

Stata 17: A guided tour

Meghan Cain | May 20th, 2021

You can download the datasets, do-file, and slides here:
<https://tinyurl.com/Stata17>

New in Stata 17

- Tables
- Bayesian econometrics
- Interval-censored Cox model
- Difference in differences (DID)
- Bayesian VAR
- Multivariate meta-analysis
- Treatment-effects lasso
- Panel-data multinomial logit
- Zero-inflated ordered logit
- Bayesian IRF and FEVD analysis
- Bayesian dynamic forecasting
- Do-file Editor enhancements
- Intel Math Kernel Library (MKL)
- Stata on Apple Silicon
- PyStata
- Jupyter Notebook with Stata
- Faster Stata
- Bayesian multilevel modeling
- New functions for dates and times
- Leave-one-out meta-analysis
- Galbraith plots
- Bayesian panel-data models
- Nonparametric tests for trend
- Lasso with clustered data
- BIC for lasso penalty selection
- Bayesian linear and nonlinear DSGEs
- H2O integration
- Java integration
- JDBC

Outline

Models and estimation

- New in Bayesian estimation and multilevel modeling
- Interval-censored Cox model
- New in lasso
- Difference in differences
- Nonparametric tests for trend
- New in meta-analysis
- Zero-inflated ordered logit model

Reporting

- Customizable tables

Workflow

- New in date/time functions
- Do-file editor enhancements
- Faster Stata

Integrations and interactions

- PyStata
- Jupyter Notebook
- Java/H2O integration
- Connecting databases
- Apple Silicon

New in Bayesian

- Bayesian VAR models
- Bayesian IRF and FEVD analysis
- Bayesian dynamic forecasting
- Bayesian linear and nonlinear DSGE models
- Bayesian longitudinal/panel-data models
- Bayesian multilevel models

bayes :

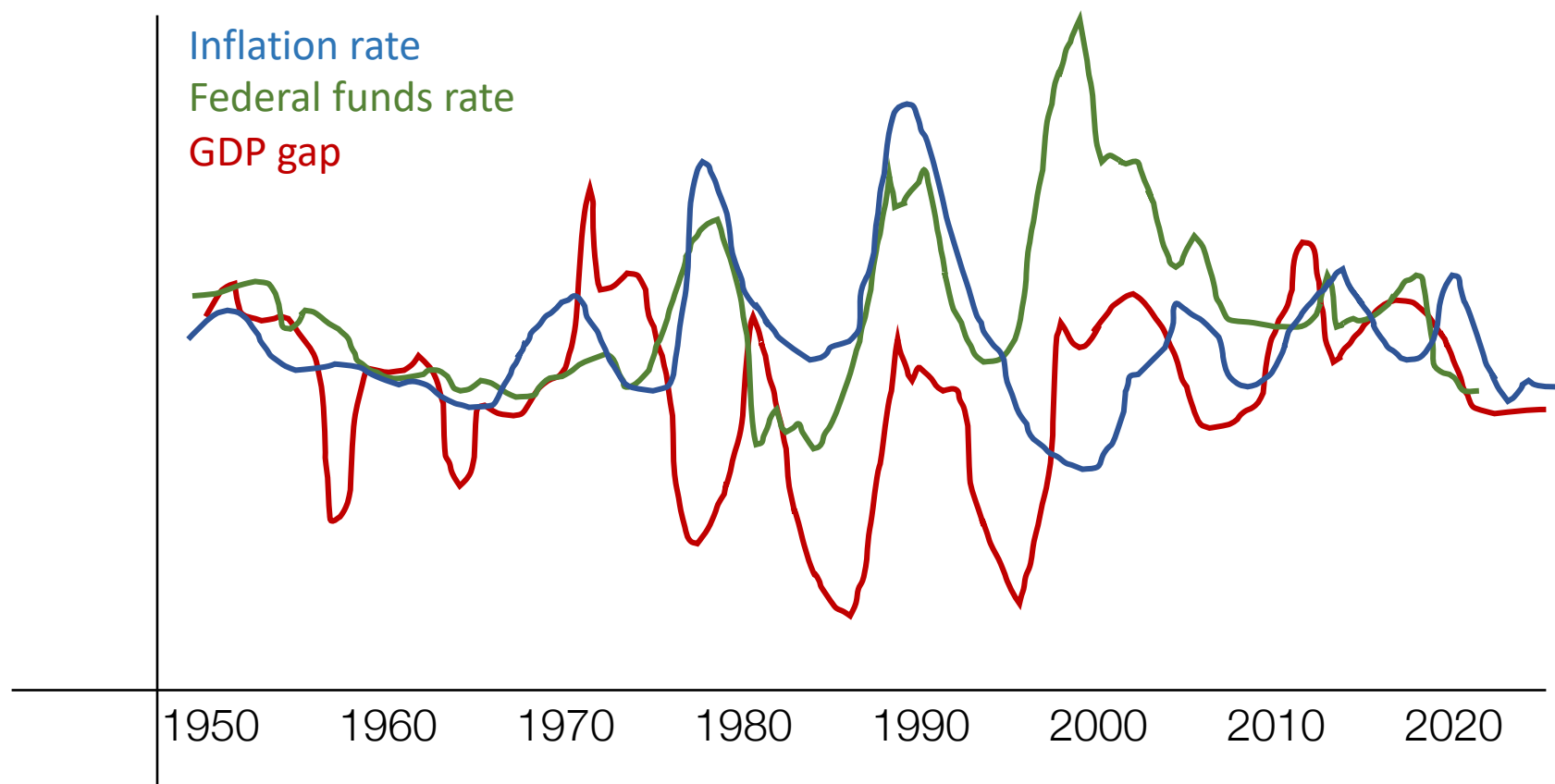
Stata 15

- Linear models
- Generalized linear models (binary, ordinal, categorical, count, fractional)
- Survival models
- Sample-selection models
- Multilevel models

New in Stata 17

- Multivariate time-series models
 - Vector autoregressive models
 - Dynamic stochastic general equilibrium models
- Longitudinal/panel-data models (xt suite)

Vector autoregressive (VAR) models



VAR(2) model

$$\begin{aligned}\text{inflation}_t &= c_1 + a_{1,1,1}\text{inflation}_{t-1} + a_{1,1,2}\text{inflation}_{t-2} \\ &\quad + a_{1,2,1}\text{ogap}_{t-1} + a_{1,2,2}\text{ogap}_{t-2} \\ &\quad + a_{1,3,1}\text{fedfunds}_{t-1} + a_{1,3,2}\text{fedfunds}_{t-2} + u_{1,t} \\ \text{ogap}_t &= c_2 + a_{2,1,1}\text{inflation}_{t-1} + a_{2,1,2}\text{inflation}_{t-2} \\ &\quad + a_{2,2,1}\text{ogap}_{t-1} + a_{2,2,2}\text{ogap}_{t-2} \\ &\quad + a_{2,3,1}\text{fedfunds}_{t-1} + a_{2,3,2}\text{fedfunds}_{t-2} + u_{1,t} \\ \text{fedfunds}_t &= c_3 + a_{3,1,1}\text{inflation}_{t-1} + a_{3,1,2}\text{inflation}_{t-2} \\ &\quad + a_{3,2,1}\text{ogap}_{t-1} + a_{3,2,2}\text{ogap}_{t-2} \\ &\quad + a_{3,3,1}\text{fedfunds}_{t-1} + a_{3,3,2}\text{fedfunds}_{t-2} + u_{1,t}\end{aligned}$$

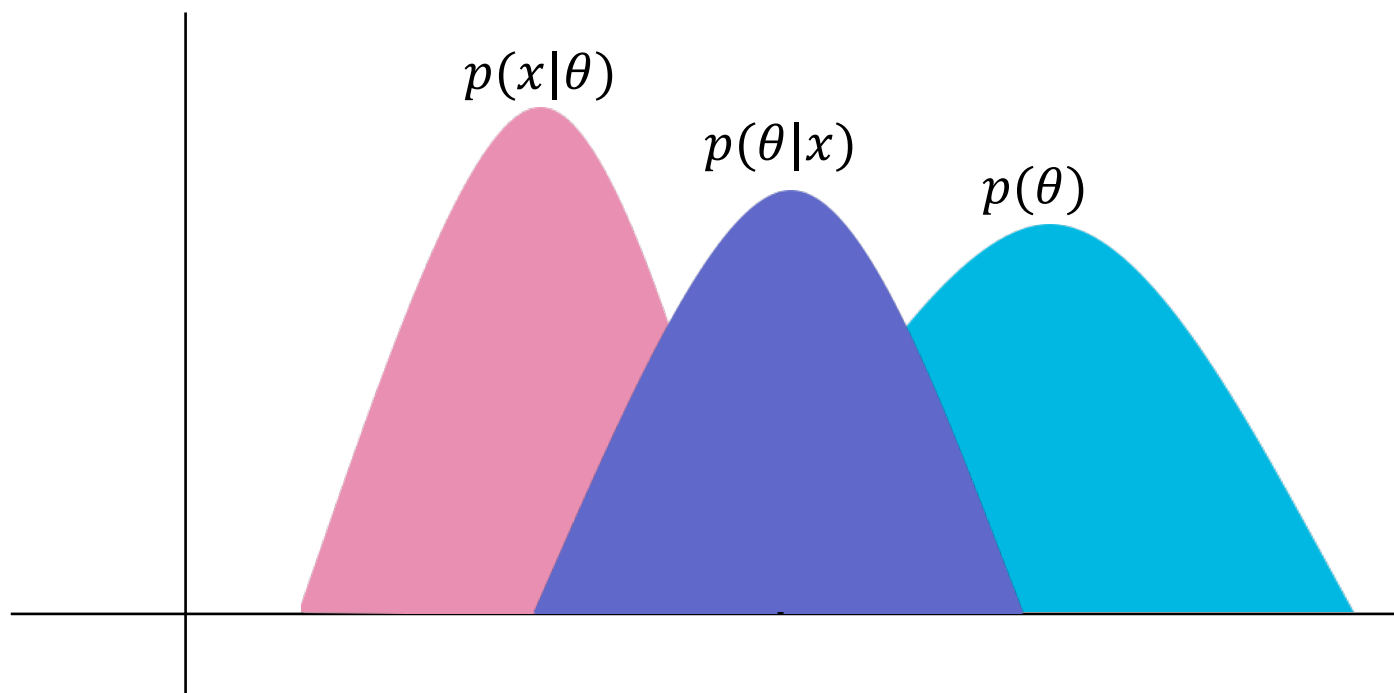
VAR(2) model

$$\begin{aligned}\text{inflation}_t &= c_1 + a_{1,1,1}\text{inflation}_{t-1} + a_{1,1,2}\text{inflation}_{t-2} \\ &\quad + a_{1,2,1}\text{ogap}_{t-1} + a_{1,2,2}\text{ogap}_{t-2} \\ &\quad + a_{1,3,1}\text{fedfunds}_{t-1} + a_{1,3,2}\text{fedfunds}_{t-2} + u_{1,t} \\ \text{ogap}_t &= c_2 + a_{2,1,1}\text{inflation}_{t-1} + a_{2,1,2}\text{inflation}_{t-2} \\ &\quad + a_{2,2,1}\text{ogap}_{t-1} + a_{2,2,2}\text{ogap}_{t-2} \\ &\quad + a_{2,3,1}\text{fedfunds}_{t-1} + a_{2,3,2}\text{fedfunds}_{t-2} + u_{2,t} \\ \text{fedfunds}_t &= c_3 + a_{3,1,1}\text{inflation}_{t-1} + a_{3,1,2}\text{inflation}_{t-2} \\ &\quad + a_{3,2,1}\text{ogap}_{t-1} + a_{3,2,2}\text{ogap}_{t-2} \\ &\quad + a_{3,3,1}\text{fedfunds}_{t-1} + a_{3,3,2}\text{fedfunds}_{t-2} + u_{3,t}\end{aligned}$$

VAR(2) model

$$\begin{aligned}\text{inflation}_t &= c_1 + a_{1,1,1}\text{inflation}_{t-1} + a_{1,1,2}\text{inflation}_{t-2} \\ &\quad + a_{1,2,1}\text{ogap}_{t-1} + a_{1,2,2}\text{ogap}_{t-2} \\ &\quad + a_{1,3,1}\text{fedfunds}_{t-1} + a_{1,3,2}\text{fedfunds}_{t-2} + u_{1,t} \\ \text{ogap}_t &= c_2 + a_{2,1,1}\text{inflation}_{t-1} + a_{2,1,2}\text{inflation}_{t-2} \\ &\quad + a_{2,2,1}\text{ogap}_{t-1} + a_{2,2,2}\text{ogap}_{t-2} \\ &\quad + a_{2,3,1}\text{fedfunds}_{t-1} + a_{2,3,2}\text{fedfunds}_{t-2} + u_{2,t} \\ \text{fedfunds}_t &= c_3 + a_{3,1,1}\text{inflation}_{t-1} + a_{3,1,2}\text{inflation}_{t-2} \\ &\quad + a_{3,2,1}\text{ogap}_{t-1} + a_{3,2,2}\text{ogap}_{t-2} \\ &\quad + a_{3,3,1}\text{fedfunds}_{t-1} + a_{3,3,2}\text{fedfunds}_{t-2} + u_{3,t}\end{aligned}$$

What is Bayesian analysis?



Why Bayesian analysis?

- Incorporation of prior knowledge
- Better small-sample performance
(with appropriate, informative priors)
- Fit underidentified models
(including models with more parameters than observations)
- Fit more complex models
- Full posterior distributions of each parameter
- Directly compare hypotheses

Bayesian VAR models

```
. bayes: var inflation ogap fedfunds
```

Bayesian vector autoregression
Gibbs sampling

Sample: 1956q1 thru 2010q4

Log marginal-likelihood = -804.25712

MCMC iterations = 12,500
Burn-in = 2,500
MCMC sample size = 10,000
Number of obs = 220
Acceptance rate = 1
Efficiency: min = .9532
 avg = .9934
 max = 1

	Mean	Std. dev.	MCSE	Median	Equal-tailed [95% cred. interval]	
inflation						
inflation						
L1.	1.050745	.0407223	.000407	1.050589	.9708352	1.130913
L2.	-.0988574	.0381619	.000382	-.0983389	-.1745514	-.0257632
ogap						

Comparing lags

```
. bayes, saving(bvarsim1): var inflation ogap fedfunds, lags(1/1)
. estimates store lag1
. bayes, saving(bvarsim2): var inflation ogap fedfunds
. estimates store lag2
. bayes, saving(bvarsim3): var inflation ogap fedfunds, lags(1/3)
. estimates store lag3
. bayes, saving(bvarsim4): var inflation ogap fedfunds, lags(1/4)
. estimates store lag4
. bayestest model lag1 lag2 lag3 lag4
```

Bayesian model tests

	log(ML)	P(M)	P(M y)
lag1	-814.1542	0.2500	0.0000
lag2	-803.6239	0.2500	0.0003
lag3	-797.0075	0.2500	0.1891
lag4	-795.5522	0.2500	0.8106

Parameter stability

. bayesvarstable

Eigenvalue stability condition

Companion matrix size = **12**

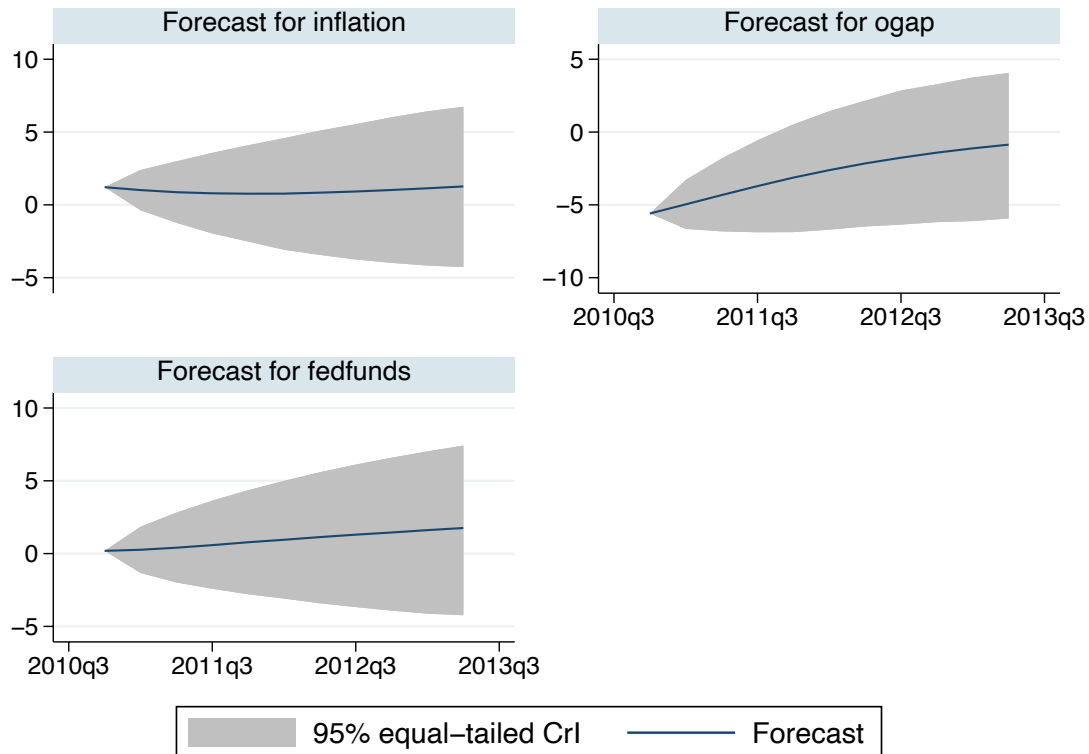
MCMC sample size = **10000**

Eigenvalue modulus	Equal-tailed					
	Mean	Std. dev.	MCSE	Median	[95% cred. interval]	
1	.9427379	.0206882	.000207	.9433122	.900931	.9827988
2	.9295124	.0290971	.000291	.934122	.8558804	.9744941
3	.8607955	.0563407	.000563	.8727556	.7227223	.938854
4	.5626488	.0905431	.000905	.5520041	.4093539	.7696678
5	.4665878	.0794672	.000795	.4642645	.3287306	.622297
6	.3717843	.0479491	.000479	.3668113	.2891107	.4823732
7	.3459041	.0361596	.000362	.3463737	.2759032	.4158424
8	.3179581	.038563	.000386	.3196635	.2388367	.3885053
9	.2986474	.039211	.000392	.300947	.2168552	.3704618
10	.2693415	.0474767	.000475	.2737034	.1635202	.3487643
11	.2332	.0557138	.000557	.2385186	.1132732	.3283775
12	.1910737	.0794654	.000795	.2066833	.015231	.3104382

Pr(eigenvalues lie inside the unit circle) = **0.9968**

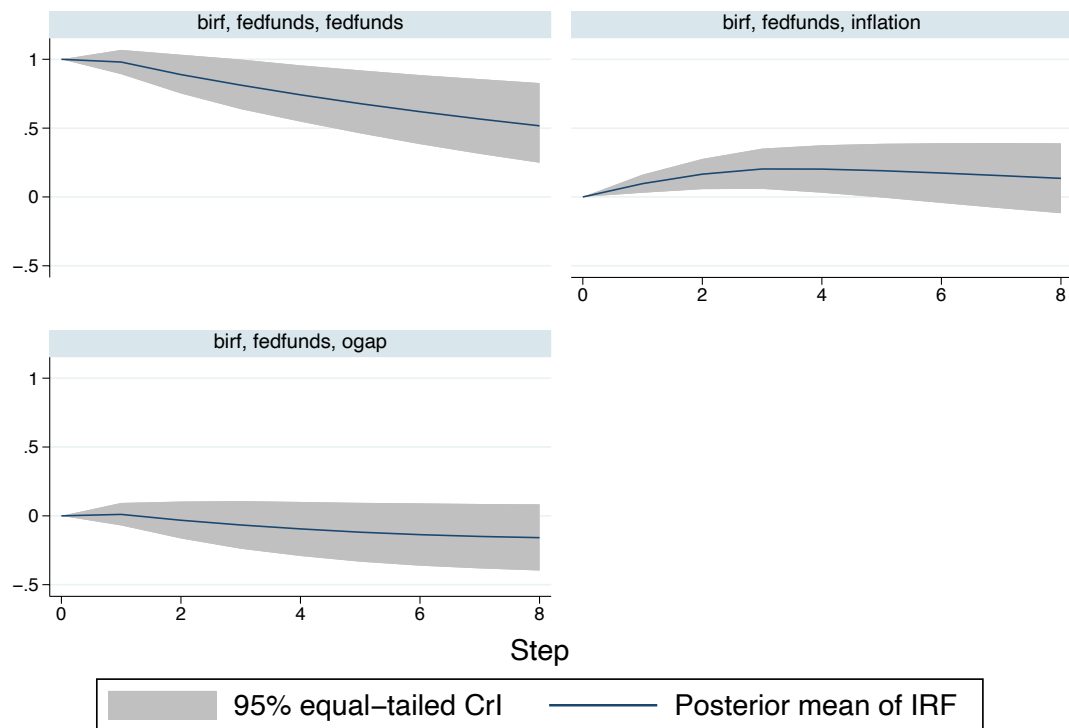
Dynamic forecasts

- . bayesfcst compute f_, step(10)
- . bayesfcst graph f_inflation f_ogap f_fedfunds



IRF analysis

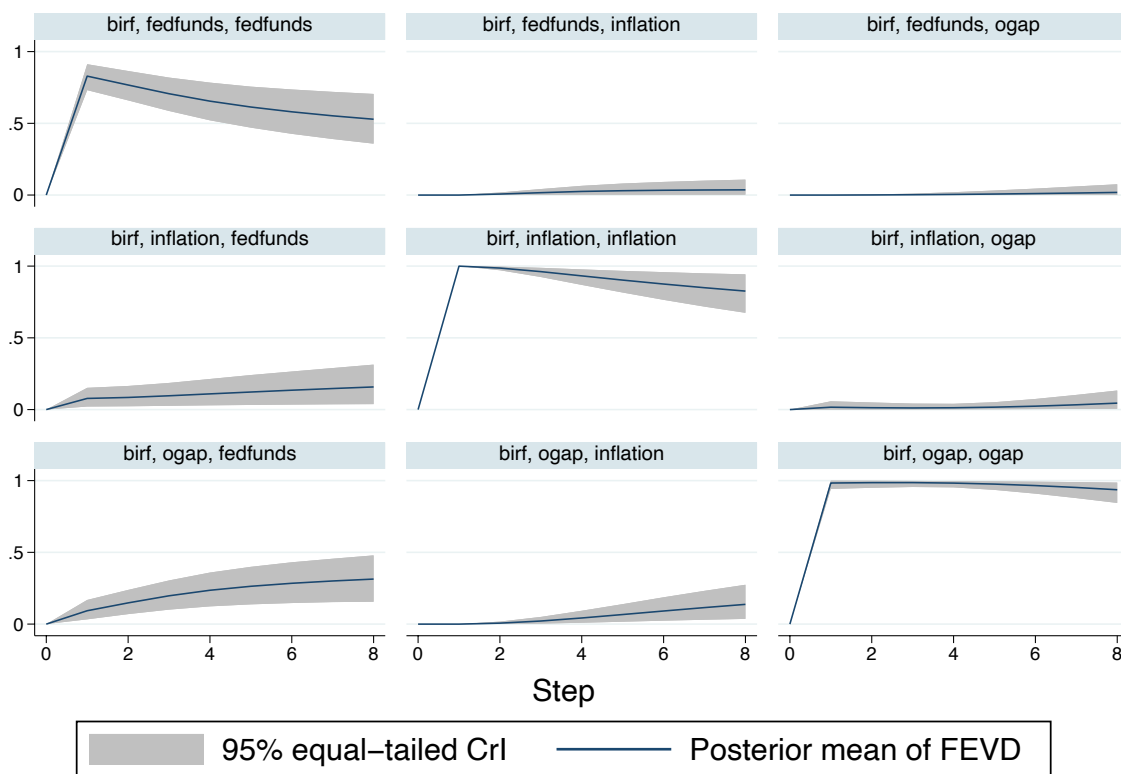
```
. bayesirf create birf, set(birfex, replace)  
. bayesirf graph irf, impulse(fedfunds)
```



Graphs by irfname, impulse variable, and response variable

FEVD analysis

. bayesirf graph fevd



Graphs by irfname, impulse variable, and response variable

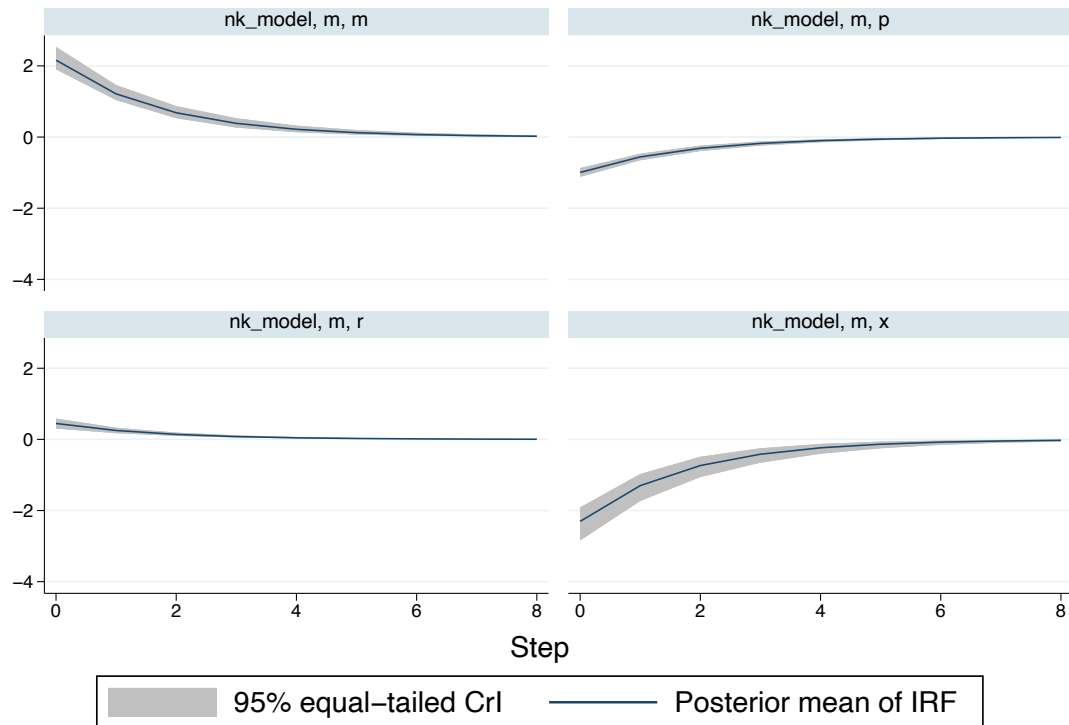
Bayesian DSGE models

```
. bayes, prior({beta}, beta(95,5)) prior({kappa}, beta(30,70)) ///
> prior({delta}, beta(60,30)) prior({rhoz}, beta(10,10)) ///
> prior({rhom}, beta(10,10)) rseed(17) : ///
> dsge ( x = F.x - (r - F.p - z), unobserved) ///
> ( p = {beta}*F.p + {kappa}*x ) ///
> ( r = 1/{delta}*p + m ) ///
> (F.z = {rhoz}*z, state ) ///
> (F.m = {rhom}*m, state )
```

	Mean	Std. dev.	MCSE	Median	Equal-tailed [95% cred. interval]	
beta	.9406005	.0243801	.001984	.9428465	.8869374	.9812481
kappa	.206337	.0327608	.001679	.2046031	.1470858	.273637
delta	.5832685	.0404188	.004897	.5835278	.497712	.6607058
rhoz	.9171911	.015764	.000978	.9170596	.8846393	.9467265
rhom	.561412	.0296339	.001667	.5621747	.5036024	.6184265
sd(e.z)	.5280986	.057275	.003639	.5255331	.4217194	.6467684
sd(e.m)	2.161816	.1585077	.020872	2.141682	1.902066	2.533669

IRF analysis

```
. bayesirf create nk_model, set(nk)  
. bayesirf graph irf, impulse(m) response(x p r m)
```



Graphs by irfname, impulse variable, and response variable

bayes :

xtreg

xtlogit

xtprobit

xtpoisson

xtnbreg

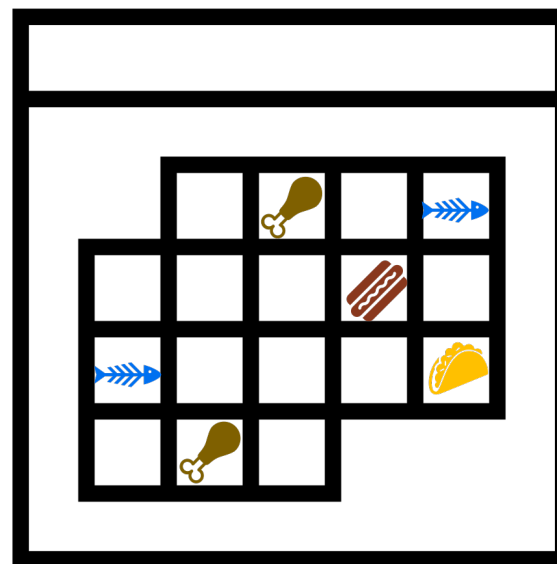
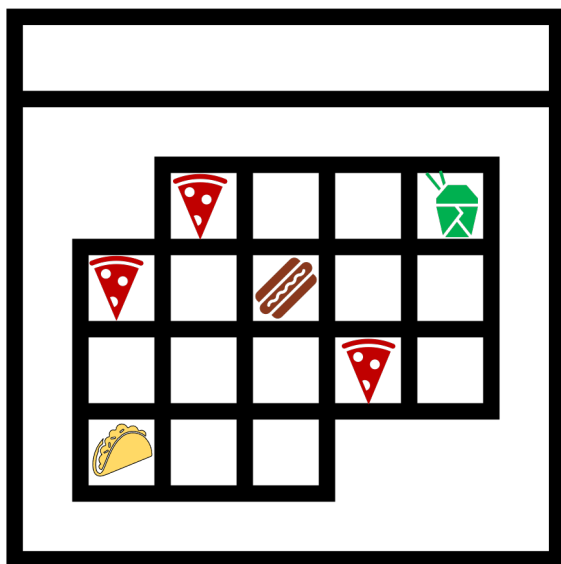
xtologit

xtoprobit

xtnlogit

Panel-data multinomial logit (MNL) models

How does a categorical outcome change over time?



Panel-data MNL model

Random effects

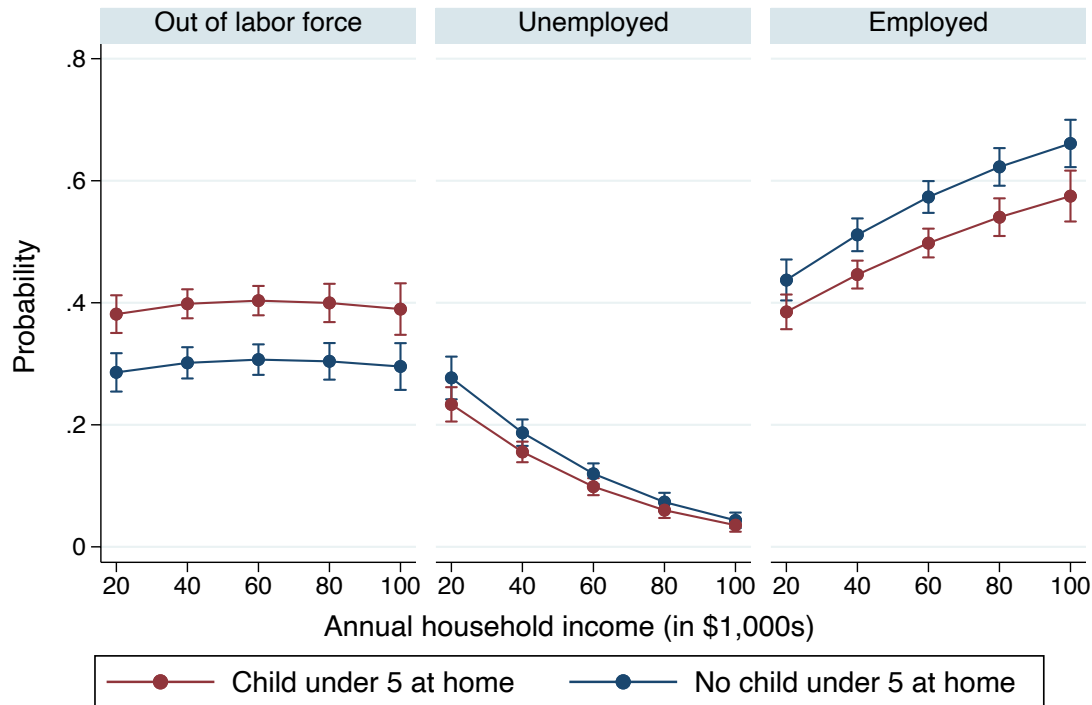
```
. xtmlogit estatus i.hhchild hhincome, rrr
```

estatus	RRR	Std. err.	z	P> z	[95% conf. interval]	
Out_of_labor_force						
hhchild						
Yes	1.626838	.1377206	5.75	0.000	1.378115	1.92045
hhincome	.9952556	.0018042	-2.62	0.009	.9917258	.998798
_cons	.6189663	.0739387	-4.02	0.000	.4897638	.7822531
Unemployed						
hhchild						
Yes	.9335288	.0980888	-0.65	0.513	.7597826	1.147007
hhincome	.9698262	.0025539	-11.63	0.000	.9648336	.9748448
_cons	1.01785	.145543	0.12	0.902	.7690775	1.347093
Employed	(base outcome)					
var(u1)	.8422651	.1068418			.6568608	1.080001
var(u2)	.7354583	.1384766			.5084973	1.06372

Marginal probabilities

```
. margins hhchild, at(hhincome=(20(20)100))  
. marginsplot
```

Marginal probabilities of employment status



Panel-data MNL model

Fixed effects

```
. xtmlogit estatus i.hhchild hhincome, fe
```

estatus	Coefficient	Std. err.	z	P> z	[95% conf. interval]	
Out_of_labor_force						
hhchild						
Yes	.5888651	.1212493	4.86	0.000	.3512209	.8265093
hhincome	-.0116242	.0063646	-1.83	0.068	-.0240986	.0008501
Unemployed						
hhchild						
Yes	.1533919	.1570554	0.98	0.329	-.1544309	.4612148
hhincome	-.02712	.0088786	-3.05	0.002	-.0445218	-.0097182
Employed	(base outcome)					

Panel-data MNL model

Bayesian

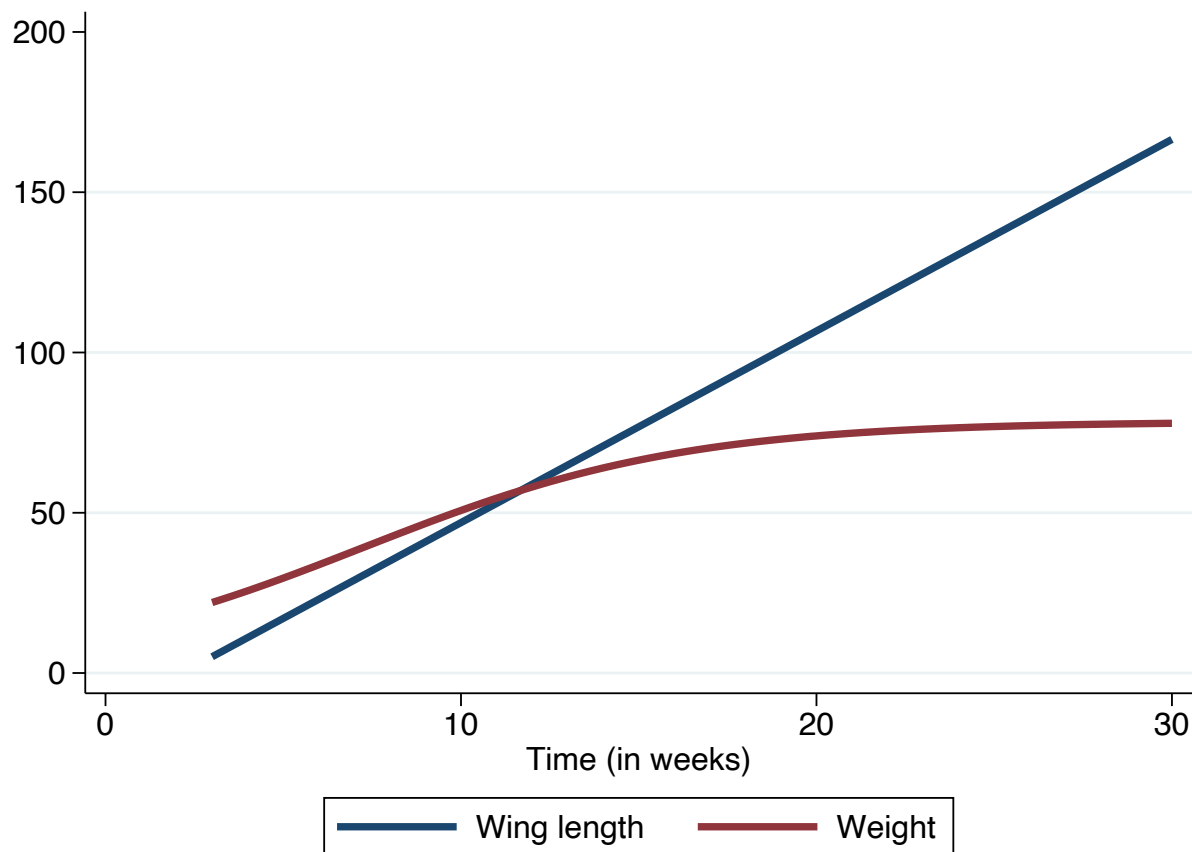
```
. bayes, prior({Out_of_labor_force:1.hhchild}, normal(.5,.5)): ///
> xtmlogit estatus i.hhchild hhincome
```

	Mean	Std. dev.	MCSE	Median	Equal-tailed [95% cred. interval]	
Out_of_labor_force						
hhchild						
Yes	.490668	.0842927	.004649	.4911029	.3232187	.6552001
hhincome	-.0047925	.001925	.000124	-.0048067	-.0084107	-.0010929
U1	1	0	0	1	1	1
_cons	-.4845118	.1235976	.007867	-.4851631	-.7257886	-.2386423
Unemployed						
hhchild						
Yes	-.0623994	.1041457	.004137	-.0635373	-.279373	.1387273
hhincome	-.0308579	.0027411	.000158	-.0307458	-.0361278	-.0255231
U2	1	0	0	1	1	1
_cons	.0078718	.144715	.007923	.0054914	-.2654459	.2965143

New in bayesmh

- Random-effects syntax!
 - Multivariate multilevel models
 - Nonlinear multilevel models
 - SEM-like models
- Exchangeable and identity covariance structures
- Parametric survival likelihoods

Chick growth



Bayesian MV multilevel model

```
. bayesmh (wing = ({U[id]} + time*{V[id]}))                               ///  
>      (weight = ({C[id]}/(1+{d}*{C[id]}*exp(-{B[id]}*time)))),           ///  
> likelihood(mvnormal({Sigma0,m}))                                       ///  
> prior({U V C B}, mvnormal(4,{u},{v},{c},{b},{Sigma,m}))              ///  
> prior({u v c b}, normal(0, 100)) prior({Sigma0,m}, iwishart(2,3,I(2))) ///  
> prior({Sigma,m}, iwishart(4,5,I(4))) prior({d}, exp(1))               ///  
> block({d u v b c}, split) block({Sigma0,m} {Sigma,m}, gibbs split)   ///  
> init({U[id] u} -10 {V[id] v} 10 {C[id] c} 100 {d} 1) mcmcsize(2500) rseed(17)
```

Burn-in 2500 aaaaaaaaaa1000aaaaaaaaa2000aaaaaa done

Simulation 25001000.....2000..... done

Model summary

Likelihood:

wing weight ~ mvnormal(2,<expr1>,<expr2>,{Sigma0,m})

Priors:

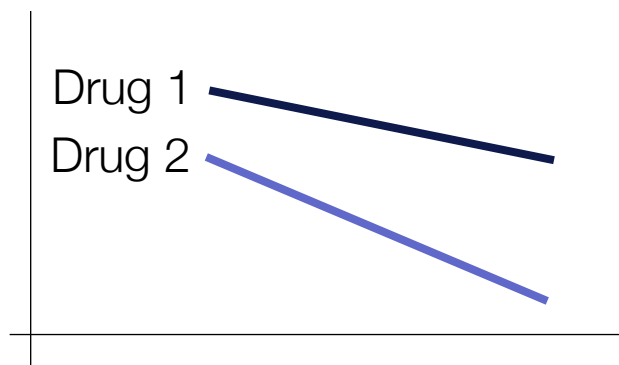
{Sigma0,m} ~ iwishart(2,3,I(2))

{U[id] V[id] C[id] B[id]} ~ mvnormal(4,{u},{v},{c},{b},{Sigma,m})

{d} ~ exponential(1)

Informative dropout model

How do symptoms of schizophrenia change over time in response to pharmaceutical intervention?



```
. misstable patterns panss*, bypattern
```

Percent	Pattern				
	1	2	3	4	5
45%	1	1	1	1	1
1:					
11	1	1	1	1	0
2:					
16	1	1	1	0	0
3:					
13	1	1	0	0	0
4:					
14	1	0	0	0	0
5:					
1	0	0	0	0	0
100%					

Informative dropout model

```
. bayesmh (panss i.treat##i.week U[id]@1, likeli(norm({var})))  
> (t1 i.treat U[id], likeli(stweibull({lnp}), failure(dropinf) ltrunc(t0))),  
> prior({panss:i.treat##i.week _cons},          normal(0,10000))  
> prior({U} ,                                   normal(0, {var_U}))  
> prior({t1:i.treat U _cons},                    normal(0,10000))  
> prior({var_U} {var},                           igamma(.01, .01))  
> prior({lnp},                                   normal(0, 10000))  
> block({panss:i.treat##i.week _cons}) block({t1: i.treat U _cons})  
> block({var_U} {var}) thinning(5) burnin(10000)
```

Model summary

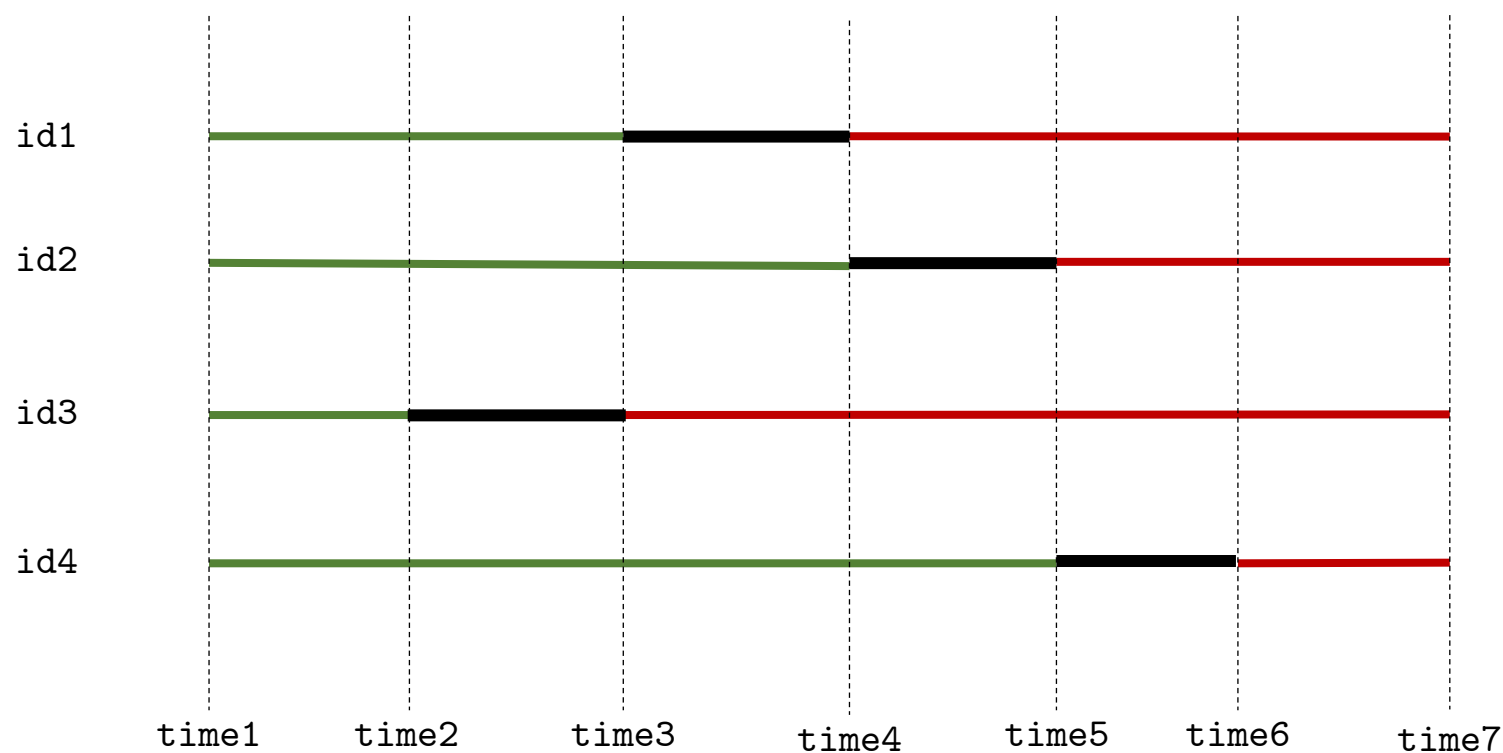
Likelihood:

```
panss ~ normal(xb_panss,{var})  
t1 ~ stweibull(xb_t1,{lnp})
```

Priors:

```
{panss:i.treat i.week i.treat#i.week _cons} ~ normal(0,10000)  
{U[id]} ~ normal(0,{var_U})  
{var} ~ igamma(.01,.01)  
{t1:i.treat cons U} ~ normal(0,10000)
```

Interval-censored Cox models



$$h(t; \mathbf{x}) = \underline{h_0(t)} \exp(\mathbf{x}\boldsymbol{\beta})$$

Interval-censored Cox models

```
. stintcox age_mean i.male i.needle, interval(ltime rtime)
```

Interval-censored Cox regression
Baseline hazard: Reduced intervals

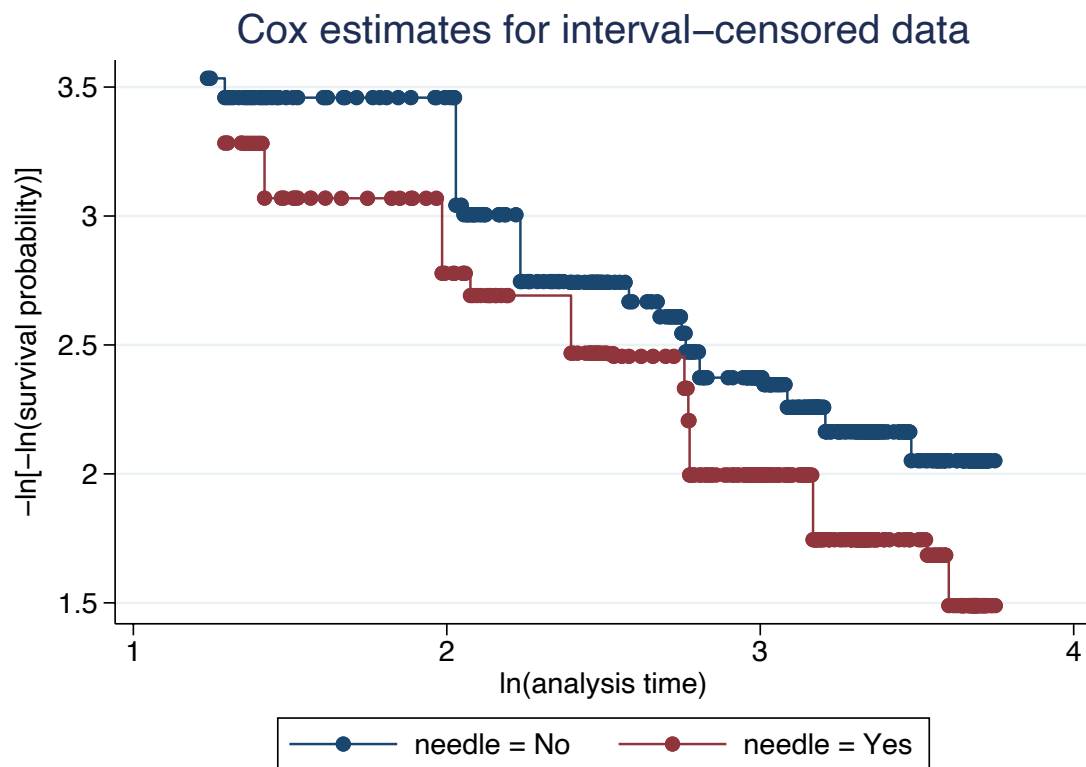
Number of obs = 1,124
Uncensored = 0
Left-censored = 41
Right-censored = 991
Interval-cens. = 92
Wald chi2(3) = 11.98
Prob > chi2 = 0.0075

Log likelihood = -601.83734

		OPG				
	Haz. ratio	std. err.	z	P> z	[95% conf. interval]	
age_mean	.975432	.0122761	-1.98	0.048	.9516657	.9997919
male						
Yes	.6421759	.176132	-1.61	0.106	.3751386	1.0993
needle						
Yes	1.419895	.2479212	2.01	0.045	1.008398	1.999312

Diagnostics

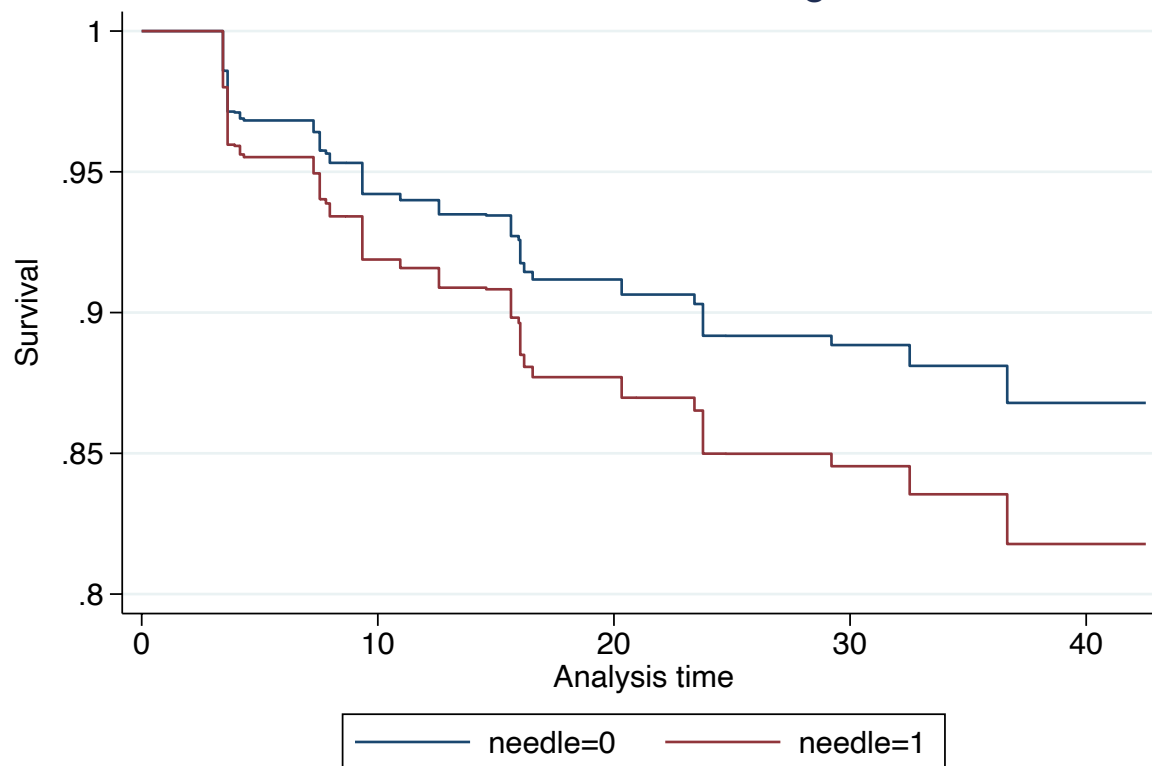
```
. stintphplot, interval(ltime rtime) by(needle) adjustfor(age_mean i.male)
```



Survivor functions

```
. stcurve, survival at(needle=(0 1))
```

Interval-censored Cox regression



New in lasso

- BIC for lasso penalty selection
- Lasso with clustered data
- Treatment-effects estimation using lasso

Prediction

$$\hat{y} = ?$$

OLS Regression

\hat{y}	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
	x_{10}	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}		
	x_{17}	x_{18}	x_{19}	x_{20}	x_{21}	x_{22}	x_{23}		
	x_{24}	x_{25}	x_{26}	x_{27}	x_{28}	x_{29}	x_{30}		
	x_{31}	x_{32}	x_{33}	x_{34}	x_{35}	x_{36}	x_{37}		
	x_{38}	x_{39}	x_{40}	x_{41}	x_{42}	x_{43}	x_{44}		
	x_{45}	x_{46}	x_{47}	x_{48}	x_{49}	x_{50}	x_{51}		
	x_{52}	x_{53}	x_{54}	x_{55}	x_{56}	x_{57}	x_{58}		
	x_{59}	x_{60}	x_{61}	x_{62}	x_{63}	x_{64}	x_{65}		
	x_{66}	x_{67}	x_{68}	x_{69}	x_{70}	x_{71}	x_{72}		
	x_{73}	x_{74}	x_{75}	x_{76}	x_{77}	...	x_{1000}		

OLS Regression $\sum_i \left(y_i - \sum_j x_{ij} \beta_j \right)^2$

$$\begin{aligned} \hat{y} = & \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + \beta_5 x_5 + \beta_6 x_6 + \beta_7 x_7 + \beta_8 x_8 + \beta_9 x_9 \\ & + \beta_{10} x_{10} + \beta_{11} x_{11} + \beta_{12} x_{12} + \beta_{13} x_{13} + \beta_{14} x_{14} + \beta_{15} x_{15} + \beta_{16} x_{16} \\ & + \beta_{17} x_{17} + \beta_{18} x_{18} + \beta_{19} x_{19} + \beta_{20} x_{20} + \beta_{21} x_{21} + \beta_{22} x_{22} + \beta_{23} x_{23} \\ & + \beta_{24} x_{24} + \beta_{25} x_{25} + \beta_{26} x_{26} + \beta_{27} x_{27} + \beta_{28} x_{28} + \beta_{29} x_{29} + \beta_{30} x_{30} \\ & + \beta_{31} x_{31} + \beta_{32} x_{32} + \beta_{33} x_{33} + \beta_{34} x_{34} + \beta_{35} x_{35} + \beta_{36} x_{36} + \beta_{37} x_{37} \\ & + \beta_{38} x_{38} + \beta_{39} x_{39} + \beta_{40} x_{40} + \beta_{41} x_{41} + \beta_{42} x_{42} + \beta_{43} x_{43} + \beta_{44} x_{44} \\ & + \beta_{45} x_{45} + \beta_{46} x_{46} + \beta_{47} x_{47} + \beta_{48} x_{48} + \beta_{49} x_{49} + \beta_{50} x_{50} + \beta_{51} x_{51} \\ & + \beta_{52} x_{52} + \beta_{53} x_{53} + \beta_{54} x_{54} + \beta_{55} x_{55} + \beta_{56} x_{56} + \beta_{57} x_{57} + \beta_{58} x_{58} \\ & + \beta_{59} x_{59} + \beta_{60} x_{60} + \beta_{61} x_{61} + \beta_{62} x_{62} + \beta_{63} x_{63} + \beta_{64} x_{64} + \beta_{65} x_{65} \\ & + \beta_{66} x_{66} + \beta_{67} x_{67} + \beta_{68} x_{68} + \beta_{69} x_{69} + \beta_{70} x_{70} + \beta_{71} x_{71} + \beta_{72} x_{72} \\ & + \beta_{73} x_{73} + \beta_{74} x_{74} + \beta_{75} x_{75} + \beta_{76} x_{76} + \beta_{77} x_{77} + \cdots + \beta_{1000} x_{1000} \end{aligned}$$

Lasso

$$\sum_i \left(y_i - \sum_j x_{ij} \beta_j \right)^2 + \lambda \sum_j |\beta_j|$$

$$\begin{aligned} \hat{y} = & \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + \beta_5 x_5 + \beta_6 x_6 + \beta_7 x_7 + \beta_8 x_8 + \beta_9 x_9 \\ & + \beta_{10} x_{10} + \beta_{11} x_{11} + \beta_{12} x_{12} + \beta_{13} x_{13} + \beta_{14} x_{14} + \beta_{15} x_{15} + \beta_{16} x_{16} \\ & + \beta_{17} x_{17} + \beta_{18} x_{18} + \beta_{19} x_{19} + \beta_{20} x_{20} + \beta_{21} x_{21} + \beta_{22} x_{22} + \beta_{23} x_{23} \\ & + \beta_{24} x_{24} + \beta_{25} x_{25} + \beta_{26} x_{26} + \beta_{27} x_{27} + \beta_{28} x_{28} + \beta_{29} x_{29} + \beta_{30} x_{30} \\ & + \beta_{31} x_{31} + \beta_{32} x_{32} + \beta_{33} x_{33} + \beta_{34} x_{34} + \beta_{35} x_{35} + \beta_{36} x_{36} + \beta_{37} x_{37} \\ & + \beta_{38} x_{38} + \beta_{39} x_{39} + \beta_{40} x_{40} + \beta_{41} x_{41} + \beta_{42} x_{42} + \beta_{43} x_{43} + \beta_{44} x_{44} \\ & + \beta_{45} x_{45} + \beta_{46} x_{46} + \beta_{47} x_{47} + \beta_{48} x_{48} + \beta_{49} x_{49} + \beta_{50} x_{50} + \beta_{51} x_{51} \\ & + \beta_{52} x_{52} + \beta_{53} x_{53} + \beta_{54} x_{54} + \beta_{55} x_{55} + \beta_{56} x_{56} + \beta_{57} x_{57} + \beta_{58} x_{58} \\ & + \beta_{59} x_{59} + \beta_{60} x_{60} + \beta_{61} x_{61} + \beta_{62} x_{62} + \beta_{63} x_{63} + \beta_{64} x_{64} + \beta_{65} x_{65} \\ & + \beta_{66} x_{66} + \beta_{67} x_{67} + \beta_{68} x_{68} + \beta_{69} x_{69} + \beta_{70} x_{70} + \beta_{71} x_{71} + \beta_{72} x_{72} \\ & + \beta_{73} x_{73} + \beta_{74} x_{74} + \beta_{75} x_{75} + \beta_{76} x_{76} + \beta_{77} x_{77} + \cdots + \beta_{1000} x_{1000} \end{aligned}$$

Lasso in Stata

```
. lasso linear y $covariates
```

Lasso linear model

No. of obs = **914**

No. of covariates = **279**

Selection: **Cross-validation**

No. of CV folds = **10**

ID	Description	lambda	No. of nonzero coef.	Out-of- sample R-squared	CV mean prediction error
1	first lambda	.9090511	0	-0.0010	18.33331
23	lambda before	.1174085	58	0.3542	11.82865
* 24	selected lambda	.1069782	64	0.3545	11.82247
25	lambda after	.0974746	66	0.3542	11.82811
28	last lambda	.0737359	80	0.3482	11.93742

* lambda selected by cross-validation.

BIC selection

```
. lasso linear y $covariates, selection(bic)
```

```
Lasso linear model                No. of obs      =      914
                                   No. of covariates =      279
```

```
Selection: Bayesian information criterion
```

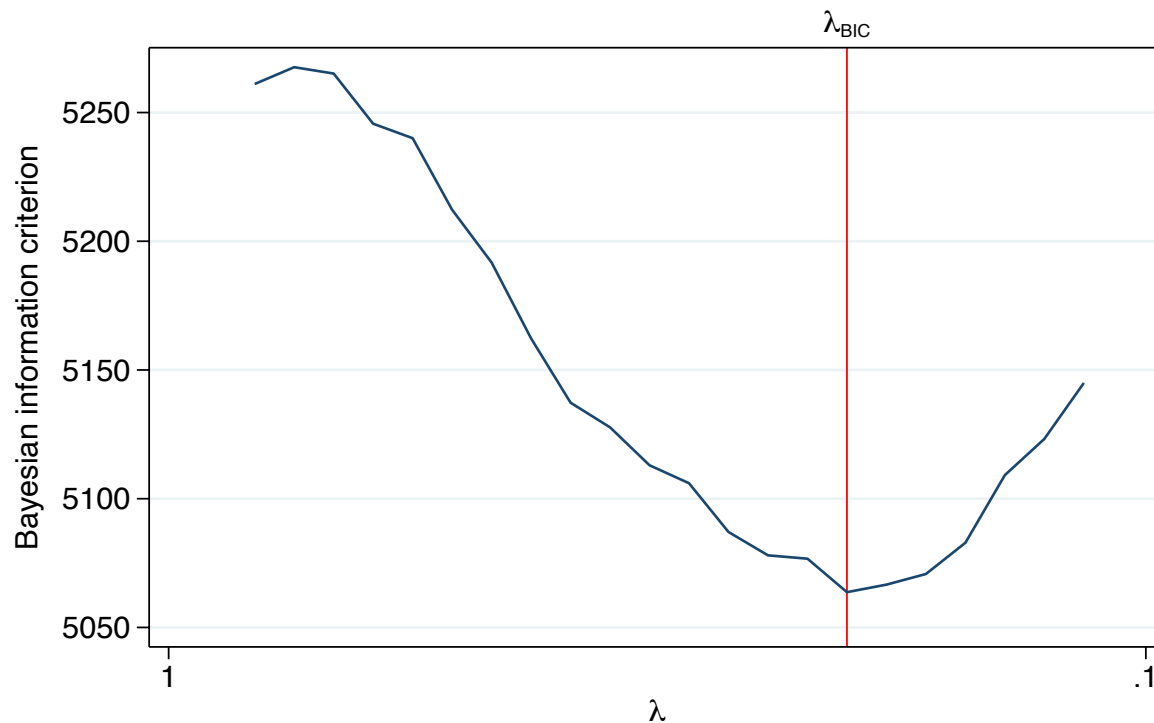
ID	Description	lambda	No. of nonzero coef.	In-sample R-squared	BIC
1	first lambda	.9090511	0	0.0000	5258.295
17	lambda before	.2051746	32	0.3750	5046.809
* 18	selected lambda	.1869475	34	0.3870	5042.754
19	lambda after	.1703396	39	0.3981	5060.244
24	last lambda	.1069782	64	0.4418	5161.662

```
* lambda selected by Bayesian information criterion
```

BIC selection

```
. bicplot
```

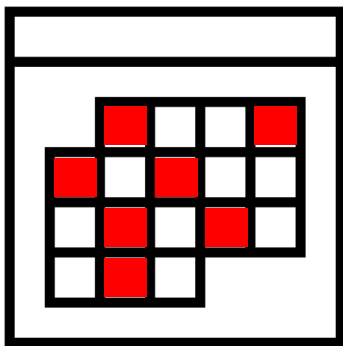
BIC plot



λ_{BIC} BIC minimum lambda. $\lambda=.2$, # Coefficients=35.

Lasso with clustered data

- Observations sampled repeatedly over time



- Observations nested in clusters



Lasso for clustered data

for prediction

```
. lasso linear ln_wage $covariates, cluster(idcode) selection(bic)
```

```
Lasso linear model                No. of obs          =      28,093
                                   No. of covariates      =           45
Cluster   : idcode                No. of clusters       =      4,699
Selection: Bayesian information criterion
```

ID	Description	lambda	No. of nonzero coef.	In-sample R-squared	BIC
1	first lambda	.2261424	0	0.0000	6884.549
66	lambda before	.0005347	22	0.3119	5313.687
* 67	selected lambda	.0004872	22	0.3121	5312.367
68	lambda after	.0004439	23	0.3123	5319.71
92	last lambda	.0000476	24	0.3131	5322.693

* lambda selected by Bayesian information criterion

Lasso for clustered data

for inference

```
. dsregress ln_wage tenure, controls($controls) vce(cluster idcode) selection(bic)
```

Estimating lasso for ln_wage using BIC

Estimating lasso for tenure using BIC

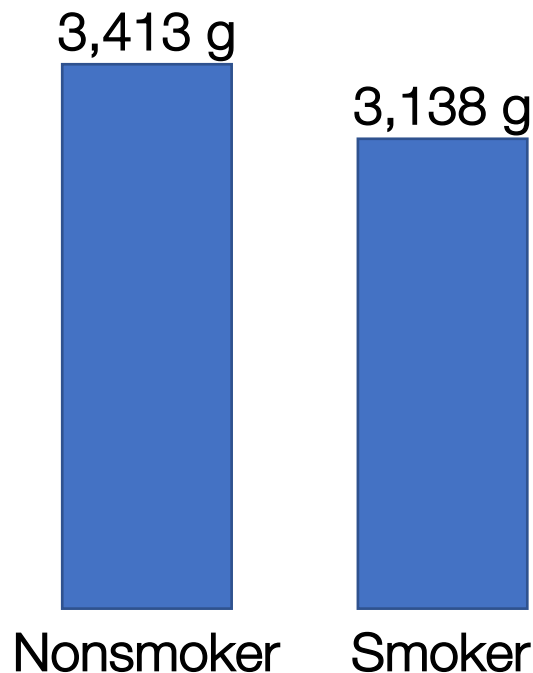
```
Double-selection linear model      Number of obs      =      28,093
                                   Number of controls      =          35
                                   Number of selected controls =          21
                                   Wald chi2(1)              =      173.33
                                   Prob > chi2               =      0.0000
```

(Std. err. adjusted for 4,699 clusters in idcode)

ln_wage	Robust					
	Coefficient	std. err.	z	P> z	[95% conf. interval]	
tenure	.0219364	.0016662	13.17	0.000	.0186707	.0252021

Treatment effects

What is the effect of smoking on birthweight?



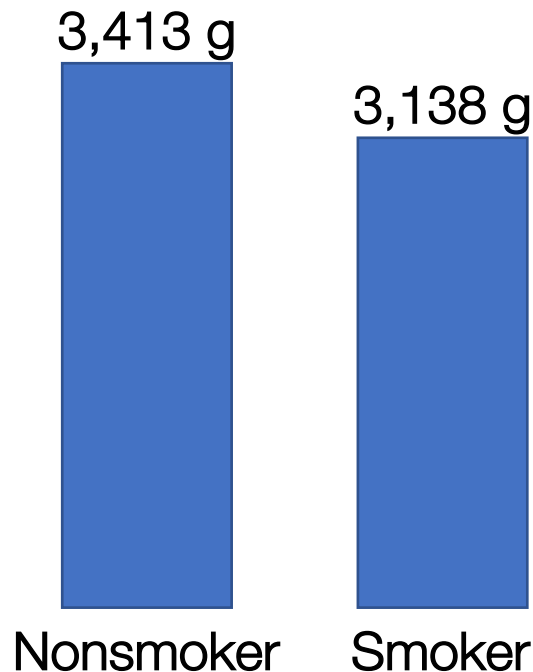
Augmented inverse-probability weighting

bweight = prenatal1 mmarried mage fbaby

$\text{logit}(\text{pr}(\text{smoking})) = \text{mmarried c.mage order}$

Treatment effects

What is the effect of smoking on birthweight?



Augmented inverse-probability weighting

```

bweight = prenatal1 mmarried age fbaby
         mhispanic fhispanic foreign alcohol
         leadkids mrace frace fbaby medu
         fedu prenatal monthslb order
         prenatal gestage gdiabetes com...

logit(pr(smoking)) = mmarried mmarried_order
                   mhispanic fhispanic foreign alcohol
                   mrace frace medu fedu
                   parentsmk peersmk cprice
                   hhincoburban prenatal...
  
```

TE lasso

```

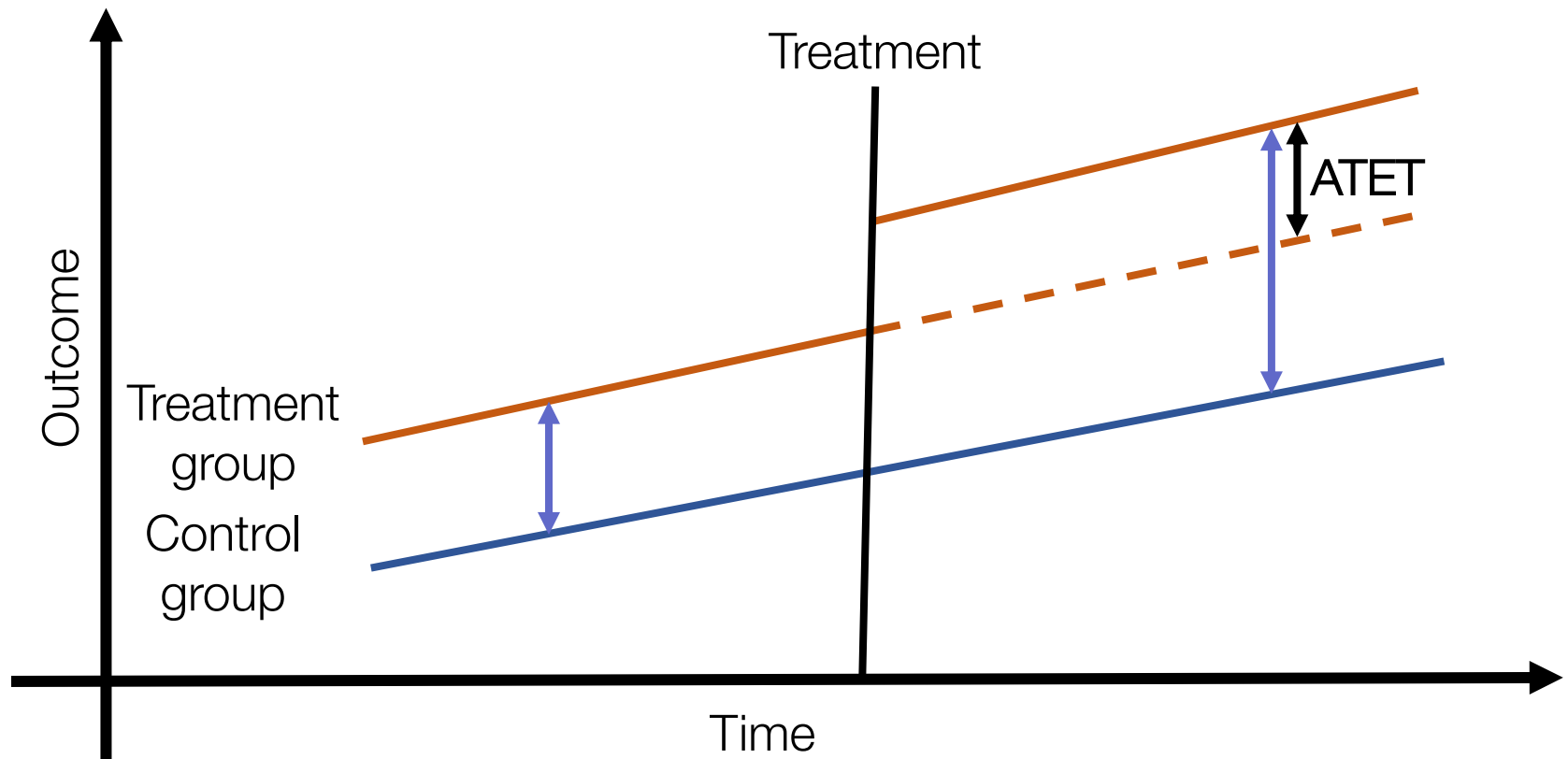
. telasso (bweight $controls) (mbsmoke $controls), selection(bic)

Estimating lasso for outcome bwei~t if mbsm~e = 0 using BIC ...
Estimating lasso for outcome bwei~t if mbsm~e = 1 using BIC ...
Estimating lasso for treatment mbsm~e using BIC ...
Estimating ATE ...
Treatment-effects lasso estimation      Number of observations      =      4,642
Outcome model:  linear                  Number of controls          =      404
Treatment model: logit                  Number of selected controls =      46

```

bweight	Robust					
	Coefficient	std. err.	z	P> z	[95% conf. interval]	
ATE						
mbsmoke						
(Smoker						
vs						
Nonsmoker)	-229.9705	28.10368	-8.18	0.000	-285.0526	-174.8883

Difference in differences (DID)



Difference in differences (DID)

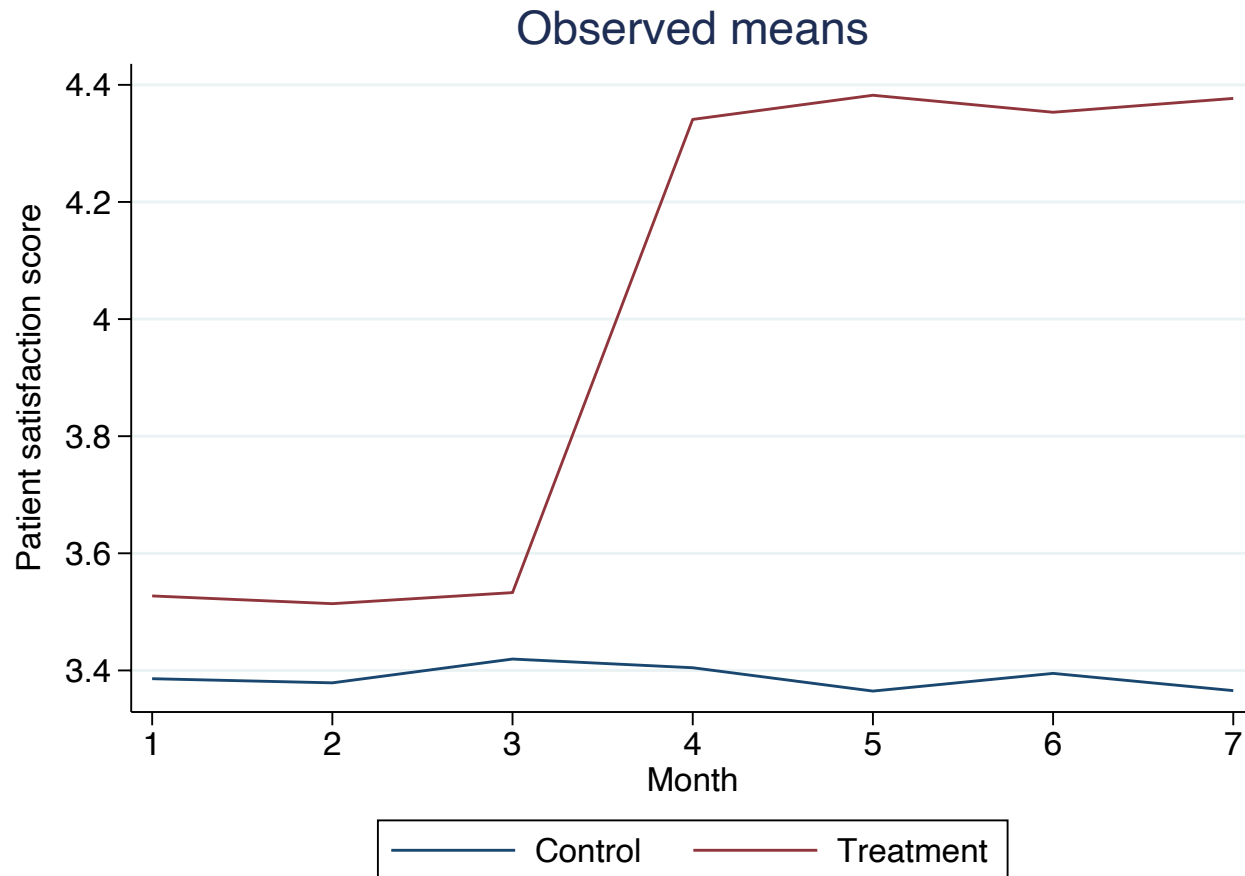
- Estimate the average treatment effect on the treated (ATET) in nonexperimental designs
- with repeated cross-sectional data – `didregress`



- with panel (longitudinal) data – `xtdidregress`



A new admissions procedure



DID in Stata

```
. didregress (satis) (procedure), group(hospital) time(month)
```

Difference-in-differences regression

Number of obs = 7,368

Data type: Repeated cross-sectional

(Std. err. adjusted for 46 clusters in hospital)

satis	Robust					
	Coefficient	std. err.	t	P> t	[95% conf. interval]	
ATET procedure (New vs Old)	.8479879	.0321121	26.41	0.000	.7833108	.912665

Note: ATET estimate adjusted for group effects and time effects.

DID in Stata

```
. didregress (satis frequency) (procedure), group(hospital) time(month)
```

Difference-in-differences regression

Number of obs = 7,368

Data type: Repeated cross-sectional

(Std. err. adjusted for 46 clusters in hospital)

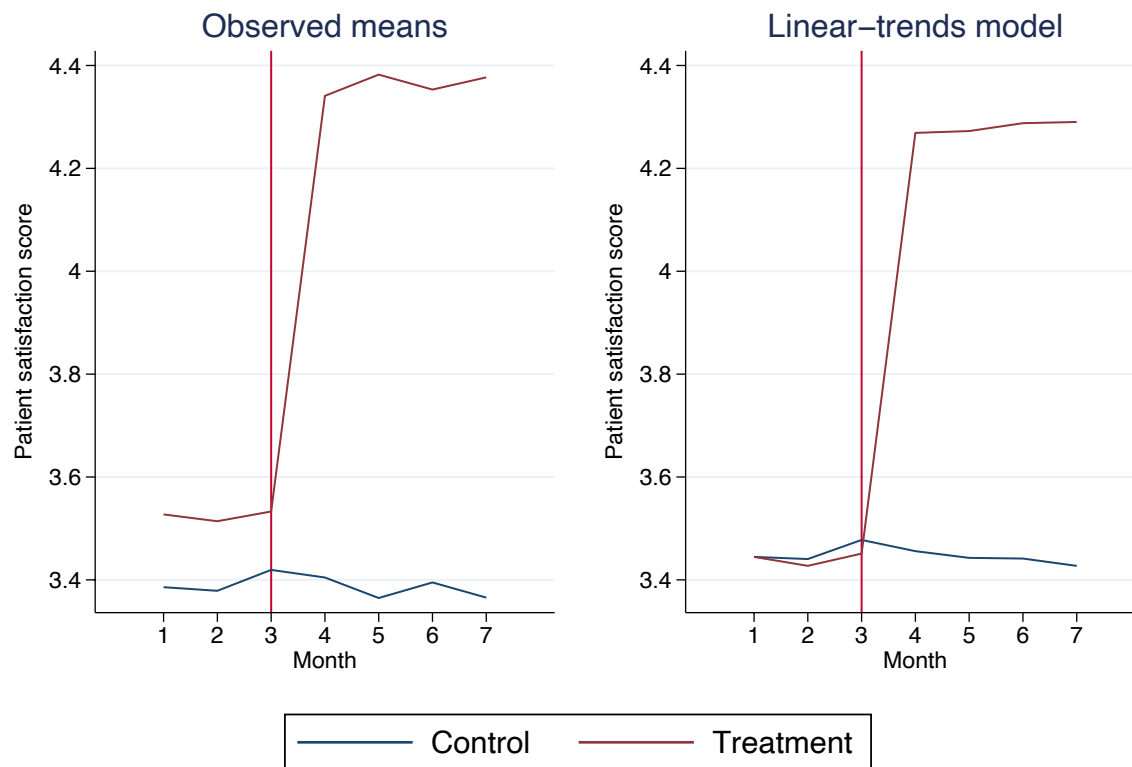
satis	Robust					
	Coefficient	std. err.	t	P> t	[95% conf. interval]	
ATET procedure (New vs Old)	.8479879	.0321143	26.41	0.000	.7833063	.9126694

Note: ATET estimate adjusted for covariates, group effects, and time effects.

DID diagnostics

```
. estat trendplots
```

Graphical diagnostics for parallel trends



DID diagnostics

```
. estat ptrends
```

Parallel-trends test (pretreatment time period)

H0: Linear trends are parallel

$F(1, 45) = 0.55$

Prob > F = 0.4616

```
. estat granger
```

Granger causality test

H0: No effect in anticipation of treatment

$F(2, 45) = 0.33$

Prob > F = 0.7240

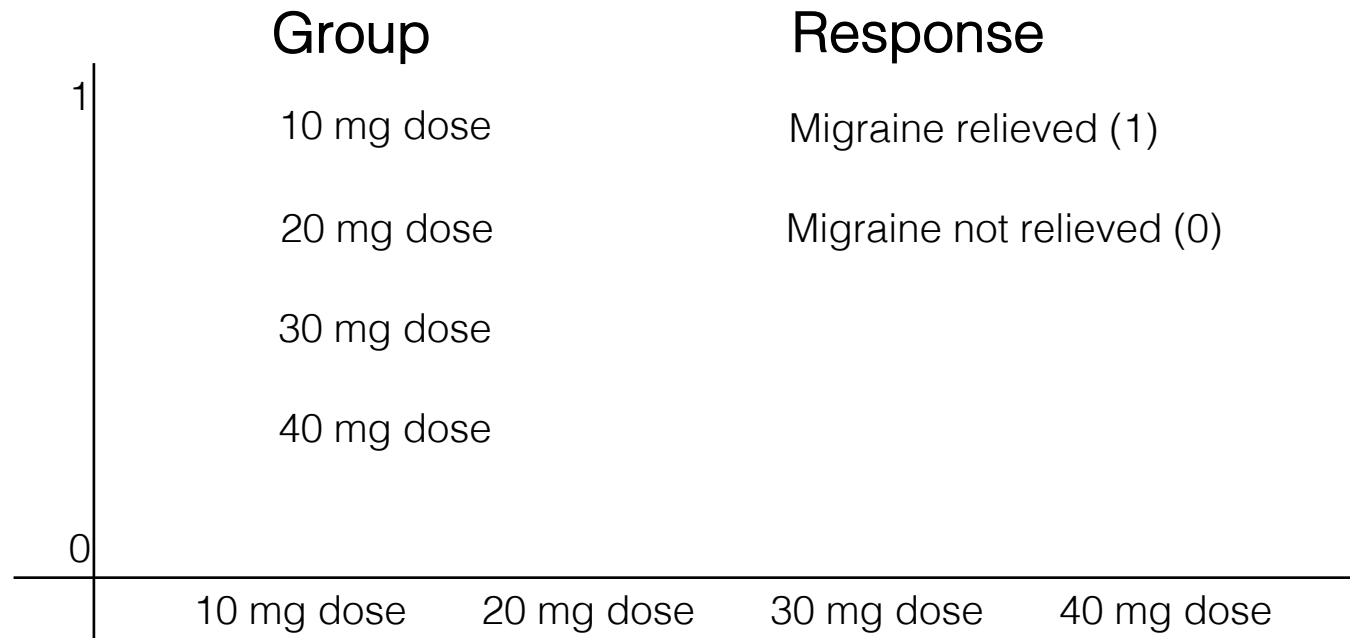
Small number of groups

- Bell and McCaffrey (2002) standard errors
 - `vce(hc2)` option
- Donald and Lang (2007) ATET estimates and standard errors
 - `aggregate(dlang)` option
 - `aggregate(dlang, varying)` option
- Wild-cluster bootstrap p -values and confidence intervals.
 - `wildbootstrap()` option

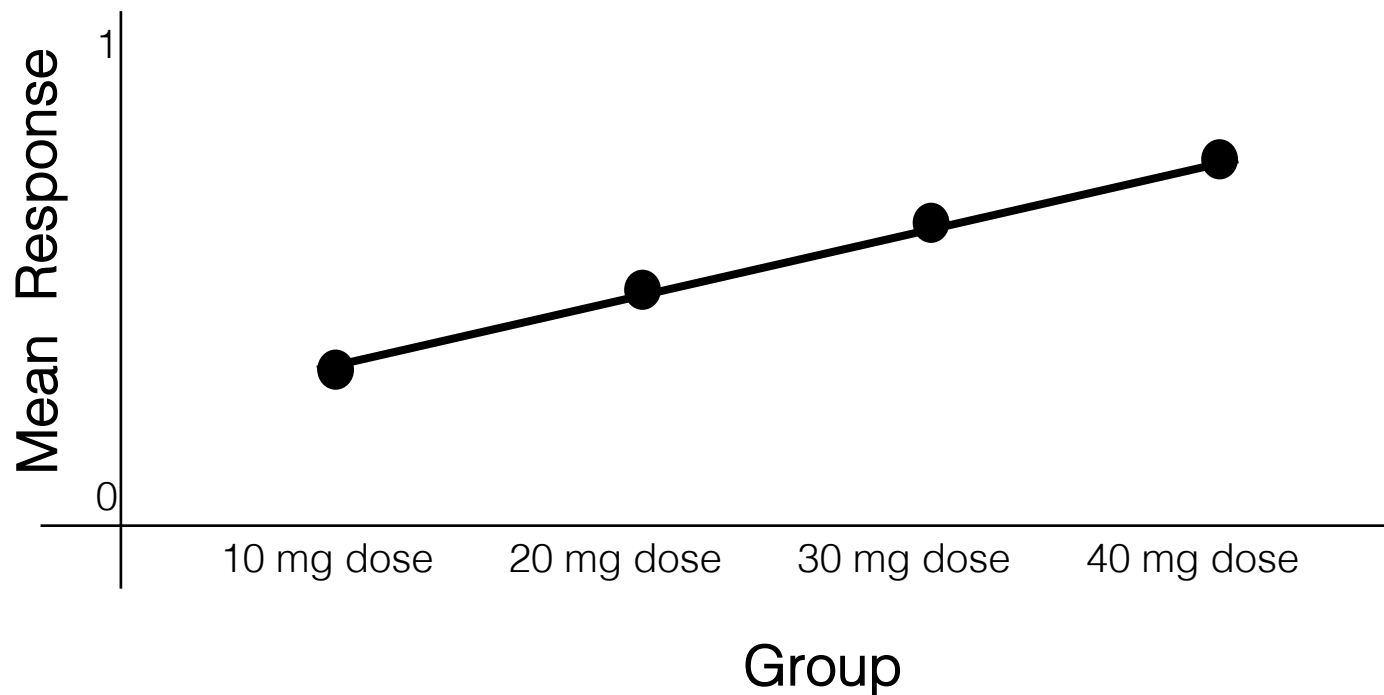
New in trend tests

- Cochran–Armitage test
- Linear-by-linear test
- Jonckheere–Terpstra test

Nonparametric tests for trend



Nonparametric tests for trend



Response types

- Cochran–Armitage test: trend in mean response
 - Binary
- Linear-by-linear test: trend in mean response
 - Ordinal
 - Continuous
- Cuzick's test with ranks: trend in mean rank response
 - Ordinal
 - Continuous
- Jonckheere–Terpstra test: association in relative ordering
 - Binary
 - Ordinal
 - Continuous

Cochran-Armitage test

```
. nptrend relief, group(dose) carmitage
```

Group	Group score	Mean response score	Number of obs
dose			
10	10	.6	200
20	20	.54	200
30	30	.585	200
40	40	.685	200

Statistic = .003

Std. err. = .0015476

z = 1.939

Prob > |z| = 0.0526

Test of departure from trend:

chi2(2) = 5.45

Prob > chi2 = 0.0656

Exact p -values

```
. nptrend relief, group(dose) carmitage notable exact(, rseed(1234))
```

Cochran–Armitage test for trend

```
Number of observations =      800
      Number of groups =        4
Number of response levels =      2

      Statistic =      .03
      Std. err. = .0154756
           z =      1.939
      Prob > |z| =      0.0526
      Exact prob =      0.0638 (10,000 Monte Carlo permutations)
```


Test of departure from trend:


```
      chi2(2) =      5.45
      Prob > chi2 =      0.0656
```


New in meta-analysis


- Galbraith plots
- Leave-one-out meta-analysis
- Multivariate meta-analysis


Meta-analysis

1  $\hat{\theta} = 0.43$


2  $\hat{\theta} = 0.51$

3  $\hat{\theta} = 0.59$

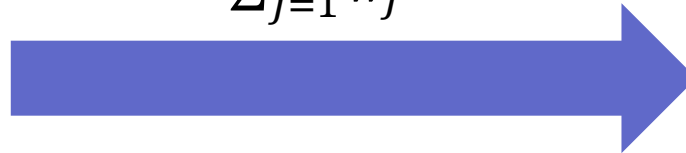
4  $\hat{\theta} = 0.45$

5  $\hat{\theta} = 0.49$

⋮

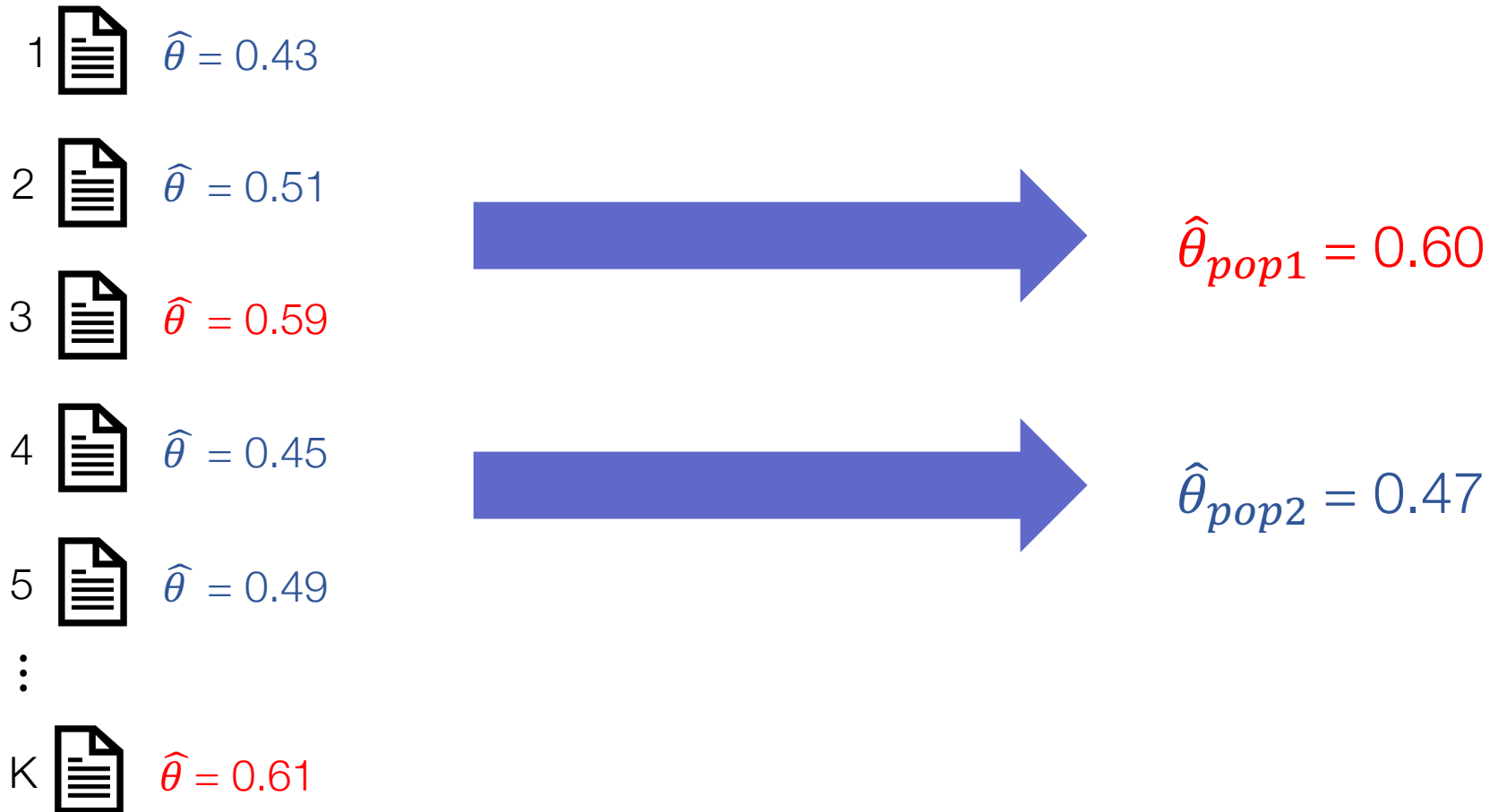
K  $\hat{\theta} = 0.61$

$$\frac{\sum_{j=1}^K w_j \hat{\theta}_j}{\sum_{j=1}^K w_j}$$



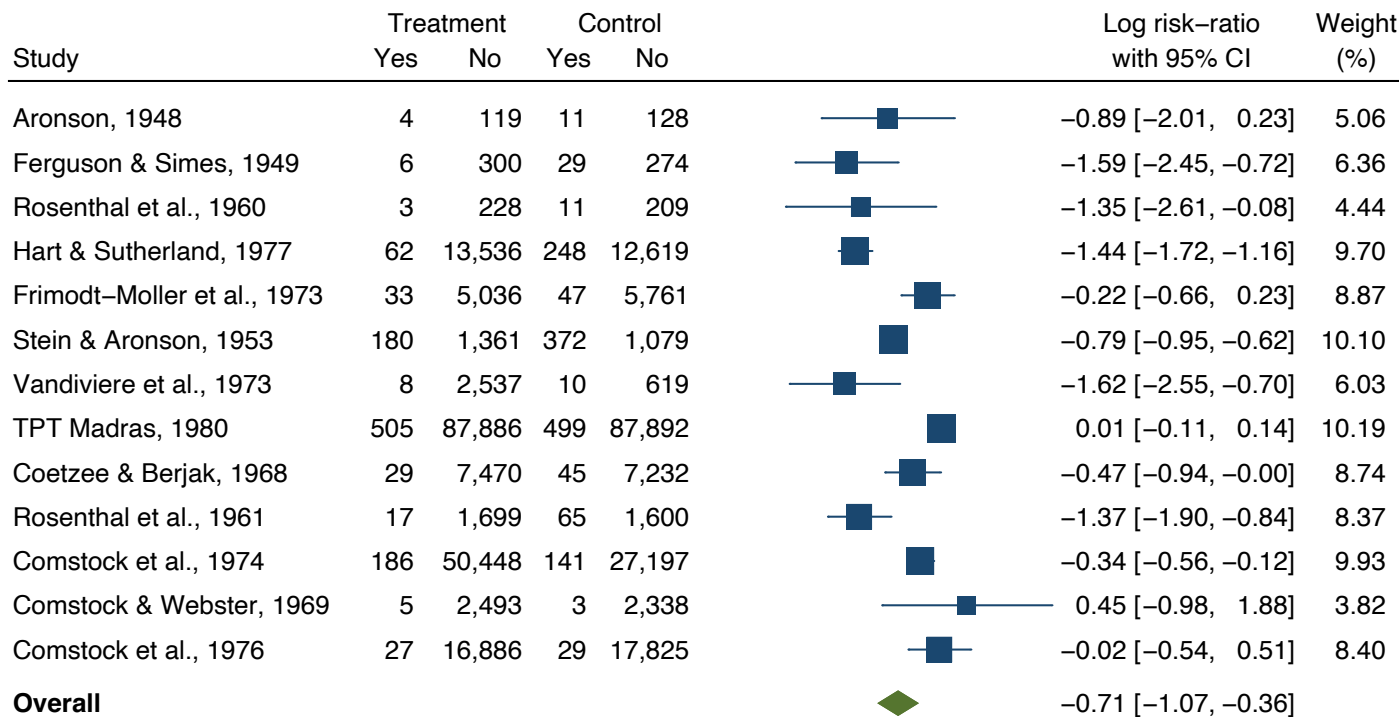
$$\hat{\theta}_{pop} = 0.52$$

Meta-analysis



Meta-analysis in Stata

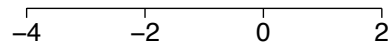
`. meta forestplot`



Heterogeneity: $\tau^2 = 0.31$, $I^2 = 92.22\%$, $H^2 = 12.86$

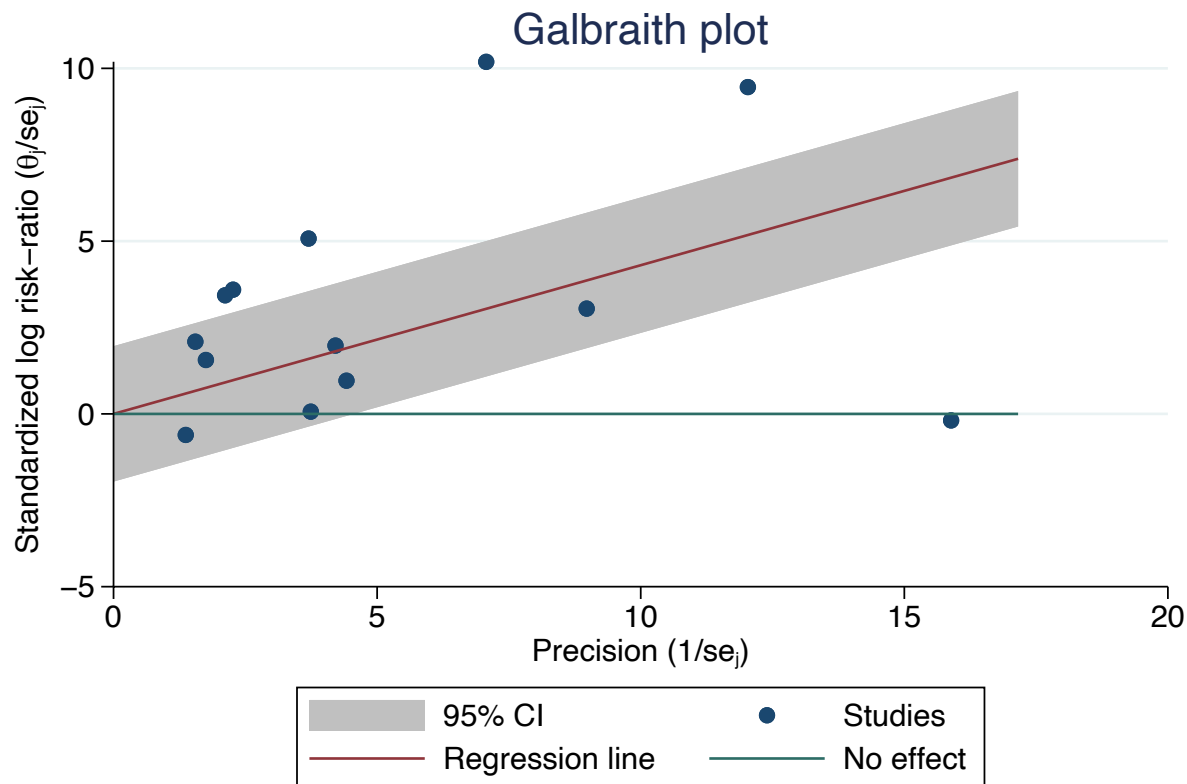
Test of $\theta_i = \theta_j$: $Q(12) = 152.23$, $p = 0.00$

Test of $\theta = 0$: $z = -3.97$, $p = 0.00$



Galbraith plots

```
. meta galbraithplot
```

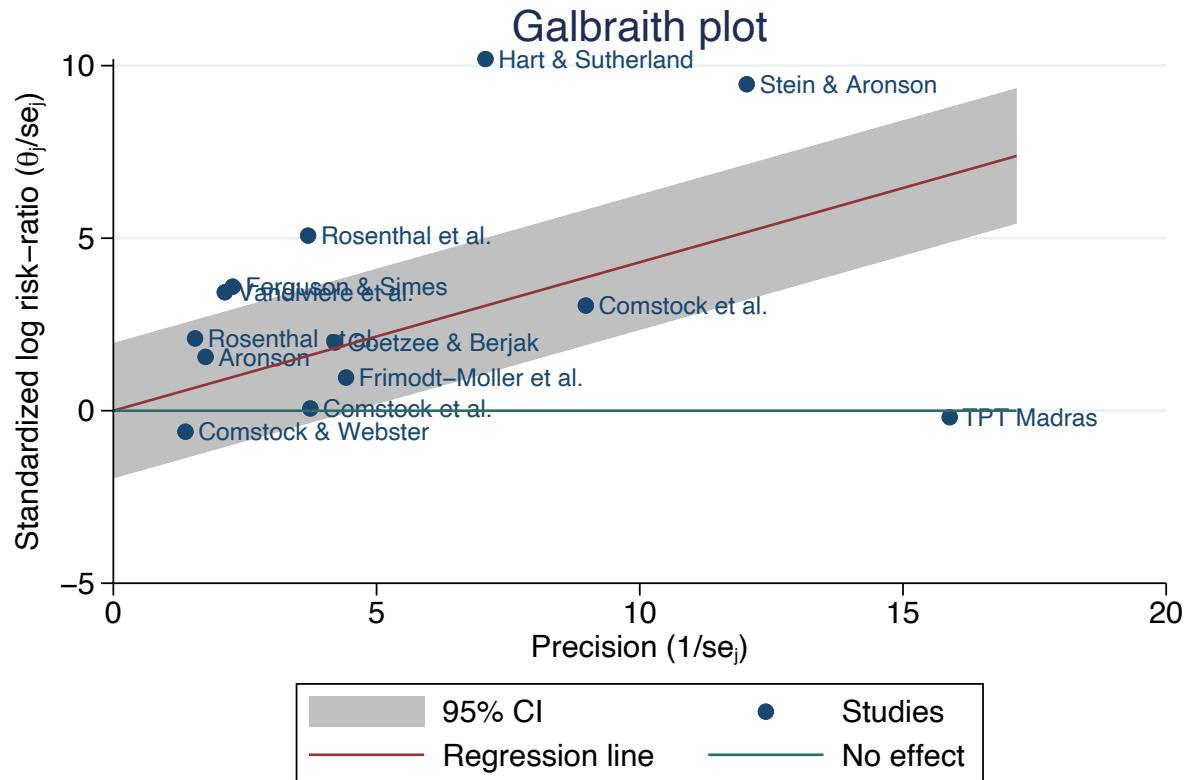


$$\hat{\theta} = 0.430$$

se_i : estimated σ_i

Galbraith plots

```
. meta galbraithplot, mlabel(author)
```

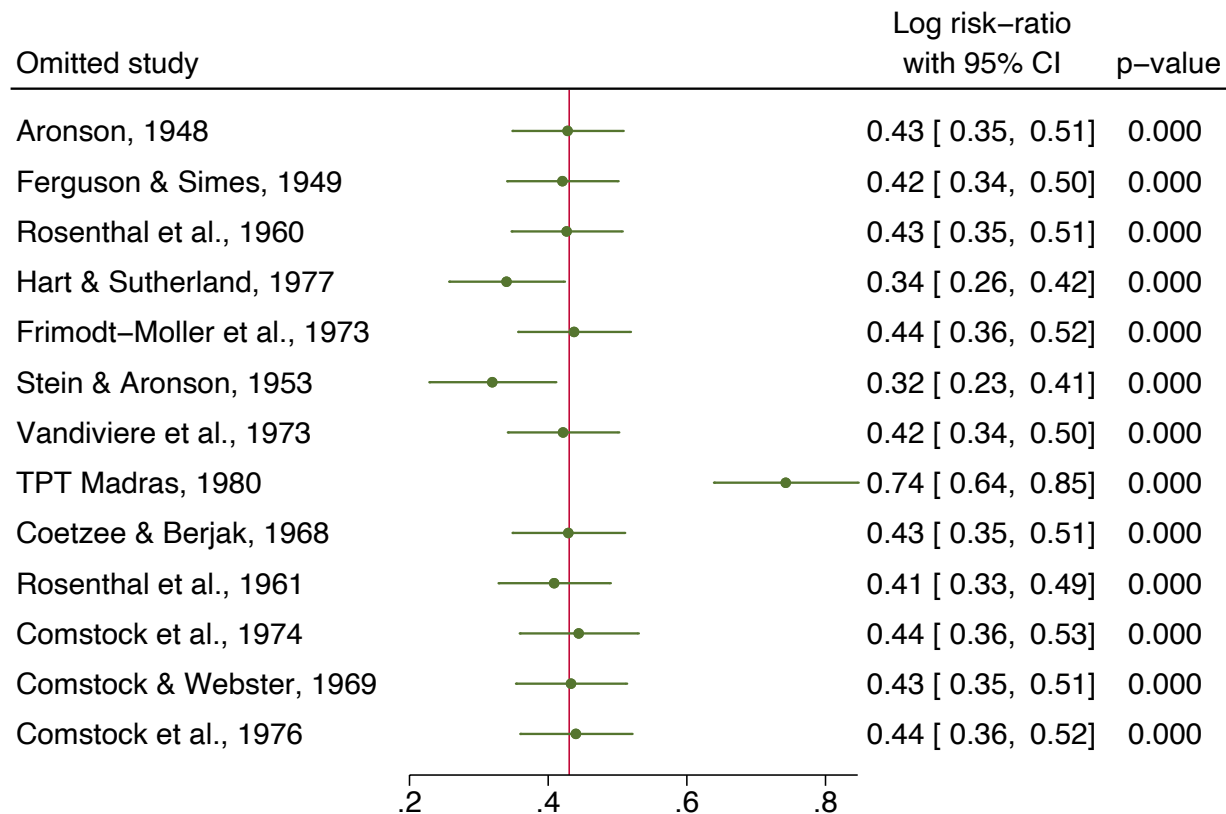


$$\hat{\theta} = 0.430$$

se_i : estimated σ_i

Leave-one-out

. meta forestplot, leaveoneout common(iv)



Common-effect inverse-variance model

Multivariate meta-analysis

Multivariate meta-analysis models the effects jointly and accounts for their dependence.

$$1 \quad \begin{array}{l} \text{document icon} \\ \swarrow \searrow \end{array} \begin{array}{l} \widehat{\theta}_{y1} = 0.43 \\ \widehat{\theta}_{y2} = 0.21 \end{array}$$

$$2 \quad \begin{array}{l} \text{document icon} \\ \swarrow \searrow \end{array} \begin{array}{l} \widehat{\theta}_{y1} = 0.51 \\ \widehat{\theta}_{y2} = 0.18 \end{array}$$

$$3 \quad \begin{array}{l} \text{document icon} \\ \swarrow \searrow \end{array} \begin{array}{l} \widehat{\theta}_{y1} = 0.47 \\ \widehat{\theta}_{y2} = 0.19 \end{array}$$

MV meta-regression in Stata

```
. meta mvregress y1 y2, wcovvariables(v11 v12 v22)
```

```

Multivariate random-effects meta-analysis      Number of obs      =          10
Method: REML                                  Number of studies   =           5
                                              Obs per study:
                                              min =              2
                                              avg =             2.0
                                              max =              2
                                              Wald chi2(0)       =           .
Log restricted-likelihood = 2.0823276          Prob > chi2         =           .
  
```

	Coefficient	Std. err.	z	P> z	[95% conf. interval]	
y1						
_cons	.3534282	.0588486	6.01	0.000	.238087	.4687694
y2						
_cons	-.3392152	.0879051	-3.86	0.000	-.5115061	-.1669243

Test of homogeneity: Q_M = chi2(8) = 128.23

Prob > Q_M = 0.0000

Heterogeneity

```
. estat heterogeneity
```

Method: Cochran

Joint:

I2 (%) = **93.76**

H2 = **16.03**

Method: Jackson–White–Riley

y1:

I2 (%) = **76.42**

R = **2.06**

y2:

I2 (%) = **95.50**

R = **4.71**

Joint:

I2 (%) = **88.66**

R = **2.97**

MV meta-regression in Stata

```
. meta mvregress y1 y2 = pubyear, wcovvariables(v11 v12 v22)
```

```
Multivariate random-effects meta-regression      Number of obs      =          10
Method: REML                                     Number of studies   =           5
                                                Obs per study:
                                                min =              2
                                                avg =             2.0
                                                max =              2
                                                Wald chi2(2)       =          0.40
Log restricted-likelihood = -3.5399567            Prob > chi2         =          0.8197
```

		Coefficient	Std. err.	z	P> z	[95% conf. interval]	
y1							
	pubyear	.0048615	.0218511	0.22	0.824	-.0379658	.0476888
	_cons	.3587569	.07345	4.88	0.000	.2147975	.5027163
y2							
	pubyear	-.0115367	.0299635	-0.39	0.700	-.070264	.0471907
	_cons	-.3357368	.0979979	-3.43	0.001	-.5278091	-.1436645

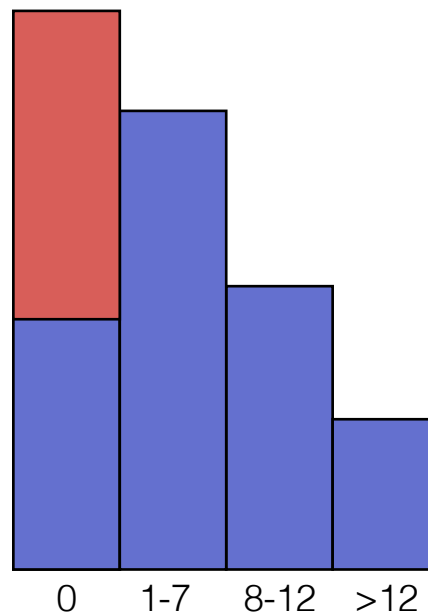
Zero-inflated ordered logit model

Within the past month, how many cigarettes have you smoked each day on average?

I would
never
smoke!



I didn't
smoke
this month



Zero-inflated ordered logit model

```
. ziologit tobacco income i.female, inflate(income i.parent)
```

Zero-inflated ordered logit regression

Number of obs = **15,000**

Wald chi2(2) = **967.16**

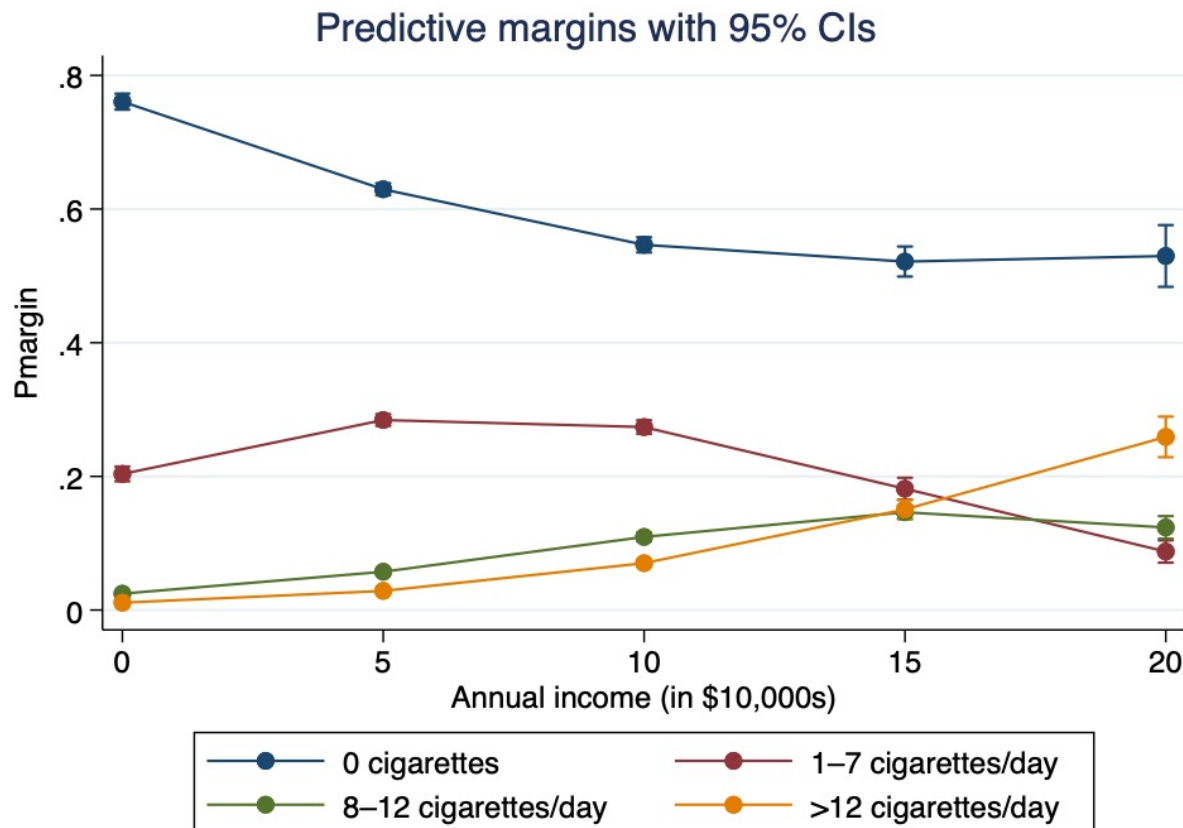
Log likelihood = **-13727.703**

Prob > chi2 = **0.0000**

tobacco	Coefficient	Std. err.	z	P> z	[95% conf. interval]	
tobacco						
income	.2078814	.007118	29.21	0.000	.1939304	.2218324
female						
Female	-.4918137	.0521513	-9.43	0.000	-.5940285	-.389599
inflate						
income	-.0251294	.0132	-1.90	0.057	-.0510008	.0007421
parent						
Smoking	.9268347	.0661451	14.01	0.000	.7971928	1.056477
_cons	-.0482399	.1608965	-0.30	0.764	-.3635912	.2671113

Zero-inflated ordered logit model

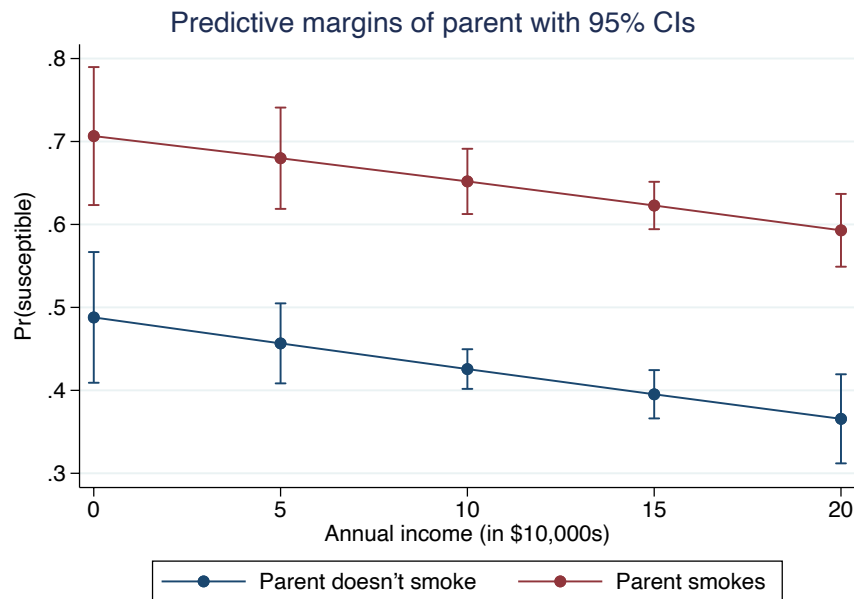
```
. margins, at(income=(0(5)20))  
. marginsplot
```



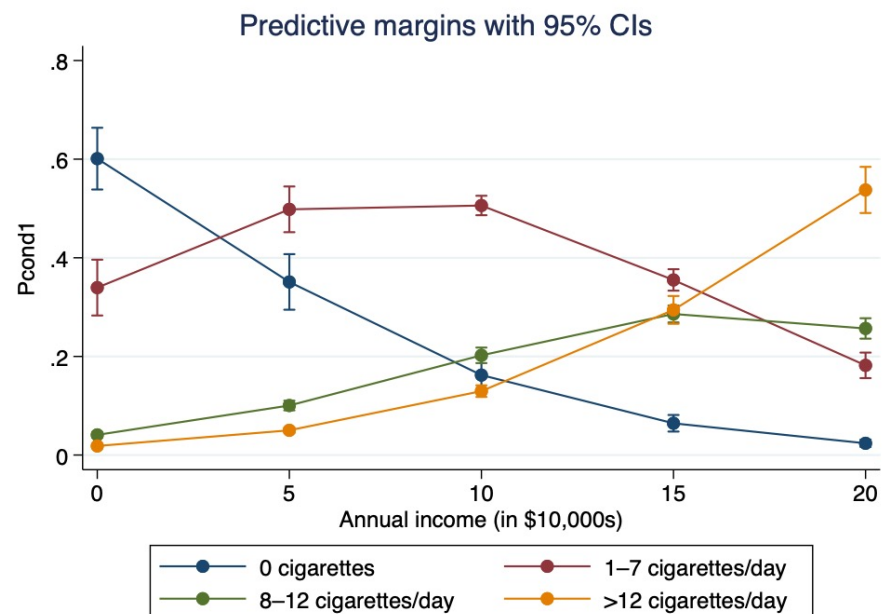
Zero-inflated ordered logit model

```
. margins parent, predict(ps)
> at(income=(0(5)20))
```

```
. marginsplot
```



```
. margins, predict(pcond1 outcome(0))
> predict(pcond1 outcome(1))
> predict(pcond1 outcome(2))
> predict(pcond1 outcome(3)) at(income=(0(5)20))
. marginsplot
```



Customizable tables

- Create, customize, and export tables with `table` and the `collect` suite

	Logistic Regression Model for Hypertension				
	Odds Ratio	Std. error	z	p-value	95% CI
Age (years)	1.03	0.00	18.78	0.0000	[1.03, 1.04]
Sex					
Female	0.15	0.02	-12.93	0.0000	[0.12, 0.21]
Sex x Age (years)					
Female	1.03	0.00	10.47	0.0000	[1.02, 1.03]
Diabetes status					
Diabetic	1.52	0.15	4.14	0.0000	[1.25, 1.86]
Intercept	0.17	0.02	-19.24	0.0000	[0.14, 0.21]

table: Reimagined

```
. table (smsa) (collgrad)
```

	College graduate		
	Not college grad	College grad	Total
Lives in SMSA			
Not SMSA	550	115	665
SMSA	1,164	417	1,581
Total	1,714	532	2,246

table: Reimagined

```
. table () (collgrad), statistic(fvpercent smsa south) style(table-1)
```

	College graduate		
	Not college grad	College grad	Total
Lives in SMSA			
Not SMSA	32.09	21.62	29.61
SMSA	67.91	78.38	70.39
Lives in the south			
Not south	57.76	59.02	58.06
South	42.24	40.98	41.94

table: Reimagined

```
. table () (collgrad), statistic(fvpercent smsa south) ///
> statistic(mean age wage) style(table-1)
```

	College graduate		
	Not college grad	College grad	Total
Lives in SMSA			
Not SMSA	32.09	21.62	29.61
SMSA	67.91	78.38	70.39
Lives in the south			
Not south	57.76	59.02	58.06
South	42.24	40.98	41.94
Age in current year	39.16569	39.11278	39.15316
Hourly wage	6.910561	10.52606	7.766949

table: Reimagined

```
. table () (collgrad), statistic(fvpercent smsa south) ///
> statistic(mean age wage) style(table-1) nformat(%5.2f)
```

	College graduate		
	Not college grad	College grad	Total
Lives in SMSA			
Not SMSA	32.09	21.62	29.61
SMSA	67.91	78.38	70.39
Lives in the south			
Not south	57.76	59.02	58.06
South	42.24	40.98	41.94
Age in current year	39.17	39.11	39.15
Hourly wage	6.91	10.53	7.77

table: Reimagined

```
. table () (collgrad), command(regress wage age tenure i.union) ///  
> style(table-reg1-fv1) nformat(%5.2f)
```

	Not college grad	College grad	Total
Age in current year	-0.01	-0.02	-0.02
Job tenure (years)	0.19	0.11	0.19
Nonunion	0.00	0.00	0.00
Union	1.12	0.19	1.13
Intercept	5.60	10.31	6.97

table: Reimagined

```
. table () (collgrad), command(_r_b _r_se: regress wage tenure i.union) ///
> style(table-reg1-fv1) nformat(%5.2f)
```

	Not college grad	College grad	Total
Coefficient			
Job tenure (years)	0.19	0.10	0.19
Nonunion	0.00	0.00	0.00
Union	1.12	0.19	1.13
Intercept	5.27	9.41	6.09
Std. error			
Job tenure (years)	0.02	0.04	0.02
Nonunion	0.00	0.00	0.00
Union	0.21	0.48	0.22
Intercept	0.14	0.38	0.15

table: Reimagined

```
. table (colname result) (collgrad), command(_r_b _r_se: ///  
> regress wage tenure i.union) style(table-reg1-fv1) nformat(%5.2f)
```

	Not college grad	College grad	Total
Job tenure (years)			
Coefficient	0.19	0.10	0.19
Std. error	0.02	0.04	0.02
Nonunion			
Coefficient	0.00	0.00	0.00
Std. error	0.00	0.00	0.00
Union			
Coefficient	1.12	0.19	1.13
Std. error	0.21	0.48	0.22
Intercept			
Coefficient	5.27	9.41	6.09
Std. error	0.14	0.38	0.15

The new collect suite

Collect results from Stata commands

```
collect:
```

```
collect get
```

Create tables by specifying rows and columns

```
collect layout
```

Customize labels, formats, borders, and more

```
collect label
```

```
collect style
```

Export to Word, PDF, HTML, L^AT_EX, and more

```
collect export
```

Use collect to customize

```
. collect style cell cell_type[column-header item], halign(center)
. collect style cell result[_r_se], sformat("(%.s)")
. collect style header result, level(hide)
. collect style showbase off
. collect style row stack, spacer
. collect style cell border_block, border(right left, pattern(nil))
. collect preview
```

	Not college grad	College grad	Total
<hr/>			
Job tenure (years)	0.19 (0.02)	0.10 (0.04)	0.19 (0.02)
Union	1.12 (0.21)	0.19 (0.48)	1.13 (0.22)
Intercept	5.27 (0.14)	9.41 (0.38)	6.09 (0.15)

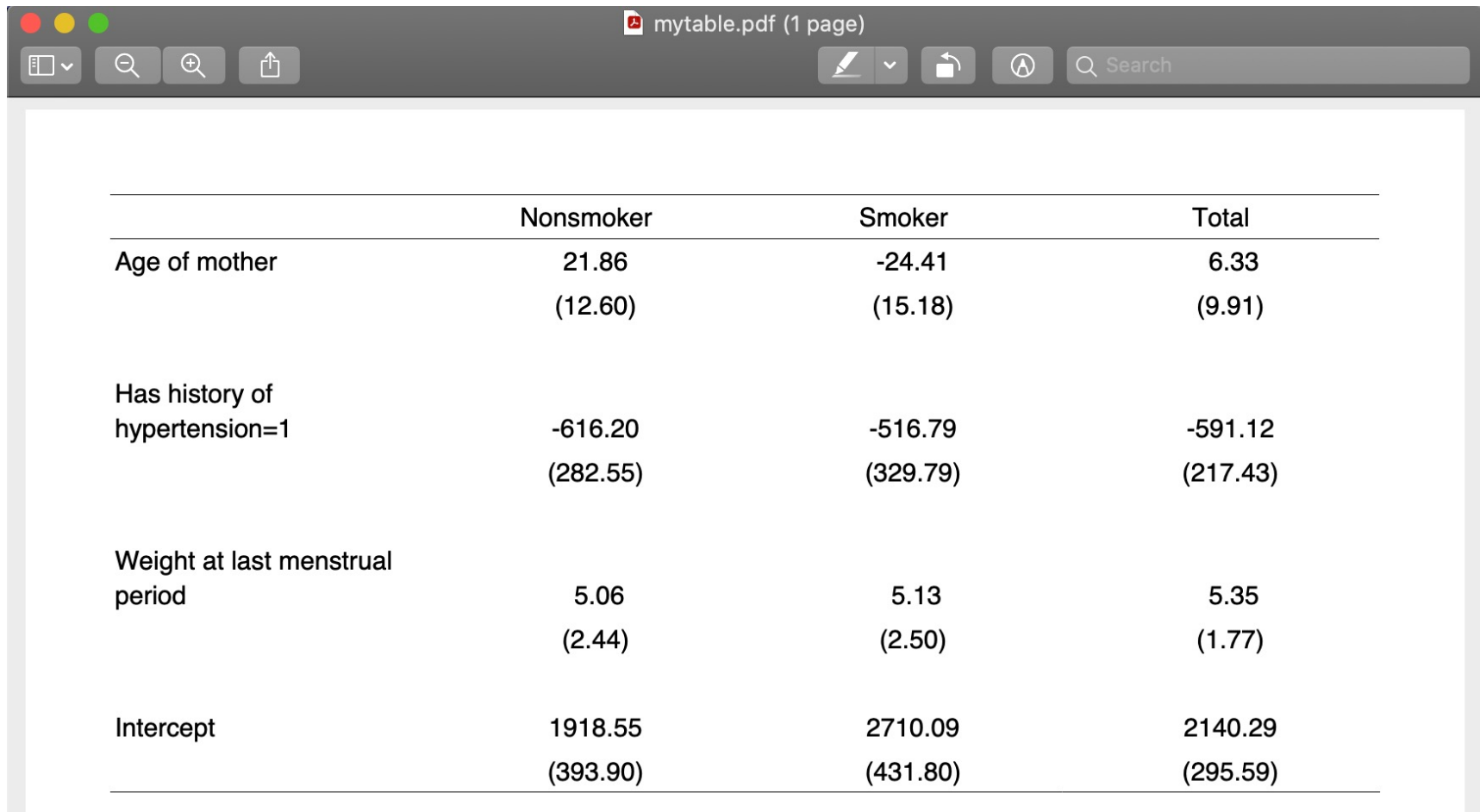
Save customized styles

```
. collect style save mystyle, replace
. webuse lbw, clear
. table (colname result) (smoke), ///
> command(_r_b _r_se: regress bwt age i.ht lwt) style(mystyle, override)
```

	Nonsmoker	Smoker	Total
Age of mother	21.86 (12.60)	-24.41 (15.18)	6.33 (9.91)
Has history of hypertension=1	-616.20 (282.55)	-516.79 (329.79)	-591.12 (217.43)
Weight at last menstrual period	5.06 (2.44)	5.13 (2.50)	5.35 (1.77)
Intercept	1918.55 (393.90)	2710.09 (431.80)	2140.29 (295.59)

Export customized tables

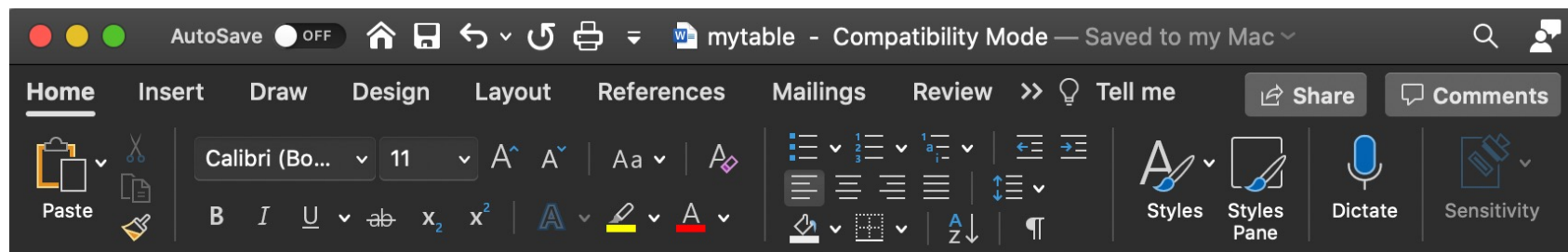
```
. collect export mytable.pdf
```



	Nonsmoker	Smoker	Total
Age of mother	21.86 (12.60)	-24.41 (15.18)	6.33 (9.91)
Has history of hypertension=1	-616.20 (282.55)	-516.79 (329.79)	-591.12 (217.43)
Weight at last menstrual period	5.06 (2.44)	5.13 (2.50)	5.35 (1.77)
Intercept	1918.55 (393.90)	2710.09 (431.80)	2140.29 (295.59)

Export customized tables

```
. collect export mytable.docx
```



	Nonsmoker	Smoker	Total
Age of mother	21.86 (12.60)	-24.41 (15.18)	6.33 (9.91)
Has history of hypertension=1	-616.20 (282.55)	-516.79 (329.79)	-591.12 (217.43)
Weight at last menstrual period	5.06 (2.44)	5.13 (2.50)	5.35 (1.77)
Intercept	1918.55 (393.90)	2710.09 (431.80)	2140.29 (295.59)

Daily reports

```
. table () (Year), statistic(sum profit) nototals name(c1)
. table () (Year) if Quarter==quarter(today()), ///
> statistic(sum profit) nototals name(c2)
. table () (Year) if month(Date)==month(today()) & day(Date)==day(today()), ///
> statistic(sum profit) nototals name(c3)
. collect combine newc = c1 c2 c3
. collect style use profit, replace
. collect label use profitlab, replace
. collect layout (collection) (Year)
```

	2020	2021
This year	\$116,109.48	\$123,361.74
This quarter	\$15,731.32	\$17,969.33
This day	\$3,898.89	\$4,892.35

12 new functions for dates/times

Clockdiff()

clockdiff()

Clockdiff_frac()

clockdiff_frac()

today()

now()

daysinmonth()

firstdayofmonth()

lastdayofmonth()

isleapsecond()

Clockpart()

clockpart()

datepart()

23 new functions in Mata

Clockdiff()

clockdiff()

Clockdiff_frac()

clockdiff_frac ()

today()

now()

daysinmonth()

firstdayofmonth()

lastdayofmonth()

Clockpart()

clockpart()

datepart()

age()

age_frac()

birthday()

previousbirthday()

nextbirthday()

datediff()

datediff_frac()

isleapyear()

previousleapyear()

nextleapyear()

isleapsecond()

Do-file Editor enhancements

- Syntax highlighting for Java and XML
- Autocompletion of quotes, parentheses, and brackets
- Bookmarks and Navigation Control!

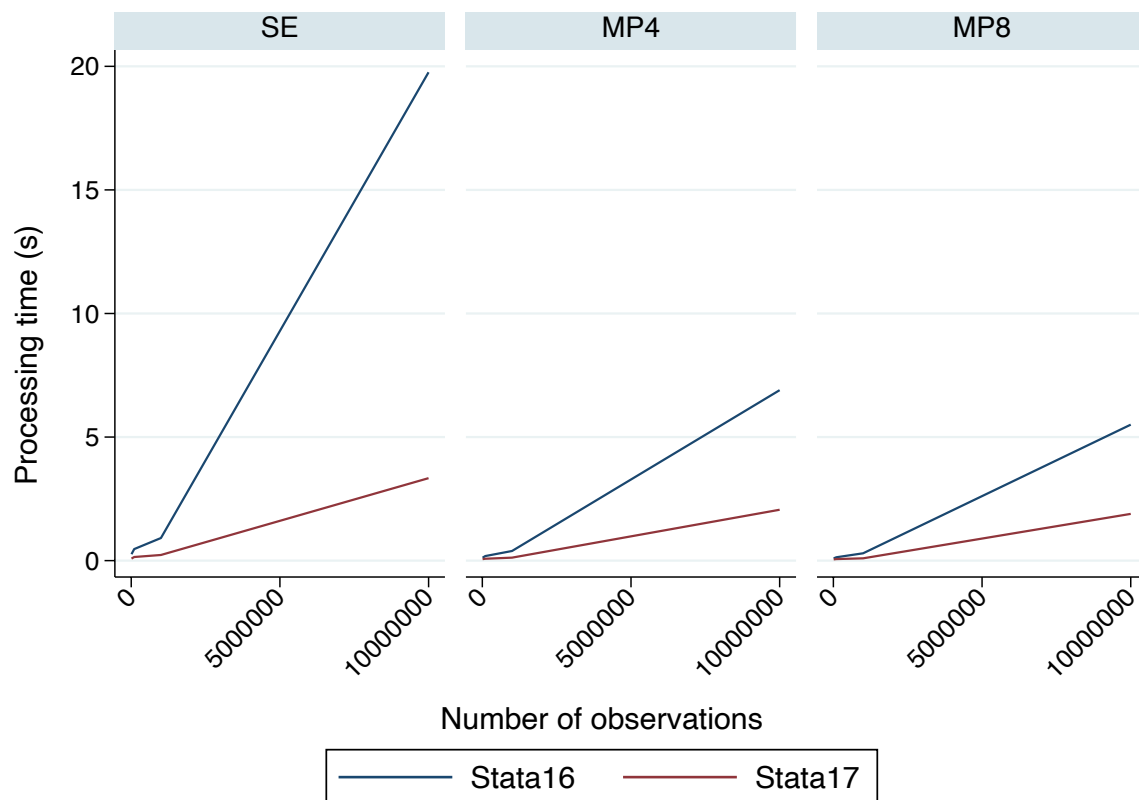
<https://www.youtube.com/watch?v=GhYZuKRgo7E>

Stata 17 is faster!

- `sort` is 1.5 to 6 times faster
- `collapse` is 6 to 13 times faster for mean computation and 40 to 70 times faster for computation of statistics
- `import delimited` is up to 4 times faster
- `mixed` is 2 to 3 times faster
- The Linear Algebra Package (LAPACK) underlying many of Stata's functions and operators is now powered by Intel Math Kernel Library (MKL)

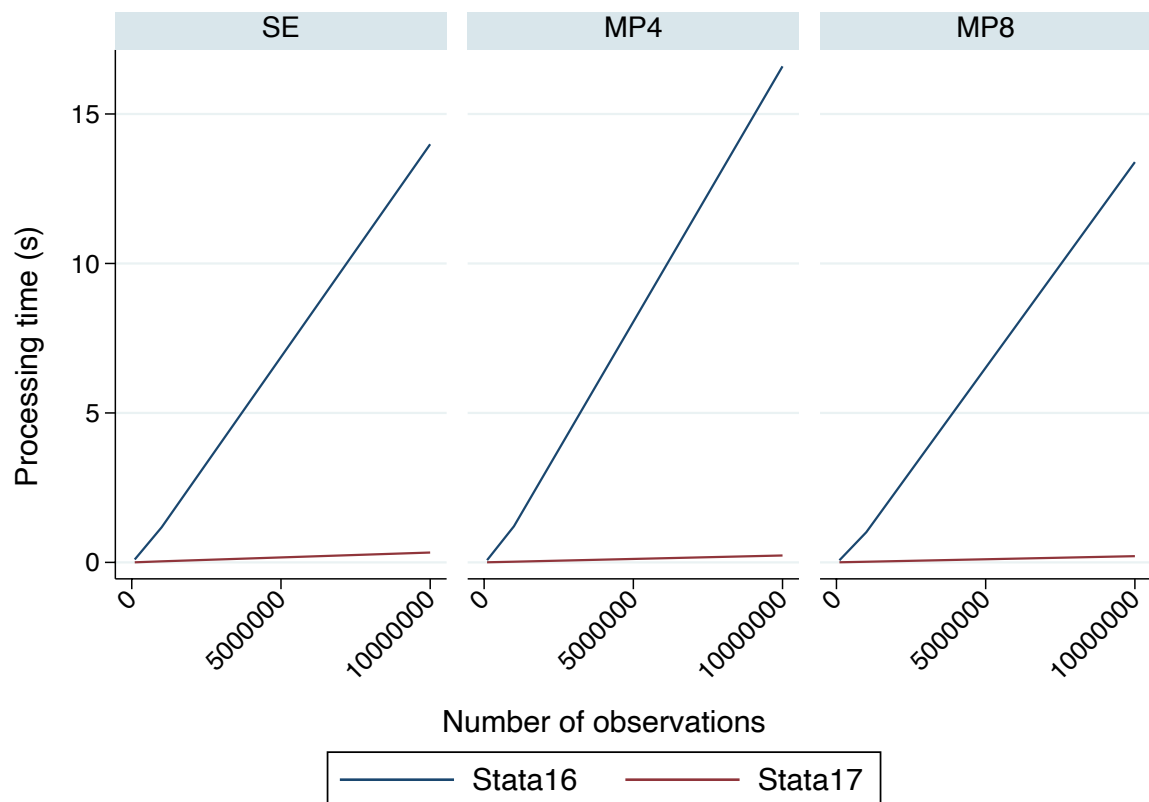
Speed improvements

sort



Speed improvements

`collapse`



Speed improvements

mixed

In Stata 16:

```
. set rmsg on

. mixed lnwage i.(married union race) educ exper || nr: exper union, cov(un) reml
r; t=8.00 14:54:44

. mixed attain c.vrq##c.sex || _all: R.pid || sid: vrq, reml
r; t=37.81 14:55:22
```

In Stata 17:

```
. set rmsg on

. mixed lnwage i.(married union race) educ exper || nr: exper union, cov(un) reml
r; t=2.40 14:52:17

. mixed attain c.vrq##c.sex || _all: R.pid || sid: vrq, reml
r; t=8.73 14:50:43
```

PyStata

PyStata

Graphical diagnostics for parallel trends

Observed means

Linear-trends model

Control Treatment

Help Plots Files

Console 1/A

STATA 17.0
MP-Parallel Edition

Statistics and Data Science

Copyright 1985-2021 StataCorp LLC
StataCorp
4905 Lakeway Drive
College Station, Texas 77845 USA
800-STATA-PC https://www.stata.com
979-696-4600 stata@stata.com

Stata license: Unlimited-user 4-core network perpetual
Serial number: 17
Licensed to: Meghan Cain
StataCorp

Notes:

1. Unicode is supported; see help unicode_advice.
2. More than 2 billion observations are allowed; see help obs_advice.
3. Maximum number of variables is set to 5,000; see help set_maxvar.

Python console History

custom (Python 3.9.4) LSP Python: ready Line 1, Col 1 ASCII CRLF RW Mem 64%

```

1 # Setup Stata
2 import stata_setup
3 stata_setup.config("/Applications/Stata17", "mp")
4 from pystata import stata
5
6 # Import other modules
7 from pandas.io.json import json_normalize
8 import json
9
10 # Import json data
11 with open("did.json") as json_file:
12     data = json.load(json_file)
13 data = json_normalize(data, 'records', ['hospital_id', 'month'])
14
15 # Load data to Stata
16 stata.pdataframe_to_data(data, True)
17
18 # Run block of Stata code
19 stata.run('''
20     destring satisfaction_score, replace
21     destring hospital_id, replace
22     destring month, replace
23
24     gen proc = 0
25     replace proc = 1 if procedure == "New"
26     label define procedure 0 "Old" 1 "New"
27     drop procedure
28     rename proc procedure
29     label value procedure procedure
30     ''', quietly=True)
31
32 stata.run('''
33     didregress (satisfaction_score) (procedure), group(hospital_id) time(month)
34     ''', echo=True)
35
36 # Load Stata results to Python
37 r = stata.get_return()['r(table)']
38
39 # Use Stata results in Python
40 print("The treatment hospitals had a %.2f-point increase."
41       % (r[0][0]), end=" ")
42 print("The result is with 95% confidence interval [%4.2f, %4.2f]."
43       % (r[4][0], r[5][0]))
44
45 # Generate Stata graph

```

PyStata

Documents — Python — 239x63

```

STATA®
Statistics and Data Science

17.0
MP-Parallel Edition

Copyright 1985-2021 StataCorp LLC
StataCorp
4905 Lakeway Drive
College Station, Texas 77845 USA
800-STATA-PC      https://www.stata.com
979-696-4600      stata@stata.com

Stata license: Unlimited-user 4-core network perpetual
Serial number: 17
Licensed to: Meghan Cain
             StataCorp

Notes:
1. Unicode is supported; see help unicode_advice.
2. More than 2 billion observations are allowed; see help obs_advice.
3. Maximum number of variables is set to 5,000; see help set_maxvar.

from pystata import stata
>>> from pystata import stata
>>> from pandas.io.json import json_normalize
>>> import json
>>> with open("did.json") as json_file:
...     data = json.load(json_file)
...
>>> data = json_normalize(data, 'records', ['hospital_id', 'month'])
<stdin>:1: FutureWarning: pandas.io.json.json_normalize is deprecated, use pandas.json_normalize instead
>>> stata.pdataframe_to_data(data, True)
>>> stata.run('
... destring satisfaction_score, replace
... destring hospital_id, replace
... destring month, replace
...
... gen proc = 0
... replace proc = 1 if procedure == "New"
... label define procedure 0 "Old" 1 "New"
... drop procedure
... rename proc procedure
... label value procedure procedure
... ', quietly=True)

>>>
>>> stata.run('
... didregress (satisfaction_score) (procedure), group(hospital_id) time(month)
... ', echo=True)
.
. didregress (satisfaction_score) (procedure), group(hospital_id) time(month)

Number of groups and treatment time

Time variable: month
Control:      procedure = 0
Treatment:    procedure = 1
-----
|      Control      Treatment
-----+-----
Group
hospital_id |      28      18
-----+-----
Time

```

Jupyter Notebook

<https://github.com/StataMeghan/JupyterNotebook/blob/main/Difference-in-differences.ipynb>

Interactions with other languages, software, and operating systems

- Java integration: embed and execute Java code directly in Stata 17
- H2O integration: access the capabilities of H2O directly from Stata 17
- Connecting to databases using JDBC: exchange data with Oracle, MySQL, Amazon Redshift, Snowflake, Microsoft SQL Server, and much more
- Stata on Apple Silicon: Stata 17 will run natively on Macs with Apple Silicon and Macs with Intel processors

New in Stata 17

- Tables
- Bayesian econometrics
- Interval-censored Cox model
- Difference in differences (DID)
- Bayesian VAR
- Multivariate meta-analysis
- Treatment-effects lasso
- Panel-data multinomial logit
- Zero-inflated ordered logit
- Bayesian IRF and FEVD analysis
- Bayesian dynamic forecasting
- Do-file Editor enhancements
- Intel Math Kernel Library (MKL)
- Stata on Apple Silicon
- PyStata
- Jupyter Notebook with Stata
- Faster Stata
- Bayesian multilevel modeling
- New functions for dates and times
- Leave-one-out meta-analysis
- Galbraith plots
- Bayesian panel-data models
- Nonparametric tests for trend
- Lasso with clustered data
- BIC for lasso penalty selection
- Bayesian linear and nonlinear DSGEs
- H2O integration
- Java integration
- JDBC

Learn more

Webinars

Customized Tables –
May 25

- PyStata/Jupyter Notebook
- Interval-censored Cox model
- DID
- Meta-analysis

Trainings

Causal inference –
June 22-25

Customized reports –
August 10-13

Thank you!

Questions?

You can download the dataset, do-file, and slides here:

<https://tinyurl.com/Stata 17>

You can contact tech support at tech-support@stata.com