant; available only with estimator regress categorical variable to be absorbed; required by and available only with estimator areg
perform inference for a subset of coefficients
specify linear test parameters
specify variable identifying clusters; required for estimators regress and areg
specify the \(p\)-value type; may be equal (the default) or symmetric specify WCB weight; default is errorweight (rademacher)
set number of bootstrap repetitions; default is reps (1000)
set random-number seed to \# or statecode
set bootstrap repetition block size; default is \(\min (r e p s(\#), 1000)\) specify stopping rule for CI computation
set confidence level; default is level(95)
control column formats and display of CIs
indepvars and coeflist may contain factor variables; see [U] 11.4.3 Factor variables.
depvar, indepvars, and coeflist may contain time-series operators; see [U] 11.4.4 Time-series varlists.
collect is allowed; see [U] 11.1.10 Prefix commands.
Any weight that is allowed by the estimator is allowed; see [U] 11.1.6 weight.

\section*{Options}

noconstant and hascons; see [R] regress. These options may be specified only when estimator regress is specified.
absorb (varname); see [R] areg. This option must be specified when estimator areg is specified.
coefficients (coeflist) performs an inference for a subset of coefficients. It reports the bootstrap \(p\)-value for a test of the subset of coefficients, coeflist, equal to 0 and the bootstrap CI. The default is to perform inference for all variables specified in indepvars. coefficients() may not be specified with test().
test (testspec) specifies a linear test. You may also specify multiple linear tests by using test ((testspecl) (testspec2) ...). The test specification must be consistent with specifying a linear constraint. See \([R]\) test and \([R]\) constraint.

\section*{Bootstrap}
cluster (clustvar) specifies the variable identifying the cluster groups. cluster() is required with estimators regress and areg. With estimator xtreg, clustvar defaults to the xtset panelvar. ptype(ptype) specifies the \(p\)-value criterion: symmetric (symmetric) or equal tailed (equal). The default is ptype(equal). See Methods and formulas for more details.
errorweight (wcbwtype) specifies the type of wild weight. wcbwtype may be one of the following:
\begin{tabular}{ll}
\begin{tabular}{l} 
wchwtype
\end{tabular} & Description \\
\hline rademacher & \begin{tabular}{c} 
two-point distribution assigns values 1 and -1 with equal \\
probability; the default
\end{tabular} \\
\(\underline{\text { mammen }}\) & \begin{tabular}{c} 
two-point distribution assigns value \(1-\phi\) with probability \\
\(\phi / \sqrt{5}\) and value \(\phi\) otherwise, where \(\phi=(1+\sqrt{5}) / 2\)
\end{tabular} \\
webb & \begin{tabular}{l} 
six-point distribution assigns probability of \(1 / 6\) to the points \\
\\
normal
\end{tabular} \\
\begin{tabular}{l} 
standard normal distribution \\
gamma distribution with shape parameter 4 and scale parameter \(1 / 2\) \\
gamma
\end{tabular} & \begin{tabular}{l} 
centered on its mean of 2
\end{tabular} \\
\hline
\end{tabular}
reps(\#) sets the number of repetitions for the bootstrap. The default is reps(1000). For the ptype (equal) option, the values of level() and reps () should be chosen so that \(\alpha / 2 \times \mathrm{reps}\) (\#) is an integer, where \(\alpha=(100-\) level (\#) )/100. For the ptype(symmetric) option, such values should be chosen so that \(\alpha \times \operatorname{reps}(\#)\) is an integer. When the product is not an integer, the number of repetitions is increased so that it is. Integer values improve the search efficiency of the wildbootstrap algorithm.
rseed (\#|statecode) sets the random-number seed to \# or statecode. See [R] set seed.
blocksize(\#) sets the bootstrap block size. This is an integer less than or equal to reps(\#) and is used to reduce the amount of memory the bootstrap computation will consume. The default is \(\min (\mathrm{reps}(\#), 1000)\).
cistop(largest|first) specifies the stopping rule for the CI computation. The bootstrap distribution is a step function, so for each bound, there is an interval of values that meet the CI level criterion. cistop(largest), the default, specifies that the largest value within the interval be selected. cistop (first) specifies that the first value the algorithm finds within the interval be selected; therefore, specifying cistop(first) will result in faster CI computation. The cistop() option may not be combined with the noci option.

Reporting
level(\#); see [R] Estimation options. The level() option will not work on replay because CIs are based on estimator-specific enumerations. To change the confidence level, you must refit the model.
display_options: noci, cformat (\%fmt), pformat (\%fmt), and sformat (\%fmt); see [R] Estimation options. The noci option may not be combined with the cistop() option.

\section*{Remarks and examples}
wildbootstrap implements the WCB, which was proposed by Cameron, Gelbach, and Miller (2008). It is an extension of the original wild bootstrap procedure proposed by Wu (1986), which was designed to work well for models with heteroskedasticity, to the case of cluster-level correlation. The wild bootstrap has proven to work well in cases where cluster-robust standard errors do not perform well. A good discussion of the methodology can be found in Cameron and Miller (2015), MacKinnon (2019), MacKinnon and Webb (2018), and MacKinnon, Nielsen, and Webb (2023), and the references therein. Specifically, the WCB is a good inference tool when one or more of the CRVE \(t\)-statistic consistency assumptions are violated. MacKinnon and Webb list the assumptions as follows:
1. The number of clusters goes to infinity.
2. The within-cluster error correlations are the same for all clusters.
3. Each cluster contains an equal number of observations.

Below, we illustrate how to use wildbootstrap; however, note that alternatives exist in the literature to address the inference problems noted above. For example, the Bell and McCaffrey (2002) \(t\)-statistic degrees-of-freedom correction is an alternative to wildbootstrap when at least one of the above assumptions is violated. The degrees-of-freedom correction is computed with option vce (hc2 clustvar, dfadjust) for regress, areg, and xtreg, fe.
> Example 1: Simple regression
Say we are interested in the effect of tenure on wages for a panel of individuals sampled from 2013 to 2016. We would like to use wild bootstrap CIs clustering at the personid level. For reproducibility, we set the seed by using option rseed().


The iteration log indicates the number of iterations used to compute the lower and upper bound for the CIs. In Methods and formulas below, we discuss how these bounds are computed. Notably, there is a separate optimization procedure used to compute each one of these bounds.

The table header also tells us the error weights used for the sampling algorithm of the wild bootstrap, which, by default, are Rademacher weights. See the errorweight () option for more details about error weights.

The column header labeled p -value indicates that the \(t\)-statistic equal-tailed \(p\)-value has been computed. The ptype (equal) option is the default. Alternatively, the symmetric \(p\)-value is computed when the ptype (symmetric) option is specified and is identified with the column header of \(\mathrm{P}>|\mathrm{t}|\).
```

. wildbootstrap regress wage tenure, cluster(personid) rseed(12345)
> ptype(symmetric)
Performing 1,000 replications for p-value for tenure = 0 ...
Computing confidence interval for tenure
Lower bound: .........10.........20.. done (22)
Upper bound: .........10..... done (15)
Wild cluster bootstrap Number of obs = 1,928
Linear regression Number of clusters = 589
Cluster variable: personid min = 1
Error weight: Rademacher
Cluster size:
avg = 3.3
max = 4

|  | wage | Estimate | t | $\mathrm{P}>\|\mathrm{t}\|$ | [95\% conf. interval] |  |
| :--- | ---: | :---: | :---: | :---: | :---: | :---: |
| constraint | tenure $=0$ | .7807403 | 27.19 | 0.000 | .7240502 | .8399896 |

```

We can always compare the CIs from the wild bootstrap with what we would have obtained using the underlying command, in this case, regress.
\begin{tabular}{|c|c|c|c|c|c|c|c|}
\hline \multicolumn{2}{|l|}{Linear regression} & \multicolumn{6}{|l|}{\begin{tabular}{l}
\begin{tabular}{llr} 
Number of obs & \(=\) & 1,928 \\
\(\mathrm{~F}(1,588)\) & \(=\) & 739.36 \\
Prob \(>\mathrm{F}\) & \(=\) & 0.0000 \\
R-squared & \(=\) & 0.4212 \\
Root MSE & \(=\) & 3.5097
\end{tabular} \\
(Std. err. adjusted for 589 clusters in personid)
\end{tabular}} \\
\hline wage & Coefficient & Robust std. err. & t & \(P>|t|\) & [95\% & & interval] \\
\hline tenure & . 7807403 & . 028713 & 27.19 & 0.000 & . 724 & & . 8371328 \\
\hline _cons & 20.89884 & . 2135686 & 97.86 & 0.000 & 20.4 & & 21.31829 \\
\hline
\end{tabular}

Note that typing this command will replace the return matrix \(r\) (table). Observe that the \(t\) statistics are the same in both tables because regress and wildbootstrap use the same CRVE, but the \(p\)-values and CIs may vary between tables.

Similarly, you can redisplay the wildbootstrap table by typing wildbootstrap, which may be abbreviated as wildboot.

\section*{> Example 2: Small number of clusters with wildly varying cluster sizes}

As in example 1, we would like to see the effect of tenure on wages; in this case, however, we would like to cluster at the industry level. Here, for year 1988, we use a wage dataset with only 12 clusters, for which cluster sizes vary wildly from 4 to 817, violating the CRVE \(t\)-statistic consistency assumptions 1 and 3 outlined previously.


\section*{> Example 3: Small number of clusters with more regressors}

Continuing with example 2, we look at a more complex and perhaps more realistic example. We add to the model explanatory variables for total work experience, ttl_exp; a college graduate indicator, collgrad; and an indicator for union membership, union.
```

. wildbootstrap regress wage c.tenure\#\#c.ttl_exp ibO.collgrad ibO.union,
> cluster(industry) rseed(12345)
Performing 1,000 replications for p-value for tenure = 0 ...
Computing confidence interval for tenure
Lower bound: .........10.........20..........30.... done (34)
Upper bound: .........10.........20.... done (24)
Performing 1,000 replications for p-value for ttl_exp = 0 ...
Computing confidence interval for ttl_exp
Lower bound: .........10..........20..... done (25)
Upper bound: .........10......... done (19)
Performing 1,000 replications for p-value for c.tenure\#c.ttl_exp = 0 ...
Computing confidence interval for c.tenure\#c.ttl_exp
Lower bound: .........10.........20... done (23)
Upper bound: .........10..........20....... done (27)
note: upper-bound CI achieved 1-F(-3.65e-04) = 0.0240, but target is 1-F(x) =
.025.
note: the sorted bootstrap t statistics have at least two tied values
adjacent to the t statistic under the null; this prevents the CI bound
from converging to the target.
Performing 1,000 replications for p-value for 1.collgrad = 0 ...
Computing confidence interval for 1.collgrad
Lower bound: .........10.........20.........30.. done (32)
note: lower-bound CI achieved F(1.66) = 0.0240, but target is F(x) = .025.
note: at least one bootstrap t statistic matches the t statistic under the
null; this prevents the CI bound from converging to the target.
Upper bound: .........10..........20...... done (26)
Performing 1,000 replications for p-value for 1.union = 0 ...
Computing confidence interval for 1.union
Lower bound: .........10.........20......... 30 done (30)
Upper bound: .........10..........20..... done (25)

```
Wild cluster bootstrap
Linear regression
Cluster variable: industry
Error weight: Rademacher

We did not specify the coefficients() option, so wildbootstrap computes \(p\)-values and CIs for all coefficients in the model, with the exception of the constant term, using the default ptype (equal) option to compute the equal-tailed \(p\)-values.

The iteration \(\log\) states that the CI-target tail values were not achieved for two of the reported constraints. This is because the sorted vector of bootstrap \(t\) statistics can have ties due to computer finite numeric precision and the finite number of draws for the Rademacher, Mammen, and Webb distributions. When ties occur with the \(t\) statistic under the null, or those adjacent to it, and the desired CI tail area cannot be achieved, wildbootstrap will choose a bound that results in a smaller tail area. This is the case when wildbootstrap searches for the c.tenure\#c.ttl_exp CI upper bound and the 1.collgrad CI lower bound.

Numerical computations will contain a roundoff error; wildbootstrap retains 13 digits on the mantissa of the \(t\) statistics before making comparisons.
> Example 4: Linear regression with an indicator-variable set
We would like to include an individual's occupation code (occupation) as an additional control in the regression to avoid potential omitted-variable bias. We keep the other control variables we introduced in example 3. For this analysis, we use the estimator areg with the absorb (occupation) option. We use the test () option to test linear combinations of regression estimates.
```

. wildbootstrap areg wage c.tenure\#\#c.ttl_exp ib0.collgrad ib0.union,
> absorb(occupation) cluster(industry) rseed(12345)
> test((tenure=ttl_exp) (1.collgrad-1.union=1)) reps(1250)
note: for equal-tailed 95% CI, better performance is obtained when
.025*reps() is an integer.
note: setting repetitions to 1,280.
Performing 1,280 replications for p-value for constraint
tenure - ttl_exp = 0 ...
Computing confidence interval for tenure - ttl_exp
Lower bound: .........10..........20... done (23)
Upper bound: .........10.........20..... done (25)
note: upper-bound CI achieved 1-F(0.15) = 0.0242, but target is 1-F(x) =
.025.
note: the sorted bootstrap t statistics have at least two tied values
adjacent to the t statistic under the null; this prevents the CI bound
from converging to the target.
Performing 1,280 replications for p-value for constraint
1.collgrad - 1.union = 1 ...

```
\begin{tabular}{|c|c|c|c|c|c|}
\hline \multicolumn{6}{|l|}{\multirow[t]{2}{*}{Computing confidence interval for 1.collgrad - 1.union
Lower bound: ........10........20...... done (26)}} \\
\hline & & & & & \\
\hline \multicolumn{6}{|l|}{Upper bound: ........10........20....... done (27)} \\
\hline \multicolumn{6}{|l|}{note: upper-bound CI achieved \(1-\mathrm{F}(\mathbf{3 . 9 1})=\mathbf{0 . 0 2 4 2}\), but target is \(1-\mathrm{F}(\mathrm{x})=\) .025.} \\
\hline \multicolumn{6}{|l|}{adjacent to the \(t\) statistic under the null; this prevents the CI bound} \\
\hline \multicolumn{4}{|l|}{Wild cluster bootstrap N} & r of obs & \(=1,851\) \\
\hline \multicolumn{3}{|l|}{Linear regression, absorbing indicators} & & r of clus er size: & \[
\text { ters }=12
\] \\
\hline \multicolumn{3}{|l|}{Cluster variable: industry} & & & min \(=\) \\
\hline \multicolumn{2}{|l|}{\multirow[t]{2}{*}{Error weight: Rademacher}} & & & & avg \(=154.2\) \\
\hline & & & & & \(\max =717\) \\
\hline wage & Estimate & & \(p\)-value & [95\% con & f. interval] \\
\hline \multicolumn{6}{|l|}{constraints} \\
\hline tenure - ttl_exp \(=0\) & -. 0391243 & -0.71 & 0.517 & -. 1915527 & . 1495055 \\
\hline 1.collgrad - 1.union = 1 & 1.356332 & 0.77 & 0.395 & -. 1882941 & 3.913359 \\
\hline
\end{tabular}

The test () option allows us to specify a linear combination of the regression coefficients that we would like to test. The syntax is the same as specifying linear constraints, constraint, or linear hypothesis tests, test. In this example, we are inquiring how probable it would be for tenure \(=\) ttl_exp and 1. collgrad -1. union \(=1\).

The reps() option allows us to set the number of bootstrap repetitions. The default is 1,000 . Because the Rademacher and Mammen distributions have two possible realizations, the maximum number of possible bootstrap samples is \(2^{G}\), where \(G\) is the number of clusters. The Webb distribution, on the other hand, has six outcomes and therefore \(6^{G}\) possible combinations.

In this example, we intentionally specify a number of bootstrap repetitions that results in a noninteger product, \(\operatorname{reps}() \times \alpha / 2: \operatorname{reps}() \times(100-\) level ()\() / 200=1250 \times 0.025=31.25\). In this case, we lose efficiency in searching for the CI bounds, so wildbootstrap adjusts the number of repetitions to 1,280 ; when we specify reps (1250) with level(95), wildbootstrap chooses \(\operatorname{reps}()=\operatorname{ceil}(1250 \times \alpha / 2) /(\alpha / 2)=32 / 0.025=1280\). If an integer alternative cannot be found, the original reps() specification will be used. For details, see Methods and formulas below.

Because the WCB distribution is a step function, there is a range of values for each CI bound. For example, the lower-bound interval of values for the linear combination tenure - ttl_exp is \(-0.191586<a_{\mathrm{lb}} \leq-0.191553\). The reported bounds are the largest value of each interval. We obtain -0.191586 by running wildbootstrap specifying level(\#) with \(\#=(0.95+2 / 1280) \times 100\). We would likely report the CIs using three digits of precision, for instance, \([-0.192,0.150]\), so this interval is negligible but significant in a numerical root search where a convergence tolerance for \(r_{l}\) might be \(10^{-8}\).

The estimate of tenure - ttl_exp is reported to be -0.0391 with a \(95 \%\) CI of \([-0.192,0.150]\). We therefore fail to reject the hypothesis that tenure \(=\) ttl_exp. Similarly, the estimate of 1 . collgrad -1 . union is 1.36 with a CI of \([-0.188,3.91]\), and we conclude the difference of 1 is feasible.

\section*{> Example 5: Fixed-effects linear regression with panels}

Our final example demonstrates the use of wildbootstrap with estimator xtreg. Continuing with the data in the previous example, we use industry as the variable defining the panels.

We specify normal error weights, errorweight (normal), thereby reducing the chance of identical draws for the error weights. Draws from the Rademacher distribution for 12 clusters have \(2^{12}=4096\) combinations.

We compute \(p\)-values and CIs on a subset of the model estimates: work tenure, work experience, and their interaction. If we replay the xtreg coefficient table, we will see that the fitted model includes tenure, ttl_exp, c.tenure\#c.ttl_exp, 1.collgrad, and 1.union.

Finally, we use the symmetric \(p\)-value, option ptype(symmetric), instead of the default equaltailed, ptype(equal). The table identifies the symmetric \(p\)-value with the header \(\mathrm{P}>|\mathrm{t}|\).

The command and the results are as follows:
```

. xtset industry
Panel variable: industry (unbalanced)
. wildbootstrap xtreg wage c.tenure\#\#c.ttl_exp i.collgrad i.union,
> rseed(12345) coef(ten ttl ten\#ttl) errorweight(normal)
> ptype(symmetric)
Panel variable: industry (unbalanced)
Performing 1,000 replications for p-value for constraint
tenure = 0 ...
Computing confidence interval for tenure
Lower bound: .........10.........20...... done (26)
Upper bound: .........10.........20. done (21)
Performing 1,000 replications for p-value for constraint
ttl_exp = 0 ...
Computing confidence interval for ttl_exp
Lower bound: .........10.........20...... done (26)
Upper bound: .........10........ done (18)
Performing 1,000 replications for p-value for constraint
c.tenure\#c.ttl_exp = 0 ...
Computing confidence interval for c.tenure\#c.ttl_exp
Lower bound: .........10.........20...... done (26)
Upper bound: .........10.........20... done (23)

```
\begin{tabular}{llr} 
Wild cluster bootstrap & Number of obs & \(=1,855\) \\
Fixed-effects linear regression & \begin{tabular}{l} 
Number of clusters
\end{tabular} & 12 \\
Cluster variable: industry & Cluster size: & \\
Error weight: Normal & & min
\end{tabular}
\begin{tabular}{rr|rrrrr}
\hline wage & Estimate & t & \(\mathrm{P}>\mid \mathrm{tl}\) & [95\% conf. interval] \\
\hline constraints & & & & & & \\
tenure \(=0\) & .2026682 & 2.95 & 0.024 & .0555676 & .3753468 \\
ttl_exp \(=0\) & .2716375 & 11.59 & 0.000 & .2227054 & .3372929 \\
c.tenure\#c.ttl_exp \(=0\) & -.0104125 & -2.55 & 0.046 & -.0201336 & -.0003384 \\
\hline
\end{tabular}

The fixed-effects model is the only xtreg model allowed with wildbootstrap, so specifying the fe option is not required.

Also note that the cluster() option is not specified. When the estimator xtreg is specified, the default cluster variable is the panel variable that is xtset. If the specified cluster variable is different
from the panel variable, then the levels of the panel variable must be nested within the cluster variable levels.

\section*{Stored results}
wildbootstrap stores the following in e():
Scalars
e(N_clust) number of clusters
e ( N -wbreps) number of bootstrap repetitions
\(e\left(w b \_b l o c k\right) \quad\) bootstrap block size
\(e\) (n_wbcns) number of bootstrap constraint restrictions
\(e\) (min_c) smallest cluster size
\(e\) (max_c) largest cluster size
\(e\) (avg_c) average cluster size
Macros
e(cmdline)
e(cmdO)
e(wb_stat)
\(e\) (wb_weight)
e(wb_ptype)
e(wb_level)
\(e\) (wb_cistop)
e(wb_rseed)
e(wb_cnsi)
e(clustvar)
command as typed
wildbootstrap
WCB statistic, t
WCB weights
WCB \(p\)-value criterion
WCB CI level
WCB CI interval type
random-number state
WCB constraint \(i\), where \(i=1, \ldots, \mathrm{e}\) (n_wbcns)
name of cluster variable
Matrices
e(wboot) WCB table
e(wb_pci) WCB CI coverage
\(e\) (wb_Cns) WCB constraint matrix
wildbootstrap will also carry forward most of the results already in e() from command.
In addition to the above, the following is stored in r() :
Matrices
\(r(\) table ) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

\section*{Methods and formulas}

Methods and formulas are presented under the following headings:
Introduction
CIs for linear combinations of coefficients
Constructing a CI inverting the hypothesis test

\section*{Introduction}

We will focus our discussion of the WCB on linear regression, which easily extends to the other fixed-effects estimators. A linear regression model with clustered errors and \(G\) clusters can be written as
\[
\mathbf{y}=\left(\begin{array}{c}
\mathbf{y}_{1}  \tag{1}\\
\mathbf{y}_{2} \\
\vdots \\
\mathbf{y}_{G}
\end{array}\right)=\mathbf{X} \beta+\epsilon=\left(\begin{array}{c}
\mathbf{X}_{1} \\
\mathbf{X}_{2} \\
\vdots \\
\mathbf{X}_{G}
\end{array}\right)\left(\begin{array}{c}
\beta_{1} \\
\beta_{2} \\
\vdots \\
\beta_{k}
\end{array}\right)+\left(\begin{array}{c}
\epsilon_{1} \\
\epsilon_{2} \\
\vdots \\
\epsilon_{G}
\end{array}\right)
\]
where for each cluster \(g, N_{g}\) is the number of observations, \(y_{g}\) is an \(N_{g} \times 1\) vector of outcomes, \(X_{g}\) is an \(N_{g} \times k\) matrix of covariates, and \(\epsilon_{g}\) is an \(N_{g} \times 1\) vector of errors. The parameter of interest is the \(k \times 1\) vector of coefficients \(\beta\). Error terms are assumed uncorrelated between observations of different clusters but possibly correlated between observations within the same cluster.

Without loss of generality, we will first focus on testing the null hypothesis \(H_{0}: \beta_{k}=0\) using the WCB algorithm. Let \(\widehat{\beta}\) denote the ordinary least-squares estimator for \(\beta\). We compute the \(t\) statistic for the \(k\) th coefficient as
\[
t_{k}=\frac{\widehat{\beta}_{k}}{\sqrt{\widehat{\mathbf{V}}_{k, k}}}
\]
where \(\widehat{\mathbf{V}}_{k, k}\) is the CRVE for \(\widehat{\beta}_{k}\). This is the \(k\) th diagonal element of the matrix
\[
\begin{equation*}
\widehat{\mathbf{V}}=\frac{G(N-1)}{(G-1)(N-k)}\left(\mathbf{X} \mathbf{X}^{\prime}\right)^{-1}\left(\sum_{g=1}^{G} \mathbf{X}_{g}^{\prime} \widehat{\epsilon}_{g} \hat{\epsilon}_{g}^{\prime} \mathbf{X}_{g}\right)\left(\mathbf{X} \mathbf{X}^{\prime}\right)^{-1} \tag{2}
\end{equation*}
\]
where \(\widehat{\epsilon}_{g}\) is the \(N_{g} \times 1\) vector of ordinary least-squares residuals for cluster \(g\) and \(N=\sum_{g=1}^{G} N_{g}\) is the total number of observations.

The WCB algorithm proceeds as follows:
1. Refit model (1) subject to the restriction \(\beta_{k}=0\). Let \(\widetilde{\beta}\) denote the restricted estimates and \(\widetilde{\epsilon}\) denote the restricted residuals.
2. For each individual bootstrap replication \(b\) (out of a total of \(B\) replications):

2a. Generate random variable \(\nu_{g}^{b}\) for each cluster \(g\) according to the distribution specified in the errorweight() option.
2b. For each cluster \(g\) and each observation in the cluster \(i=1,2, \ldots, N_{g}\), generate a new bootstrap-dependent variable \(y_{i g}^{b}\) using the data-generating process:
\[
y_{i g}^{b}=X_{i g} \widetilde{\beta}+\widetilde{\epsilon}_{i g} \nu_{g}^{b}
\]

2c. Fit model (1) using the bootstrap variable \(y_{i g}^{b}\) as regressand. Calculate the \(t\) statistic for \(\beta_{k}=0\),
\[
t_{k}^{b}=\frac{\widehat{\beta}_{k}^{b}}{\sqrt{\widehat{\mathbf{V}}_{k, k}^{b}}}
\]
where \(\widehat{\beta}_{k}^{b}\) and \(\widehat{\mathbf{V}}_{k, k}^{b}\) are the ordinary least-squares coefficient and the CRVE for that coefficient in the bootstrap replication, respectively. In this case, the CRVE is the \(k\) th diagonal element of matrix (2) obtained when using the residuals from bootstrap replication \(b\).
3. The \(p\)-values for the one-sided alternative hypotheses \(H_{1}: \beta_{k}>0\) and \(H_{2}: \beta_{k}<0\) are given by
\[
\begin{aligned}
& p_{1}=\frac{1}{B} \sum_{b=1}^{B} I\left(t_{k}^{b}>t_{k}\right) \\
& p_{2}=\frac{1}{B} \sum_{b=1}^{B} I\left(t_{k}^{b}<t_{k}\right)
\end{aligned}
\]

For the alternative hypothesis \(H_{3}: \beta_{k} \neq 0\), the \(p\)-value assuming that the distribution of the \(t\) statistic is symmetric around 0 is given by
\[
p_{S}=\frac{1}{B} \sum_{b=1}^{B} I\left(\left|t_{k}^{b}\right|>\left|t_{k}\right|\right)
\]

If the assumption of symmetry is not appropriate, then the \(p\)-value is given by the equal-tailed \(p\)-value \(p_{e}=2 \min \left(p_{1}, p_{2}\right)\). See Djogbenou, MacKinnon, and Nielsen (2019).

To increase computational speed, at the expense of computer memory usage, the wildbootstrap command uses the matrix algebra of Roodman et al. (2019). By doing this, the computational complexity (run time) of the WCB algorithm is reduced from the order \(O(N B)\) to the order \(O(G B)\). The wCB uses a \(G \times B\) matrix of random variables. By organizing these variables in column-major order, wildbootstrap reduces overall memory usage with minimal extra computation by breaking up the number of bootstrap replicates into blocks (the blocksize(\#) option).

\section*{Cls for linear combinations of coefficients}

By inverting hypotheses tests, we can apply the WCB algorithm to find CIs for any linear combination of coefficients. Suppose we wanted to compute a CI for the linear combination of parameters \(\mathbf{R} \beta\), where \(\mathbf{R}\) is a \(1 \times k\) vector. In this case, the associated null hypothesis is \(H_{0}: \mathbf{R} \beta=r\), where \(r\) is an arbitrary scalar. We can test this null hypothesis with the WCB algorithm described in the previous section by using the bootstrap \(t\) statistics
\[
t^{b}(r)=\frac{\mathbf{R} \widehat{\beta}^{b}-r}{\sqrt{\mathbf{R} \widehat{\mathbf{V}}^{b} \mathbf{R}^{\prime}}}
\]
for \(b=1, \ldots, B\). For more on restricted regression with linear constraints on multiple coefficients, see [P] makecns.

The associated one-sided alternative hypotheses are now \(H_{1}: \mathbf{R} \beta>r\) and \(H_{2}: \mathbf{R} \beta<r\). As before, their respective bootstrap \(p\)-values are given by
\[
\begin{aligned}
& p_{1}(r)=\frac{1}{B} \sum_{b=1}^{B} I\left\{t^{b}(r)>t(r)\right\} \\
& p_{2}(r)=\frac{1}{B} \sum_{b=1}^{B} I\left\{t^{b}(r)<t(r)\right\}
\end{aligned}
\]
where \(t(r)\) is the \(t\) statistic from the original sample:
\[
t(r)=\frac{\mathbf{R} \widehat{\beta}-r}{\sqrt{\mathbf{R} \widehat{\mathbf{V}} \mathbf{R}^{\prime}}}
\]

The associated two-sided alternative hypothesis is \(H_{3}: \mathbf{R} \beta \neq r\), and its bootstrap \(p\)-value under the assumption of symmetry is given by
\[
p_{S}(r)=\frac{1}{B} \sum_{b=1}^{B} I\left\{\left|t^{b}(r)\right|>|t(r)|\right\}
\]

When using the equal-tailed criterion (the ptype (equal) option), the \(100(1-\alpha) \%\) CI for \(\mathbf{R} \beta\) is given by \(\left[r_{l}, r_{u}\right]\), where \(r_{l}\) and \(r_{u}\) satisfy \(p_{1}\left(r_{l}\right)=p_{2}\left(r_{u}\right)=\alpha / 2\). On the other hand, when using the symmetric criterion (the ptype (symmetric) option), \(r_{l}\) and \(r_{u}\) satisfy \(p_{S}\left(r_{l}\right)=p_{S}\left(r_{u}\right)=\alpha\). The parameter \(\alpha\) can be specified with the level() option, that is, \(\alpha=(100-\mathrm{level}()) / 100\).

The bootstrap \(p\)-values are step functions on \(r\), and therefore a range of values for \(r_{l}\) and for \(r_{u}\) will solve the \(p\)-value conditions above. Because distribution functions are right continuous, wildbootstrap chooses the rightmost point in each range of solutions.

For the equal-tailed CIs, the starting values for the search of solutions to the \(p\)-value conditions are chosen as follows. First, the bootstrap \(t\) statistics are sorted \(t^{(1)} \leq t^{(2)} \leq \ldots<t^{(B)}\). Second, we define \(b_{l}=\operatorname{ceil}\{B(1-\alpha / 2)\}\) so that \(t^{\left(b_{l}\right)}\) is smaller than \(B(\alpha / 2)\) of the \(t\) statistics. Similarly, we define \(b_{u}=\) floor \(\{B(\alpha / 2)\}\) so that \(t^{\left(b_{u}\right)}\) is larger than \(B(\alpha / 2)\) of the \(t\) statistics. Third and finally, the initial lower and upper bounds of the CI are given by
\[
\begin{aligned}
& r_{l}=\mathbf{R} \widehat{\beta}-t^{\left(b_{l}\right)} \sqrt{\mathbf{R} \widehat{\mathbf{V}} \mathbf{R}^{\prime}} \\
& r_{u}=\mathbf{R} \widehat{\beta}-t^{\left(b_{u}\right)} \sqrt{\mathbf{R} \widehat{\mathbf{V}} \mathbf{R}^{\prime}}
\end{aligned}
\]
where \(\widehat{\beta}\) and \(\widehat{\mathbf{V}}\) are the unrestricted estimator for \(\beta\) and the unrestricted CRVE in the original sample.

\section*{Constructing a \(\mathbf{C l}\) inverting the hypothesis test}

For a linear combination of coefficients \(\sum R_{j} \beta_{j}\), we can construct a CI by inverting the hypothesis test \(H_{0}: \mathbf{R} \beta=r\), where \(\mathbf{R}\) is a row vector that includes the coefficients \(R_{j}\) and \(r\) is an arbitrary scalar. To do this, wildbootstrap searches for the CI lower bound \(r_{l}\) such that \(\operatorname{Pr}\left(\mathbf{R} \beta \leq r_{l}\right)=\alpha / 2\) (equal-tailed criterion, the ptype (equal) option). The WCB distribution of \(\mathbf{R} \beta\) is a step function with a step size of \(1 /\) reps (\#). When reps (\#) \(\times \alpha / 2\) is an integer, call it \(S\), then we search for the \(r_{l}\) that produces the ordered bootstrapped \(t\) statistics
\[
t^{(b)}=\frac{\mathbf{R} \widehat{\boldsymbol{\beta}}^{(b)}-r_{l}}{\sqrt{\mathbf{R}^{\prime} \mathbf{V}^{(b)} \mathbf{R}}}
\]
with the property \(\sum_{b=1}^{\text {reps }()} I\left(t^{(b)}>t\right)=S\) and therefore \(\operatorname{Pr}\left(\mathbf{R} \beta \leq r_{l}\right)=S /\) reps \((\#)=\alpha / 2\).

\section*{Acknowledgments}

We thank David Roodman of Open Philanthropy for his insights on the wild bootstrap computation.
We also acknowledge previous and ongoing contributions to this area from the Stata community. In particular, we acknowledge David Roodman, James G. MacKinnon, Morten Ørregaard Nielsen, and Matthew D. Webb for their command boottest and Mitchell Petersen, Douglas Miller, and Judson Caskey for their command cgmwildboot.

\section*{References}

Bell, R. M., and D. F. McCaffrey. 2002. Bias reduction in standard errors for linear regression with multi-stage samples. Survey Methodology 28: 169-181.
Cameron, A. C., J. B. Gelbach, and D. L. Miller. 2008. Bootstrap-based improvements for inference with clustered errors. Review of Economics and Statistics 90: 414-427. https://doi.org/10.1162/rest.90.3.414.

Cameron, A. C., and D. L. Miller. 2015. A practitioner's guide to cluster-robust inference. Journal of Human Resources 50: 317-372. https://doi.org/10.3368/jhr.50.2.317.
Djogbenou, A. A., J. G. MacKinnon, and M. Ø. Nielsen. 2019. Asymptotic theory and wild bootstrap inference with clustered errors. Journal of Econometrics 212: 393-412. https://doi.org/10.1016/j.jeconom.2019.04.035.

MacKinnon, J. G. 2019. How cluster-robust inference is changing applied econometrics. Canadian Journal of Economics 52: 851-881. https://doi.org/10.1111/caje. 12388.
MacKinnon, J. G., M. Ø. Nielsen, and M. D. Webb. 2023. Cluster-robust inference: A guide to empirical practice. Journal of Econometrics 232: 272-299. https://doi.org/10.1016/j.jeconom.2022.04.001.

MacKinnon, J. G., and M. D. Webb. 2017. Wild bootstrap inference for wildly different cluster sizes. Journal of Applied Econometrics 32: 233-254. https://doi.org/10.1002/jae.2508.
2018. The wild bootstrap for few (treated) clusters. Econometrics Journal 21: 114-135. https://doi.org/10.1111/ectj. 12107.

Roodman, D., J. G. MacKinnon, M. Ø. Nielsen, and M. D. Webb. 2019. Fast and wild: Bootstrap inference in Stata using boottest. Stata Journal 19: 4-60.
Wu, C. F. J. 1986. Jackknife, bootstrap and other resampling methods in regression analysis. Annals of Statistics 14: 1261-1350 (including discussions and rejoinder). https://doi.org/10.1214/aos/1176350142.

\section*{Also see}
[R] areg - Linear regression with many indicator variables \({ }^{+}\)
\([R]\) regress - Linear regression
[XT] xtreg - Linear models for panel data \({ }^{+}\)
[U] 20 Estimation and postestimation commands

\section*{Title}
xi - Interaction expansion
\begin{tabular}{llll} 
Description & Menu & Syntax & Options \\
Remarks and examples & Stored results & Also see &
\end{tabular}

\section*{Description}
xi expands terms containing categorical variables into indicator (also called dummy) variable sets by creating new variables and, in the second syntax (xi: any_stata_command), executes the specified command with the expanded terms. The dummy variables created are
\begin{tabular}{|c|c|}
\hline i. varname & creates dummies for categorical variable varname \\
\hline i. varname \(_{1} *\) i. varname \(_{2}\) & \begin{tabular}{l}
creates dummies for categorical variables varname \(_{1}\) and varname 2 : \\
all interactions and main effects
\end{tabular} \\
\hline i. varname \(_{1} *\) varname \(_{3}\) & \begin{tabular}{l}
creates dummies for categorical variable varname \(_{1}\) and continuous variable varname \(3_{3}\) : \\
all interactions and main effects
\end{tabular} \\
\hline i. varname \(_{1}\) | varname \(_{3}\) & \begin{tabular}{l}
creates dummies for categorical variable varname \({ }_{1}\) and continuous variable varname \(e_{3}\) : \\
all interactions and main effect of varname \({ }_{3}\), but no main effect of varname \(_{1}\)
\end{tabular} \\
\hline
\end{tabular}

\section*{Menu}

Data \(>\) Create or change data \(>\) Other variable-creation commands \(>\) Interaction expansion

Most commands in Stata now allow factor variables; see [U] 11.4.3 Factor variables. To determine if a command allows factor variables, see the information printed below the options table for the command. If the command allows factor variables, it will say something like "indepvars may contain factor variables".

We recommend that you use factor variables instead of xi if a command allows factor variables. We include \([R]\) xi in our documentation so that readers can consult it when using a Stata command that does not allow factor variables.

\section*{Syntax}
```

xi [, prefix(string) noomit] term(s)
xi [, prefix(string) noomit] : any_stata_command varlist_with_terms ...

```
where a term has the form
\begin{tabular}{|c|c|c|}
\hline i. varname & or & I. varname \\
\hline i. varname \(_{1}\) *i. varname \(_{2}\) & & I. varname \({ }_{1} *\) I. varname \\
\hline i. varname \(_{1} *\) varname \(_{3}\) & & I. varname \({ }_{1} *\) varname \(_{3}\) \\
\hline i. varname \({ }_{1}\) | varname \(_{3}\) & & I. varname \({ }_{1}\) | varname \(^{2}\) \\
\hline
\end{tabular}
varname, varname \(_{1}\), and varname \({ }_{2}\) denote numeric or string categorical variables. varname \({ }_{3}\) denotes a continuous, numeric variable.

\section*{Options}
prefix (string) allows you to choose a prefix other than _I for the newly created interaction variables. The prefix cannot be longer than four characters. By default, xi will create interaction variables starting with _I. When you use xi, it drops all previously created interaction variables starting with the prefix specified in the prefix (string) option or with _I by default. Therefore, if you want to keep the variables with a certain prefix, specify a different prefix in the prefix (string) option.
noomit prevents xi from omitting groups. This option provides a way to generate an indicator variable for every category having one or more variables, which is useful when combined with the noconstant option of an estimation command.

\section*{Remarks and examples}

Remarks are presented under the following headings:
```

Background
Indicator variables for simple effects
Controlling the omitted dummy
Categorical variable interactions
Interactions with continuous variables
Using xi: Interpreting output
How xi names variables
xi as a command rather than a command prefix
Warnings

```
xi provides a convenient way to include dummy or indicator variables when fitting a model (say, with regress or logistic). For instance, assume that the categorical variable agegrp contains 1 for ages 20-24, 2 for ages 25-39, 3 for ages 40-44, etc. Typing
```

. xi: logistic outcome weight i.agegrp bp

```
estimates a logistic regression of outcome on weight, dummies for each agegrp category, and bp. That is, xi searches out and expands terms starting with "i." or "I." but ignores the other variables. xi will expand both numeric and string categorical variables, so if you had a string variable race containing "white", "black", and "other", typing
```

. xi: logistic outcome weight bp i.agegrp i.race

```
would include indicator variables for the race group as well.

The i. indicator variables xi expands may appear anywhere in the varlist, so
. xi: logistic outcome i.agegrp weight i.race bp
would fit the same model.
You can also create interactions of categorical variables; typing
```

xi: logistic outcome weight bp i.agegrp*i.race

```
fits a model with indicator variables for all agegrp and race combinations, including the agegrp and race main-effect terms (that is, the terms that are created when you just type i.agegrp i.race).

You can interact dummy variables with continuous variables; typing
xi: logistic outcome bp i.agegrp*weight i.race
fits a model with indicator variables for all agegrp categories interacted with weight, plus the main-effect terms weight and i. agegrp.

You can get the interaction terms without the agegrp main effect (but with the weight main effect) by typing
```

xi: logistic outcome bp i.agegrp|weight i.race

```

You can also include multiple interactions:
xi: logistic outcome bp i.agegrp*weight i.agegrp*i.race
We will now back up and describe the construction of dummy variables in more detail.

\section*{Background}

The terms continuous, categorical, and indicator or dummy variables are used below. Continuous variables measure something - such as height or weight - and at least conceptually can take on any real number over some range. Categorical variables, on the other hand, take on a finite number of values, each denoting membership in a subclass-for example, excellent, good, and poor, which might be coded 0, 1, 2, or 1, 2, 3, or even "Excellent", "Good", and "Poor". An indicator or dummy variable-the terms are used interchangeably-is a special type of two-valued categorical variable that contains values 0 , denoting false, and 1 , denoting true. The information contained in any \(k\)-valued categorical variable can be equally well represented by \(k\) indicator variables. Instead of one variable recording values representing excellent, good, and poor, you can have three indicator variables, indicating the truth or falseness of "result is excellent", "result is good", and "result is poor".
xi provides a convenient way to convert categorical variables to dummy or indicator variables when you fit a model (say, with regress or logistic).

\section*{> Example 1}

For instance, assume that the categorical variable agegrp contains 1 for ages \(20-24,2\) for ages \(25-39\), and 3 for ages \(40-44\). (There is no one over 44 in our data.) As it stands, agegrp would be a poor candidate for inclusion in a model even if we thought age affected the outcome. The reason is that the coding would restrict the effect of being in the second age group to be twice the effect of being in the first, and, similarly, the effect of being in the third to be three times the first. That is, if we fit the model,
\[
y=\beta_{0}+\beta_{1} \operatorname{agegrp}+X \beta_{2}
\]
the effect of being in the first age group is \(\beta_{1}\), the second \(2 \beta_{1}\), and the third \(3 \beta_{1}\). If the coding 1,2 , and 3 is arbitrary, we could just as well have coded the age groups 1,4 , and 9 , making the effects \(\beta_{1}, 4 \beta_{1}\), and \(9 \beta_{1}\).

The solution is to convert the categorical variable agegrp to a set of indicator variables, \(a_{1}, a_{2}\), and \(a_{3}\), where \(a_{i}\) is 1 if the individual is a member of the \(i\) th age group and 0 otherwise. We can then fit the model
\[
y=\beta_{0}+\beta_{11} a_{1}+\beta_{12} a_{2}+\beta_{13} a_{3}+X \beta_{2}
\]

The effect of being in age group 1 is now \(\beta_{11} ; 2, \beta_{12}\); and \(3, \beta_{13}\); and these results are independent of our (arbitrary) coding. The only difficulty at this point is that the model is unidentified in the sense that there are an infinite number of \(\left(\beta_{0}, \beta_{11}, \beta_{12}, \beta_{13}\right)\) that fit the data equally well.

To see this, pretend that \(\left(\beta_{0}, \beta_{11}, \beta_{12}, \beta_{13}\right)=(1,1,3,4)\). The predicted values of \(y\) for the various age groups are
\[
y= \begin{cases}1+1+X \beta_{2}=2+X \beta_{2} & (\text { age group 1) } \\ 1+3+X \beta_{2}=4+X \beta_{2} & (\text { age group } 2) \\ 1+4+X \beta_{2}=5+X \beta_{2} & (\text { age group } 3)\end{cases}
\]

Now, pretend that \(\left(\beta_{0}, \beta_{11}, \beta_{12}, \beta_{13}\right)=(2,0,2,3)\). The predicted values of \(y\) are
\[
y=\left\{\begin{array}{lc}
2+0+X \beta_{2}=2+X \beta_{2} & (\text { age group 1) } \\
2+2+X \beta_{2}=4+X \beta_{2} & (\text { age group 2) } \\
2+3+X \beta_{2}=5+X \beta_{2} & (\text { age group 3) }
\end{array}\right.
\]

These two sets of predictions are indistinguishable: for age group \(1, y=2+X \beta_{2}\) regardless of the coefficient vector used, and similarly for age groups 2 and 3 . This arises because we have three equations and four unknowns. Any solution is as good as any other, and, for our purposes, we merely need to choose one of them. The popular selection method is to set the coefficient on the first indicator variable to 0 (as we have done in our second coefficient vector). This is equivalent to fitting the model
\[
y=\beta_{0}+\beta_{12} a_{2}+\beta_{13} a_{3}+X \beta_{2}
\]

How we select a particular coefficient vector (identifies the model) does not matter. It does, however, affect the interpretation of the coefficients.

For instance, we could just as well choose to omit the second group. In our artificial example, this would yield \(\left(\beta_{0}, \beta_{11}, \beta_{12}, \beta_{13}\right)=(4,-2,0,1)\) instead of \((2,0,2,3)\). These coefficient vectors are the same in the sense that
\[
y= \begin{cases}2+0+X \beta_{2}=2+X \beta_{2}=4-2+X \beta_{2} & \text { (age group 1) } \\ 2+2+X \beta_{2}=4+X \beta_{2}=4+0+X \beta_{2} & \text { (age group 2) } \\ 2+3+X \beta_{2}=5+X \beta_{2}=4+1+X \beta_{2} & \text { (age group 3) }\end{cases}
\]

But what does it mean that \(\beta_{13}\) can just as well be 3 or 1 ? We obtain \(\beta_{13}=3\) when we set \(\beta_{11}=0\), so \(\beta_{13}=\beta_{13}-\beta_{11}\) and \(\beta_{13}\) measures the difference between age groups 3 and 1 .

In the second case, we obtain \(\beta_{13}=1\) when we set \(\beta_{12}=0\), so \(\beta_{13}-\beta_{12}=1\) and \(\beta_{13}\) measures the difference between age groups 3 and 2 . There is no inconsistency. According to our \(\beta_{12}=0\) model, the difference between age groups 3 and 1 is \(\beta_{13}-\beta_{11}=1-(-2)=3\), the same result we got in the \(\beta_{11}=0\) model.
> Example 2
The issue of interpretation is important because it can affect the way we discuss results. Imagine that we are studying recovery after a coronary bypass operation. Assume that the age groups are children under 13 (we have two of them), young adults under 25 (we have a handful of them), adults under 46 (of which we have even more), mature adults under 56 , older adults under 65 , and elderly adults. We follow the prescription of omitting the first group, so all our results are reported relative to children under 13. While there is nothing statistically wrong with this, readers will be suspicious when we make statements like "compared with young children, older and elder adults . . .". Moreover, we will probably have to end each statement with "although results are not statistically significant" because we have only two children in our comparison group. Of course, even with results reported in this way, we can do reasonable comparisons (say, with mature adults), but we will have to do extra work to perform the appropriate linear hypothesis test using Stata's test command.

Here it would be better to force the omitted group to be more reasonable, such as mature adults. There is, however, a generic rule for automatic comparison group selection that, although less popular, tends to work better than the omit-the-first-group rule. That rule is to omit the most prevalent group. The most prevalent is usually a reasonable baseline.

In any case, the prescription for categorical variables is
1. Convert each \(k\)-valued categorical variable to \(k\) indicator variables.
2. Drop one of the \(k\) indicator variables; any one will do, but dropping the first is popular, dropping the most prevalent is probably better in terms of having the computer guess at a reasonable interpretation, and dropping a specified one often eases interpretation the most.
3. Fit the model on the remaining \(k-1\) indicator variables.
xi automates this procedure.
We will now consider each of xi's features in detail.

\section*{Indicator variables for simple effects}

When you type i.varname, xi internally tabulates varname (which may be a string or a numeric variable) and creates indicator (dummy) variables for each observed value, omitting the indicator for the smallest value. For instance, say that agegrp takes on the values 1, 2, 3, and 4. Typing
```

xi: logistic outcome i.agegrp

```
creates indicator variables named _Iagegrp_2, _Iagegrp_3, and _Iagegrp_4. (xi chooses the names and tries to make them readable; xi guarantees that the names are unique.) The expanded logistic model is
```

. logistic outcome _Iagegrp_2 _Iagegrp_3 _Iagegrp_4

```

Afterward, you can drop the new variables xi leaves behind by typing 'drop _I*' (note the capitalization).
xi provides the following features when you type i.varname:
- varname may be string or numeric.
- Dummy variables are created automatically.
- By default, the dummy-variable set is identified by dropping the dummy corresponding to the smallest value of the variable (how to specify otherwise is discussed below).
- The new dummy variables are left in your dataset. By default, the names of the new dummy variables start with _I; therefore, you can drop them by typing 'drop _I*'. You do not have to do this; each time you use xi, any automatically generated dummies with the same prefix as the one specified in the prefix (string) option, or _I by default, are dropped and new ones are created.
- The new dummy variables have variable labels so that you can determine what they correspond to by typing 'describe'.
- xi may be used with any Stata command (not just logistic).

\section*{Controlling the omitted dummy}

By default, i.varname omits the dummy corresponding to the smallest value of varname; for a string variable, this is interpreted as dropping the first in an alphabetical, case-sensitive sort. xi provides two alternatives to dropping the first: xi will drop the dummy corresponding to the most prevalent value of varname, or xi will let you choose the particular dummy to be dropped.

To change xi's behavior to dropping the most prevalent dummy, type
```

. char _dta[omit] prevalent

```
although whether you type "prevalent" or "yes" or anything else does not matter. Setting this characteristic affects the expansion of all categorical variables in the dataset. If you resave your dataset, the prevalent preference will be remembered. If you want to change the behavior back to the default drop-the-first rule, type
```

. char _dta[omit]

```
to clear the characteristic.
Once you set _dta[omit], i.varname omits the dummy corresponding to the most prevalent value of varname. Thus, the coefficients on the dummies have the interpretation of change from the most prevalent group. For example,
. char _dta[omit] prevalent
. xi: regress y i.agegrp
might create _Iagegrp_1 through _Iagegrp_4, resulting in _Iagegrp_2 being omitted if agegrp \(=\) 2 is most common (as opposed to the default dropping of _Iagegrp_1). The model is then
\[
\mathrm{y}=b_{0}+b_{1} \text { _Iagegrp_1 }+b_{3} \text { _Iagegrp_3 }+b_{4} \text { _Iagegrp_ } 4+u
\]

Then,
\[
\begin{array}{ll}
\text { Predicted y for agegrp } 1=b_{0}+b_{1} & \text { Predicted y for agegrp } 3=b_{0}+b_{3} \\
\text { Predicted y for agegrp } 2=b_{0} & \text { Predicted y for agegrp } 4=b_{0}+b_{4}
\end{array}
\]

Thus, the model's reported \(t\) or \(Z\) statistics are for a test of whether each group is different from the most prevalent group.

Perhaps you wish to omit the dummy for agegrp 3 instead. You do this by setting the variable's omit characteristic:

This overrides _dta[omit] if you have set it. Now, when you type
```

. xi: regress y i.agegrp

```
_Iagegrp_3 will be omitted, and you will fit the model
\[
\mathrm{y}=b_{0}^{\prime}+b_{1}^{\prime} \text { _Iagegrp_1 }+b_{2}^{\prime} \text { _Iagegrp_2 }+b_{4}^{\prime} \text { _Iagegrp_ } 4+u
\]

Later, if you want to return to the default omission, type
```

. char agegrp[omit]

```
to clear the characteristic.
In summary, i.varname omits the first group by default, but if you define
. char _dta[omit] prevalent
the default behavior changes to dropping the most prevalent group. Either way, if you define a characteristic of the form
```

. char varname[omit] \#

```
or, if varname is a string,
- char varname[omit] string-literal
the specified value will be omitted.
\[
\begin{array}{lll}
\text { Examples: } & \text {. char agegrp[omit] } 1 & \\
& \text {. char race[omit] White } & \\
& \text { (for race, a string variable) } \\
& \text {. char agegrp[omit] } & \text { (to restore default for agegr }
\end{array}
\]

\section*{Categorical variable interactions}
i. varname \(_{1} *\) i. varname \(_{2}\) creates the dummy variables associated with the interaction of the categorical variables varname \(_{1}\) and varname \(_{2}\). The identification rules - which categories are omitted - are the same as those for i.varname. For instance, assume that agegrp takes on four values and race takes on three values. Typing
```

. xi: regress y i.agegrp*i.race

```
results in
model :
\[
\begin{array}{rlrl}
y= & a & +b_{2} \text { _Iagegrp_2 }+b_{3} \text { _Iagegrp_3 }+b_{4} \text { _Iagegrp_4 } & \\
& & \text { (agegrp) } \\
& +c_{2} \text {-Irace_2 }+c_{3} \_ \text {Irace_3 } & & \text { (race) } \\
& +d_{22} \text { _IageXrac_2_2 }+d_{23} \text {-IageXrac_2_3 } & \\
& +d_{32} \text { _IageXrac_3_2 }+d_{33} \text { _IageXrac_3_3 } & & \text { (agegrp* } \\
& +d_{42} \text {-IageXrac_4_2 }+d_{43} \text { _IageXrac_4_3 } & \\
& +u & &
\end{array}
\]

That is, typing
```

. xi: regress y i.agegrp*i.race

```
is the same as typing
```

. xi: regress y i.agegrp i.race i.agegrp*i.race

```

Although there are many other ways the interaction could have been parameterized, this method has the advantage that you can test the joint significance of the interactions by typing
```

. testparm _IageXrac*

```

When you perform the estimation step, whether you specify i.agegrp*i.race or i.race*i.agegrp makes no difference (other than in the names given to the interaction terms; in the first case, the names will begin with _IageXrac; in the second, _IracXage). Thus,
```

. xi: regress y i.race*i.agegrp

```
fits the same model.
You may also include multiple interactions simultaneously:
```

. xi: regress y i.agegrp*i.race i.agegrp*i.sex

```

The model fit is
model:
```

$y=a+b_{2}$ _Iagegrp_2 $+b_{3}$ _Iagegrp_3 $+b_{4}$ _Iagegrp_4
$+c_{2}$ _Irace_ $2+c_{3}$ _Irace_3
$+d_{22}$ _IageXrac_2_2 $+d_{23}$ _IageXrac_2_3
$+d_{32}$ _IageXrac_3_2 $+d_{33}$ _IageXrac_3_3
$+d_{42}$ _IageXrac_4_2 $+d_{43}$ _IageXrac_4_3
$+e_{2}$ _Isex_2
$+f_{22}$ _IageXsex_2_2 $+f_{23}$ _IageXsex_2_3 $+f_{24}$ _IageXsex_2_4
$+u$

```

The agegrp dummies are (correctly) included only once.

\section*{Interactions with continuous variables}
i. varname \({ }_{1}\) varname \(_{2}\) (as distinguished from i.varname \({ }_{1} *\) i. varname \({ }_{2}\) —note the second i.) specifies an interaction of a categorical variable with a continuous variable. For instance,
. xi: regress y i.agegr*wgt
results in the model
\[
\begin{aligned}
y=a & +b_{2} \text { _Iagegrp_2 }+b_{3} \_ \text {Iagegrp_3 }+b_{4} \_ \text {Iagegrp_4 } & & \text { (agegrp dummies) } \\
& +c \text { wgt } & & \text { (continuous wgt effect) } \\
& +d_{2} \text { _IageXwgt_2 }+d_{3} \text { _IageXwgt_3 }+d_{4} \_ \text {IageXwgt_4 } & & \text { (agegrp*wgt interactions) } \\
& +u & &
\end{aligned}
\]

A variation on this notation, using \(\mid\) rather than \(*\), omits the agegrp dummies. Typing . xi: regress y i.agegrplwgt
fits the model
\[
\begin{aligned}
y=a^{\prime} & +c^{\prime} \text { wgt } & & \text { (continuous wgt effect) } \\
& +d_{2}^{\prime} \_ \text {IageXwgt_2 }+d_{3}^{\prime} \_ \text {IageXwgt_3 }+d_{4}^{\prime} \_ \text {IageXwgt_4 } & & \text { (agegrp*wgt interactions) } \\
& +u^{\prime} & &
\end{aligned}
\]

The predicted values of y are
\begin{tabular}{cll} 
agegrp*wgt model & agegrp \(\mid\) wgt model & \\
\hline\(y=a+c\) wgt & \(a^{\prime}+c^{\prime}\) wgt & if agegrp \(=1\) \\
\(a+c\) wgt \(+b_{2}+d_{2}\) wgt & \(a^{\prime}+c^{\prime}\) wgt \(+d_{2}^{\prime}\) wgt & if agegrp \(=2\) \\
\(a+c\) wgt \(+b_{3}+d_{3} \mathrm{wgt}\) & \(a^{\prime}+c^{\prime}\) wgt \(+d_{3}^{\prime} \mathrm{wgt}\) & if agegrp \(=3\) \\
\(a+c\) wgt \(+b_{4}+d_{4} \mathrm{wgt}\) & \(a^{\prime}+c^{\prime}\) wgt \(+d_{4}^{\prime} \mathrm{wgt}\) & if agegrp \(=4\)
\end{tabular}

That is, typing
. xi: regress y i.agegrp*wgt
is equivalent to typing
```

. xi: regress y i.agegrp i.agegrplwgt

```

In either case, you do not need to specify separately the continuous variable wgt; it is included automatically.

\section*{Using xi: Interpreting output}
```

. xi: regress mpg i.rep78
i.rep78 _Irep78_1-5 (naturally coded; _Irep78_1 omitted)
(output from regress appears)

```

Interpretation: i.rep78 expanded to the dummies _Irep78_1, _Irep78_2, ..., _Irep78_5. The numbers on the end are "natural" in the sense that _Irep78_1 corresponds to rep78 = 1, _Irep78_2 to \(\operatorname{rep} 78=2\), and so on. Finally, the dummy for rep78 \(=1\) was omitted.
```

. xi: regress mpg i.make
i.make _Imake_1-74 (_Imake_1 for make==AMC Concord omitted)
(output from regress appears)

```

Interpretation: i.make expanded to _Imake_1, _Imake_2, ..., _Imake_74. The coding is not natural because make is a string variable. _Imake_1 corresponds to one make, _Imake_2 to another, and so on. You can find out the coding by typing describe. _Imake_1 for the AMC Concord was omitted.

\section*{How xi names variables}

By default, xi assigns to the dummy variables it creates names having the form
_Istub_groupid

You may subsequently refer to the entire set of variables by typing 'Istub*'. For example,
\begin{tabular}{lllll} 
name & \(=\) _I + stub & \(+\ldots+\) groupid & Entire set \\
\hline -Iagegrp_1 & _I & agegrp & -1 & -Iagegrp* \\
_Iagegrp_2 & -I agegrp & -2 & -Iagegrp* \\
-IageXwgt_1 & -I ageXwgt & -1 & -IageXwgt* \\
_IageXrac_1_2 & -I ageXrac & \(-1 \_2\) & -IageXrac* \\
_IageXrac_2_1 & _I ageXrac & \(-2 \_1\) & _IageXrac*
\end{tabular}

If you specify a prefix in the prefix(string) option, say, \(\quad\) S, then xi will name the variables starting with the prefix

\section*{xi as a command rather than a command prefix}
xi can be used as a command prefix or as a command by itself. In the latter form, xi merely creates the indicator and interaction variables. Typing
```

. xi: regress y i.agegrp*wgt
i.agegrp _Iagegrp_1-4 (naturally coded; _Iagegrp_1 omitted)
i.agegrp*wgt _IageXwgt_1-4 (coded as above)
(output from regress appears)

```
is equivalent to typing
```

. xi i.agegrp*wgt
i.agegrp _Iagegrp_1-4 (naturally coded; _Iagegrp_1 omitted)
i.agegrp*wgt _IageXwgt_1-4 (coded as above)
. regress y _Iagegrp* _IageXwgt*
(output from regress appears)

```

\section*{Warnings}
1. xi creates new variables in your dataset; most are bytes, but interactions with continuous variables will have the storage type of the underlying continuous variable. You may get the error message "no room to add more variables" or "insufficient memory". You may need to adjust the maxvar setting or reset max_memory if it has been set too low; see [U] 6 Managing memory.
2. When using xi with an estimation command, you may get the error message "unable to allocate matrix". This usually occurs because you attempted to create a matrix that is too large; see \([R]\) Limits.

\section*{Stored results}
xi stores the following characteristics:
```

_dta[__xi__Vars__Prefix__] prefix names
_dta[__xi__Vars__To__Drop__] variables created

```

\section*{Also see}
[U] 11.1.10 Prefix commands
[U] 20 Estimation and postestimation commands

\section*{Title}
zinb - Zero-inflated negative binomial regression
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
zinb fits a zero-inflated negative binomial (ZINB) model to overdispersed count data with excess zero counts. The ZINB model assumes that the excess zero counts come from a logit or probit model and the remaining counts come from a negative binomial model.

\section*{Quick start}

Zero-inflated negative binomial model of y on x 1 and x 2 with inflation modeled using x 3 zinb y x1 x2, inflate(x3)

And conduct likelihood-ratio test against ZIP model
zinb y x1 x2, inflate(x3) zip
Use a probit model instead of a logit model to predict excess zeros
zinb y x1 x2, inflate(x3) probit

\section*{Menu}

Statistics \(>\) Count outcomes \(>\) Zero-inflated negative binomial regression

\section*{Syntax}
\[
\begin{aligned}
& \text { zinb depvar }[\text { indepvars }][\text { if }][\text { in }][\text { weight }], \\
& \text { inflate(varlist }[, \underline{\text { offset }(\text { varname })] \mid \text { _cons) }[\text { options }]}
\end{aligned}
\]
options
Model
```

*inflate()
noconstant
exposure(\mp@subsup{varname e}{e}{})
offset(\mp@subsup{varname o}{o}{)}
constraints(constraints)
probit

```

SE/Robust
vce (vcetype)

Reporting
level(\#)
irr
zip
nocnsreport
display_options

Description
equation that determines whether the count is zero suppress constant term
include \(\ln \left(\right.\) varname \(\left._{e}\right)\) in model with coefficient constrained to 1
include varname \(_{o}\) in model with coefficient constrained to 1
apply specified linear constraints
use probit model to characterize excess zeros; default is logit
vcetype may be oim, robust, cluster clustvar, opg, bootstrap, or jackknife
set confidence level; default is level (95)
report incidence-rate ratios
perform ZIP likelihood-ratio test
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling

\section*{Maximization}
maximize_options
collinear
coeflegend
control the maximization process; seldom used
keep collinear variables
display legend instead of statistics
*inflate (varlist \([\), offset(varname) \(] \mid\) _cons) is required.
indepvars and varlist may contain factor variables; see [U] 11.4.3 Factor variables.
bayes, bootstrap, by, collect, fp, jackknife, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: zinb.

Weights are not allowed with the bootstrap prefix; see \([R]\) bootstrap.
vce(), zip, and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options}
\(\qquad\) Model
inflate(varlist \([\), offset(varname) \(] \mid\) _cons) specifies the equation that determines whether the observed count is zero. Conceptually, omitting inflate() would be equivalent to fitting the model with nbreg.
inflate (varlist [, offset (varname) \(]\) ) specifies the variables in the equation. You may optionally include an offset for this varlist.
inflate(_cons) specifies that the equation determining whether the count is zero contains only an intercept. To run a zero-inflated model of depvar with only an intercept in both equations, type zinb depvar, inflate(_cons).
noconstant, exposure (varname ), offset ( varname \(_{o}\) ), constraints (constraints); see [R] Estimation options.
probit requests that a probit, instead of logit, model be used to characterize the excess zeros in the data.

\section*{SE/Robust}
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.

\section*{Reporting}
level(\#); see [R] Estimation options.
irr reports estimated coefficients transformed to incidence-rate ratios, that is, \(e^{\beta_{i}}\) rather than \(\beta_{i}\). Standard errors and confidence intervals are similarly transformed. This option affects how results are displayed, not how they are estimated or stored. irr may be specified at estimation or when replaying previously estimated results.
zip requests that a likelihood-ratio test comparing the ZINB model with the zero-inflated Poisson model be included in the output.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate(\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from (init_specs); see [R] Maximize. These options are seldom used.
Setting the optimization type to technique (bhhh) resets the default vcetype to vce (opg).
The following options are available with zinb but are not shown in the dialog box:
collinear, coeflegend; see [R] Estimation options.

\section*{Remarks and examples}

Zero-inflated negative binomial (ZINB) models are used to model count data that have a higher fraction of zeros than is likely to be generated by a standard negative binomial model. To account for excess zeros, ZINB models assume that these excess zeros come from a model other than the negative binomial model. A zero that comes from this other model is known as a "degenerate zero".

The negative binomial overdispersion parameter, \(\alpha\), differentiates the ZINB model from the zeroinflated Poisson (ZIP) model (see [R] zip). Here overdispersion refers to the fact that the negative binomial variance is greater than its mean, whereas the Poisson variance is equal to its mean. Thus, values of \(\alpha>1\) indicate overdispersion. The larger the \(\alpha\), the greater the negative binomial variance. See Methods and formulas in [R] nbreg for further discussion of negative binomial overdispersion.

The zinb command fits ZINB models and provides two choices for modeling the excess zeros: the default logit function or, when the probit option is specified, the probit function. Both functions are symmetric about zero, but the logistic function has more area under the tails.

See Long (1997, 242-247) and Cameron and Trivedi (2005, 680-681) for a discussion of zeromodified count models.

\section*{> Example 1: Fitting a ZINB model}

In example 1 of [R] zip, we fit a zero-inflated Poisson model using the zip command to the fictional data on the number of fish caught by visitors to a national park. Let's fit a ZINB model to these data.

Just like with zip, we use the required option inflate() to model whether a visitor fishes as a function of the number of accompanying children (child) and whether the visitor is camping (camper). Next, we assume the response variable, count, depends on whether the visitor used a live bait (livebait) and the number of persons in the party (persons), which includes the visitor plus other adults and children.
```

. use https://www.stata-press.com/data/r18/fish
(Fictional fishing data)
. zinb count persons livebait, inflate(child camper)
Fitting constant-only model:
Iteration 0: Log likelihood = -519.33992
Iteration 1: Log likelihood = -451.38662
Iteration 2: Log likelihood = -444.49118
Iteration 3: Log likelihood = -442.96272
Iteration 4: Log likelihood = -442.71065
Iteration 5: Log likelihood = -442.66718
Iteration 6: Log likelihood = -442.6631
Iteration 7: Log likelihood = -442.66299
Iteration 8: Log likelihood = -442.66299
Fitting full model:
Iteration 0: Log likelihood = -442.66299 (not concave)
Iteration 1: Log likelihood = -432.83107 (not concave)
Iteration 2: Log likelihood = -426.32934
Iteration 3: Log likelihood = -413.75019
Iteration 4: Log likelihood = -403.09586
Iteration 5: Log likelihood = -401.56013
Iteration 6: Log likelihood = -401.54781
Iteration 7: Log likelihood = -401.54776
Iteration 8: Log likelihood = -401.54776

```
Zero-inflated negative binomial regression \(\quad\) Number of obs \(=250\)
Inflation model: logit Nonzero obs = 108
    Zero obs \(=142\)
    LR chi2 \((2)=82.23\)
Log likelihood \(=-401.5478\)
\begin{tabular}{llr} 
Number of obs & \(=\) & 250 \\
Nonzero obs & \(=\) & 108 \\
Zero obs & \(=\) & 142 \\
LR chi2 \((2)\) & \(=\) & 82.23 \\
Prob \(>\) chi2 & \(=0.0000\)
\end{tabular}
\begin{tabular}{r|rrrrrr}
\hline count & Coefficient & Std. err. & z & P>|z| & [95\% conf. interval] \\
\hline count & & & & & & \\
persons \\
livebait \\
_cons & .9742984 & .1034938 & 9.41 & 0.000 & .7714543 & 1.177142 \\
\hline inflate & -2.730064 & .476953 & -5.72 & 0.000 & -3.664874 & -1.795253 \\
child & 3.185999 & .7468551 & 4.27 & 0.000 & 1.72219 & 4.649808 \\
camper & -2.020951 & .872054 & -2.32 & 0.020 & -3.730146 & -.3117567 \\
_cons & -2.695385 & .8929071 & -3.02 & 0.003 & -4.44545 & -.9453189 \\
\hline /lnalpha & .5110429 & .1816816 & 2.81 & 0.005 & .1549535 & .8671323 \\
\hline alpha & 1.667029 & .3028685 & & & 1.167604 & 2.380076 \\
\hline
\end{tabular}

The coefficients in the first equation of the coefficient table, labeled count, correspond to the negative binomial model for individuals who fished. For instance, among visitors who fished, using a live bait increases the expected number of caught fish by a factor of \(\exp (1.5575) \approx 4.7\), holding other covariates constant.

The confidence interval for alpha indicates that the ZINB model is more appropriate than the ZIP model. To confirm this, you can run zinb and specify the zip option to obtain the ZIP likelihood-ratio test.

The inflate equation models whether the visitor does not fish. We can use margins to obtain a better understanding of how the inflate equation affects the occurrence of the excess zero counts. We specify margins's options dydx (child camper) and predict(pr). pr is predict's option
for estimating the probability of a degenerate zero or, in our example, the probability of not fishing; see the margins section in \([R]\) zinb postestimation.
```

. margins, dydx(child camper) predict(pr)
Average marginal effects Number of obs = 250
Model VCE: OIM
Expression: Pr(count=0), predict(pr)
dy/dx wrt: child camper

```
\begin{tabular}{r|rrrrrrr}
\hline & \multicolumn{4}{|c}{ Delta-method } \\
& dy/dx & std. err. & z & P>|z| & [95\% conf. interval] \\
\hline child & .257531 & .029941 & 8.60 & 0.000 & .1988477 & .3162144 \\
camper & -.1633578 & .0503938 & -3.24 & 0.001 & -.2621277 & -.0645878 \\
\hline
\end{tabular}

The margins output tells us that a visitor is less likely to be visiting the park to fish if accompanied by children and more likely to fish if camping.

You also may want to evaluate whether a standard negative binomial model is adequate to fit the data. This can be done using information criteria; see example 2 in [R] zip.

\section*{Stored results}
zinb stores the following in e():
Scalars
e(N)
e(N_zero)
e(k)
\(e\left(k \_\right.\)eq)
e(k_eq_model)
e(k_aux)
e(k_dv)
e(df_m)
e(11)
e(11_0)
e(df_c)
e(N_clust)
e(chi2)
e(p)
e(chi2_cp)
e(rank)
e(ic)
\(e(r c)\)
e(converged)
Macros
e(cmd)
e(cmdline)
e(depvar)
e(inflate)
\(e\) (wtype)
e(wexp)
e(title)
e(clustvar)
e(offset1)
e(offset2)
e(chi2type)
number of observations
number of zero observations
number of parameters
number of equations in \(e(b)\)
number of equations in overall model test
number of auxiliary parameters
number of dependent variables
model degrees of freedom
log likelihood
log likelihood, constant-only model
degrees of freedom for comparison test
number of clusters
\(\chi^{2}\)
\(p\)-value for model test
\(\chi^{2}\) for test of \(\alpha=0\)
rank of e(V)
number of iterations
return code
1 if converged, 0 otherwise
zinb
command as typed
name of dependent variable
logit or probit
weight type
weight expression
title in estimation output
name of cluster variable
offset
offset for inflate()
Wald or LR; type of model \(\chi^{2}\) test
```

e(chi2_cpt)
e(vce)
e(vcetype)
e(opt)
e(which)
e(ml_method)
e(user)
e(technique)
e(properties)
e(predict)
e(asbalanced)
e(asobserved)
e(b)
e(Cns)
e(ilog)
e(gradient)
e(V)
e(V_modelbased)
e(sample) marks estimation sample

```

Wald or LR; type of model \(\chi^{2}\) test corresponding to e(chi2_cp)
vcetype specified in vce()
title used to label Std. err.
type of optimization
max or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement predict
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
constraints matrix
iteration \(\log\) (up to 20 iterations)
gradient vector
variance-covariance matrix of the estimators
model-based variance
marks estimation sample
Matrices
Functions

In addition to the above, the following is stored in r() :

Matrices \(r\) (table)
matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

\section*{Methods and formulas}

The zinb command maximizes a likelihood function that is a mixture of the logistic (or probit) and negative binomial distributions. The logistic distribution models the unobserved process that creates the excess zeros, and the negative binomial distribution models the counts. Define
\[
\begin{aligned}
\xi_{j}^{\beta} & =\mathbf{x}_{j} \boldsymbol{\beta}+\operatorname{offset}_{j}^{\beta} \\
\xi_{j}^{\gamma} & =\mathbf{z}_{j} \gamma+\operatorname{offset}_{j}^{\gamma} \\
\mu_{j} & =\exp \left(\xi_{j}^{\beta}\right) \\
p_{j} & =1 /\left(1+\alpha \mu_{j}\right) \\
m & =1 / \alpha
\end{aligned}
\]

Here the vector \(\mathbf{x}_{j}\) contains the covariates specified in indepvars for the \(j\) th observation, and \(\mathbf{z}_{j}\) contains the covariates specified in the inflate() option. Similarly, estimates for \(\boldsymbol{\beta}\) are found in the first equation of the zinb coefficient table (labeled after depvar), and the estimates for \(\gamma\) are found in the second equation of the coefficient table (labeled inflate). The parameter \(\alpha\) is the negative binomial overdispersion parameter, and its estimate is the ancillary parameter labeled alpha in the coefficient table. Parameters \(p_{j}, m\), and \(\mu_{j}\) are parameters of a negative binomial distribution; see Methods and formulas in [R] nbreg for details.

The log likelihood maximized by zinb is
\[
\begin{aligned}
\ln L= & \sum_{j \in S} w_{j} \ln \left\{F_{j}+\left(1-F_{j}\right) p_{j}^{m}\right\}+ \\
& \sum_{j \notin S} w_{j}\left\{\ln \left(1-F_{j}\right)+\ln \Gamma\left(m+y_{j}\right)-\ln \Gamma\left(y_{j}+1\right)\right. \\
& \left.\quad-\ln \Gamma(m)+m \ln p_{j}+y_{j} \ln \left(1-p_{j}\right)\right\}
\end{aligned}
\]
where \(w_{j}\) are the weights, \(S\) is the set of observations for which the observed outcome \(y_{j}=0\), and \(F_{j}\) is the logistic distribution function
\[
F_{j}=F\left(\xi_{j}^{\gamma}\right)=\exp \left(\xi_{j}^{\gamma}\right) /\left\{1+\exp \left(\xi_{j}^{\gamma}\right)\right\}
\]
or, if the probit option is specified, the standard normal distribution function
\[
F_{j}=F\left(\xi_{j}^{\gamma}\right)=\Phi\left(\xi_{j}^{\gamma}\right)
\]

From Long (1997), the variance of the mixture distribution is
\[
\operatorname{Var}\left(y_{j} \mid \mathbf{x}_{i}, \mathbf{z}_{i}\right)=\mu_{j}\left(1-F_{j}\right)\left\{1+\mu_{j}\left(F_{j}+\alpha\right)\right\}
\]

When \(F_{j}\) is zero, we have the variance of the negative binomial distribution; when \(F_{j}>0\), the variance can exceed that of the negative binomial distribution.

This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce(robust) and vce(cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.
zinb also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

\section*{References}

Cameron, A. C., and P. K. Trivedi. 2005. Microeconometrics: Methods and Applications. New York: Cambridge University Press.
Cummings, T. H., and J. W. Hardin. 2019. Modeling count data with marginalized zero-inflated distributions. Stata Journal 19: 499-509.

Desmarais, B. A., and J. J. Harden. 2013. Testing for zero inflation in count models: Bias correction for the Vuong test. Stata Journal 13: 810-835.
Harris, T., J. M. Hilbe, and J. W. Hardin. 2014. Modeling count data with generalized distributions. Stata Journal 14: 562-579.

Long, J. S. 1997. Regression Models for Categorical and Limited Dependent Variables. Thousand Oaks, CA: Sage.
Long, J. S., and J. Freese. 2001. Predicted probabilities for count models. Stata Journal 1: 51-57.
__. 2014. Regression Models for Categorical Dependent Variables Using Stata. 3rd ed. College Station, TX: Stata Press.

Mullahy, J. 1986. Specification and testing of some modified count data models. Journal of Econometrics 33: 341-365. https://doi.org/10.1016/0304-4076(86)90002-3.
Xia, Y., Y. Zhou, and T. Cai. 2019. gidm: A command for generalized inflated discrete models. Stata Journal 19: 698-718.

\section*{Also see}
[R] zinb postestimation - Postestimation tools for zinb
[R] zip - Zero-inflated Poisson regression
[R] nbreg - Negative binomial regression
[R] poisson - Poisson regression
[R] tnbreg - Truncated negative binomial regression
[R] tpoisson - Truncated Poisson regression
[BAYES] bayes: zinb - Bayesian zero-inflated negative binomial regression
[SVY] svy estimation - Estimation commands for survey data
[XT] xtnbreg - Fixed-effects, random-effects, \& population-averaged negative binomial models
[U] 20 Estimation and postestimation commands

\section*{Title}
zinb postestimation - Postestimation tools for zinb
\begin{tabular}{ll} 
Postestimation commands & predict margins Methods and formulas \\
References & Also see
\end{tabular}

\section*{Postestimation commands}

The following postestimation commands are available after zinb:
\(\left.\begin{array}{ll}\text { Command } & \text { Description } \\
\hline \text { contrast } & \begin{array}{l}\text { contrasts and ANOVA-style joint tests of estimates } \\
\text { estat ic }\end{array} \\
\text { Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian in- } \\
\text { formation criteria (AIC, CAIC, AICc, and BIC) }\end{array}\right]\)\begin{tabular}{ll} 
estat summarize & summary statistics for the estimation sample \\
estat (svy) & variance-covariance matrix of the estimators (VCE) \\
estimates & \begin{tabular}{l} 
potestimation statistics for survey data \\
catable estimation results
\end{tabular} \\
* table of estimation results
\end{tabular}

\footnotetext{
*forecast, hausman, and lrtest are not appropriate with svy estimation results.
}

\section*{predict}

\section*{Description for predict}
predict creates a new variable containing predictions such as numbers of events, incidence rates, probabilities, linear predictions, and standard errors.

\section*{Menu for predict}

Statistics \(>\) Postestimation

\section*{Syntax for predict}
predict \([\) type \(]\) newvar \([\) if \(][\) in \(][\), statistic nooffset \(]\)
predict \([\) type \(]\) stub* \([\) if \(][\mathrm{in}]\), scores
statistic

Main
n number of events; the default
ir incidence rate
pr probability of a degenerate zero
\(\operatorname{pr}(n) \quad\) probability \(\operatorname{Pr}\left(y_{j}=n\right)\)
\(\operatorname{pr}(a, b) \quad\) probability \(\operatorname{Pr}\left(a \leq y_{j} \leq b\right)\)
\(\mathrm{xb} \quad\) linear prediction
stdp standard error of the linear prediction

These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

\section*{Options for predict}
\(\qquad\) Main
n , the default, calculates the predicted number of events, which is \(\left(1-F_{j}\right) \exp \left(\mathbf{x}_{j} \boldsymbol{\beta}\right)\) if neither offset() nor exposure() was specified when the model was fit, where \(F_{j}\) is the predicted probability of a zero outcome; \(\left(1-F_{j}\right) \exp \left(\mathbf{x}_{j} \boldsymbol{\beta}+\operatorname{offset}_{j}^{\beta}\right)\) if offset() was specified; or \(\left(1-F_{j}\right)\left\{\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}\right) \times\right.\) exposure \(\left._{j}\right\}\) if exposure() was specified.
ir calculates the incidence rate, which is the predicted number of events when exposure is 1 . This is equivalent to specifying both the n and the nooffset options.
pr calculates the probability of a degenerate zero, predicted from the fitted degenerate distribution \(F_{j}=\) \(F\left(\mathbf{z}_{j} \gamma\right)\). If offset () was specified within the inflate() option, then \(F_{j}=F\left(\mathbf{z}_{j} \gamma+\right.\) offset \(\left._{j}^{\gamma}\right)\) is calculated.
\(\operatorname{pr}(n)\) calculates the probability \(\operatorname{Pr}\left(y_{j}=n\right)\), where \(n\) is a nonnegative integer that may be specified as a number or a variable. Note that pr is not equivalent to \(\mathrm{pr}(0)\).
\(\operatorname{pr}(a, b)\) calculates the probability \(\operatorname{Pr}\left(a \leq y_{j} \leq b\right)\), where \(a\) and \(b\) are nonnegative integers that may be specified as numbers or variables;
\(b\) missing ( \(b \geq\).) means \(+\infty\);
\(\operatorname{pr}(20,\).\() calculates \operatorname{Pr}\left(y_{j} \geq 20\right)\);
\(\operatorname{pr}(20, b)\) calculates \(\operatorname{Pr}\left(y_{j} \geq 20\right)\) in observations for which \(b \geq\). and calculates
\(\operatorname{Pr}\left(20 \leq y_{j} \leq b\right)\) elsewhere.
\(\operatorname{pr}(., b)\) produces a syntax error. A missing value in an observation of the variable \(a\) causes a missing value in that observation for \(\mathrm{pr}(a, b)\).
xb calculates the linear prediction, which is \(\mathbf{x}_{j} \boldsymbol{\beta}\) if neither offset() nor exposure() was specified; \(\mathbf{x}_{j} \boldsymbol{\beta}+\operatorname{offset}_{j}^{\beta}\) if offset () was specified; or \(\mathbf{x}_{j} \boldsymbol{\beta}+\ln\) (exposure \(_{j}\) ) if exposure() was specified; see nooffset below.
stdp calculates the standard error of the linear prediction.
nooffset is relevant only if you specified offset() or exposure() when you fit the model. It modifies the calculations made by predict so that they ignore the offset or exposure variable; the linear prediction is treated as \(\mathbf{x}_{j} \boldsymbol{\beta}\) rather than as \(\mathbf{x}_{j} \boldsymbol{\beta}+\operatorname{offset}_{j}^{\beta}\) or \(\mathbf{x}_{j} \boldsymbol{\beta}+\ln \left(\right.\) exposure \(\left._{j}\right)\). Specifying predict..., nooffset is equivalent to specifying predict..., ir.
scores calculates equation-level score variables.
The first new variable will contain \(\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)\).
The second new variable will contain \(\partial \ln L / \partial\left(\mathbf{z}_{j} \gamma\right)\).
The third new variable will contain \(\partial \ln L / \partial \ln \alpha\).

\section*{margins}

\section*{Description for margins}
margins estimates margins of response for number of events, incidence rates, probabilities, and linear predictions.

\section*{Menu for margins}

Statistics \(>\) Postestimation

\section*{Syntax for margins}
```

margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...) ...] [options]

```
statistic
Description
n
ir
pr
\(\operatorname{pr}(n)\)
\(\operatorname{pr}(a, b)\)
xb
stdp
number of events; the default incidence rate probability of a degenerate zero probability \(\operatorname{Pr}\left(y_{j}=n\right)\) probability \(\operatorname{Pr}\left(a \leq y_{j} \leq b\right)\) linear prediction not allowed with margins

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

\section*{Methods and formulas}

See Methods and formulas in [R] zinb for the model definition and notation.
The probabilities calculated using the \(\operatorname{pr}(n)\) option are the probability \(\operatorname{Pr}\left(y_{j}=n\right)\). These are calculated using
\[
\begin{aligned}
& \operatorname{Pr}\left(y_{j}=0 \mid \mathbf{x}_{j}, \mathbf{z}_{j}\right)=F_{j}+\left(1-F_{j}\right) p_{2}\left(0 \mid \mathbf{x}_{j}\right) \\
& \operatorname{Pr}\left(y_{j}=n \mid \mathbf{x}_{j}, \mathbf{z}_{j}\right)=\left(1-F_{j}\right) p_{2}\left(n \mid \mathbf{x}_{j}\right) \quad \text { for } n=1,2, \ldots
\end{aligned}
\]
where \(F_{j}\) is the probability of obtaining an observation from the degenerate distribution whose mass is concentrated at zero, and \(p_{2}\left(n \mid \mathbf{x}_{j}\right)\) is the probability of \(y_{j}=n\) from the nondegenerate, negative binomial distribution. \(F_{j}\) can be obtained by using the pr option.

See Cameron and Trivedi (2013, sec. 4.6) for further details.

\section*{References}

Cameron, A. C., and P. K. Trivedi. 2013. Regression Analysis of Count Data. 2nd ed. New York: Cambridge University Press.

Manjón, M., and O. Martínez. 2014. The chi-squared goodness-of-fit test for count-data models. Stata Journal 14: 798-816.

\section*{Also see}
[R] zinb - Zero-inflated negative binomial regression
[U] 20 Estimation and postestimation commands

\section*{Title}
ziologit — Zero-inflated ordered logit regression
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
ziologit fits a model for an ordinal outcome with excess zeros, a higher fraction of zeros than would be expected from a standard ordered logit model, also known as zero inflation. This model is known as a zero-inflated ordered logit (ZIOL) model. In the context of ZIOL models, zero is the lowest outcome category. The ZIOL model accounts for the zero inflation by assuming that the zero-valued outcomes come from both a logit model and an ordered logit model, allowing potentially different sets of covariates for each model.

\section*{Quick start}

Zero-inflated ordered logit model of y on x 1 and categorical variable a with excess zeros modeled using x2
ziologit y x1 i.a, inflate(x2)
Add offset \(x 3\) to the ordered logit model ziologit \(y\) x1 i.a, inflate(x2) offset(x3)

Model excess zeros using only a constant
ziologit y x1 i.a, inflate(_cons)
Model excess zeros with x 2 , and offset x 5 while suppressing the constant term
ziologit y x1 i.a, inflate(x2, offset(x5) noconstant)
Account for complex sampling design using svyset data
svy: ziologit y x1 i.a, inflate(x2)

\section*{Menu}

Statistics \(>\) Ordinal outcomes \(>\) Zero-inflated ordered logit regression

\section*{Syntax}
\[
\begin{aligned}
& \text { ziologit depvar }[\text { indepvars }][\text { if }][\text { in }][\text { weight }], \\
& \text { inflate(varlist }[, \underline{\text { noconstant offset }(\text { varname })] \mid \text { _cons) }[\text { options }]}
\end{aligned}
\]

\section*{options}

Description
Model
*inflate ()
offset (varname)
constraints (constraints)
SE/Robust
vce(vcetype)

Reporting
level(\#)
nocnsreport
or
display_options

Maximization
maximize_options
collinear
coeflegend or jackknife
report odds ratios
inflation equation that determines excess zero values include varname in model with coefficient constrained to 1 apply specified linear constraints
vcetype may be oim, robust, cluster clustvar, opg, bootstrap,
set confidence level; default is level (95)
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
control the maximization process; seldom used
keep collinear variables
display legend instead of statistics
*inflate (varlist \([\), noconstant offset(varname) \(] \mid\) _cons) is required.
indepvars and varlist may contain factor variables; see [U] 11.4.3 Factor variables.
bayes, bootstrap, by, collect, fp, jackknife, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: ziologit.
Weights are not allowed with the bootstrap prefix; see [R] bootstrap.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options}
inflate (varlist [, noconstant offset (varname) \(] \mid\) _cons) specifies the inflation equation for the logit model that determines the excess zero values; this option is required. Conceptually, omitting inflate() would be equivalent to fitting the model with ologit; see [R] ologit.
inflate (varlist [, noconstant offset(varname)]) specifies the independent variables in the inflation equation. To suppress the constant in this equation, specify the noconstant suboption. You may optionally include an offset for this varlist; see offset (varname) in [R] Estimation options.
inflate(_cons) specifies that the inflation equation contains only an intercept. To run a zeroinflated model of depvar with only an intercept in both equations, type ziologit depvar, inflate(_cons).
offset (varname), constraints (constraints); see [R] Estimation options.

\section*{SE/Robust}
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.

\section*{Reporting}
level(\#), nocnsreport; see [R] Estimation options.
or reports the estimated coefficients transformed to odds ratios, that is, \(e^{b}\) rather than \(b\). Standard errors and confidence intervals are similarly transformed. This option affects how results are displayed, not how they are estimated. or may be specified at estimation or when replaying previously estimated results.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fmt), and nolstretch; see [R] Estimation options.

\section*{Maximization}
maximize_options: difficult, technique (algorithm_spec), iterate(\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from(init_specs); see [R] Maximize. These options are seldom used.

Setting the optimization type to technique (bhhh) resets the default vcetype to vce (opg).
The following options are available with ziologit but are not shown in the dialog box:
collinear, coeflegend; see [R] Estimation options.

\section*{Remarks and examples}

ZIOL models (Kelley and Anderson 2008) are used when the outcome of interest is an ordinal response with a higher fraction of observations in the lowest level than would be expected from a standard ordered logit model. The observations in the lowest category are often referred to as zeros because they typically correspond to the absence of a behavior or trait. Examples include measurements of symptoms, diet, exercise, drug use, and other ordinal outcomes where the lowest level is "none" or "never".

With ordinal outcomes, larger values of the response variable represent higher outcome levels, but the precise numeric value is irrelevant. For consistency, we will refer to the lowest category as zero regardless of whether that outcome takes the number 0 in the dataset.

The presence of a high fraction of zeros, also known as zero inflation, is driven by the fact that the lowest-level observations represent two latent (unobservable) groups: subjects with excess zeros and subjects with conditional zeros. Excess zeros are subjects who are not susceptible to the outcome, whereas conditional zeros are susceptible to higher outcomes even though they currently demonstrate
zero response. For example, if the outcome is alcohol consumption, a teetotaler who completely abstains from alcohol would be an excess zero, whereas a drinker who chose not to drink on the day of the survey would be a conditional zero.

To determine susceptibility (that is, to identify excess zeros), ZIOL regression uses a logit model for the inflation equation. For the main regression that determines the ordinal outcome of susceptible subjects, ZIOL uses an ordered logit model. The equation for the main regression is sometimes referred to as the intensity equation. These two equations can have different sets of covariates and different offsets. Parameters can be reported as coefficients or odds ratios with option or. See Methods and formulas for more details.

ZIOL models are similar in spirit to zero-inflated ordered probit (ZIOP) models (see [R] zioprobit), which use a probit model for the inflation equation and an ordered probit model for the intensity equation. Unlike ZIOP models, the results from ZIOL models can be presented as odds ratios. As such, ZIOL models often appeal to epidemiologists and health researchers, whereas ZIOP models are traditionally used by economists and social scientists. In general, the two types of models tend to produce similar predictions and yield similar inference.

In the context of ZIOL models, the excess-zero or "always-zero" group is often called the nonsusceptible group, and, likewise, the other group is called the susceptible group. In the context of ZIOP models, the terms "nonparticipation" and "participation" are used to refer to the groups. We will use both terminologies interchangeably depending on the context.

The ordered logit model is not nested within the ZIOL model, which means that a likelihood-ratio test cannot be used to check whether a standard ordered logit model is adequate. Instead, one can use information criteria; see example 2 in [ R\(]\) zip.
\(>\) Example 1: Zero-inflated ordered logit model
We use the fictional data on cigarette consumption from example 1 in [R] zioprobit. In that example, we fit a ZIOP model to these data; here we fit a ZIOL model. Both models are appropriate for analyzing this zero-inflated ordinal outcome, but the ZIOP model is more likely to be used by an economist studying consumer purchasing patterns, while the ZIOL model is more likely to be used by an epidemiologist studying smoking behavior.

The outcome of interest, tobacco, represents daily cigarette consumption as an ordinal response with four levels: 0 indicates "no cigarettes", and responses 1 through 3 indicate increasing cigarette consumption. More than half of the respondents reported no cigarette consumption, and we suspect that these respondents belong to one of two latent groups. Individuals in the first group are excess zeros, which is to say that they are genuine nonsmokers and thus are not susceptible to cigarette consumption. Individuals in the second group are would-be smokers with no current smoking activity but who might smoke, say, if the price of cigarettes falls or their income increases. These individuals are called conditional zeros because they demonstrate zero cigarette consumption, conditional on being susceptible to smoking. We suspect that these two types of zeros are driven by different patterns of behavior; hence, we choose the ZIOL model over the traditional ordered logit model.

Here the inflation (logit) equation corresponds to the decision to smoke. The intensity (ordered logit) equation corresponds to the level of cigarette consumption by a smoker or would-be smoker. The intensity equation includes covariates for education in years (education), annual income in tens of thousands of dollars (income), age in decades (age), and gender (female). The inflation equation includes all the covariates from the intensity equation, plus indicators for whether either of the respondent's parents smoked (parent) and whether the respondent's religion discourages smoking (religion). Covariates for the intensity equation are specified in the ziologit command statement directly after the dependent variable tobacco. Covariates for the inflation equation are specified in the required inflate() option.
```

. use https://www.stata-press.com/data/r18/tobacco
(Fictional tobacco consumption data)
. ziologit tobacco education income age i.female,
> inflate(education income age i.female i.parent i.religion)
Iteration 0: Log likelihood = -15877.562 (not concave)
Iteration 1: Log likelihood = -13082.531 (not concave)
Iteration 2: Log likelihood = -12316.574 (not concave)
Iteration 3: Log likelihood = -11686.438 (not concave)
Iteration 4: Log likelihood = -11214.794
Iteration 5: Log likelihood = -9857.4914
Iteration 6: Log likelihood = -8355.1808
Iteration 7: Log likelihood = -7805.3902
Iteration 8: Log likelihood = -7658.2562
Iteration 9: Log likelihood = -7652.5121
Iteration 10: Log likelihood = -7652.4891
Iteration 11: Log likelihood = -7652.4891

```
Zero-inflated ordered logit regression Number of obs \(=15,000\)
Wald chi2(4) \(=2143.69\)
Prob > chi2 \(=0.0000\)
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline tobacco & Coefficient & Std. err. & z & \(P>|z|\) & [95\% conf & interval] \\
\hline \multicolumn{7}{|l|}{tobacco} \\
\hline education & . 9239328 & . 0202141 & 45.71 & 0.000 & . 8843139 & . 9635516 \\
\hline income & 1.289223 & . 0285828 & 45.10 & 0.000 & 1.233202 & 1.345244 \\
\hline age & -1.393183 & . 0352799 & -39.49 & 0.000 & -1.462331 & -1.324036 \\
\hline \begin{tabular}{l}
female \\
Female
\end{tabular} & -. 7190566 & . 0755389 & -9.52 & 0.000 & -. 8671101 & -. 5710032 \\
\hline \multicolumn{7}{|l|}{inflate} \\
\hline education & -. 1602718 & . 0046406 & -34.54 & 0.000 & -. 1693671 & -. 1511765 \\
\hline income & -. 1922987 & . 007553 & -25.46 & 0.000 & -. 2071024 & -. 177495 \\
\hline age & . 313469 & . 0152296 & 20.58 & 0.000 & . 2836196 & . 3433184 \\
\hline \begin{tabular}{l}
female \\
Female
\end{tabular} & -. 4400725 & . 0516756 & -8.52 & 0.000 & -. 5413548 & -. 3387902 \\
\hline parent Smoking & 1.282534 & . 0527674 & 24.31 & 0.000 & 1.179112 & 1.385956 \\
\hline religion & & & & & & \\
\hline Discourag.. & -. 5387562 & . 0832958 & -6.47 & 0.000 & -. 7020129 & -. 3754995 \\
\hline _cons & 2.088609 & . 0984716 & 21.21 & 0.000 & 1.895609 & 2.28161 \\
\hline /cut1 & 5.327041 & . 1431499 & & & 5.046472 & 5.607609 \\
\hline /cut2 & 14.6561 & . 3250778 & & & 14.01896 & 15.29324 \\
\hline /cut3 & 20.28054 & . 4446689 & & & 19.409 & 21.15207 \\
\hline
\end{tabular}

The first section of the coefficient table, labeled tobacco, corresponds to the tobacco intensity equation and contains coefficients from the ordered logit model fit to susceptible subjects. The second section, labeled inflate, corresponds to the inflation equation and contains coefficients from the logit model for susceptibility. ZIOL models allow different sets of covariates in the inflation and intensity equations, but when a variable is included in both equations, it is possible for the coefficients to have opposite signs. The third section of the coefficient table contains the cutpoints from the ordered logit model. See Remarks and examples in [R] ologit for an explanation of cutpoints.

Compared with the ZIOP model from example 1 in [R] zioprobit, the coefficients here are larger in absolute value, but the \(z\) scores and \(p\)-values are similar. This is consistent with the fact that the
logistic distribution used by ZIOL has heavier tails than the normal distribution used by ZIOP. Perhaps the most important difference between these two models is that the estimated coefficients from the ZIOP model are difficult to interpret, but the coefficients from the ZIOL model can be exponentiated and reported as odds ratios with the or option.

We replay our results specifying or:
\begin{tabular}{ll}
. ziologit, or & \\
Zero-inflated ordered logit regression & Number of obs \(=15,000\) \\
& Wald chi2 \((4)=2143.69\) \\
Log likelihood \(=-7652.4891\) & Prob \(>\) chi2 \(=0.0000\)
\end{tabular}
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline tobacco & Odds ratio & Std. err. & z & \(P>|z|\) & [95\% con & interval] \\
\hline \multicolumn{7}{|l|}{tobacco} \\
\hline education & 2.519178 & . 0509228 & 45.71 & 0.000 & 2.421323 & 2.620989 \\
\hline income & 3.629966 & . 1037545 & 45.10 & 0.000 & 3.432202 & 3.839125 \\
\hline age & . 2482837 & . 0087594 & -39.49 & 0.000 & . 2316957 & . 2660593 \\
\hline \begin{tabular}{l}
female \\
Female
\end{tabular} & . 4872117 & . 0368034 & -9.52 & 0.000 & . 420164 & . 5649584 \\
\hline \multicolumn{7}{|l|}{inflate} \\
\hline education & . 8519122 & . 0039533 & -34.54 & 0.000 & . 8441989 & . 8596959 \\
\hline income & . 8250604 & . 0062317 & -25.46 & 0.000 & . 8129364 & . 8373652 \\
\hline age & 1.368163 & . 0208365 & 20.58 & 0.000 & 1.327928 & 1.409618 \\
\hline \begin{tabular}{l}
female \\
Female
\end{tabular} & . 6439897 & . 0332785 & -8.52 & 0.000 & . 5819593 & . 712632 \\
\hline parent Smoking & 3.605764 & . 1902668 & 24.31 & 0.000 & 3.251484 & 3.998647 \\
\hline religion & & & & & & \\
\hline Discourag.. & . 5834735 & . 0486009 & -6.47 & 0.000 & . 4955867 & . 6869461 \\
\hline _cons & 8.07368 & . 7950279 & 21.21 & 0.000 & 6.656599 & 9.792435 \\
\hline /cut1 & 5.327041 & . 1431499 & & & 5.046472 & 5.607609 \\
\hline /cut2 & 14.6561 & . 3250778 & & & 14.01896 & 15.29324 \\
\hline /cut3 & 20.28054 & . 4446689 & & & 19.409 & 21.15207 \\
\hline
\end{tabular}

Note: Estimates are transformed only in the first 2 equations to odds ratios. Note: _cons estimates baseline odds.

Examining the odds ratios from the tobacco intensity equation, we see that a one-unit increase in income, which corresponds to an increase of \(\$ 10,000\) in annual income, raises a smoker's odds of increased cigarette consumption by a factor of 3.63 . Looking at the inflation equation, we see that a one-unit increase in income lowers the odds of being susceptible to smoking by a factor of 0.825 . This suggests that wealthier individuals are less likely to smoke (consistent with income acting as a proxy for health consciousness), but if they do decide to smoke, they tend to smoke more cigarettes (perhaps because they can afford them).

\section*{Stored results}
ziologit stores the following in e() :

Scalars
```

e(N)
e(N_zero)
e(k_cat)
e(k)
e(k_eq)
e(k_eq_model)
e(k_aux)
e(k_dv)
e(df_m)
e(ll)
e(N_clust)
e(chi2)
e(p)
e(rank)
e(ic)
e(rc)
e(converged)

```

Macros
    e(cmd)
    e(cmdline)
    e(depvar)
    e(wtype)
    e(wexp)
    e(title)
    e(clustvar)
    e(offset1)
    e(offset2)
    e(chi2type)
    e(vce)
    e(vcetype)
    e(opt)
    e(which)
    e(ml_method)
    e(user)
    e(technique)
    e(properties)
    e(predict)
    e(marginsok)
    e(marginsnotok)
    e(marginsdefault)
    \(e\) (asbalanced)
    e(asobserved)

Matrices
        e(b)
    e(Cns)
    e(ilog)
    e(gradient)
    e(cat)
    e(V)
    e(V_modelbased)

Functions
        e(sample)
number of observations
number of zeros or lowest-category observations
number of categories
number of parameters
number of equations in \(e(b)\)
number of equations in overall model test
number of auxiliary parameters
number of dependent variables
model degrees of freedom
log likelihood
number of clusters
\(\chi^{2}\)
\(p\)-value for model test
rank of e(V)
number of iterations
return code
1 if converged, 0 otherwise
ziologit
command as typed
name of dependent variable
weight type
weight expression
title in estimation output
name of cluster variable
offset
offset for inflate()
Wald or LR; type of model \(\chi^{2}\) test
vcetype specified in vce()
title used to label Std. err.
type of optimization
max or min; whether optimizer is to perform maximization or minimization
type of ml method
name of likelihood-evaluator program
maximization technique
b V
program used to implement predict
predictions allowed by margins
predictions disallowed by margins
default predict() specification for margins
factor variables fvset as asbalanced
factor variables fvset as asobserved
coefficient vector
constraints matrix
iteration \(\log\) (up to 20 iterations)
gradient vector
category values
variance-covariance matrix of the estimators
model-based variance
marks estimation sample

In addition to the above, the following is stored in r() :

Matrices
\(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

\section*{Methods and formulas}

The modern class of zero-inflated models was originally developed by Lambert (1992) to address the problem of excess zeros in count data (see [R] zip). This model framework has been extended to ordinal outcomes with the ZIOL model by Kelley and Anderson (2008).

The precise numeric value of an ordinal outcome is irrelevant, so without loss of generality we consider an ordinal response variable \(Y\) with levels coded as \(0,1,2, \ldots, H\). The first step in the ZIOL model is to determine susceptibility. Let \(s_{j}=1\) if the \(j\) th individual is susceptible to exhibiting a nonzero response, and let \(s_{j}=0\) if the \(j\) th individual is an excess zero. (Similarly, using alternative terminology, let individuals with \(s_{j}=1\) belong to the participation group and those with \(s_{j}=0\) belong to the nonparticipation group.) The ZIOL inflation equation uses a logit model to determine the probability of susceptibility (or participation) as
\[
\begin{equation*}
\operatorname{Pr}\left(s_{j}=1 \mid \mathbf{z}_{j}\right)=F\left(\mathbf{z}_{j} \gamma\right) \tag{1}
\end{equation*}
\]
\(\mathbf{z}_{j}\) is a vector of covariates that determines susceptibility, \(\gamma\) is a vector of coefficients that have to be estimated, and \(F(\cdot)\) is the logistic distribution function: \(F(x)=e^{x} /\left(1+e^{x}\right)\).

Next, conditioning on \(s_{j}=1\), outcome intensity levels \(\widetilde{y}_{j}\) are modeled using an ordered logit model whose levels may also include 0 . The corresponding probabilities are given by
\[
\begin{equation*}
\operatorname{Pr}\left(\widetilde{y}_{j}=h \mid s_{j}=1, \mathbf{x}_{j}\right)=F\left(\kappa_{h}-\mathbf{x}_{j} \boldsymbol{\beta}\right)-F\left(\kappa_{h-1}-\mathbf{x}_{j} \boldsymbol{\beta}\right) \quad h=0,1, \ldots, H \tag{2}
\end{equation*}
\]
\(\mathbf{x}_{j}\) is a vector of covariates that determine intensity (which can be different from \(\mathbf{z}_{j}\) ), \(\boldsymbol{\beta}\) is a vector of coefficients to be estimated, and cutpoints \(\kappa_{h}\) are boundary parameters to be estimated (subject to \(\left.\kappa_{-1}=-\infty, \kappa_{H}=+\infty\right)\). These cutpoints take the place of an intercept.

The observed response variable is \(y_{j}=s_{j} \widetilde{y}_{j}\). Thus, a zero outcome occurs when \(s_{j}=0\) (the individual is an excess zero) or when \(s_{j}=1\) and \(\widetilde{y}_{j}=0\) (the individual is a conditional zero). To observe a positive \(y_{j}\), there is a joint requirement that \(s_{j}=1\) and \(\widetilde{y}_{j}>0\).

The distribution of \(Y\) is given by
\[
\begin{align*}
\operatorname{Pr}(Y) & =\left\{\begin{array}{l}
\operatorname{Pr}\left(y_{j}=0 \mid \mathbf{z}_{j}, \mathbf{x}_{j}\right) \\
\operatorname{Pr}\left(y_{j}=h \mid \mathbf{z}_{j}, \mathbf{x}_{j}\right) \quad h=1,2, \ldots, H
\end{array}\right.  \tag{3}\\
& =\left\{\begin{array}{l}
\operatorname{Pr}\left(s_{j}=0 \mid \mathbf{z}_{j}\right)+\operatorname{Pr}\left(s_{j}=1 \mid \mathbf{z}_{j}\right) \operatorname{Pr}\left(\widetilde{y}_{j}=0 \mid s_{j}=1, \mathbf{x}_{j}\right) \\
\operatorname{Pr}\left(s_{j}=1 \mid \mathbf{z}_{j}\right) \operatorname{Pr}\left(\widetilde{y}_{j}=h \mid s_{j}=1, \mathbf{x}_{j}\right) \quad h=1,2, \ldots, H
\end{array}\right.
\end{align*}
\]

The probability of zero outcome has been inflated because it is the sum of the probability of zero intensity from the ordered logit model and the probability of nonsusceptibility from the logit model.

Substituting (1) and (2) in (3), we get
\[
\left.\begin{array}{rl}
\operatorname{Pr}(Y) & =\left\{\begin{array}{l}
\operatorname{Pr}\left(y_{j}=0 \mid \mathbf{z}_{j}, \mathbf{x}_{j}\right) \\
\operatorname{Pr}\left(y_{j}=h \mid \mathbf{z}_{j}, \mathbf{x}_{j}\right) \\
\operatorname{Pr}\left(y_{j}=H \mid \mathbf{z}_{j}, \mathbf{x}_{j}\right)
\end{array} \quad h=1,2, \ldots, H-1\right.
\end{array}\right\} \begin{aligned}
& \left\{1-F\left(\mathbf{z}_{j} \boldsymbol{\gamma}\right)\right\}+F\left(\mathbf{z}_{j} \boldsymbol{\gamma}\right) F\left(\kappa_{0}-\mathbf{x}_{j} \boldsymbol{\beta}\right) \\
& F\left(\mathbf{z}_{j} \boldsymbol{\gamma}\right)\left\{F\left(\kappa_{h}-\mathbf{x}_{j} \boldsymbol{\beta}\right)-F\left(\kappa_{h-1}-\mathbf{x}_{j} \boldsymbol{\beta}\right)\right\} \quad h=1,2, \ldots, H-1  \tag{4}\\
& F\left(\mathbf{z}_{j} \gamma\right)\left\{1-F\left(\kappa_{H-1}-\mathbf{x}_{j} \boldsymbol{\beta}\right)\right\}
\end{aligned}
\]

If the offset () option is specified, \(\mathbf{x}_{j} \boldsymbol{\beta}\) in the intensity equation is replaced with \(\mathbf{x}_{j} \boldsymbol{\beta}+\operatorname{offset}_{j}^{\beta}\). If the offset () suboption is specified in option inflate(), \(\mathbf{z}_{j} \gamma\) in the inflation equation is replaced with \(\mathbf{z}_{j} \gamma+\) offset \(_{j}^{\gamma}\).

The log-likelihood function is
\[
\ln L=\sum_{j=1}^{N} w_{j} \sum_{h=0}^{H} I\left(y_{j}=h\right) \ln \left\{\operatorname{Pr}\left(y_{j}=h \mid \mathbf{z}_{j}, \mathbf{x}_{j}\right)\right\}
\]
where \(w_{j}\) is an optional weight for the \(j\) th observation and
\[
I\left(y_{j}=h\right)= \begin{cases}1 & \text { if } y_{j}=h \\ 0 & \text { otherwise }\end{cases}
\]

The choice between the ZIOL model and the ordered logit model cannot be made using a likelihoodratio test because the two hypotheses are not nested in the usual sense of parameter restrictions. The inflation effect is removed, and all subjects are deemed susceptible when \(\mathbf{z}_{j} \gamma \rightarrow \infty\), a condition that cannot be imposed. To compare the fits of nonnested models, you can use estat ic to display information criteria (see [R] estat ic).

This command supports the Huber/White/sandwich estimator of the variance and its clustered version with vce(robust) and vce(cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.
ziologit also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

\section*{References}

Kelley, M. E., and S. J. Anderson. 2008. Zero inflation in ordinal data: Incorporating susceptibility to response through the use of a mixture model. Statistics in Medicine 27: 3674-3688. https://doi.org/10.1002/sim.3267.
Lambert, D. 1992. Zero-inflated Poisson regression, with an application to defects in manufacturing. Technometrics 34: 1-14. https://doi.org/10.2307/1269547.

\section*{Also see}
[ R ] ziologit postestimation - Postestimation tools for ziologit
[R] logit - Logistic regression, reporting coefficients
\([\mathrm{R}]\) ologit - Ordered logistic regression
[R] zioprobit - Zero-inflated ordered probit regression
[BAYES] bayes: ziologit - Bayesian zero-inflated ordered logit regression
[SVY] svy estimation - Estimation commands for survey data
[U] 20 Estimation and postestimation commands

\section*{Title}
ziologit postestimation - Postestimation tools for ziologit
\begin{tabular}{lll} 
Postestimation commands & predict & margins \\
Remarks and examples & Methods and formulas & Reference
\end{tabular}

\section*{Postestimation commands}

The following postestimation commands are available after ziologit:
\begin{tabular}{|c|c|}
\hline Command & Description \\
\hline contrast & contrasts and ANOVA-style joint tests of estimates \\
\hline estat ic & Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) \\
\hline estat summarize & summary statistics for the estimation sample \\
\hline estat vce & variance-covariance matrix of the estimators (VCE) \\
\hline estat (svy) & postestimation statistics for survey data \\
\hline estimates & cataloging estimation results \\
\hline etable & table of estimation results \\
\hline *forecast & dynamic forecasts and simulations \\
\hline * hausman & Hausman's specification test \\
\hline lincom & point estimates, standard errors, testing, and inference for linear combinations of coefficients \\
\hline * lrtest & likelihood-ratio test \\
\hline margins & marginal means, predictive margins, marginal effects, and average marginal effects \\
\hline marginsplot & graph the results from margins (profile plots, interaction plots, etc.) \\
\hline nlcom & point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients \\
\hline predict & probabilities, linear predictions and their SEs, etc. \\
\hline predictnl & point estimates, standard errors, testing, and inference for generalized predictions \\
\hline pwcompare & pairwise comparisons of estimates \\
\hline suest & seemingly unrelated estimation \\
\hline test & Wald tests of simple and composite linear hypotheses \\
\hline testnl & Wald tests of nonlinear hypotheses \\
\hline
\end{tabular}

\footnotetext{
*forecast, hausman, and lrtest are not appropriate with svy estimation results.
}

\section*{predict}

\section*{Description for predict}
predict creates a new variable containing predictions such as probabilities, linear predictions, and standard errors.

\section*{Menu for predict}

Statistics \(>\) Postestimation

\section*{Syntax for predict}
```

predict [type] {stub*|newvar|newvarlist } [if] [in] [, statistic
outcome(outcome) nooffset]
predict [type] stub* [if] [in], scores

```
statistic Description

\section*{Main}
\begin{tabular}{ll} 
pmargin & \begin{tabular}{l} 
marginal probabilities of levels, \(\operatorname{Pr}\left(y_{j}=h\right) ;\) the default \\
joint probabilities of levels and susceptibility, \(\operatorname{Pr}\left(y_{j}=h, s_{j}=1\right)\) \\
pjoint1
\end{tabular} \\
pcond1 & \begin{tabular}{l} 
probabilities of levels conditional on susceptibility, \(\operatorname{Pr}\left(y_{j}=h \mid s_{j}=1\right)\) \\
ps
\end{tabular} \\
probability of susceptibility, \(\operatorname{Pr}\left(s_{j}=1\right)\) \\
pns & probability of nonsusceptibility, \(\operatorname{Pr}\left(s_{j}=0\right)\) \\
xb & linear prediction \\
xbinfl & \begin{tabular}{l} 
linear prediction for inflation equation
\end{tabular} \\
stdp & \begin{tabular}{l} 
standard error of the linear prediction
\end{tabular} \\
stdpinf1 & standard error of the linear prediction for inflation equation
\end{tabular}

If you do not specify outcome(), pmargin, pjoint1, and pcond1 (with one new variable specified) assume outcome(\#1).

You specify one or \(k\) new variables with pmargin, pjoint1, and pcond1, where \(k\) is the number of outcomes.
You specify one new variable with ps, pns, xb, xbinfl, stdp, and stdpinfl.
These statistics are available both in and out of sample; type predict ... if e(sample)... if wanted only for the estimation sample.

\section*{Options for predict}
\(\qquad\) Main
pmargin, the default, calculates the predicted marginal probabilities of outcome levels, \(\operatorname{Pr}\left(y_{j}=h\right)\). pjoint1 calculates the predicted joint probabilities of outcome levels and susceptibility, \(\operatorname{Pr}\left(y_{j}=\right.\) \(\left.h, s_{j}=1\right)\).
pcond1 calculates the predicted probabilities of outcome levels conditional on susceptibility, \(\operatorname{Pr}\left(y_{j}=\right.\) \(\left.h \mid s_{j}=1\right)\).

With pmargin, pjoint1, and pcond1, you can compute predicted probabilities for one or for all outcome levels. When you specify one new variable, predict computes probabilities for the first outcome level. You can specify the outcome ( \(\# i\) ) option to obtain probabilities for the \(i\) th level. When you specify multiple new variables or a stub, predict computes probabilities for all outcome levels. The behavior of predict with one new variable is equivalent to specifying outcome (\#1).
ps and pns calculate the predicted marginal probability of susceptibility \(\left[\operatorname{Pr}\left(s_{j}=1\right)\right]\) and of nonsusceptibility \(\left[\operatorname{Pr}\left(s_{j}=0\right)\right]\), respectively.
In econometrics literature, probabilities of susceptibility and nonsusceptibility are known as probabilities of participation and nonparticipation. Similarly to predict after zioprobit, you can use options ppar and pnpar to compute these probabilities. Options ppar and pnpar produce identical results to the respective options ps and pns but label new variables as \(\operatorname{Pr}\) (participation) and \(\operatorname{Pr}\) (nonparticipation) instead of \(\operatorname{Pr}\) (susceptible) and \(\operatorname{Pr}\) (nonsusceptible).
xb calculates the linear prediction for the ordered logit equation, which is \(\mathbf{x}_{j} \boldsymbol{\beta}\) if offset () was not specified with ziologit and is \(\mathbf{x}_{j} \boldsymbol{\beta}+\operatorname{offset}_{j}^{\beta}\) if offset() was specified.
xbinfl calculates the linear prediction for the inflation equation, which is \(\mathbf{z}_{j} \gamma\) if offset () was not specified in inflate() and is \(\mathbf{z}_{j} \gamma+\operatorname{offset}_{j}^{\gamma}\) if offset() was specified in inflate().
stdp calculates the standard error of the linear prediction for the ordered logit equation.
stdpinfl calculates the standard error of the linear prediction for the inflation equation.
outcome(outcome) specifies the outcome for which predicted probabilities are to be calculated. outcome () should contain either one value of the dependent variable or one of \#1, \#2, ..., with \#1 meaning the first category of the dependent variable, \#2 meaning the second category, etc. outcome() is allowed only with pmargin, pjoint1, and pcond1.
nooffset is relevant only if you specified offset (varname) with ziologit or within the inflate() option. It modifies the calculations made by predict so that they ignore the offset variable; that is, the linear prediction for the main regression equation is treated as \(\mathbf{x}_{j} \boldsymbol{\beta}\) rather than as \(\mathbf{x}_{j} \boldsymbol{\beta}+\operatorname{offset}{ }_{j}^{\beta}\) and the linear prediction for the inflation equation is treated as \(\mathbf{z}_{j} \gamma\) rather than as \(\mathbf{z}_{j} \gamma+\) offset \(_{j}^{\gamma}\).
scores calculates equation-level score variables.
The first new variable will contain \(\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)\). In the absence of independent variables in the main equation, this variable is not stored.
The second new variable will contain \(\partial \ln L / \partial\left(\mathbf{z}_{j} \gamma\right)\).
When the dependent variable takes \(k\) different values, the third new variable through new variable \(k+1\) will contain \(\partial \ln L / \partial\left(\kappa_{h}\right)\) for \(h=0,1, \ldots, k-2\).

\section*{margins}

\section*{Description for margins}
margins estimates margins of response for probabilities and linear predictions.

\section*{Menu for margins}

Statistics \(>\) Postestimation

\section*{Syntax for margins}
```

margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...) ...] [options]

```
\begin{tabular}{ll} 
statistic & Description \\
\hline default & marginal probabilities for each outcome \\
pmargin & marginal probabilities of levels, \(\operatorname{Pr}\left(y_{j}=h\right) ;\) the default \\
pjoint1 & joint probabilities of levels and susceptibility, \(\operatorname{Pr}\left(y_{j}=h, s_{j}=1\right)\) \\
pcond1 & probabilities of levels conditional on susceptibility, \(\operatorname{Pr}\left(y_{j}=h \mid s_{j}=1\right)\) \\
ps & probability of susceptibility, \(\operatorname{Pr}\left(s_{j}=1\right)\) \\
pns & probability of nonsusceptibility, \(\operatorname{Pr}\left(s_{j}=0\right)\) \\
xb & linear prediction \\
xbinfl & linear prediction for inflation equation \\
stdp & not allowed with margins \\
stdpinfl & not allowed with margins
\end{tabular}
pmargin, pjoint1, and pcond1 default to the first outcome.
Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

\section*{Remarks and examples}

The ZIOL model allows all the predictions and marginal effects available with the standard ologit model (see [R] ologit postestimation), along with additional predictions and marginal effects related to the inflation equation for susceptibility. The probabilities of susceptibility and nonsusceptibility can be calculated using options ps and pns, respectively. If you prefer an alternative terminology of probabilities of participation and nonparticipation, you can instead use options ppar and pnpar, which will produce identical numerical results but label variables as \(\operatorname{Pr}\) (participation) and \(\operatorname{Pr}\) (nonparticipation) instead of \(\operatorname{Pr}\) (susceptible) and \(\operatorname{Pr}\) (nonsusceptible).
> Example 1: Average marginal effect of gender on probability of nonsusceptibility
In example 1 of [R] ziologit, we fit a model for levels of cigarette consumption.
```

. use https://www.stata-press.com/data/r18/tobacco
(Fictional tobacco consumption data)
. ziologit tobacco education income age i.female,
> inflate(education income age i.female i.parent i.religion)
(output omitted)

```

This model parallels the zero-inflated ordered probit (ZIOP) model that was fit in example 1 of [R] zioprobit.

To continue the comparison between the ZIOL and ZIOP models, we re-create example 1 from \([\mathrm{R}]\) zioprobit postestimation by using margins to estimate the average marginal effect of gender on the probability of nonsusceptibility (being an excess zero) for individuals with a college degree (17 years of education) and a smoking parent.
```

. margins, predict(pns) dydx(female) at(education = 17 parent = 1)
Average marginal effects Number of obs = 15,000
Model VCE: OIM
Expression: Pr(nonsusceptible), predict(pns)
dy/dx wrt: 1.female
At: education = 17
parent = 1

```
\begin{tabular}{c|cccccc}
\hline & dy/dx \begin{tabular}{c} 
Delta-method \\
std. err.
\end{tabular} & \(z\) & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf. interval] \\
\hline \begin{tabular}{c} 
female \\
Female
\end{tabular} & .085421 & .010096 & 8.46 & 0.000 & .0656333 & .1052087 \\
\hline
\end{tabular}

Note: dy/dx for factor levels is the discrete change from the base level.
Despite the differences between the ZIOL and ZIOP models, the conclusion is the same: women with a college degree and a smoking parent are expected to have an approximately \(8.5 \%\) higher chance of being genuine nonsmokers (excess zeros) than comparable men.

\section*{> Example 2: Predicted probabilities of conditional zeros}

Next, we consider the effect of income on the probability of zero tobacco consumption, conditional on susceptibility. These would-be smokers are known as conditional zeros. In example 1 of \([R]\) ziologit, we saw that increasing income raises a smoker's odds of increased tobacco consumption dramatically, so we expect to see a larger fraction of conditional zeros at the lower end of the income scale.

We examine conditional probabilities of zero consumption for incomes ranging from \(\$ 10,000\) to \(\$ 60,000\), and we use the noatlegend option to suppress the default legend because we know the values 1 to 6 correspond to income in tens of thousands of dollars.
```

. margins, predict(pcond1 outcome(0)) at(income = (1/6)) noatlegend
Predictive margins Number of obs = 15,000
Model VCE: OIM
Expression: Pr(tobacco=0|susceptible=1), predict(pcond1 outcome(0))

```
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline & \multicolumn{3}{|c|}{Delta-method} & \(P>|z|\) & \multicolumn{2}{|l|}{[95\% conf. interval]} \\
\hline _at & & & & & & \\
\hline 1 & . 5923634 & . 0027586 & 214.73 & 0.000 & . 5869566 & . 5977702 \\
\hline 2 & . 5393818 & . 0025948 & 207.87 & 0.000 & . 534296 & . 5444676 \\
\hline 3 & . 4854668 & . 0024651 & 196.94 & 0.000 & . 4806354 & . 4902982 \\
\hline 4 & . 4306299 & . 0023953 & 179.78 & 0.000 & . 4259352 & . 4353245 \\
\hline 5 & . 3741538 & . 0024547 & 152.42 & 0.000 & . 3693427 & . 3789649 \\
\hline 6 & . 3152985 & . 0026294 & 119.91 & 0.000 & . 3101449 & . 320452 \\
\hline
\end{tabular}

The influence of income is dramatic: susceptible individuals (potential smokers) who earn \(\$ 10,000\) a year are almost twice as likely to refrain from smoking as potential smokers who earn \(\$ 60,000\) per year (59\% versus 32\%).

\section*{Methods and formulas}

See Methods and formulas in [R] ziologit for the model definition and notation. Specifically, see (1) for the formula for the probability of susceptibility, \(\operatorname{Pr}\left(s_{j}=1 \mid \mathbf{z}_{j}\right)\); see (2) for the formula for the probabilities of outcome levels conditional on susceptibility, \(\operatorname{Pr}\left(y_{j}=h \mid s_{j}=1, \mathbf{x}_{j}\right)\); and see (4) for the formula for the marginal probabilities of outcome levels, \(\operatorname{Pr}\left(y_{j}=h \mid \mathbf{z}_{j}, \mathbf{x}_{j}\right)\).

The joint probability of susceptibility and outcome \(y_{j}=h\) can be expressed as
\[
\operatorname{Pr}\left(y_{j}=h, s_{j}=1 \mid \mathbf{z}_{j}, \mathbf{x}_{j}\right)=\operatorname{Pr}\left(s_{j}=1 \mid \mathbf{z}_{j}\right) \operatorname{Pr}\left(y_{j}=h \mid s_{j}=1, \mathbf{x}_{j}\right)
\]
for \(h=0,1, \ldots, H\).

\section*{Reference}

Kelley, M. E., and S. J. Anderson. 2008. Zero inflation in ordinal data: Incorporating susceptibility to response through the use of a mixture model. Statistics in Medicine 27: 3674-3688. https://doi.org/10.1002/sim.3267.

\section*{Also see}
[R] ziologit - Zero-inflated ordered logit regression
[U] 20 Estimation and postestimation commands

\section*{Title}
zioprobit — Zero-inflated ordered probit regression
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
zioprobit fits a model for an ordinal outcome with excess zeros, a higher fraction of zeros than would be expected from a standard ordered probit model, also known as zero inflation. This model is known as a zero-inflated ordered probit (ZIOP) model. In the context of ZIOP models, zero is an actual 0 value or the lowest outcome category. The ZIOP model accounts for the zero inflation by assuming that the zero-valued outcomes come from both a probit model and an ordered probit model, allowing potentially different sets of covariates for each model.

\section*{Quick start}

Zero-inflated ordered probit model of y on x1 and categorical variable a with excess zeros modeled using x2
zioprobit y x1 i.a, inflate(x2)
Add offset \(x 3\) to the ordered probit model
zioprobit \(y\) x1 i.a, inflate(x2) offset(x3)
Model excess zeros using only a constant
zioprobit y x1 i.a, inflate(_cons)
Model excess zeros with x 2 , and offset x 5 while suppressing the constant term
zioprobit y x1 i.a, inflate(x2, offset(x5) noconstant)
Account for complex sampling design using svyset data
svy: zioprobit y x1 i.a, inflate(x2)

\section*{Menu}

Statistics \(>\) Ordinal outcomes \(>\) Zero-inflated ordered probit regression

\section*{Syntax}
zioprobit depvar [indepvars] [if] [in] [weight],
inflate(varlist[, noconstant offset(varname)]|_cons) [options]
options
Model
*inflate ( )
offset (varname)
constraints (constraints)
SE/Robust
vce (vcetype)

Reporting
level(\#)
nocnsreport
display_options

Maximization
maximize_options
collinear
coeflegend

Description
inflation equation that determines excess zero values include varname in model with coefficient constrained to 1
apply specified linear constraints
vcetype may be oim, robust, cluster clustvar, opg, bootstrap, or jackknife
set confidence level; default is level (95)
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
control the maximization process; seldom used
keep collinear variables
display legend instead of statistics
*inflate (varlist \([\), noconstant offset (varname) \(] \mid\) _cons) is required.
indepvars and varlist may contain factor variables; see [U] 11.4.3 Factor variables.
bayes, bootstrap, by, collect, fp, jackknife, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: zioprobit.
Weights are not allowed with the bootstrap prefix; see \([R]\) bootstrap.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options}

Model
inflate (varlist \([\), noconstant offset (varname) \(] \mid\) _cons) specifies the inflation equation for the probit model that determines the excess zero values; this option is required. Conceptually, omitting inflate() would be equivalent to fitting the model with oprobit; see [R] oprobit.
inflate (varlist [, noconstant offset(varname)]) specifies the independent variables in the inflation equation. To suppress the constant in this equation, specify the noconstant suboption. You may optionally include an offset for this varlist; see offset (varname) in [R] Estimation options.
inflate(_cons) specifies that the inflation equation contains only an intercept. To run a zeroinflated model of depvar with only an intercept in both equations, type zioprobit depvar, inflate (_cons).
offset (varname), constraints (constraints); see [R] Estimation options.

SE/Robust
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
```

            Reporting
    level(\#), nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fmt), pformat (\%fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.

```
\(\underset{\text { maximize_options: difficult, technique(algorithm_spec), iterate (\#), [no log, trace, }}{\sqrt{\text { Maximization }} \text {, }}\) gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from (init_specs); see [R] Maximize. These options are seldom used.

Setting the optimization type to technique (bhhh) resets the default vcetype to vce (opg).
The following options are available with zioprobit but are not shown in the dialog box: collinear, coeflegend; see [R] Estimation options.

\section*{Remarks and examples}

ZIOP models are used when the outcome of interest is an ordinal response variable and the data exhibit a high fraction of observations in the lowest category or what we will refer to from now on as "zero" but without quotes. Like the ordered probit model, the actual values taken by the ordinal response variable are irrelevant. While the outcome is typically coded as \(0,1,2, \ldots, H\), zioprobit interprets the lowest value present in the dataset as 0 to be consistent with the original derivation of the model and subsequent applications.

Like all zero-inflated models, the ZIOP model is an alternative when the data exhibit a higher fraction of zero-valued outcomes than is likely compatible with an ordered probit model. This concentration of zeros is referred to as zero inflation. Inflation is assumed to occur in the lowest value to ensure that shifting the levels of the ordinal response variable by a constant will not affect the estimated parameters in the model. This is common in ordered probit models; see [R] oprobit.

Without loss of generality, we consider an ordinal response variable with levels \(0,1,2, \ldots, H\). Traditional ordered probit models treat all observations with zero-valued outcomes as a homogeneous group. By contrast, ZIOP models assume that zeros could occur in the data as members of two latent (unobservable) groups. Individuals in the "always-zero" group have outcome 0 as the only possible value. This first group is often called the nonparticipation group. The second group, in addition to 0 , may also assume any of the other values, \(1,2, \ldots, H\). This group is often called the participation
group. Some disciplines, such as public health and medicine, refer to the process that determines the zeros rather than the groups. In this case, there is an incidence or occurrence process that determines whether an observation belongs to the always-zero group and a severity or intensity process that determines the level in the second group. The result of having two groups or processes is an inflation in the proportion of zero-valued observations in the data.

The ZIOP model has been used in studies of international and domestic conflicts (Bagozzi et al. 2015), sports participation (Downward, Lera-Lopez, and Rasciute 2011), and the adoption of new building technologies (Ganguly, Koebel, and Cantrell 2010), to name a few. See Kelley and Anderson (2008) for a discussion of zero-inflated ordinal models in the context of health.

The classic application of the ZIOP model is the study of tobacco use by Harris and Zhao (2007). Like the zero-inflated Poisson models in the count-data literature (Lambert 1992), Harris and Zhao derived the ZIOP model using a two-stage decision process. An individual must decide whether to participate in an activity (for example, smoking or drug consumption) and, conditional on participating, must decide on the level of participation, which also includes zero participation. The first decision is a binary choice and is modeled using a probit model, while the second is an ordered choice and is modeled using an ordered probit model. In other terms, to account for the excess of zeros, Harris and Zhao allowed for zero observations to occur in two ways: as a realization of the probit model (nonparticipants) and as a realization of the ordered probit model when the binary random variable in the probit model is 1 (participant with zero activity). See Methods and formulas for more details. For a Bayesian derivation of the ZIOP model, see Gurmu and Dagne (2012).

You may want to check whether a standard ordered probit model is adequate to fit the data. You can do this, for instance, using information criteria; see example 2 in [ R\(]\) zip.
\(>\) Example 1: Zero-inflated ordered probit model
We have fictional data on cigarette consumption per day for 15,000 subjects between ages 14 and 84. The outcome of interest, tobacco, is an ordinal response with four levels coded as 0 for "no cigarettes", 1 for "up to 8 cigarettes/day", 2 for " 8 to 12 cigarettes/day", and 3 for "more than 12 cigarettes/day". The exact number of daily consumed cigarettes is unknown.

About \(63 \%\) of the respondents identified themselves as current nonsmokers. We suspect that these self-identified current nonsmokers belong to one of two groups. Individuals in the first group are genuine nonsmokers (always-zero group) who have never smoked and will never smoke. Individuals in the second group are smokers with no smoking activity who could be the corner solution of a standard consumer demand problem and who may smoke, say, if the price of tobacco falls or their income increases. It is likely that these two types of zeros are driven by different patterns of consumer behavior and a ZIOP model is a good candidate in this case.

We model tobacco consumption levels for subjects who choose to smoke as a function of years of education (education), annual income in tens of thousands of dollars (income), age in tens of years (age), and whether the respondent is a female (female). In addition to education, income, age, and female, the decision to smoke is modeled as a function of whether either of the respondent's parents smoked (parent) and whether the respondent's religion discourages smoking (religion). We list all the covariates in the required inflate() option.
```

. use https://www.stata-press.com/data/r18/tobacco
(Fictional tobacco consumption data)
. zioprobit tobacco education income i.female age,
> inflate(education income i.parent age i.female i.religion)
Iteration 0: Log likelihood = -14820.211 (not concave)
Iteration 1: Log likelihood = -12819.475 (not concave)
Iteration 2: Log likelihood = -12078.843 (not concave)
Iteration 3: Log likelihood = -10926.037
Iteration 4: Log likelihood = -9549.5112
Iteration 5: Log likelihood = -8662.3141
Iteration 6: Log likelihood = -7749.9803
Iteration 7: Log likelihood = -7647.1348
Iteration 8: Log likelihood = -7640.5027
Iteration 9: Log likelihood = -7640.4738
Iteration 10: Log likelihood = -7640.4738
Zero-inflated ordered probit regression Number of obs = 15,000
Wald chi2(4) = 2574.27
Prob > chi2 = 0.0000

```
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline tobacco & Coefficient & Std. err. & z & \(\mathrm{P}>|\mathrm{z}|\) & [95\% conf & interval] \\
\hline \multicolumn{7}{|l|}{tobacco} \\
\hline education & . 5112664 & . 0102407 & 49.92 & 0.000 & . 491195 & . 5313378 \\
\hline income & . 712975 & . 0144803 & 49.24 & 0.000 & . 6845942 & . 7413559 \\
\hline female & & & & & & \\
\hline Female & -. 3975341 & . 0416675 & -9.54 & 0.000 & -. 4792009 & -. 3158674 \\
\hline age & -. 7709896 & . 0182554 & -42.23 & 0.000 & -. 8067695 & -. 7352097 \\
\hline \multicolumn{7}{|l|}{inflate} \\
\hline education & -. 0966613 & . 0026422 & -36.58 & 0.000 & -. 1018398 & -. 0914827 \\
\hline income & -. 1157545 & . 0043787 & -26.44 & 0.000 & -. 1243365 & -. 1071725 \\
\hline \multicolumn{7}{|l|}{parent} \\
\hline Smoking & . 7655798 & . 0307553 & 24.89 & 0.000 & . 7053006 & . 825859 \\
\hline age & . 1873904 & . 0088643 & 21.14 & 0.000 & . 1700168 & . 204764 \\
\hline \multicolumn{7}{|l|}{female} \\
\hline & . 2639665 & . 0307184 & & & . 3241735 & \\
\hline \multicolumn{7}{|l|}{religion} \\
\hline Discourag.. & -. 3223335 & . 0496827 & -6.49 & 0.000 & -. 4197098 & -. 2249572 \\
\hline _cons & 1.27051 & . 0584794 & 21.73 & 0.000 & 1.155892 & 1.385127 \\
\hline /cut1 & 2.959808 & . 0753035 & & & 2.812216 & 3.1074 \\
\hline /cut2 & 8.111228 & . 1648965 & & & 7.788037 & 8.43442 \\
\hline /cut3 & 11.20791 & . 2247711 & & & 10.76736 & 11.64845 \\
\hline
\end{tabular}

In the output table, the first set of coefficients, labeled tobacco, corresponds to the participation (smoking) levels. These coefficients are interpreted in the same way as coefficients from an ordered probit model. The second set of coefficients, labeled inflate, corresponds to the equation for the participation decision. These are interpreted in the same way as coefficients from a binary probit model. ZIOP models do not require the variables to be the same in the participation level and decision equations. However, the same variables can appear in both. If the same variables are included, it is not uncommon for the coefficients to have opposite signs. For example, income and education in the model above have positive signs in the level equation and negative signs in the decision equation.

The estimated coefficients are not particularly informative, and as with all discrete choice models, marginal effects are better to interpret. We use margins to estimate the average marginal effect of
having a smoking parent on the probability of being a genuine nonsmoker. Specifying pnpar within the predict () option means that we are requesting the predicted probability of nonparticipation, which in our example is equivalent to being a genuine nonsmoker.
```

. margins, predict(pnpar) dydx(parent)

```
Average marginal effects \(\quad\) Number of obs \(=15,000\)
Model VCE: OIM
Expression: \(\operatorname{Pr}(\) nonparticipation), predict(pnpar)
dy/dx wrt: 1.parent
\begin{tabular}{c|rccccc}
\hline & \multicolumn{5}{|c}{\begin{tabular}{c} 
Delta-method \\
std. err.
\end{tabular}} & \(z\) \\
& dy/dx & P>|z| & [95\% conf. interval] \\
\hline \begin{tabular}{c} 
parent \\
Smoking
\end{tabular} & -.1797895 & .0071967 & -24.98 & 0.000 & -.1938948 & -.1656843 \\
\hline
\end{tabular}

Note: dy/dx for factor levels is the discrete change from the base level.
On average, individuals whose parents are smokers are about \(18 \%\) less likely to be nonsmokers themselves than individuals whose parents did not use tobacco. See [R] zioprobit postestimation for more information and examples.

\section*{Stored results}
zioprobit stores the following in e():
Scalars
e(N)
e(N_zero)
e(k_cat)
e(k)
e(k_eq)
e(k_eq_model)
e(k_aux)
e(k_dv)
e(df_m)
e(11)
e(N_clust)
e(chi2)
e (p)
e(rank)
e(ic)
e(rc)
e(converged)
number of observations
number of zeros or lowest-category observations
number of categories
number of parameters
number of equations in \(e(b)\)
number of equations in overall model test
number of auxiliary parameters
number of dependent variables
model degrees of freedom
log likelihood
number of clusters
\(\chi^{2}\)
\(p\)-value for model test
rank of e(v)
number of iterations
return code
1 if converged, 0 otherwise

Macros
\[
e(c m d)
\]
e(cmdline)
e(depvar)
e(wtype)
e(wexp)
e(title)
e(clustvar)
e(offset1)
e(offset2)
e(chi2type)
e(vce)
e(vcetype)
e(opt)
e(which)
e(ml_method)
e(user)
e(technique)
e(properties)
e(predict)
e(marginsok)
e(marginsnotok)
e(marginsdefault)
e(asbalanced)
e(asobserved)
Matrices
e(b)
e(Cns)
e(ilog)
e(gradient)
e(cat)
e(V)
e(V_modelbased)
Functions
e(sample) marks estimation sample
In addition to the above, the following is stored in r() :

Matrices
\(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

\section*{Methods and formulas}

Consider an ordinal response variable \(Y\) with levels coded as \(0,1,2, \ldots, H\). For notational simplicity, we assume that the zeros are inflated, but the following derivation may be adapted to accommodate inflation in the lowest outcome category. Harris and Zhao (2007) derived the ZIOP model in two steps. First, the group membership (participants versus nonparticipants) can be modeled using a probit model. Let \(s_{j}=1\) if the \(j\) th individual belongs to the participation group or let \(s_{j}=0\) otherwise. With the probit model, the probability of participation is given by
\[
\begin{equation*}
\operatorname{Pr}\left(s_{j}=1 \mid \mathbf{z}_{j}\right)=\Phi\left(\mathbf{z}_{j} \gamma\right) \tag{1}
\end{equation*}
\]
\(\mathbf{z}_{j}\) is a vector of covariates that determines group membership, \(\gamma\) is a vector of coefficients that have to be estimated, and \(\Phi(\cdot)\) is the standard normal distribution function. Next, conditioning on \(s_{j}=1\), participation levels \(\widetilde{y}_{j}\) are modeled using an ordered probit model; these levels may also include 0 . The corresponding probabilities are given by
\[
\begin{equation*}
\operatorname{Pr}\left(\widetilde{y}_{j}=h \mid s_{j}=1, \mathbf{x}_{j}\right)=\Phi\left(\kappa_{h}-\mathbf{x}_{j} \boldsymbol{\beta}\right)-\Phi\left(\kappa_{h-1}-\mathbf{x}_{j} \boldsymbol{\beta}\right) \quad h=0,1, \ldots, H \tag{2}
\end{equation*}
\]
where \(\kappa_{-1}=-\infty, \kappa_{H}=+\infty\), and \(\mathbf{x}_{j}\) is a vector of covariates that could be different from \(\mathbf{z}_{j} . \kappa_{h}\) are boundary parameters that need to be estimated in addition to the coefficients vector \(\boldsymbol{\beta}\).

The intercept \(\beta_{0}\) is set equal to 0 in (2) for identification. Note that \(s_{j}\) and \(\widetilde{y}_{j}\) are both unobservable in terms of the zeros. The observed response variable is \(y_{j}=s_{j} \widetilde{y}_{j}\). Thus, the zero outcome occurs when \(s_{j}=0\) (the individual is not a participant) or occurs when \(s_{j}=1\) and \(\widetilde{y}_{j}=0\) (the individual is a participant with zero activity). To observe a positive \(y_{j}\), it is a joint requirement that \(s_{j}=1\) and \(\widetilde{y}_{j}>0\).

The distribution of \(Y\) is given by
\[
\begin{align*}
\operatorname{Pr}(Y) & =\left\{\begin{array}{l}
\operatorname{Pr}\left(y_{j}=0 \mid \mathbf{z}_{j}, \mathbf{x}_{j}\right) \\
\operatorname{Pr}\left(y_{j}=h \mid \mathbf{z}_{j}, \mathbf{x}_{j}\right) \quad h=1,2, \ldots, H
\end{array}\right. \\
& =\left\{\begin{array}{l}
\operatorname{Pr}\left(s_{j}=0 \mid \mathbf{z}_{j}\right)+\operatorname{Pr}\left(s_{j}=1 \mid \mathbf{z}_{j}\right) \operatorname{Pr}\left(\widetilde{y}_{j}=0 \mid s_{j}=1, \mathbf{x}_{j}\right) \\
\operatorname{Pr}\left(s_{j}=1 \mid \mathbf{z}_{j}\right) \operatorname{Pr}\left(\widetilde{y}_{j}=h \mid s_{j}=1, \mathbf{x}_{j}\right) \quad h=1,2, \ldots, H
\end{array}\right. \tag{3}
\end{align*}
\]

The probability of zero outcome has been inflated because it is the sum of the probability of zero activity from the ordered probit model and the probability of nonparticipation from the probit model.

Substituting (1) and (2) in (3), we get
\[
\begin{aligned}
\operatorname{Pr}(Y) & =\left\{\begin{array}{l}
\operatorname{Pr}\left(y_{j}=0 \mid \mathbf{z}_{j}, \mathbf{x}_{j}\right) \\
\operatorname{Pr}\left(y_{j}=h \mid \mathbf{z}_{j}, \mathbf{x}_{j}\right) \quad h=1,2, \ldots, H-1 \\
\operatorname{Pr}\left(y_{j}=H \mid \mathbf{z}_{j}, \mathbf{x}_{j}\right)
\end{array}\right. \\
& =\left\{\begin{array}{l}
\left\{1-\Phi\left(\mathbf{z}_{j} \boldsymbol{\gamma}\right)\right\}+\Phi\left(\mathbf{z}_{j} \gamma\right) \Phi\left(\kappa_{0}-\mathbf{x}_{j} \boldsymbol{\beta}\right) \\
\Phi\left(\mathbf{z}_{j} \gamma\right)\left\{\Phi\left(\kappa_{h}-\mathbf{x}_{j} \boldsymbol{\beta}\right)-\Phi\left(\kappa_{h-1}-\mathbf{x}_{j} \boldsymbol{\beta}\right)\right\} \quad h=1,2, \ldots, H-1 \\
\Phi\left(\mathbf{z}_{j} \gamma\right)\left\{1-\Phi\left(\kappa_{H-1}-\mathbf{x}_{j} \boldsymbol{\beta}\right)\right\}
\end{array}\right.
\end{aligned}
\]

If the respective offset () option is specified, \(\mathbf{x}_{j} \boldsymbol{\beta}\) and \(\mathbf{z}_{j} \gamma\) are replaced with \(\mathbf{x}_{j} \boldsymbol{\beta}+\operatorname{offset}_{j}^{\beta}\) and \(\mathbf{z}_{j} \gamma+\operatorname{offset}_{j}^{\gamma}\), respectively.

The log-likelihood function is
\[
\ln L=\sum_{j=1}^{N} w_{j} \sum_{h=0}^{H} I\left(y_{j}=h\right) \ln \left\{\operatorname{Pr}\left(y_{j}=h \mid \mathbf{z}_{j}, \mathbf{x}_{j}\right)\right\}
\]
where \(w_{j}\) is an optional weight for the \(j\) th observation and
\[
I\left(y_{j}=h\right)= \begin{cases}1 & \text { if } y_{j}=h \\ 0 & \text { otherwise }\end{cases}
\]

The choice between the ZIOP model and the ordered probit model cannot be made using a likelihoodratio test because the two hypotheses are not nested in the usual sense of parameter restrictions. The restriction \(\gamma=0\) does not eliminate the inflation effect; it makes the group membership probabilities both equal to 0.5 [see (1)]. What is needed to remove the inflation effect is \(\mathbf{z}_{j} \gamma \rightarrow \infty\), which cannot be imposed. Because ZIOP and ordered probit models are not nested, you can compare the fits of the two models using information criteria.

This command supports the Huber/White/sandwich estimator of the variance and its clustered version with vce(robust) and vce(cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.
zioprobit also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

\section*{References}

Bagozzi, B. E., D. W. Hill, Jr., W. H. Moore, and B. Mukherjee. 2015. Modeling two types of peace: The zero-inflated ordered probit (ZiOP) model in conflict research. Journal of Conflict Resolution 59: 728-752. https://doi.org/10.1177/0022002713520530.
Dale, D., and A. Sirchenko. 2021. Estimation of nested and zero-inflated ordered probit models. Stata Journal 21: 3-38.

Downward, P., F. Lera-Lopez, and S. Rasciute. 2011. The zero-inflated ordered probit approach to modelling sports participation. Economic Modelling 28: 2469-2477. https://doi.org/10.1016/j.econmod.2011.06.024.
Ganguly, I., C. T. Koebel, and R. A. Cantrell. 2010. A categorical modeling approach to analyzing new product adoption and usage in the context of the building-materials industry. Technological Forecasting and Social Change 77: 662-677. https://doi.org/10.1016/j.techfore.2009.10.011.
Greene, W. H., and D. A. Hensher. 2010. Modeling Ordered Choices: A Primer. New York: Cambridge University.
Gurmu, S., and G. A. Dagne. 2012. Bayesian approach to zero-inflated bivariate ordered probit regression model, with an application to tobacco use. Journal of Probability and Statistics 2012: 1-26. https://doi.org/10.1155/2012/617678.

Harris, M. N., and X. Zhao. 2007. A zero-inflated ordered probit model, with an application to modelling tobacco consumption. Journal of Econometrics 141: 1073-1099. https://doi.org/10.1016/j.jeconom.2007.01.002.
Kelley, M. E., and S. J. Anderson. 2008. Zero inflation in ordinal data: Incorporating susceptibility to response through the use of a mixture model. Statistics in Medicine 27: 3674-3688. https://doi.org/10.1002/sim.3267.

Lambert, D. 1992. Zero-inflated Poisson regression, with an application to defects in manufacturing. Technometrics 34: 1-14. https://doi.org/10.2307/1269547.
Long, J. S., and J. Freese. 2014. Regression Models for Categorical Dependent Variables Using Stata. 3rd ed. College Station, TX: Stata Press.

Xia, Y., Y. Zhou, and T. Cai. 2019. gidm: A command for generalized inflated discrete models. Stata Journal 19: 698-718.

\footnotetext{
Also see
[R] zioprobit postestimation - Postestimation tools for zioprobit
[R] oprobit - Ordered probit regression
[R] probit - Probit regression
[R] ziologit - Zero-inflated ordered logit regression
[BAYES] bayes: zioprobit - Bayesian zero-inflated ordered probit regression
[SVY] svy estimation - Estimation commands for survey data
[U] 20 Estimation and postestimation commands
}

\section*{Title}
zioprobit postestimation - Postestimation tools for zioprobit
\begin{tabular}{lll} 
Postestimation commands & predict & margins \\
Remarks and examples & Methods and formulas & Also see
\end{tabular}

\section*{Postestimation commands}

The following postestimation commands are available after zioprobit:
\begin{tabular}{|c|c|}
\hline Command & Description \\
\hline contrast & contrasts and ANOVA-style joint tests of estimates \\
\hline estat ic & Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) \\
\hline estat summarize & summary statistics for the estimation sample \\
\hline estat vce & variance-covariance matrix of the estimators (VCE) \\
\hline estat (svy) & postestimation statistics for survey data \\
\hline estimates & cataloging estimation results \\
\hline etable & table of estimation results \\
\hline * forecast & dynamic forecasts and simulations \\
\hline * hausman & Hausman's specification test \\
\hline lincom & point estimates, standard errors, testing, and inference for linear combinations of coefficients \\
\hline * 1 rtest & likelihood-ratio test \\
\hline margins & marginal means, predictive margins, marginal effects, and average marginal effects \\
\hline marginsplot & graph the results from margins (profile plots, interaction plots, etc.) \\
\hline nlcom & point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients \\
\hline predict & probabilities, linear predictions and their SEs, etc. \\
\hline predictnl & point estimates, standard errors, testing, and inference for generalized predictions \\
\hline pwcompare & pairwise comparisons of estimates \\
\hline suest & seemingly unrelated estimation \\
\hline test & Wald tests of simple and composite linear hypotheses \\
\hline testnl & Wald tests of nonlinear hypotheses \\
\hline
\end{tabular}

\footnotetext{
*forecast, hausman, and lrtest are not appropriate with svy estimation results.
}

\section*{predict}

\section*{Description for predict}
predict creates a new variable containing predictions such as probabilities, linear predictions, and standard errors.

\section*{Menu for predict}

Statistics \(>\) Postestimation

\section*{Syntax for predict}
```

predict [type] {stub*|newvar|newvarlist } [if] [in] [, statistic
outcome(outcome) nooffset]
predict [type] stub* [if] [in], scores
statistic Description

```

\section*{Main}
\begin{tabular}{ll} 
pmargin & \begin{tabular}{l} 
marginal probabilities of levels, \(\operatorname{Pr}\left(y_{j}=h\right) ;\) the default \\
pjoint1
\end{tabular} \\
joint probabilities of levels and participation, \(\operatorname{Pr}\left(y_{j}=h, s_{j}=1\right)\) \\
pcond1 & probabilities of levels conditional on participation, \(\operatorname{Pr}\left(y_{j}=h \mid s_{j}=1\right)\) \\
ppar & probability of participation, \(\operatorname{Pr}\left(s_{j}=1\right)\) \\
pnpar & probability of nonparticipation, \(\operatorname{Pr}\left(s_{j}=0\right)\) \\
xb & linear prediction \\
xbinf1 & linear prediction for inflation equation \\
stdp & standard error of the linear prediction \\
stdpinfl & standard error of the linear prediction for inflation equation
\end{tabular}

If you do not specify outcome(), pmargin, pjoint1, and pcond1 (with one new variable specified) assume outcome(\#1).
You specify one or \(k\) new variables with pmargin, pjoint1, and pcond1, where \(k\) is the number of outcomes.
You specify one new variable with ppar, pnpar, xb, xbinfl, stdp, and stdpinfl.
These statistics are available both in and out sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

\section*{Options for predict}
\(\qquad\) Main
pmargin, the default, calculates the predicted marginal probabilities of outcome levels, \(\operatorname{Pr}\left(y_{j}=h\right)\). pjoint1 calculates the predicted joint probabilities of outcome levels and participation, \(\operatorname{Pr}\left(y_{j}=\right.\) \(\left.h, s_{j}=1\right)\).
pcond1 calculates the predicted probabilities of outcome levels conditional on participation, \(\operatorname{Pr}\left(y_{j}=\right.\) \(\left.h \mid s_{j}=1\right)\).

With pmargin, pjoint1, and pcond1, you can compute predicted probabilities for one or for all outcome levels. When you specify one new variable, predict computes probabilities for the first outcome level. You can specify the outcome ( \(\# i\) ) option to obtain probabilities for the \(i\) th level. When you specify multiple new variables or a stub, predict computes probabilities for all outcome levels. The behavior of predict with one new variable is equivalent to specifying outcome (\#1).
ppar and pnpar calculate the predicted marginal probability of participation \(\left[\operatorname{Pr}\left(s_{j}=1\right)\right]\) and of nonparticipation \(\left[\operatorname{Pr}\left(s_{j}=0\right)\right]\), respectively.
In health-related fields, probabilities of participation and nonparticipation are known as probabilities of susceptibility and nonsusceptibility. Similarly to predict after ziologit, you can use options ps and pns to compute these probabilities. Options ps and pns produce identical results to the respective options ppar and pnpar but label new variables as \(\operatorname{Pr}\) (susceptible) and \(\operatorname{Pr}\) (nonsusceptible) instead of \(\operatorname{Pr}\) (participation) and \(\operatorname{Pr}\) (nonparticipation).
xb calculates the linear prediction for the ordered probit equation, which is \(\mathbf{x}_{j} \boldsymbol{\beta}\) if offset () was not specified with zioprobit and is \(\mathbf{x}_{j} \boldsymbol{\beta}+\operatorname{offset}_{j}^{\beta}\) if offset() was specified.
xbinfl calculates the linear prediction for the inflation equation, which is \(\mathbf{z}_{j} \gamma\) if offset () was not specified in inflate() and is \(\mathbf{z}_{j} \gamma+\operatorname{offset}_{j}^{\gamma}\) if offset() was specified in inflate().
stdp calculates the standard error of the linear prediction for the ordered probit equation.
stdpinfl calculates the standard error of the linear prediction for the inflation equation.
outcome(outcome) specifies the outcome for which predicted probabilities are to be calculated. outcome () should contain either one value of the dependent variable or one of \#1, \#2, ..., with \#1 meaning the first category of the dependent variable, \#2 meaning the second category, etc. outcome() is allowed only with pmargin, pjoint1, and pcond1.
nooffset is relevant only if you specified offset(varname) with zioprobit or within the inflate() option. It modifies the calculations made by predict so that they ignore the offset variable; that is, the linear prediction for the main regression equation is treated as \(\mathbf{x}_{j} \boldsymbol{\beta}\) rather than as \(\mathbf{x}_{j} \boldsymbol{\beta}+\operatorname{offset}_{j}^{\beta}\) and the linear prediction for the inflation equation is treated as \(\mathbf{z}_{j} \gamma\) rather than as \(\mathbf{z}_{j} \gamma+\operatorname{offset}_{j}^{\gamma}\).
scores calculates equation-level score variables.
The first new variable will contain \(\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)\). In the absence of independent variables in the main equation, this variable is not stored.
The second new variable will contain \(\partial \ln L / \partial\left(\mathbf{z}_{j} \gamma\right)\).
When the dependent variable takes \(k\) different values, the third new variable through new variable \(k+1\) will contain \(\partial \ln L / \partial\left(\kappa_{h}\right)\) for \(h=0,1, \ldots, k-2\).

\section*{margins}

\section*{Description for margins}
margins estimates margins of response for probabilities and linear predictions.

\section*{Menu for margins}

Statistics \(>\) Postestimation

\section*{Syntax for margins}
```

margins [marginlist] [, options]
margins [marginlist] , predict(statistic ...) [predict(statistic ...)...] [options]

```
statistic Description
\begin{tabular}{ll} 
default & \begin{tabular}{l} 
marginal probabilities for each outcome \\
marginal probabilities of levels, \(\operatorname{Pr}\left(y_{j}=h\right) ;\) the default
\end{tabular} \\
pmargin & joint probabilities of levels and participation, \(\operatorname{Pr}\left(y_{j}=h, s_{j}=1\right)\) \\
pcond1 & probabilities of levels conditional on participation, \(\operatorname{Pr}\left(y_{j}=h \mid s_{j}=1\right)\) \\
ppar & probability of participation, \(\operatorname{Pr}\left(s_{j}=1\right)\) \\
pnpar & probability of nonparticipation, \(\operatorname{Pr}\left(s_{j}=0\right)\) \\
xb & linear prediction \\
xbinfl & linear prediction for inflation equation \\
stdp & not allowed with margins \\
stdpinfl & not allowed with margins
\end{tabular}
pmargin, pjoint1, and pcond1 default to the first outcome.

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [R] margins.

\section*{Remarks and examples}

Various sets of predictions and marginal effects may be of interest for the ZIOP model. For instance, we may want to investigate the marginal effects of a covariate on the probability of participation, or on the probabilities for levels of consumption conditional on participation, or on the overall probabilities for different consumption levels. We explore these options in greater detail in the following examples.
> Example 1: Average marginal effects on probability of nonparticipation
In example 1 of \([R]\) zioprobit, we fit a model for level of cigarette consumption.
```

. use https://www.stata-press.com/data/r18/tobacco
(Fictional tobacco consumption data)
. zioprobit tobacco education income i.female age,
> inflate(education income i.parent age i.female i.religion)
(output omitted)

```

We can use margins to estimate the expected marginal effect of gender for individuals with a college degree (17 years of education) and a smoking parent on the probability of nonparticipation (being a genuine nonsmoker). To do this, we specify predict (pnpar) with margins as follows:
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline \multicolumn{5}{|l|}{Average marginal effects Model VCE: OIM} & \multicolumn{2}{|l|}{\multirow[t]{2}{*}{Number of obs \(=15,000\)}} \\
\hline \multicolumn{5}{|l|}{```
Expression: Pr(nonparticipation), predict(pnpar)
dy/dx wrt: 1.female
At: education = 17
    parent = 1
```} & & \\
\hline & \[
d y / d x
\] & \begin{tabular}{l}
lta-meth \\
std. err
\end{tabular} & z & \(P>|z|\) & [95\% conf & terval] \\
\hline female
Female & . 0855995 & . 0100239 & 8.54 & 0.000 & . 0659531 & . 105246 \\
\hline
\end{tabular}

Note: dy/dx for factor levels is the discrete change from the base level.
Women with a college degree and a smoking parent are expected to have about an \(8.5 \%\) higher chance of being genuine nonsmokers than do men.

\section*{\(>\) Example 2: Predicted probabilities of zero-valued outcomes}

In example 1 of \([R]\) zioprobit, we found that the coefficient on income was positive in the level equation but negative in the participation equation. In the case of our tobacco consumption example, economic theory offers a reasonable interpretation for this. Higher income may act as an indicator for health awareness, which accounts for its association with an increased probability of being a genuine nonsmoker. However, if cigarettes are a normal good-that is, something for which demand increases when income increases-then smokers with higher income should have a lower probability of having zero consumption at the time of the survey.

We first consider the effect of income at six prespecified values ranging from \(\$ 10,000\) to \(\$ 60,000\) on the probability of being a genuine nonsmoker (nonparticipation). Because we know the values 1 to 6 correspond to income in tens of thousands, we conserve space and suppress the default legend by using the noatlegend option.
```

. margins, predict(pnpar) at(income = (1/6)) noatlegend
Predictive margins Number of obs = 15,000
Model VCE: OIM
Expression: Pr(nonparticipation), predict(pnpar)

```
\begin{tabular}{c|rrrrrr}
\hline & \multicolumn{4}{|c}{\begin{tabular}{c} 
Delta-method \\
std. err.
\end{tabular}} & z & \(\mathrm{P}>|\mathrm{z}|\) \\
\hline Margin & [95\% conf. interval] \\
\hline 1 & & & & & & \\
2 & .1727928 & .0056963 & 30.33 & 0.000 & .1616283 & .1839573 \\
3 & .2176784 & .0055236 & 35.20 & 0.000 & .1836304 & .2052824 \\
4 & .0052981 & 41.09 & 0.000 & .2072945 & .2280629 \\
5 & .2424141 & .0050413 & 48.09 & 0.000 & .2325333 & .2522948 \\
6 & .2961309 & .0047857 & 56.12 & 0.000 & .259215 & .2779745 \\
\hline
\end{tabular}

The probability of being a genuine nonsmoker increases with income. For instance, for individuals who earn \(\$ 10,000\) a year, the expected increase in the probability of being a genuine nonsmoker is about \((0.1945-0.1728) \times 100 \%=2.17 \%\) if they earn an additional \(\$ 10,000\).

We next investigate the effect of income on the joint probability of being a smoker (participation equals 1) with zero consumption. We do this by specifying predict(pjoint1 outcome(0)) with margins:
```

. margins, predict(pjoint1 outcome(0)) at(income = (1/6)) noatlegend
Predictive margins Number of obs = 15,000
Model VCE: OIM
Expression: Pr(tobacco=0, participation=1), predict(pjoint1 outcome(0))

```
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline & \multicolumn{3}{|l|}{\begin{tabular}{l}
Delta-method \\
Margin std. err.
\end{tabular}} & \(P>|z|\) & \multicolumn{2}{|l|}{[95\% conf. interval]} \\
\hline _at & & & & & & \\
\hline 1 & . 5595131 & . 0040167 & 139.30 & 0.000 & . 5516405 & . 5673856 \\
\hline 2 & . 5080066 & . 0037793 & 134.42 & 0.000 & . 5005993 & . 5154139 \\
\hline 3 & . 4558656 & . 0035553 & 128.22 & 0.000 & . 4488975 & . 4628338 \\
\hline 4 & . 4029089 & . 0033683 & 119.62 & 0.000 & . 3963072 & . 4095106 \\
\hline 5 & . 3485692 & . 0032481 & 107.32 & 0.000 & . 3422031 & . 3549352 \\
\hline 6 & . 2929155 & . 00317 & 92.40 & 0.000 & . 2867023 & . 2991287 \\
\hline
\end{tabular}

For individuals who are smokers, the probability of zero consumption decreases as income increases, suggesting that tobacco is like a normal good for smokers. For example, for individuals who earn \(\$ 10,000\) a year, earning an additional \(\$ 10,000\) will decrease their probability of being a smoker with zero consumption by about \((0.5595-0.5080) \times 100 \%=5.15 \%\).

If we wanted to compute the effect of income on the overall probability of zero consumption instead of the probability of zero consumption among smokers, we would omit pjoint1 from within the predict option.
```

. margins, predict(outcome(0)) at(income = (1/6)) noatlegend
(output omitted)

```

This version of the margins command gives the sum of the probability of nonparticipation and the joint probability of participation with zero consumption.

\section*{Methods and formulas}

The participation equation is
\[
s_{j}=I\left(\mathbf{z}_{j} \gamma+u_{1 j}>0\right) \quad j=1,2, \ldots, n
\]
where \(s_{j}\) is 1 if the \(j\) th subject belongs to the participation group (for example, smokers) and is 0 if the subject belongs to the nonparticipation group (for example, genuine nonsmokers), \(\mathbf{z}_{j}\) are the covariates used to model the group membership, \(\gamma\) is a vector of coefficients, and \(u_{1 j}\) is a random-error term following a standard normal distribution.

The ordinal outcome equation is
\[
\widetilde{y}_{j}=\sum_{h=0}^{H} h I\left(\kappa_{h-1}<\mathbf{x}_{j} \boldsymbol{\beta}+u_{2 j} \leq \kappa_{h}\right)
\]
where \(\widetilde{y}_{j}\) is the ordinal outcome conditional on participation, \(\mathbf{x}_{j}\) are the outcome covariates, \(\boldsymbol{\beta}\) are the coefficients, and \(u_{2 j}\) is a random-error term following a standard normal distribution. The observed outcome values are \(0,1, \ldots, H . \kappa_{0}, \kappa_{1}, \ldots, \kappa_{H-1}\) are real numbers such that \(\kappa_{i}<\kappa_{m}\) for \(i<m\). \(\kappa_{-1}\) is taken as \(-\infty\) and \(\kappa_{H}\) is taken as \(+\infty\). We assume that the error terms \(u_{1 j}\) and \(u_{2 j}\) are independent. We observe \(y_{j}=s_{j} \widetilde{y}_{j}\).

The probability of participation is
\[
\begin{equation*}
\operatorname{Pr}\left(s_{j}=1 \mid \mathbf{z}_{j}\right)=\Phi\left(\mathbf{z}_{j} \gamma\right) \tag{1}
\end{equation*}
\]
where \(\Phi(\cdot)\) is the standard normal distribution function.
The probability of nonparticipation is
\[
\begin{equation*}
\operatorname{Pr}\left(s_{j}=0 \mid \mathbf{z}_{j}\right)=1-\Phi\left(\mathbf{z}_{j} \boldsymbol{\gamma}\right) \tag{2}
\end{equation*}
\]

The probability of outcome \(y_{j}=h\) given that the \(j\) th subject belongs to the participation group is
\[
\begin{align*}
\operatorname{Pr}\left(y_{j}=h \mid s_{j}=1, \mathbf{x}_{j}\right) & =\operatorname{Pr}\left(\widetilde{y}_{j}=h \mid \mathbf{x}_{j}\right) \\
& =\Phi\left(\kappa_{h}-\mathbf{x}_{j} \boldsymbol{\beta}\right)-\Phi\left(\kappa_{h-1}-\mathbf{x}_{j} \boldsymbol{\beta}\right) \tag{3}
\end{align*}
\]
for \(h=0,1, \ldots, H\), where \(\Phi\left(\kappa_{-1}-\mathbf{x}_{j} \boldsymbol{\beta}\right)=0\) and \(\Phi\left(\kappa_{H}-\mathbf{x}_{j} \boldsymbol{\beta}\right)=1\).
The joint probability of outcome \(y_{j}=h\) and participation can be expressed as
\[
\operatorname{Pr}\left(y_{j}=h, s_{j}=1 \mid \mathbf{z}_{j}, \mathbf{x}_{j}\right)=\operatorname{Pr}\left(s_{j}=1 \mid \mathbf{z}_{j}\right) \operatorname{Pr}\left(y_{j}=h \mid s_{j}=1, \mathbf{x}_{j}\right)
\]
for \(h=0,1, \ldots, H\), and computed using (1) and (3).
The marginal probabilities of the outcome \(y_{j}\) are
\[
\begin{aligned}
\operatorname{Pr}\left(y_{j}=0 \mid \mathbf{z}_{j}, \mathbf{x}_{j}\right) & =\operatorname{Pr}\left(s_{j}=0 \mid \mathbf{z}_{j}\right)+\operatorname{Pr}\left(s_{j}=1 \mid \mathbf{z}_{j}\right) \operatorname{Pr}\left(\widetilde{y}_{j}=0 \mid \mathbf{x}_{j}\right) \\
\operatorname{Pr}\left(y_{j}=h \mid \mathbf{z}_{j}, \mathbf{x}_{j}\right) & =\operatorname{Pr}\left(s_{j}=1 \mid \mathbf{z}_{j}\right) \operatorname{Pr}\left(\widetilde{y}_{j}=h \mid \mathbf{x}_{j}\right) \quad h=1,2, \ldots, H-1 \\
\operatorname{Pr}\left(y_{j}=H \mid \mathbf{z}_{j}, \mathbf{x}_{j}\right) & =\operatorname{Pr}\left(s_{j}=1 \mid \mathbf{z}_{j}\right) \operatorname{Pr}\left(\widetilde{y}_{j}=H \mid \mathbf{x}_{j}\right)
\end{aligned}
\]
and can be computed using (1), (2), and (3).
If the offset () option is specified with zioprobit, \(\mathbf{x}_{j} \boldsymbol{\beta}\) is replaced with \(\mathbf{x}_{j} \boldsymbol{\beta}+\operatorname{offset}_{j}^{\beta}\). If the offset () option is specified within the inflate() option, \(\mathbf{z}_{j} \gamma\) is replaced with \(\mathbf{z}_{j} \gamma+\operatorname{offset}_{j}^{\gamma}\).

\section*{Also see}
[R] zioprobit - Zero-inflated ordered probit regression
[U] 20 Estimation and postestimation commands

\section*{Title}
zip - Zero-inflated Poisson regression
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
zip fits a zero-inflated Poisson (ZIP) model to count data with excess zero counts. The ZIP model assumes that the excess zero counts come from a logit or probit model and the remaining counts come from a Poisson model.

\section*{Quick start}

Zero-inflated Poisson model of y on x 1 and x 2 with inflation modeled using x 3
zip y x1 x2, inflate(x3)
Use a probit model instead of a logit model to predict excess zeros
zip y x1 x2, inflate(x3) probit

\section*{Menu}

Statistics \(>\) Count outcomes \(>\) Zero-inflated Poisson regression

\section*{Syntax}
```

zip depvar [indepvars] [if][in] [weight],
inflate(varlist[, offset(varname)]|_cons) [options]

```
options
Model
*inflate ( )
noconstant
exposure (varname \({ }_{e}\) )
offset (varname \({ }_{o}\) )
constraints (constraints)
probit
SE/Robust
vce(vcetype)

Reporting
level (\#)
irr
nocnsreport
display_options

Description
equation that determines whether the count is zero
suppress constant term
include \(\ln \left(\right.\) varname \(\left._{e}\right)\) in model with coefficient constrained to 1 include varname \(_{o}\) in model with coefficient constrained to 1
apply specified linear constraints
use probit model to characterize excess zeros; default is logit
vcetype may be oim, robust, cluster clustvar, opg, bootstrap, or jackknife
set confidence level; default is level (95)
report incidence-rate ratios
do not display constraints
control columns and column formats, row spacing, line width, display of omitted variables and base and empty cells, and factor-variable labeling
control the maximization process; seldom used
keep collinear variables
display legend instead of statistics

\section*{Maximization}
maximize_options
collinear
coeflegend
*inflate (varlist \([\), offset(varname) \(] \mid\) _cons) is required. indepvars and varlist may contain factor variables; see [U] 11.4.3 Factor variables.
bayes, bootstrap, by, collect, fp, jackknife, rolling, statsby, and svy are allowed; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: zip.
Weights are not allowed with the bootstrap prefix; see [R] bootstrap.
vce() and weights are not allowed with the svy prefix; see [SVY] svy.
fweights, iweights, and pweights are allowed; see [U] 11.1.6 weight.
collinear and coeflegend do not appear in the dialog box.
See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

\section*{Options}
\(\qquad\) Model
inflate(varlist [, offset(varname)] _ cons) specifies the equation that determines whether the observed count is zero. Conceptually, omitting inflate() would be equivalent to fitting the model with poisson; see [R] poisson.
inflate (varlist \([\), offset (varname) \(]\) ) specifies the variables in the equation. You may optionally include an offset for this varlist.
inflate (_cons) specifies that the equation determining whether the count is zero contains only an intercept. To run a zero-inflated model of depvar with only an intercept in both equations, type zip depvar, inflate(_cons).
noconstant, exposure (varname \({ }_{e}\) ), offset ( varname \(_{o}\) ), constraints (constraints); see [R] Estimation options.
probit requests that a probit, instead of logit, model be used to characterize the excess zeros in the data.

SE/Robust
vce (vcetype) specifies the type of standard error reported, which includes types that are derived from asymptotic theory (oim, opg), that are robust to some kinds of misspecification (robust), that allow for intragroup correlation (cluster clustvar), and that use bootstrap or jackknife methods (bootstrap, jackknife); see [R] vce_option.
\(\qquad\) Reporting
level (\#); see [R] Estimation options.
irr reports estimated coefficients transformed to incidence-rate ratios, that is, \(e^{b}\) rather than \(b\). Standard errors and confidence intervals are similarly transformed. This option affects how results are displayed, not how they are estimated or stored. irr may be specified at estimation or when replaying previously estimated results.
nocnsreport; see [R] Estimation options.
display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap (\#), fvwrapon (style), cformat (\%fmt), pformat (\% fmt), sformat (\% fmt), and nolstretch; see [R] Estimation options.

Maximization
maximize_options: difficult, technique (algorithm_spec), iterate (\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from (init_specs); see [R] Maximize. These options are seldom used.
Setting the optimization type to technique (bhhh) resets the default vcetype to vce (opg).
The following options are available with zip but are not shown in the dialog box:
collinear, coeflegend; see [R] Estimation options.

\section*{Remarks and examples}

Zero-inflated Poisson (ZIP) models address the case when the data contain a higher fraction of zeros than is likely to be generated from a Poisson model. Having a large proportion of zero observations, in itself, does not necessarily mean that we have the excess zeros problem. For instance, a Poisson model with a mean value of 0.2 predicts that \(P(Y=0)=\exp (-0.2) \approx 0.82\). However, the range of possible outcomes is restricted because of the small variance, 0.2 (mean and variance are equal for a Poisson model), as shown by \(P(Y>3) \approx 0.00006\). Unlike Poisson models, ZIP models allow us to have a large fraction of zeros without restricting the range of outcomes.

ZIP models assume that an observation is 0 with a probability \(p\) or is a realization of a Poisson random variable, which can also be 0 , with a probability \(1-p\). For instance, you might count how many fish each visitor to a park catches. Many visitors may catch zero, because they do not fish (as opposed to being unsuccessful). Using a logit or probit model, you may model the probability \(p\) of whether a person does not fish depending on several covariates related to fishing. Using a Poisson distribution, you may model how many fish a person catches depending on several covariates having to do with the success of catching fish (type of lure or bait, time of day, temperature, season, etc.). This is the type of data for which the zip command is useful.

See Long (1997, 242-247) and Cameron and Trivedi (2005, 680-681) for a discussion of the ZIP model and other zero-modified count models.

\section*{> Example 1: Fitting a ZIP model}

We have fictional data on the number of fish caught (count) by visitors to a national park on a particular day. Some of the visitors do not fish, but we do not have the data on whether a person fished; we merely have data on how many fish were caught together with several covariates.

Variable count exhibits an excess of zero observations (142 of 250 observations), beyond what would be expected from a Poisson model. We suspect that the number of zeros may be inflated because many visitors are not fishing. That is, a zero observation may be the result of a visitor who was unfortunate and caught no fish but may also be because the visitor did not fish. A standard Poisson model (see [R] poisson) treats these two types of zero observations as a homogeneous group, which typically leads to biased statistical results. We would like to distinguish between the two types of zeros and possibly draw inference for them separately (see example 1 in [R] zip postestimation).

The zip command allows us to model the two types of zeros. First, using the required option inflate(), we model whether a visitor fishes as a function of the number of children accompanying him or her (child) and whether he or she is camping (camper). Next, we assume the response variable, count, depends on whether the visitor used a live bait (livebait) and the number of persons (persons), which includes the visitor and any adults or children accompanying him. Note that persons is always greater than child.
```

. use https://www.stata-press.com/data/r18/fish
(Fictional fishing data)
. zip count persons livebait, inflate(child camper)
Fitting constant-only model:
Iteration 0: Log likelihood = -1347.807
Iteration 1: Log likelihood = -1305.3245
Iteration 2: Log likelihood = -1104.3005
Iteration 3: Log likelihood = -1103.9426
Iteration 4: Log likelihood = -1103.9425
Fitting full model:
Iteration 0: Log likelihood = -1103.9425
Iteration 1: Log likelihood = -896.2346
Iteration 2: Log likelihood = -851.61723
Iteration 3: Log likelihood = -850.70435
Iteration 4: Log likelihood = -850.70142
Iteration 5: Log likelihood = -850.70142
Zero-inflated Poisson regression Number of obs = 250
Inflation model: logit Nonzero obs = 108
Zero obs = 142
LR chi2(2) = 506.48
Log likelihood = -850.7014

| Number of obs | $=$ | 250 |
| :--- | :--- | ---: |
| Nonzero obs | $=$ | 108 |
| Zero obs | $=$ | 142 |
| LR chi2 $(2)$ | $=506.48$ |  |
| Prob $>$ chi2 | $=0.0000$ |  |

```
\begin{tabular}{r|rrrrrr}
\hline count & Coefficient & Std. err. & \(z\) & P>|z| & [95\% conf. interval] \\
\hline count & & & & & & \\
persons & .8068853 & .0453288 & 17.80 & 0.000 & .7180424 & .8957281 \\
livebait & 1.757289 & .2446082 & 7.18 & 0.000 & 1.277866 & 2.236713 \\
_cons & -2.178472 & .2860289 & -7.62 & 0.000 & -2.739078 & -1.617865 \\
\hline inflate & & & & & & \\
child & 1.602571 & .2797719 & 5.73 & 0.000 & 1.054228 & 2.150913 \\
camper & -1.015698 & .365259 & -2.78 & 0.005 & -1.731593 & -.2998038 \\
_cons & -.4922872 & .3114562 & -1.58 & 0.114 & -1.10273 & .1181558 \\
\hline
\end{tabular}

Coefficients in the upper half of the table correspond to the Poisson model for individuals who fished. For instance, among visitors who fished, using a live bait increases the expected number of caught fish by a factor of \(\exp (1.7572) \approx 5.8\), holding other covariates constant.

\section*{> Example 2: Comparing model fit}

When you have count data, you may want to test whether a conventional count data model or a zero-inflated count data model is preferable. The classical likelihood-ratio test cannot be used here because the models are not nested. But we can use information criteria such as the AIC and BIC to check whether the standard or zero-inflated model is more appropriate.

Continuing with our fishing example, let's check whether the standard Poisson or ZIP model is more appropriate for our data. First, we store the estimation results from the previous ZIP model by typing
```

. estimates store zip

```

Next, we fit the Poisson model corresponding to the main equation of the ZIP model and store its results as pois:
```

. poisson count persons livebait
(output omitted)

```
. estimates store pois
We use estimates stats to display the AIC and BIC values for the two models.
. estimates stats pois zip
Akaike's information criterion and Bayesian information criterion
\begin{tabular}{r|rrrrrr}
\hline Model & N & ll(null) & ll(model) & df & AIC & BIC \\
\hline pois & 250 & -1647.716 & -1312.178 & 3 & 2630.356 & 2640.92 \\
zip & 250 & -1103.942 & -850.7014 & 6 & 1713.403 & 1734.532 \\
\hline
\end{tabular}

Note: BIC uses \(N=\) number of observations. See [R] IC note.
The ZIP model has smaller AIC and BIC values; we thus conclude that it fits our data better than the standard Poisson model.

\section*{Stored results}
zip stores the following in e():
Scalars
```

e(N)
e(N_zero)
e(k)
e(k_eq)
e(k_eq_model)
e(k_dv)
e(df_m)
e(ll)
e(ll_0)
e(N_clust)
e(chi2)
e(p)
e(rank)
e(ic)
e(rc)
e(converged)

```

Macros
e(cmd)
    e(cmdline)
    e(depvar)
    e(inflate)
    e(wtype)
    e(wexp)
    e(title)
    e(clustvar)
    e(offset1)
    e(offset2)
    e(chi2type)
    e(vce)
    e(vcetype)
    e(opt)
number of observations
number of zero observations
number of parameters
number of equations in \(e(b)\)
number of equations in overall model test
number of dependent variables
model degrees of freedom
log likelihood
log likelihood, constant-only model
number of clusters
\(\chi^{2}\)
\(p\)-value for model test
rank of \(e(V)\)
number of iterations
return code
1 if converged, 0 otherwise
zip
command as typed
name of dependent variable
logit or probit
weight type
weight expression
title in estimation output
name of cluster variable
offset
offset for inflate()
Wald or LR; type of model \(\chi^{2}\) test
vcetype specified in vce()
title used to label Std. err.
type of optimization
```

e(which)
e(ml_method)
e(user)
e(technique)
e(properties)
e(predict)
e(asbalanced)
e(asobserved)
Matrices
e(b)
e(Cns)
e(ilog)
e(gradient)
e(V)
e(V_modelbased)

```

Functions
e(sample) marks estimation sample

In addition to the above, the following is stored in r() :

Matrices
\(r\) (table) matrix containing the coefficients with their standard errors, test statistics, \(p\)-values, and confidence intervals

Note that results stored in \(r()\) are updated when the command is replayed and will be replaced when any r-class command is run after the estimation command.

\section*{Methods and formulas}

Consider the formulation of a zero-inflated model as presented in Lambert (1992). Define
\[
\begin{aligned}
\xi_{j}^{\beta} & =\mathbf{x}_{j} \boldsymbol{\beta}+\operatorname{offset}_{j}^{\beta} \\
\xi_{j}^{\gamma} & =\mathbf{z}_{j} \gamma+\operatorname{offset}_{j}^{\gamma} \\
\lambda_{j} & =\exp \left(\xi_{j}^{\beta}\right) \\
F_{j} & =F\left(\xi_{j}^{\gamma}\right)
\end{aligned}
\]
where \(F(\cdot)\) is the inverse of the logit function or, if the probit option was specified, the inverse of the probit function (or the standard normal cumulative distribution function). All subjects are assumed to be independent with the \(j\) th response determined as follows:
\[
\begin{gathered}
Y_{j}=0 \quad \text { with probability } F_{j} \\
Y_{j} \sim \operatorname{Poisson}\left(\lambda_{j}\right) \text { with probability } 1-F_{j}
\end{gathered}
\]

In other words,
\[
\begin{aligned}
& \operatorname{Pr}\left(Y_{j}=0 \mid \mathbf{x}_{j}, \mathbf{z}_{j}\right)=F_{j}+\left(1-F_{j}\right) \exp \left(-\lambda_{j}\right) \\
& \operatorname{Pr}\left(Y_{j}=n \mid \mathbf{x}_{j}, \mathbf{z}_{j}\right)=\left(1-F_{j}\right) \exp \left(-\lambda_{j}\right) \frac{\lambda_{j}^{n}}{n!} \quad \text { for } n=1,2, \ldots
\end{aligned}
\]

The zip command maximizes the \(\log\)-likelihood \(\ln L\), defined by
\[
\begin{aligned}
\ln L= & \sum_{j \in S} w_{j} \ln \left\{F_{j}+\left(1-F_{j}\right) \exp \left(-\lambda_{j}\right)\right\} \\
& +\sum_{j \notin S} w_{j}\left\{\ln \left(1-F_{j}\right)-\lambda_{j}+\xi_{j} y_{j}-\ln \left(y_{j}!\right)\right\}
\end{aligned}
\]
where \(w_{j}\) are the weights and \(S\) is the set of observations for which the observed outcome \(y_{j}=0\).
This command supports the Huber/White/sandwich estimator of the variance and its clustered version using vce(robust) and vce(cluster clustvar), respectively. See [P] _robust, particularly Maximum likelihood estimators and Methods and formulas.
zip also supports estimation with survey data. For details on VCEs with survey data, see [SVY] Variance estimation.

\section*{References}

Cameron, A. C., and P. K. Trivedi. 2005. Microeconometrics: Methods and Applications. New York: Cambridge University Press.
Cummings, T. H., and J. W. Hardin. 2019. Modeling count data with marginalized zero-inflated distributions. Stata Journal 19: 499-509.

Desmarais, B. A., and J. J. Harden. 2013. Testing for zero inflation in count models: Bias correction for the Vuong test. Stata Journal 13: 810-835.
Lambert, D. 1992. Zero-inflated Poisson regression, with an application to defects in manufacturing. Technometrics 34: 1-14. https://doi.org/10.2307/1269547.
Long, J. S. 1997. Regression Models for Categorical and Limited Dependent Variables. Thousand Oaks, CA: Sage.
Long, J. S., and J. Freese. 2001. Predicted probabilities for count models. Stata Journal 1: 51-57.
——. 2014. Regression Models for Categorical Dependent Variables Using Stata. 3rd ed. College Station, TX: Stata Press.
Mullahy, J. 1986. Specification and testing of some modified count data models. Journal of Econometrics 33: 341-365. https://doi.org/10.1016/0304-4076(86)90002-3.
Xia, Y., Y. Zhou, and T. Cai. 2019. gidm: A command for generalized inflated discrete models. Stata Journal 19: 698-718.

\section*{Also see}
[R] zip postestimation - Postestimation tools for zip
[R] zinb - Zero-inflated negative binomial regression
[R] nbreg - Negative binomial regression
[R] poisson - Poisson regression
[R] tnbreg - Truncated negative binomial regression
[R] tpoisson - Truncated Poisson regression
[BAYES] bayes: zip - Bayesian zero-inflated Poisson regression
[SVY] svy estimation - Estimation commands for survey data
[XT] xtpoisson - Fixed-effects, random-effects, and population-averaged Poisson models
[U] 20 Estimation and postestimation commands

\section*{Title}
zip postestimation - Postestimation tools for zip

Postestimation commands
Remarks and examples
Also see
predict margins
Methods and formulas References

\section*{Postestimation commands}

The following postestimation commands are available after zip:

Command Description
\begin{tabular}{|c|c|}
\hline contrast & contrasts and ANOVA-style joint tests of estimates \\
\hline estat ic & Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC) \\
\hline estat summarize & summary statistics for the estimation sample \\
\hline estat vce & variance-covariance matrix of the estimators (VCE) \\
\hline estat (svy) & postestimation statistics for survey data \\
\hline estimates & cataloging estimation results \\
\hline etable & table of estimation results \\
\hline * forecast & dynamic forecasts and simulations \\
\hline * hausman & Hausman's specification test \\
\hline lincom & point estimates, standard errors, testing, and inference for linear combinations of coefficients \\
\hline * Irtest & likelihood-ratio test \\
\hline margins & marginal means, predictive margins, marginal effects, and average marginal effects \\
\hline marginsplot & graph the results from margins (profile plots, interaction plots, etc.) \\
\hline nlcom & point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients \\
\hline predict & number of events, incidence rates, probabilities, etc. \\
\hline predictnl & point estimates, standard errors, testing, and inference for generalized predictions \\
\hline pwcompare & pairwise comparisons of estimates \\
\hline suest & seemingly unrelated estimation \\
\hline test & Wald tests of simple and composite linear hypotheses \\
\hline testnl & Wald tests of nonlinear hypotheses \\
\hline
\end{tabular}

\footnotetext{
*forecast, hausman, and lrtest are not appropriate with svy estimation results.
}

\section*{predict}

\section*{Description for predict}
predict creates a new variable containing predictions such as numbers of events, incidence rates, probabilities, linear predictions, and standard errors.

\section*{Menu for predict}

\author{
Statistics \(>\) Postestimation
}

\section*{Syntax for predict}
```

    predict [type] newvar [if] [in] [, statistic nooffset]
    predict [type] stub* [if] [in], scores
    statistic Description

```
Main
    n number of events; the default
    ir incidence rate
    pr probability of a degenerate zero
    \(\operatorname{pr}(n) \quad\) probability \(\operatorname{Pr}\left(y_{j}=n\right)\)
    \(\operatorname{pr}(a, b) \quad\) probability \(\operatorname{Pr}\left(a \leq y_{j} \leq b\right)\)
    xb linear prediction
    stdp standard error of the linear prediction

These statistics are available both in and out of sample; type predict ... if e(sample) ... if wanted only for the estimation sample.

\section*{Options for predict}
\(\qquad\) Main
n , the default, calculates the predicted number of events, which is \(\left(1-F_{j}\right) \exp \left(\mathbf{x}_{j} \boldsymbol{\beta}\right)\) if neither offset() nor exposure() was specified when the model was fit, where \(F_{j}\) is the predicted probability of a zero outcome; \(\left(1-F_{j}\right) \exp \left(\mathbf{x}_{j} \boldsymbol{\beta}+\operatorname{offset}_{j}^{\beta}\right)\) if offset() was specified; or \(\left(1-F_{j}\right)\left\{\exp \left(\mathbf{x}_{j} \boldsymbol{\beta}\right) \times\right.\) exposure \(\left._{j}\right\}\) if exposure() was specified.
ir calculates the incidence rate, which is the predicted number of events when exposure is 1 . This is equivalent to specifying both the n and the nooffset options.
pr calculates the probability of a degenerate zero, predicted from the fitted degenerate distribution \(F_{j}=\) \(F\left(\mathbf{z}_{j} \gamma\right)\). If offset() was specified within the inflate() option, then \(F_{j}=F\left(\mathbf{z}_{j} \gamma+\operatorname{offset}_{j}^{\gamma}\right)\) is calculated.
\(\operatorname{pr}(n)\) calculates the probability \(\operatorname{Pr}\left(y_{j}=n\right)\), where \(n\) is a nonnegative integer that may be specified as a number or a variable. Note that pr is not equivalent to \(\mathrm{pr}(0)\).
\(\operatorname{pr}(a, b)\) calculates the probability \(\operatorname{Pr}\left(a \leq y_{j} \leq b\right)\), where \(a\) and \(b\) are nonnegative integers that may be specified as numbers or variables;
\(b\) missing ( \(b \geq\).) means \(+\infty\);
\(\operatorname{pr}(20,\).\() calculates \operatorname{Pr}\left(y_{j} \geq 20\right)\);
\(\operatorname{pr}(20, b)\) calculates \(\operatorname{Pr}\left(y_{j} \geq 20\right)\) in observations for which \(b \geq\). and calculates
\(\operatorname{Pr}\left(20 \leq y_{j} \leq b\right)\) elsewhere.
\(\operatorname{pr}(., b)\) produces a syntax error. A missing value in an observation of the variable \(a\) causes a missing value in that observation for \(\operatorname{pr}(a, b)\).
xb calculates the linear prediction, which is \(\mathbf{x}_{j} \boldsymbol{\beta}\) if neither offset () nor exposure() was specified; \(\mathbf{x}_{j} \boldsymbol{\beta}+\operatorname{offset}_{j}^{\beta}\) if offset () was specified; or \(\mathbf{x}_{j} \boldsymbol{\beta}+\ln\) (exposure \(_{j}\) ) if exposure() was specified; see nooffset below.
stdp calculates the standard error of the linear prediction.
nooffset is relevant only if you specified offset() or exposure() when you fit the model. It modifies the calculations made by predict so that they ignore the offset or exposure variable; the linear prediction is treated as \(\mathbf{x}_{j} \boldsymbol{\beta}\) rather than as \(\mathbf{x}_{j} \boldsymbol{\beta}+\operatorname{offset}_{j}^{\beta}\) or \(\mathbf{x}_{j} \boldsymbol{\beta}+\ln \left(\right.\) exposure \(\left._{j}\right)\). Specifying predict..., nooffset is equivalent to specifying predict..., ir.
scores calculates equation-level score variables.
The first new variable will contain \(\partial \ln L / \partial\left(\mathbf{x}_{j} \boldsymbol{\beta}\right)\).
The second new variable will contain \(\partial \ln L / \partial\left(\mathbf{z}_{j} \gamma\right)\).

\section*{margins}

\section*{Description for margins}
margins estimates margins of response for the numbers of events, incidence rates, probabilities, and linear predictions.

\section*{Menu for margins}

Statistics \(>\) Postestimation

\section*{Syntax for margins}
```

margins [marginlist] [, options]

```

statistic Description
\begin{tabular}{ll}
n & number of events; the default \\
ir & incidence rate \\
pr & probability of a degenerate zero \\
\(\operatorname{pr}(n)\) & probability \(\operatorname{Pr}\left(y_{j}=n\right)\) \\
\(\operatorname{pr}(a, b)\) & probability \(\operatorname{Pr}\left(a \leq y_{j} \leq b\right)\) \\
xb & linear prediction \\
\(\operatorname{stdp}\) & not allowed with margins \\
\hline
\end{tabular}

Statistics not allowed with margins are functions of stochastic quantities other than e(b).
For the full syntax, see [ R\(]\) margins.

\section*{Remarks and examples}
> Example 1: Obtaining predicted counts
Continuing with example 1 from \([\mathrm{R}]\) zip, we will use predict to compute the predicted number of fish captured by each individual.
```

. use https://www.stata-press.com/data/r18/fish
(Fictional fishing data)
. zip count persons livebait, inflate(child camper)
(output omitted)
. predict numfish
(option n assumed; predicted number of events)
. summarize numfish

| Variable | Obs | Mean | Std. dev. | Min | Max |
| ---: | ---: | ---: | ---: | ---: | ---: |
| numfish | 250 | 2.770999 | 3.269588 | .079269 | 13.55015 |

```

The average predicted number of fish caught by all visitors, regardless of whether or not they fished, is 2.77 fish.

\section*{> Example 2: Obtaining predicted probabilities}
predict with the pr option computes the probability that an individual does not fish. - predict pr, pr

On the other hand, predict with the \(\operatorname{pr}(n)\) option computes the probability of catching \(n\) fish; particularly, the probability of catching zero fish will be
- predict pr0, pr(0)
. list pr pro in 1


Notice that pr0 is always equal to or greater than pr. For example, for the first individual, the probability of not fishing is 0.38 ; on the other hand, the probability of catching zero fish (0.86) is equal to the sum of the probability of not fishing and the probability of fishing but not catching any fish. pr0 can be also computed as one minus the probability of catching at least one fish, that is:
. predict pr_catch, pr(1,.)
. gen prOb = 1-pr_catch

\section*{Methods and formulas}

See Methods and formulas in \([R]\) zip for the model definition and notation.
The probabilities calculated using the \(\operatorname{pr}(n)\) option are the probability \(\operatorname{Pr}\left(y_{j}=n\right)\). These are calculated using
\[
\begin{aligned}
& \operatorname{Pr}\left(y_{j}=0 \mid \mathbf{x}_{j}, \mathbf{z}_{j}\right)=F_{j}+\left(1-F_{j}\right) \exp \left(-\lambda_{j}\right) \\
& \operatorname{Pr}\left(y_{j}=n \mid \mathbf{x}_{j}, \mathbf{z}_{j}\right)=\left(1-F_{j}\right) \frac{\lambda_{j}^{n} \exp \left(-\lambda_{j}\right)}{n!} \quad \text { for } n=1,2, \ldots
\end{aligned}
\]
where \(F_{j}\) is the probability of obtaining an observation from the degenerate distribution whose mass is concentrated at zero. \(F_{j}\) can be obtained by using the pr option.

See Cameron and Trivedi (2013, sec. 4.6) for further details.

\section*{References}

Cameron, A. C., and P. K. Trivedi. 2013. Regression Analysis of Count Data. 2nd ed. New York: Cambridge University Press.
Manjón, M., and O. Martínez. 2014. The chi-squared goodness-of-fit test for count-data models. Stata Journal 14: 798-816.

\section*{Also see}
[R] zip - Zero-inflated Poisson regression
[U] 20 Estimation and postestimation commands

\section*{Title}
ztest - \(z\) tests (mean-comparison tests, known variance)
\begin{tabular}{llll} 
Description & Quick start & Menu & Syntax \\
Options & Remarks and examples & Stored results & Methods and formulas \\
References & Also see & &
\end{tabular}

\section*{Description}
ztest performs \(z\) tests on the equality of means, assuming known variances. The test can be performed for one sample against a hypothesized population value or for no difference in population means estimated from two samples. Two-sample tests can be conducted for paired and unpaired data. Clustered data are also supported.
ztesti is the immediate form of ztest; see [U] 19 Immediate commands.
For the comparison of means when variances are unknown, use ttest; see \([R]\) ttest.

\section*{Quick start}

One-sample test that the mean of v1 is 3 at the \(90 \%\) confidence level
ztest v1 == 3, level (90)
Same as above, and adjust for clustering with clusters defined by cvar and an intraclass correlation of 0.5
ztest v1 == 3, level(90) cluster(cvar) rho(0.5)
Unpaired \(z\) test that the mean of v 1 is equal between two groups defined by catvar
ztest v1, by(catvar)
Same as above, and adjust for clustering with clusters defined by cvar and an intraclass correlation of 0.5 in the two groups
ztest v1, by(catvar) cluster(cvar) rho(0.5)
Unpaired test of equality of the means of v2 and v3
ztest v2 == v3, unpaired
Paired test of equality of the means of v2 and v3 with standard deviation of the differences between paired observations of 2.4
ztest v2 == v3, sddiff(2.4)
Same as above, specified using a common standard deviation of 2 and correlation between observations of 0.28
ztest v2 == v3, sd(2) corr (0.28)
Immediate form unpaired test of \(\mu_{1}=\mu_{2}\) if \(\bar{x}_{1}=3.2, \operatorname{sd}_{1}=0.1, \bar{x}_{2}=3.4\), and \(\operatorname{sd}_{2}=0.15\) with \(n_{1}=n_{2}=120\)
ztesti 1203.20 .11203 .40 .15

\section*{Menu}

\section*{ztest}

Statistics \(>\) Summaries, tables, and tests \(>\) Classical tests of hypotheses \(>\mathrm{z}\) test (mean-comparison test, known variance)

\section*{ztesti}

Statistics \(>\) Summaries, tables, and tests \(>\) Classical tests of hypotheses \(>\mathrm{z}\) test calculator

\section*{Syntax}

One-sample z test
\[
\text { ztest varname }==\#[\text { if }][\text { in }][\text {, onesampleopts }]
\]

Two-sample \(z\) test using groups
```

ztest varname [if] [in], by(groupvar) [twosamplegropts]

```

Two-sample z test using variables
```

ztest varname }\mp@subsup{\mp@code{N}}{== varname2 [if] [in], unpaired [twosamplevaropts]}{

```

Paired z test
```

ztest varname }\mp@subsup{1}{1}{==}\mp@subsup{\mathrm{ varname }}{2}{}[\mathrm{ [if] [in], sddiff(\#) [llevel(\#)]

```


Immediate form of one-sample z test
```

ztesti \# obs \#mean \#

```

Immediate form of two-sample unpaired \(z\) test
```

ztesti \#\#obs1 \#mean1 \# \#sd1 \# obs2 \#mean2 \# \#sd2 [, level (\#)]

```
onesampleopts
Description

\section*{Main}
sd(\#) one-population standard deviation; default is \(\operatorname{sd}(1)\)
level(\#) confidence level; default is level(95)
cluster(varname) variable defining the clusters
rho(\#) intraclass correlation
```

Main

* by(groupvar) variable defining the groups
unpaired unpaired test; implied when by() is specified
sd(\#) two-population common standard deviation; default is sd(1)
sd1(\#) standard deviation of the first population; requires sd2() and may not be
combined with sd()
sd2(\#) standard deviation of the second population; requires sd1() and may not
be combined with sd()
level(\#)
cluster(varname)
rho(\#)
rho1(\#) intraclass correlation for group 1
rho2(\#) intraclass correlation for group 2

```
*by (groupvar) is required.
twosamplevaropts Description
Main
\begin{tabular}{ll} 
* unpaired & \begin{tabular}{l} 
unpaired test \\
sd(\#) \\
sd1 (\#)
\end{tabular} \\
\begin{tabular}{l} 
two-population common standard deviation; default is \(\operatorname{sd}(1)\) \\
standard deviation of the first population; requires \(\operatorname{sd2}()\) and may not be \\
combined with \(s d()\)
\end{tabular} \\
level(\#) & \begin{tabular}{l} 
standard deviation of the second population; requires sd1() and may not \\
be combined with sd() \\
confidence level; default is level (95)
\end{tabular}
\end{tabular}
*unpaired is required.
pairedopts Description
Main
* corr (\#) correlation between paired observations
sd(\#)
two-population common standard deviation; default is sd(1); may not be combined with sd1(), sd2(), or sddiff()
sd1 (\#) standard deviation of the first population; requires corr () and \(\operatorname{sd2}()\) and may not be combined with \(\operatorname{sd}()\) or sddiff()
sd2 (\#) standard deviation of the second population; requires corr () and sd1() and may not be combined with sd() or sddiff()
level(\#) confidence level; default is level(95)

\footnotetext{
* \(\operatorname{corr}(\#)\) is required.
}
by and collect are allowed with ztest and ztesti; see [U] 11.1.10 Prefix commands.

\section*{Options}
\(\qquad\)
by (groupvar) specifies the groupvar that defines the two groups that ztest will use to test the hypothesis that their means are equal. Specifying by (groupvar) implies an unpaired (two-sample) \(z\) test. Do not confuse the by () option with the by prefix; you can specify both.
unpaired specifies that the data be treated as unpaired. The unpaired option is used when the two sets of values to be compared are in different variables.
sddiff (\#) specifies the population standard deviation of the differences between paired observations for a paired \(z\) test. For this kind of test, either \(\operatorname{sddiff}()\) or corr () must be specified.
\(\operatorname{corr}(\#)\) specifies the correlation between paired observations for a paired \(z\) test. This option along with \(\operatorname{sd1}()\) and \(\operatorname{sd2()}\) ) or with \(\operatorname{sd}()\) is used to compute the standard deviation of the differences between paired observations unless that standard deviation is supplied directly in the sddiff() option. For a paired \(z\) test, either \(\operatorname{sddiff}()\) or corr() must be specified.
\(\operatorname{sd}(\#)\) specifies the population standard deviation for a one-sample \(z\) test or the common population standard deviation for a two-sample \(z\) test. The default is \(\operatorname{sd}(1) . \mathrm{sd}()\) may not be combined with sd1(), sd2(), or sddiff().
sd1(\#) specifies the standard deviation of the first population or group. When sd1() is specified with by (groupvar), the first group is defined by the first category of the sorted groupvar. sd1() requires \(\operatorname{sd2}()\) and may not be combined with \(s d()\) or \(\operatorname{sddiff}()\).
sd2(\#) specifies the standard deviation of the second population or group. When sd2() is specified with by (groupvar), the second group is defined by the second category of the sorted groupvar. \(\operatorname{sd2}()\) requires sd1() and may not be combined with sd() or sddiff().
level(\#) specifies the confidence level, as a percentage, for confidence intervals. The default is level (95) or as set by set level; see [U] 20.8 Specifying the width of confidence intervals.
cluster (varname) specifies the variable that identifies clusters. The cluster() option is required to adjust the computation for clustering.
rho(\#) specifies the intraclass correlation for a one-sample test or the common intraclass correlation for a two-sample test. The rho() option is required to adjust the computation for clustering for a one-sample test.
rho1(\#) specifies the intraclass correlation of the first group for a two-sample test using groups. The rho() option or both rho1() and rho2() options are required to adjust the computation for clustering.
rho2(\#) specifies the intraclass correlation of the second group for a two-sample test using groups. The rho() option or both rho1() and rho2() options are required to adjust the computation for clustering.

When by () is used, sd1() and sd2() or sd() is used to specify the population standard deviations of the two groups defined by groupvar for an unpaired two-sample \(z\) test (using groups). By default, a common standard deviation of one, \(\operatorname{sd}(1)\), is assumed.

When unpaired is used, sd1() and sd2() or \(s d()\) is used to specify the population standard deviations of varname \({ }_{1}\) and varname \(_{2}\) for an unpaired two-sample \(z\) test (using variables). By default, a common standard deviation of one, \(\operatorname{sd}(1)\), is assumed.

Options corr(), sd1(), and sd2() or corr() and \(\operatorname{sd}()\) are used for a paired \(z\) test to compute the standard deviation of the differences between paired observations. By default, a common standard
deviation of one, \(\operatorname{sd}(1)\), is assumed for both populations. Alternatively, the standard deviation of the differences between paired observations may be supplied directly with the sddiff () option.

\section*{Remarks and examples}

Remarks are presented under the following headings:

> One-sample \(z\) test
> Two-sample z test
> Paired z test
> Adjust for clustering Immediate form

For the purpose of illustration, we assume that variances are known in all the examples below.

\section*{One-sample z test}
> Example 1
In the first form, ztest tests whether the mean of the sample is equal to a known constant under the assumption of known variance. Assume that we have a sample of 74 automobiles. We know each automobile's average mileage rating and wish to test whether the overall average for the sample is 20 miles per gallon. We also assume that the population standard deviation is 6 .
```

. use https://www.stata-press.com/data/r18/auto
(1978 automobile data)
. ztest mpg==20, sd(6)
One-sample z test

| Variable | Obs | Mean | Std. err. | Std. dev. | [95\% conf. interval] |  |
| ---: | :---: | ---: | :---: | ---: | ---: | ---: | ---: |
| $\operatorname{mpg}$ | 74 | 21.2973 | .6974858 | 6 | 19.93025 | 22.66434 |
| mean $=$ mean $(\mathrm{mpg})$ |  |  |  | $\mathrm{z}=$ | 1.8600 |  |

HO: mean = 20
Ha: mean < 20
Ha: mean != 20
Ha: mean > 20
Pr}(Z<z)=0.968
Pr}(|Z|>|z|)=0.062
Pr}(Z > z) = 0.0314

```

The \(p\)-value for the two-sided test is 0.0629 , so we do not have statistical evidence to reject the null hypothesis that the mean equals 20 at a \(5 \%\) significance level, but we would reject the null hypothesis at a \(10 \%\) level.

\section*{Two-sample \(z\) test}

\section*{> Example 2: Two-sample \(z\) test using groups}

We are testing the effectiveness of a new fuel additive. We run an experiment in which 12 cars are given the fuel treatment and 12 cars are not. The results of the experiment are as follows:
\begin{tabular}{cc} 
treated & mpg \\
\hline 0 & 20 \\
0 & 23 \\
0 & 21 \\
0 & 25 \\
0 & 18 \\
0 & 17 \\
0 & 18 \\
0 & 24 \\
0 & 20 \\
0 & 24 \\
0 & 23 \\
0 & 19 \\
1 & 24 \\
1 & 25 \\
1 & 21 \\
1 & 22 \\
1 & 23 \\
1 & 18 \\
1 & 17 \\
1 & 28 \\
1 & 24 \\
1 & 27 \\
1 & 21 \\
1 & 23
\end{tabular}

The treated variable is coded as 1 if the car received the fuel treatment and 0 otherwise.
We can test the equality of means of the treated and untreated group by typing
```

. use https://www.stata-press.com/data/r18/fuel3
. ztest mpg, by(treated) sd(3)
Two-sample z test

```
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline Group & Obs & Mean & Std. err. & Std. dev. & [95\% con & interval] \\
\hline 0 & 12 & 21 & . 8660254 & 3 & 19.30262 & 22.69738 \\
\hline 1 & 12 & 22.75 & . 8660254 & 3 & 21.05262 & 24.44738 \\
\hline diff & & -1.75 & 1.224745 & & -4.150456 & . 6504558 \\
\hline \multicolumn{3}{|l|}{\multirow[t]{2}{*}{\[
\begin{aligned}
\text { diff } & =\text { mean }(0)-\operatorname{mean}(1) \\
\text { H0: } \operatorname{diff} & =0
\end{aligned}
\]}} & & & \multicolumn{2}{|r|}{\(z=-1.4289\)} \\
\hline & & & & & & \\
\hline \multicolumn{2}{|l|}{\multirow[t]{2}{*}{\[
\begin{gathered}
\text { Ha: diff }<0 \\
\operatorname{Pr}(Z<z)=0.0765
\end{gathered}
\]}} & \multicolumn{3}{|r|}{\multirow[t]{2}{*}{\[
\begin{gathered}
\text { Ha: diff }!=0 \\
\operatorname{Pr}(|z|>|z|)=0.1530
\end{gathered}
\]}} & \multicolumn{2}{|l|}{\multirow[t]{2}{*}{\[
\begin{gathered}
\text { Ha: diff }>0 \\
\operatorname{Pr}(Z>z)=0.9235
\end{gathered}
\]}} \\
\hline & & & & & & \\
\hline
\end{tabular}

We do not have evidence to reject the null hypothesis that the means of the two groups are equal at a 5\% significance level.

In the above, we assumed that the two groups have the same standard deviation of 3 . If the standard deviations for the two groups are different, we can specify group-specific standard deviations in options sd1() and sd2():
. ztest mpg, by(treated) sd1(2.7) sd2(3.2)
Two-sample z test
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline Group & Obs & Mean & Std. err. & Std. dev. & [95\% con & interval] \\
\hline 0 & 12 & 21 & . 7794229 & 2.7 & 19.47236 & 22.52764 \\
\hline 1 & 12 & 22.75 & . 9237604 & 3.2 & 20.93946 & 24.56054 \\
\hline diff & & -1.75 & 1.208649 & & -4.118909 & . 6189093 \\
\hline \multicolumn{3}{|l|}{\multirow[t]{2}{*}{\[
\begin{aligned}
\text { diff } & =\text { mean }(0)-\operatorname{mean}(1) \\
\text { H0: } \operatorname{diff} & =0
\end{aligned}
\]}} & & & \multicolumn{2}{|r|}{\(z=-1.4479\)} \\
\hline & & & & & & \\
\hline \multicolumn{2}{|l|}{Ha: diff < 0} & \multicolumn{3}{|c|}{Ha: diff != 0} & \multicolumn{2}{|r|}{Ha: diff > 0} \\
\hline \(\operatorname{Pr}(\mathrm{Z}<\mathrm{z})\) & 0738 & \multicolumn{3}{|c|}{\(\operatorname{Pr}(|Z|>|z|)=0.1476\)} & \multicolumn{2}{|l|}{\(\operatorname{Pr}(\mathrm{Z} \mathrm{>} \mathrm{z} \mathrm{)}=0.9262\)} \\
\hline
\end{tabular}

\section*{- Technical note}

In two-sample randomized designs, subjects will sometimes refuse the assigned treatment but still be measured for an outcome. In this case, take care to specify the group properly. You might be tempted to let varname contain missing where the subject refused and thus let ztest drop such observations from the analysis. Zelen (1979) argues that it would be better to specify that the subject belongs to the group in which he or she was randomized, even though such inclusion will dilute the measured effect.

\section*{> Example 3: Two-sample \(z\) test using variables}

There is a second, inferior way to organize the data in the preceding example. We ran a test on 24 cars, 12 without the additive and 12 with. We now create two new variables, mpg1 and mpg2.
\begin{tabular}{cl} 
mpg1 & mpg2 \\
\hline 20 & 24 \\
23 & 25 \\
21 & 21 \\
25 & 22 \\
18 & 23 \\
17 & 18 \\
18 & 17 \\
24 & 28 \\
20 & 24 \\
24 & 27 \\
23 & 21 \\
19 & 23
\end{tabular}

This method is inferior because it suggests a connection that is not there. There is no link between the car with 20 mpg and the car with 24 mpg in the first row of the data. Each column of data could be arranged in any order. Nevertheless, if our data are organized like this, ztest can accommodate us.
. use https://www.stata-press.com/data/r18/fuel
. ztest mpg1==mpg2, unpaired sd(3)
Two-sample z test
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline Variable & Obs & Mean & Std. err. & Std. dev. & [95\% con & interval] \\
\hline mpg1 & 12 & 21 & . 8660254 & 3 & 19.30262 & 22.69738 \\
\hline mpg2 & 12 & 22.75 & . 8660254 & 3 & 21.05262 & 24.44738 \\
\hline diff & & -1.75 & 1.224745 & & -4.150456 & . 6504558 \\
\hline \multicolumn{7}{|l|}{\multirow[t]{2}{*}{```
    diff = mean(mpg1) - mean(mpg2)
                z = -1.4289
HO: diff = 0
```}} \\
\hline & & & & & & \\
\hline \multicolumn{2}{|l|}{Ha: diff < 0} & \multicolumn{3}{|c|}{Ha: diff != 0} & \multicolumn{2}{|r|}{Ha: diff > 0} \\
\hline \(\operatorname{Pr}(\mathrm{Z}<\mathrm{z})\) & . 0765 & \multicolumn{3}{|r|}{\(\operatorname{Pr}(|Z|>|z|)=0.1530\)} & \multicolumn{2}{|l|}{\(\operatorname{Pr}(\mathrm{Z}>\mathrm{z})=0.9235\)} \\
\hline
\end{tabular}

\section*{Paired z test}

\section*{> Example 4}

Suppose that the preceding data were actually collected by running a test on 12 cars. Each car was run once with the fuel additive and once without. Our data are stored in the same manner as in example 3, but this time, there is most certainly a connection between the mpg values that appear in the same row. These come from the same car. The variables mpg1 and mpg2 represent mileage without and with the treatment, respectively. Suppose that the two variables have a common standard deviation of 2 and the correlation between them is 0.4 .
```

. use https://www.stata-press.com/data/r18/fuel
. ztest mpg1==mpg2, sd(2) corr(0.4)
Paired z test

```
\begin{tabular}{r|rrrrrr}
\hline Variable & Obs & Mean & Std. err. & Std. dev. & [95\% conf. interval] \\
\hline mpg1 & 12 & 21 & .5773503 & 2 & 19.86841 & 22.13159 \\
mpg2 & 12 & 22.75 & .5773503 & 2 & 21.61841 & 23.88159 \\
\hline diff & 12 & -1.75 & .6324555 & 2.19089 & -2.98959 & -.5104099 \\
\hline
\end{tabular}
    \(\operatorname{mean}(\) diff \()=\operatorname{mean}(m p g 1-\operatorname{mpg} 2) \quad z=-2.7670\)
HO: mean(diff) \(=0\)
Ha: mean(diff) < \(0 \quad\) Ha: mean(diff) \(!=0 \quad\) Ha: mean(diff) > 0
\(\operatorname{Pr}(Z<z)=0.0028 \quad \operatorname{Pr}(|Z|>|z|)=0.0057 \quad \operatorname{Pr}(Z>z)=0.9972\)

The \(p\)-value for the two-sided test is 0.0057 , so we reject, for example, the null hypothesis that the two means are equal at a \(5 \%\) significance level.

Equivalently, we could specify directly the standard deviation of the differences between paired observations with the sddiff() option:
. ztest mpg1==mpg2, sddiff(2.191)
Paired z test
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline Variable & Obs & Mean & Std. err. & Std. dev. & [95\% conf & interval] \\
\hline diff & 12 & -1.75 & . 6324872 & 2.191 & -2.989652 & -. 5103478 \\
\hline \multicolumn{5}{|l|}{\multirow[t]{2}{*}{\[
\begin{aligned}
\text { Hean }(\operatorname{diff}) & =\text { mean }(\operatorname{mpg} 1-\operatorname{mpg} 2) \\
\text { mean }(\operatorname{diff}) & =0
\end{aligned}
\]}} & \multicolumn{2}{|r|}{\multirow[t]{2}{*}{\(z=-2.7669\)}} \\
\hline & & & & & & \\
\hline \multicolumn{2}{|l|}{Ha: mean(diff) < 0} & \multicolumn{3}{|c|}{Ha: mean(diff) != 0} & \multicolumn{2}{|l|}{Ha: mean(diff) > 0} \\
\hline \multicolumn{2}{|l|}{\(\operatorname{Pr}(\mathrm{Z}<\mathrm{z})=0.0028\)} & \multicolumn{3}{|c|}{\(\operatorname{Pr}(|Z|>|z|)=0.0057\)} & \multicolumn{2}{|l|}{\(\operatorname{Pr}(\mathrm{Z} \mathrm{>} \mathrm{z})=0.9972\)} \\
\hline
\end{tabular}

\section*{Adjust for clustering}

When observations are not independent and can be grouped into clusters, we need to adjust for clustering in a \(z\) test. For example, in a cluster randomized design, groups of individuals are randomized instead of individuals. To adjust for clustering, we need to specify the cluster identifier variable in the cluster () option. In the case of a one-sample \(z\) test, we need to also specify the intraclass correlation in the rho() option. In the case of a two-sample \(z\) test, we need to also specify the common population intraclass correlation in the rho() option or group-specific population intraclass correlations in the rho1() and rho2() options.

\section*{> Example 5: One-sample \(z\) test, adjusting for clusters}

Consider data on the SAT score of 75 students from 15 classes, with 5 students in each class. We want to test whether the mean verbal SAT score is different from 600 . We assume a known standard deviation of 132 and a known intraclass correlation of 0.7 . To perform the test, we specify the options cluster (class), rho (0.7), and sd(132):


We find statistical evidence to reject the null hypothesis of \(H_{0}: \mu_{\text {SAT }}=600\) versus a two-sided alternative \(H_{a}: \mu_{\text {SAT }} \neq 600\) at the \(5 \%\) significance level; the \(p\)-value \(=0.0014<0.05\).

\section*{> Example 6: Two-sample \(z\) test using groups, adjusting for clusters}

Consider a cluster randomized control trial that studies the effect of additional training of nurses and general practitioners in patient-centered care on the well-being and future disease risk of patients with type 2 diabetes (Kinmonth et al. [1998] and Campbell and Walters [2014]). Practices (practice) are randomly allocated to two groups-one trained to give patient-centered care (intervention group) and another trained to give routine care (comparison or control group). In our analysis, we transform the original bmi using the formula \(\ln (\mathrm{bmi}-14.67355)\) to obtain a variable that is approximately normally distributed, lbmi. We want to test the equality of the means of lbmi for the two groups. We assume a known common standard deviation of 0.35 and a known common intraclass correlation of 0.028 .

To perform the test, we need to specify the rho ( 0.028 ) and \(s d(0.35)\) options. We also need to specify the cluster identifier practice in the cluster() option and the group identifier group in the by () option.


We find statistical evidence to reject the null hypothesis of \(H_{0}: \mu_{\text {diff }}=0\) versus a two-sided alternative \(H_{a}: \mu_{\text {diff }} \neq 0\) at the \(5 \%\) significance level; the \(p\)-value \(=0.0167<0.05\).

\section*{Immediate form}

\section*{> Example 7: One-sample \(z\) test}
ztesti is like ztest, except that we specify summary statistics rather than variables as arguments. For instance, we are reading an article that reports the mean number of sunspots per month as 62.6 with a standard deviation of 15.8 . We assume this standard deviation is the population standard deviation. There are 24 months of data. We wish to test whether the mean is 75 :
. ztesti 2462.615 .875
One-sample z test
\begin{tabular}{|c|c|c|c|c|c|c|}
\hline & Obs & Mean & Std. err. & Std. dev. & [95\% conf & nterval] \\
\hline x & 24 & 62.6 & 3.225161 & 15.8 & 56.2788 & 68.9212 \\
\hline \multicolumn{2}{|l|}{\[
\begin{aligned}
\text { mean } & =\operatorname{mean}(x) \\
\text { Hean } & =75
\end{aligned}
\]} & & & & & -3.8448 \\
\hline \[
\begin{array}{r}
\mathrm{Ha}: m \\
\operatorname{Pr}(\mathrm{Z}<\mathrm{z}
\end{array}
\] & \[
\begin{aligned}
& 75 \\
& .0001
\end{aligned}
\] & \multicolumn{3}{|c|}{Ha: mean ! \(=75\)} & \multicolumn{2}{|l|}{Ha: mean > 75} \\
\hline
\end{tabular}

\section*{Example 8: Two-sample \(z\) test}

There is no immediate form of ztest with paired data because the test is also a function of the covariance, a number unlikely to be reported in any published source. For unpaired data, however, we might type
. ztesti 2020532154
Two-sample z test
\begin{tabular}{r|rrrrrr}
\hline & Obs & Mean & Std. err. & Std. dev. & [95\% conf. interval] \\
\hline x & 20 & 20 & 1.118034 & 5 & 17.80869 & 22.19131 \\
y & 32 & 15 & .7071068 & 4 & 13.6141 & 16.3859 \\
\hline diff & 5 & 1.322876 & & 2.407211 & 7.592789 \\
\hline diff \(=\operatorname{mean}(\mathrm{x})-\operatorname{mean}(\mathrm{y})\) & & & 3.7796
\end{tabular}

H0: diff \(=0\)

Ha: diff < 0
Ha: diff != 0
\(\operatorname{Pr}(Z<z)=0.9999\)
\(\operatorname{Pr}(|Z|>|z|)=0.0002\)

Ha: diff > 0
\(\operatorname{Pr}(Z>z)=0.0001\)

\section*{Stored results}

One-sample ztest and ztesti store the following in r() :
Scalars
```

r(N)
r(mu) sample mean
r(sd) standard deviation
r(se) standard error
r(lb) lower confidence bound of one-sample mean
r(ub) upper confidence bound of one-sample mean
r(z) z statistic
r(p_l) lower one-sided p-value
r(p) two-sided p-value
r(p_u) upper one-sided p-value
r(level) confidence level

```

Cluster-adjusted one-sample ztest also stores the following in \(r()\) :
Scalars
```

r(K)
r(M)
r(rho)
r(CV_cluster)

```
number of clusters \(K\)
cluster size \(M\)
intraclass correlation
coefficient of variation for cluster sizes
Two-sample ztest and ztesti store the following in r() :
Scalars
```

r(N1) sample size of population one
r(N2) sample size of population two
r(mu1) sample mean for population one
r(mu2) sample mean for population two
r(mu_diff) difference of means
r(corr) correlation between paired observations; if the corr() option is specified
r(sd)
r(sd1)
r(sd2)
r(sd_diff)
r(se1)
r(se2)
r(se_diff)
r(lb1)
r(ub1)
r(lb2)
r(ub2)
r(lb_diff)
r(ub_diff)
r(z)
r(p_l) lower one-sided p-value
r(p) two-sided p-value
r(p_u) upper one-sided p-value
r(level)
confidence level

```

Cluster-adjusted two-sample ztest using the by () option also stores the following in \(\mathrm{r}(\mathrm{)}\) :
Scalars
r(K1)
\(r(\mathrm{~K} 2)\)
population-one cluster size \(M_{1}\)
\(r\) (M2) population-two cluster size \(M_{2}\)
\(r\) (rho) common intraclass correlation
\(r\) (rho1) population-one intraclass correlation
\(r\) (rho2) population-two intraclass correlation
\(r\) (CV_cluster1) population-one coefficient of variation for cluster sizes
\(r\) (CV_cluster2) population-two coefficient of variation for cluster sizes

\section*{Methods and formulas}

Methods and formulas are presented under the following headings:
One-sample z test
Two-sample unpaired z test
Paired z test

For all the tests below, the test statistic \(z\) is distributed as standard normal, and the \(p\)-value is computed as
\[
p= \begin{cases}1-\Phi(z) & \text { for an upper one-sided test } \\ \Phi(z) & \text { for a lower one-sided test } \\ 2\{1-\Phi(|z|)\} & \text { for a two-sided test }\end{cases}
\]
where \(\Phi(\cdot)\) is the cdf of a standard normal distribution and \(|z|\) is an absolute value of \(z\).
Also see, for instance, Hoel (1984, 140-161), Dixon and Massey (1983, 100-130), and Tamhane and Dunlop (2000, 237-290) for more information about \(z\) tests.

\section*{One-sample z test}

Suppose that we observe a random sample \(x_{1}, x_{2}, \ldots, x_{n}\) of size \(n\), which follows a normal distribution with mean \(\mu\) and standard deviation \(\sigma\). We are interested in testing the null hypothesis \(H_{0}: \mu=\mu_{0}\) versus the two-sided alternative hypothesis \(H_{a}: \mu \neq \mu_{0}\), the upper one-sided alternative \(H_{a}: \mu>\mu_{0}\), or the lower one-sided alternative \(H_{a}: \mu<\mu_{0}\). Assuming a known standard deviation \(\sigma\), we use the following test statistic,
\[
z=\frac{\left(\bar{x}-\mu_{0}\right)}{s}
\]
where \(\bar{x}=\left(\sum_{i=1}^{n} x_{i}\right) / n\) is the sample mean and \(s=\sigma / \sqrt{n}\) is the standard error of \(\bar{x}\).
The \(100(1-\alpha) \%\) confidence interval for \(\bar{x}\) is given by
\[
\bar{x} \pm z_{1-\alpha / 2} s
\]
where \(z_{1-\alpha / 2}\) is the \((1-\alpha / 2)\) th quantile of the standard normal distribution.
With clustered data, suppose that there are \(K\) clusters. The \(i\) th cluster of size \(M_{i}\) contains the observations \(x_{i 1}, x_{i 2}, \ldots, x_{i M_{i}}\), such that \(n=\sum_{i=1}^{K} M_{i}\) and \(\bar{x}=\frac{1}{n} \sum_{i=1}^{K} \sum_{j=1}^{M_{i}} x_{i j}\). Let \(\rho\) be the intraclass correlation. Following Ahn, Heo, and Zhang (2015), we assume that the cluster sizes \(M_{i}\) are independent and identically distributed. Let \(C_{\text {adj }}\) be the adjustment to the standard error for clustered data,
\[
C_{\mathrm{adj}}=\sqrt{\sum_{i=1}^{K} M_{i}\left\{1+\rho\left(M_{i}-1\right)\right\} / n}
\]
such that \(s_{\mathrm{cl}}=C_{\mathrm{adj}} s\).
\(C_{\text {adj }}\) can be equivalently written as
\[
C_{\mathrm{adj}}=\sqrt{1+\rho(\bar{M}-1)+\rho \bar{M} \mathrm{CV}_{\mathrm{cl}}^{2}}
\]
where \(\bar{M}=\sum_{i=1}^{K} M_{i} / K\) is the average cluster size and \(\mathrm{CV}_{\mathrm{cl}}\) is the coefficient of variation for cluster sizes:
\[
\mathrm{CV}_{\mathrm{cl}}=\frac{\sqrt{\sum_{i=1}^{K}\left(M_{i}-\bar{M}\right)^{2} / K}}{\bar{M}}
\]

To adjust the test statistic \(z\) and the confidence interval for clustering, replace \(s\) with \(s_{\mathrm{cl}}\) in the corresponding formulas. In the presence of clustering, the test statistic \(z\) is asymptotically normally distributed conditional on the empirical distribution of \(M_{i}\) 's.

\section*{Two-sample unpaired \(z\) test}

Suppose that we observe a random sample \(x_{11}, x_{12}, \ldots, x_{1 n_{1}}\) of size \(n_{1}\), which follows a normal distribution with mean \(\mu_{1}\) and standard deviation \(\sigma_{1}\), and another random sample \(x_{21}, x_{22}, \ldots, x_{2 n_{2}}\) of size \(n_{2}\), which follows a normal distribution with mean \(\mu_{2}\) and standard deviation \(\sigma_{2}\). We are interested in testing the null hypothesis \(H_{0}: \mu_{2}=\mu_{1}\) versus the two-sided alternative hypothesis \(H_{a}: \mu_{2} \neq \mu_{1}\), the upper one-sided alternative \(H_{a}: \mu_{2}>\mu_{1}\), or the lower one-sided alternative \(H_{a}: \mu_{2}<\mu_{1}\). Assuming known standard deviations \(\sigma_{1}\) and \(\sigma_{2}\), we use the following test statistic,
\[
z=\frac{\bar{x}_{2}-\bar{x}_{1}}{\sqrt{s_{1}^{2}+s_{2}^{2}}}
\]
where \(\bar{x}_{1}=\left(\sum_{i=1}^{n_{1}} x_{1 i}\right) / n_{1}\) and \(\bar{x}_{2}=\left(\sum_{i=1}^{n_{2}} x_{2 i}\right) / n_{2}\) are the two sample means and \(s_{1}=\sigma_{1} / \sqrt{n_{1}}\) and \(s_{2}=\sigma_{2} / \sqrt{n_{2}}\) are the corresponding two standard errors.

The \(100(1-\alpha) \%\) confidence intervals for \(\bar{x}_{1}\) and \(\bar{x}_{2}\) are given by
\[
\begin{aligned}
& \bar{x}_{1} \pm z_{1-\alpha / 2} s_{1} \\
& \bar{x}_{2} \pm z_{1-\alpha / 2} s_{2}
\end{aligned}
\]
where \(z_{1-\alpha / 2}\) is the \((1-\alpha / 2)\) th quantile of the standard normal distribution.
The \(100(1-\alpha) \%\) confidence interval for \(\bar{x}_{1}-\bar{x}_{2}\) is given by
\[
\bar{x}_{1}-\bar{x}_{2} \pm z_{1-\alpha / 2} \sqrt{s_{1}^{2}+s_{2}^{2}}
\]

With clustered data, similar to the discussion for the one-sample test, suppose that population one has \(K_{1}\) clusters and population two has \(K_{2}\) clusters. Let \(\rho_{1}\) and \(\rho_{2}\) be the intraclass correlations, \(\bar{M}_{1}\) and \(\bar{M}_{2}\) be the average cluster sizes, \(\bar{x}_{1}=\left(1 / n_{1}\right) \sum_{i=1}^{K_{1}} \sum_{j=1}^{M_{1 i}} x_{1 i j}\) and \(\bar{x}_{2}=\left(1 / n_{2}\right) \sum_{i=1}^{K_{2}} \sum_{j=1}^{M_{2 i}} x_{2 i j}\) be the sample means, and \(\mathrm{CV}_{\mathrm{cl}, 1}\) and \(\mathrm{CV}_{\mathrm{cl}, 2}\) be the coefficients of variation for cluster sizes for population one and population two. Let \(s_{1, \mathrm{cl}}=C_{\mathrm{adj}, 1} s_{1}\) and \(s_{2, \mathrm{cl}}=C_{\mathrm{adj}, 2} s_{2}\) be the standard errors of the population-specific sample means adjusted for clustered data, where the population-specific adjustment factors are defined as described for the one-sample test. To adjust the two-sample test statistic and the confidence intervals for clustering, replace \(s_{1}\) with \(s_{1, \mathrm{cl}}\) and \(s_{2}\) with \(s_{2, \mathrm{cl}}\) in the corresponding formulas.

\section*{Paired z test}

Some experiments have paired observations (also known as matched observations, correlated pairs, or permanent components). Consider a sequence of \(n\) paired observations denoted by \(x_{i j}\) for subjects \(i=1,2, \ldots, n\) and groups \(j=1,2\). An individual observation corresponds to the pair ( \(x_{i 1}, x_{i 2}\) ), and inference is made on the differences within the pairs. Let \(\mu_{d}=\mu_{2}-\mu_{1}\) denote the mean difference, where \(\mu_{j}\) is the population mean of group \(j\), and let \(D_{i}=x_{i 2}-x_{i 1}\) denote the difference between individual observations. \(D_{i}\) follows a normal distribution with mean \(\mu_{2}-\mu_{1}\) and standard deviation \(\sigma_{d}\), where \(\sigma_{d}=\sqrt{\sigma_{1}{ }^{2}+\sigma_{2}{ }^{2}-2 \rho_{\text {pair }} \sigma_{1} \sigma_{2}}, \sigma_{j}\) is the population standard deviation of group \(j\) and \(\rho_{\text {pair }}\) is the correlation between paired observations.

We are interested in testing the null hypothesis \(H_{0}: \mu_{2}=\mu_{1}\) versus the two-sided alternative hypothesis \(H_{a}: \mu_{2} \neq \mu_{1}\), the upper one-sided alternative \(H_{a}: \mu_{2}>\mu_{1}\), or the lower one-sided alternative \(H_{a}: \mu_{2}<\mu_{1}\). Assuming the standard deviation of the differences \(\sigma_{d}\) is known, we use the following test statistic,
\[
z=\frac{\bar{d}}{s_{d}}
\]
where \(\bar{d}=\left(\sum_{i=1}^{n} D_{i}\right) / n\) is the sample mean of the differences between paired observations and \(s_{d}=\sigma_{d} / \sqrt{n}\) is the standard error of \(\bar{d}\).

The \(100(1-\alpha) \%\) confidence interval for \(\bar{d}\) is given by
\[
\bar{d} \pm z_{1-\alpha / 2} s_{d}
\]

\section*{References}

Ahn, C., M. Heo, and S. Zhang. 2015. Sample Size Calculations for Clustered and Longitudinal Outcomes in Clinical Research. Boca Raton, FL: CRC Press.
Campbell, M. J., and S. J. Walters. 2014. How to Design, Analyse and Report Cluster Randomised Trials in Medicine and Health Related Research. Chichester, UK: Wiley.

Dixon, W. J., and F. J. Massey, Jr. 1983. Introduction to Statistical Analysis. 4th ed. New York: McGraw-Hill.
Hoel, P. G. 1984. Introduction to Mathematical Statistics. 5th ed. New York: Wiley.
Kinmonth, A. L., A. Woodcock, S. Griffin, N. Spiegal, and M. J. Campbell. 1998. Randomised controlled trial of patient centred care of diabetes in general practice: Impact on current wellbeing and future disease risk. BMJ 317: 1202-1208. https://doi.org/10.1136/bmj.317.7167.1202.
Tamhane, A. C., and D. D. Dunlop. 2000. Statistics and Data Analysis: From Elementary to Intermediate. Upper Saddle River, NJ: Prentice Hall.

Zelen, M. 1979. A new design for randomized clinical trials. New England Journal of Medicine 300: 1242-1245. https://doi.org/10.1056/NEJM197905313002203.

\section*{Also see}
\([\mathrm{R}]\) ci - Confidence intervals for means, proportions, and variances
[R] esize - Effect size based on mean comparison
[R] mean - Estimate means
\([R]\) oneway - One-way analysis of variance
[R] ttest - \(t\) tests (mean-comparison tests)
[MV] hotelling - Hotelling's \(T^{2}\) generalized means test
[PSS-2] power onemean - Power analysis for a one-sample mean test
[PSS-2] power onemean, cluster - Power analysis for a one-sample mean test, CRD
[PSS-2] power pairedmeans - Power analysis for a two-sample paired-means test
[PSS-2] power twomeans - Power analysis for a two-sample means test
[PSS-2] power twomeans, cluster - Power analysis for a two-sample means test, CRD

\section*{Subject and author index}

See the combined subject index and the combined author index in the Stata Index.```


[^0]:    * 5 was added to the variables prior to calculating the results.

    Note: Missing values in confidence intervals for harmonic mean indicate that confidence interval is undefined for corresponding variables.

[^1]:    Also see
    [R] ci - Confidence intervals for means, proportions, and variances
    [R] mean - Estimate means
    [R] summarize - Summary statistics
    [SVY] svy estimation - Estimation commands for survey data

[^2]:    ANCOVA
    You can include multiple explanatory variables with the anova command, but unless you explicitly state otherwise by using the c . factor-variable operator, all the variables are interpreted as categorical variables. Using the c. operator, you can designate variables as continuous and thus perform ANCOVA.

    ## > Example 9: ANCOVA (ANOVA with a continuous covariate)

    We have census data recording the deathrate (drate) and median age (age) for each state. The dataset also includes the region of the country in which each state is located (region):

[^3]:    *forecast is not appropriate with mi estimation results.

[^4]:    *forecast, hausman, and lrtest are not appropriate with svy estimation results.

[^5]:    *estat ic and lrtest are not appropriate after binreg, irls.
    $\dagger_{\text {forecast }}$ is not appropriate with mi estimation results.

[^6]:    . biprobit (private=years logptax loginc) (vote=years logptax loginc)

[^7]:    *hausman and lrtest are not appropriate with svy estimation results.

[^8]:    * Inference is valid only for hypotheses concerning $\lambda$ and $\theta$.

[^9]:    *forecast, hausman, and lrtest are not appropriate with svy estimation results.

[^10]:    *hausman and lrtest are not appropriate with svy estimation results.

[^11]:    Model
    noconstant, offset (varname); see [R] Estimation options.
    asis forces retention of perfect predictor variables and their associated perfectly predicted observations and may produce instabilities in maximization; see [R] probit. constraints (constraints); see [R] Estimation options.

[^12]:    *forecast, hausman, and lrtest are not appropriate with svy estimation results. forecast is also not appropriate with mi estimation results.

[^13]:    *forecast, hausman, and lrtest are not appropriate with svy estimation results. forecast is also not appropriate with mi estimation results.

[^14]:    Also see
    [R] copyright - Display copyright information

[^15]:    Also see
    $[R]$ copyright - Display copyright information

[^16]:    Also see
    [R] copyright - Display copyright information

[^17]:    *forecast, hausman, and lrtest are not appropriate with svy estimation results.

[^18]:    . db regress

[^19]:    Note: alpha estimates sum to 1.
    Shares: w_dairy w_proteins w_fruitveg w_flours w_misc
    Prices: p_dairy p_proteins p_fruitveg p_flours p_misc
    Expenditure: expfd

[^20]:    warn does not appear in the dialog box.

[^21]:    Incidence-rate data
    Stratified incidence-rate data
    Standardized estimates with stratified incidence-rate data
    Cumulative incidence data
    Stratified cumulative incidence data
    Standardized estimates with stratified cumulative incidence data
    Case-control data
    Stratified case-control data
    Case-control data with multiple levels of exposure
    Case-control data with confounders and possibly multiple levels of exposure
    Standardized estimates with stratified case-control data
    Matched case-control data
    Video examples
    Glossary

[^22]:    collect is allowed; see [U] 11.1.10 Prefix commands.

[^23]:    noopen does not appear in the dialog box.

[^24]:    Model
    condvars ( varlist $_{c}$ ) specifies variables whose parameter estimates are not of interest to you. You can save substantial computer time and memory by moving such variables from indepvars to condvars(). Understand that you will get the same results for x 1 and x 3 whether you type
    . expoisson y x1 x2 x3 x4
    or
    . expoisson y x1 x3, condvars(x2 x4)
    group (varname) specifies the variable defining the strata, if any. A constant term is assumed for each stratum identified in varname, and the sufficient statistics for indepvars are conditioned on the observed number of successes within each group (as well as other variables in the model). The group variable must be integer valued.

[^25]:    Note: Mid-p-value computed for the probabilities and CIs.

[^26]:    - predict fit2
    (option xb assumed; fitted values)

[^27]:    fp plot, residuals(none)

[^28]:    *forecast and hausman are not appropriate with svy estimation results. forecast is also not appropriate with mi estimation results.

[^29]:    Reporting
    level(\#), nocnsreport; see [R] Estimation options.
    display_options: noci, nopvalues, noomitted, vsquish, noemptycells, baselevels, allbaselevels, nofvlabel, fvwrap(\#), fvwrapon(style), cformat (\%fint), pformat (\%fmt), sformat (\% fint), and nolstretch; see [R] Estimation options.

[^30]:    *You may not specify both hasderivatives and haslfderivatives.
    $\dagger_{\text {You must specify equations (namelist) or nequations(\#); you may specify both. }}$
    $\ddagger_{\text {You must specify parameters (namelist) or nparameters(\#); you may specify both. }}$

[^31]:    *lrtest is not appropriate with svy estimation results.

[^32]:    Note: _cons estimates baseline relative risk for each outcome.

[^33]:    *forecast, hausman, and lrtest are not appropriate with svy estimation results. forecast is also not appropriate with mi estimation results.

[^34]:    *forecast, hausman, and lrtest are not appropriate with svy estimation results. forecast is also not appropriate with mi estimation results.

[^35]:    *forecast, hausman, and lrtest are not appropriate with svy estimation results. forecast is also not appropriate with mi estimation results.

[^36]:    Mean-dispersion model
    Constant-dispersion model

[^37]:    Definition of a package
    The purpose of the net and ado commands
    Content pages
    Package-description pages
    Where packages are installed
    A summary of the net command
    A summary of the ado command
    Relationship of net and ado to the point-and-click interface
    Creating your own site
    Format of content and package-description files
    Example 1
    Example 2
    Additional package directives
    SMCL in content and package-description files
    Error-free file delivery

[^38]:    *forecast, hausman, and lrtest are not appropriate with svy estimation results.
    $\dagger$ You must specify the variables() option with nl.

[^39]:    *You must specify the variables() option with nlsur.

[^40]:    Also see
    [ R$]$ npregress kernel - Nonparametric kernel regression
    [R] bootstrap postestimation - Postestimation tools for bootstrap
    [U] 20 Estimation and postestimation commands

[^41]:    Reporting
    citype (citype) specifies the type of confidence interval to be computed. By default, bootstrap percentile confidence intervals are reported as recommended by Cattaneo and Jansson (2018). citype may be one of percentile, bc, or normal.

[^42]:    Maximization
    maximize_options: difficult, technique (algorithm_spec), iterate(\#), [no]log, trace, gradient, showstep, hessian, showtolerance, tolerance(\#), ltolerance(\#), nrtolerance(\#), nonrtolerance, and from (init_specs); see [R] Maximize. These options are seldom used.

[^43]:    *forecast, hausman, and lrtest are not appropriate with svy estimation results. forecast is also not appropriate with mi estimation results.

[^44]:    Sorted by:

[^45]:    *forecast, hausman, and lrtest are not appropriate with svy estimation results. forecast is also not appropriate with mi estimation results.

[^46]:    Model
    noconstant, exposure (varname ${ }_{e}$ ), offset ( varname $_{o}$ ), constraints (constraints); see [R] Estimation options.

[^47]:    *forecast, hausman, and lrtest are not appropriate with svy estimation results. forecast is also not appropriate with mi estimation results.

[^48]:    ${ }^{+}$These features are part of StataNow.
    indepvars may contain factor variables; see [U] 11.4.3 Factor variables.
    by, collect, mi estimate, rolling, and statsby, are allowed by qreg, iqreg, sqreg, and bsqreg; bayes ${ }^{+}$, mfp, nestreg, and stepwise are allowed only with qreg; see [U] 11.1.10 Prefix commands. For more details, see [BAYES] bayes: qreg.
    qreg allows fweights, iweights, and pweights; see [U] 11.1.6 weight.
    See [U] 20 Estimation and postestimation commands for more capabilities of estimation commands.

[^49]:    Median regression
    Quantile regression
    Estimated standard errors
    Interquantile and simultaneous-quantile regression
    What are the parameters?

[^50]:    *hausman is not appropriate after bsqreg, isqreg, or sqreg.
    $\dagger_{\text {forecast }}$ is not appropriate with mi estimation results.

[^51]:    *estat ic is not appropriate after reg3, 2sls.

[^52]:    *forecast, hausman, and lrtest are not appropriate with svy estimation results. forecast is also not appropriate with mi estimation results.

[^53]:    Add plots
    addplot (plot) provides a way to add other plots to the generated graph; see [G-3] addplot_option.

[^54]:    collect is allowed; see [U] 11.1.10 Prefix commands.

[^55]:    Also see
    [ R ] regress - Linear regression
    $[R]$ regress postestimation - Postestimation tools for regress
    [R] regress postestimation diagnostic plots - Postestimation plots for regress
    [TS] tsset - Declare data to be time-series data

[^56]:    Note: _cons estimates baseline risk.

[^57]:    Also see
    [R] logistic postestimation - Postestimation tools for logistic
    $[R]$ roc - Receiver operating characteristic (ROC) analysis
    [R] rocfit - Parametric ROC models
    $[R]$ rocreg - Receiver operating characteristic (ROC) regression
    [R] roctab - Nonparametric ROC analysis

[^58]:    *See Using lincom and test below.

[^59]:    *probit is required.

[^60]:    Bootstrap
    nobootstrap specifies that bootstrap standard errors not be calculated.

[^61]:    *bfile() and btype() are only allowed with parametric analysis using bootstrap inference.

[^62]:    Parametric model: Summary parameter definition
    Maximum likelihood estimation
    Estimating equations estimation

[^63]:    *Only one of roc() or invroc() may be specified.

[^64]:    * bfile() is only allowed with parametric analysis using bootstrap inference; in which case this option is required with roc() or invroc().

[^65]:    - 1983. A method of comparing the areas under receiver operating characteristic curves derived from the same cases. Radiology 148: 839-843. https://doi.org/10.1148/radiology.148.3.6878708.

[^66]:    __. 1991b. ssi2: Bootstrap programming. Stata Technical Bulletin 4: 18-27. Reprinted in Stata Technical Bulletin Reprints, vol. 1, pp. 208-220. College Station, TX: Stata Press.

[^67]:    *forecast is not appropriate with mi estimation results.

[^68]:    *forecast and lrtest are not appropriate with svy estimation results.

[^69]:    . set showemptycells off

[^70]:    Also see
    [R] search - Search Stata documentation and other resources
    [R] net - Install and manage community-contributed additions from the Internet
    $[R]$ net search - Search the Internet for installable packages
    [R] update - Check for official updates
    [U] 3.4 The Stata Journal
    [U] 29 Using the Internet to keep up to date
    [GSM] 19 Updating and extending Stata-Internet functionality
    [GSU] 19 Updating and extending Stata-Internet functionality
    [GSW] 19 Updating and extending Stata-Internet functionality

[^71]:    *hausman and lrtest are not appropriate with svy estimation results.

