Title

regress postestimation — Postestimation tools for regress

Description

The following postestimation commands are of special interest after `regress`:

command	description
`dfbeta`	DFBETA influence statistics
`estat hettest`	tests for heteroskedasticity
`estat imtest`	information matrix test
`estat ovtest`	Ramsey regression specification-error test for omitted variables
`estat szroeter`	Szroeter's rank test for heteroskedasticity
`estat vif`	variance inflation factors for the independent variables
`acprplot`	augmented component-plus-residual plot
`avplot`	added-variable plot
`avplots`	all added-variables plots in one image
`cprplot`	component-plus-residual plot
`lvr2plot`	leverage-versus-squared-residual plot
`rvfplot`	residual-versus-fitted plot
`rvpplot`	residual-versus-predictor plot

These commands are not appropriate after the `svy` prefix.

For information about these commands, see below.

The following standard postestimation commands are also available:

command	description
`estat`	AIC, BIC, VCE, and estimation sample summary
`estat` (svy)	postestimation statistics for survey data
`estimates`	cataloging estimation results
`hausman`	Hausman's specification test
`lincom`	point estimates, standard errors, testing, and inference for linear combinations of coefficients
`linktest`	link test for model specification
`lrtest`[1]	likelihood-ratio test
`margins`	marginal means, predictive margins, marginal effects, and average marginal effects
`nlcom`	point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients
`predict`	predictions, residuals, influence statistics, and other diagnostic measures
`predictnl`	point estimates, standard errors, testing, and inference for generalized predictions
`suest`	seemingly unrelated estimation
`test`	Wald tests of simple and composite linear hypotheses
`testnl`	Wald tests of nonlinear hypotheses

[1] `lrtest` is not appropriate with `svy` estimation results.

See the corresponding entries in the *Base Reference Manual* for details, but see [SVY] **estat** for details about `estat` (svy).

For postestimation tests specific to time series, see [R] **regress postestimation time series**.

Special-interest postestimation commands

These commands provide tools for diagnosing sensitivity to individual observations, analyzing residuals, and assessing specification.

`dfbeta` will calculate one, more than one, or all the DFBETAs after `regress`. Although `predict` will also calculate DFBETAs, `predict` can do this for only one variable at a time. `dfbeta` is a convenience tool for those who want to calculate DFBETAs for multiple variables. The names for the new variables created are chosen automatically and begin with the letters `_dfbeta_`.

`estat hettest` performs three versions of the Breusch–Pagan (1979) and Cook–Weisberg (1983) test for heteroskedasticity. All three versions of this test present evidence against the null hypothesis that $\mathbf{t} = \mathbf{0}$ in $\mathrm{Var}(e) = \sigma^2 \exp(\mathbf{zt})$. In the `normal` version, performed by default, the null hypothesis also includes the assumption that the regression disturbances are independent-normal draws with variance σ^2. The normality assumption is dropped from the null hypothesis in the `iid` and `fstat` versions, which respectively produce the score and F tests discussed in *Methods and formulas*. If *varlist* is not specified, the fitted values are used for $\mathbf{z}$. If *varlist* or the `rhs` option is specified, the variables specified are used for $\mathbf{z}$.

`estat imtest` performs an information matrix test for the regression model and an orthogonal decomposition into tests for heteroskedasticity, skewness, and kurtosis due to Cameron and Trivedi (1990); White's test for homoskedasticity against unrestricted forms of heteroskedasticity (1980) is available as an option. White's test is usually similar to the first term of the Cameron–Trivedi decomposition.

`estat ovtest` performs two versions of the Ramsey (1969) regression specification-error test (RESET) for omitted variables. This test amounts to fitting $y = \mathbf{xb} + \mathbf{zt} + u$ and then testing $\mathbf{t} = \mathbf{0}$. If the `rhs` option is not specified, powers of the fitted values are used for $\mathbf{z}$. If `rhs` is specified, powers of the individual elements of $\mathbf{x}$ are used.

`estat szroeter` performs Szroeter's rank test for heteroskedasticity for each of the variables in *varlist* or for the explanatory variables of the regression if `rhs` is specified.

`estat vif` calculates the centered or uncentered variance inflation factors (VIFs) for the independent variables specified in a linear regression model.

`acprplot` graphs an augmented component-plus-residual plot (a.k.a. augmented partial residual plot) as described by Mallows (1986). This seems to work better than the component-plus-residual plot for identifying nonlinearities in the data.

`avplot` graphs an added-variable plot (a.k.a. partial-regression leverage plot, partial regression plot, or adjusted partial residual plot) after `regress`. *indepvar* may be an independent variable (a.k.a. predictor, carrier, or covariate) that is currently in the model or not.

`avplots` graphs all the added-variable plots in one image.

`cprplot` graphs a component-plus-residual plot (a.k.a. partial residual plot) after `regress`. *indepvar* must be an independent variable that is currently in the model.

`lvr2plot` graphs a leverage-versus-squared-residual plot (a.k.a. L-R plot).

`rvfplot` graphs a residual-versus-fitted plot, a graph of the residuals against the fitted values.

`rvpplot` graphs a residual-versus-predictor plot (a.k.a. independent variable plot or carrier plot), a graph of the residuals against the specified predictor.

(*Continued on next page*)

Syntax for predict

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

statistic	description
`xb`	linear prediction; the default
`residuals`	residuals
`score`	score; equivalent to `residuals`
`rstandard`	standardized residuals
`rstudent`	studentized (jackknifed) residuals
`cooksd`	Cook's distance
`leverage` | `hat`	leverage (diagonal elements of hat matrix)
`pr(`*a*,*b*`)`	$\Pr(y_j \mid a < y_j < b)$
`e(`*a*,*b*`)`	$E(y_j \mid a < y_j < b)$
`ystar(`*a*,*b*`)`	$E(y_j^*)$, $y_j^* = \max\{a, \min(y_j, b)\}$
* `dfbeta(`*varname*`)`	DFBETA for *varname*
`stdp`	standard error of the linear prediction
`stdf`	standard error of the forecast
`stdr`	standard error of the residual
* `covratio`	COVRATIO
* `dfits`	DFITS
* `welsch`	Welsch distance

Unstarred statistics are available both in and out of sample; type `predict ... if e(sample) ...` if wanted only for the estimation sample. Starred statistics are calculated only for the estimation sample, even when `if e(sample)` is not specified.

`rstandard`, `rstudent`, `cooksd`, `leverage`, `dfbeta()`, `stdf`, `stdr`, `covratio`, `dfits`, and `welsch` are not available if any `vce()` other than `vce(ols)` was specified with `regress`.

`xb`, `residuals`, `score`, and `stdp` are the only options allowed with `svy` estimation results.

where *a* and *b* may be numbers or variables; *a* missing ($a \geq .$) means $-\infty$, and *b* missing ($b \geq .$) means $+\infty$; see [U] **12.2.1 Missing values**.

Menu

Statistics > Postestimation > Predictions, residuals, etc.

Options for predict

`xb`, the default, calculates the linear prediction.

`residuals` calculates the residuals.

`score` is equivalent to `residuals` in linear regression.

`rstandard` calculates the standardized residuals.

`rstudent` calculates the studentized (jackknifed) residuals.

`cooksd` calculates the Cook's D influence statistic (Cook 1977).

`leverage` or `hat` calculates the diagonal elements of the projection hat matrix.

`pr(`*a*`,`*b*`)` calculates $\Pr(a < \mathbf{x}_j\mathbf{b} + u_j < b)$, the probability that $y_j|\mathbf{x}_j$ would be observed in the interval (a, b).

a and *b* may be specified as numbers or variable names; *lb* and *ub* are variable names;
`pr(20,30)` calculates $\Pr(20 < \mathbf{x}_j\mathbf{b} + u_j < 30)$;
`pr(`*lb*`,`*ub*`)` calculates $\Pr(lb < \mathbf{x}_j\mathbf{b} + u_j < ub)$;
and `pr(20,`*ub*`)` calculates $\Pr(20 < \mathbf{x}_j\mathbf{b} + u_j < ub)$.

a missing ($a \geq .$) means $-\infty$; `pr(.,30)` calculates $\Pr(-\infty < \mathbf{x}_j\mathbf{b} + u_j < 30)$;
`pr(`*lb*`,30)` calculates $\Pr(-\infty < \mathbf{x}_j\mathbf{b} + u_j < 30)$ in observations for which $lb \geq .$
and calculates $\Pr(lb < \mathbf{x}_j\mathbf{b} + u_j < 30)$ elsewhere.

b missing ($b \geq .$) means $+\infty$; `pr(20,.)` calculates $\Pr(+\infty > \mathbf{x}_j\mathbf{b} + u_j > 20)$;
`pr(20,`*ub*`)` calculates $\Pr(+\infty > \mathbf{x}_j\mathbf{b} + u_j > 20)$ in observations for which $ub \geq .$
and calculates $\Pr(20 < \mathbf{x}_j\mathbf{b} + u_j < ub)$ elsewhere.

`e(`*a*`,`*b*`)` calculates $E(\mathbf{x}_j\mathbf{b} + u_j \mid a < \mathbf{x}_j\mathbf{b} + u_j < b)$, the expected value of $y_j|\mathbf{x}_j$ conditional on $y_j|\mathbf{x}_j$ being in the interval (a, b), meaning that $y_j|\mathbf{x}_j$ is censored.
a and *b* are specified as they are for `pr()`.

`ystar(`*a*`,`*b*`)` calculates $E(y_j^*)$, where $y_j^* = a$ if $\mathbf{x}_j\mathbf{b} + u_j \leq a$, $y_j^* = b$ if $\mathbf{x}_j\mathbf{b} + u_j \geq b$, and $y_j^* = \mathbf{x}_j\mathbf{b} + u_j$ otherwise, meaning that y_j^* is truncated. *a* and *b* are specified as they are for `pr()`.

`dfbeta(`*varname*`)` calculates the DFBETA for *varname*, the difference between the regression coefficient when the jth observation is included and excluded, said difference being scaled by the estimated standard error of the coefficient. *varname* must have been included among the regressors in the previously fitted model. The calculation is automatically restricted to the estimation subsample.

`stdp` calculates the standard error of the prediction, which can be thought of as the standard error of the predicted expected value or mean for the observation's covariate pattern. The standard error of the prediction is also referred to as the standard error of the fitted value.

`stdf` calculates the standard error of the forecast, which is the standard error of the point prediction for 1 observation. It is commonly referred to as the standard error of the future or forecast value. By construction, the standard errors produced by `stdf` are always larger than those produced by `stdp`; see *Methods and formulas*.

`stdr` calculates the standard error of the residuals.

`covratio` calculates COVRATIO (Belsley, Kuh, and Welsch 1980), a measure of the influence of the jth observation based on considering the effect on the variance–covariance matrix of the estimates. The calculation is automatically restricted to the estimation subsample.

`dfits` calculates DFITS (Welsch and Kuh 1977) and attempts to summarize the information in the leverage versus residual-squared plot into one statistic. The calculation is automatically restricted to the estimation subsample.

`welsch` calculates Welsch distance (Welsch 1982) and is a variation on `dfits`. The calculation is automatically restricted to the estimation subsample.

Syntax for dfbeta

`dfbeta` [*indepvar* [*indepvar* [...]]] [, `stub(`*name*`)`]

Menu

Statistics > Linear models and related > Regression diagnostics > Residual-versus-fitted plot

Option for dfbeta

stub(*name*) specifies the leading characters dfbeta uses to name the new variables to be generated. The default is stub(_dfbeta_).

Syntax for estat hettest

```
estat hettest [varlist] [, rhs [normal|iid|fstat] mtest[(spec)]]
```

Menu

Statistics > Postestimation > Reports and statistics

Options for estat hettest

rhs specifies that tests for heteroskedasticity be performed for the right-hand-side (explanatory) variables of the fitted regression model. The rhs option may be combined with a *varlist*.

normal, the default, causes estat hettest to compute the original Breusch–Pagan/Cook–Weisberg test, which assumes that the regression disturbances are normally distributed.

iid causes estat hettest to compute the $N * R^2$ version of the score test that drops the normality assumption.

fstat causes estat hettest to compute the F statistic version that drops the normality assumption.

mtest[(*spec*)] specifies that multiple testing be performed. The argument specifies how p-values are adjusted. The following specifications, *spec*, are supported:

bonferroni	Bonferroni's multiple testing adjustment
holm	Holm's multiple testing adjustment
sidak	Šidák's multiple testing adjustment
noadjust	no adjustment is made for multiple testing

mtest may be specified without an argument. This is equivalent to specifying mtest(noadjust), that is, tests for the individual variables should be performed with unadjusted p-values. By default, estat hettest does not perform multiple testing. mtest[(*spec*)] may not be specified with iid or fstat.

Syntax for estat imtest

```
estat imtest [, preserve white]
```

Menu

Statistics > Postestimation > Reports and statistics

Options for estat imtest

`preserve` specifies that the data in memory be preserved, all variables and cases that are not needed in the calculations be dropped, and at the conclusion the original data be restored. This option is costly for large datasets. However, because `estat imtest` has to perform an auxiliary regression on $k(k+1)/2$ temporary variables, where k is the number of regressors, it may not be able to perform the test otherwise.

`white` specifies that White's original heteroskedasticity test also be performed.

Syntax for estat ovtest

`estat ovtest [, rhs]`

Menu

Statistics > Postestimation > Reports and statistics

Option for estat ovtest

`rhs` specifies that powers of the right-hand-side (explanatory) variables be used in the test rather than powers of the fitted values.

Syntax for estat szroeter

`estat szroeter` [*varlist*] [`, rhs mtest(`*spec*`)`]

Either *varlist* or `rhs` must be specified.

Menu

Statistics > Postestimation > Reports and statistics

Options for estat szroeter

`rhs` specifies that tests for heteroskedasticity be performed for the right-hand-side (explanatory) variables of the fitted regression model. Option `rhs` may be combined with a *varlist*.

`mtest(`*spec*`)` specifies that multiple testing be performed. The argument specifies how p-values are adjusted. The following specifications *spec* are supported:

`bonferroni`	Bonferroni's multiple testing adjustment
`holm`	Holm's multiple testing adjustment
`sidak`	Šidák's multiple testing adjustment
`noadjust`	no adjustment is made for multiple testing

`estat szroeter` always performs multiple testing. By default, it does not adjust the p-values.

Syntax for estat vif

`estat vif` [, uncentered]

Menu

Statistics > Postestimation > Reports and statistics

Option for estat vif

`uncentered` requests that the computation of the uncentered variance inflation factors. This option is often used to detect the collinearity of the regressors with the constant. `estat vif, uncentered` may be used after regression models fit without the constant term.

Syntax for acprplot

`acprplot` *indepvar* [, *acprplot_options*]

acprplot_options	description
Plot	
marker_options	change look of markers (color, size, etc.)
marker_label_options	add marker labels; change look or position
Reference line	
`rlopts(`*cline_options*`)`	affect rendition of the reference line
Options	
`lowess`	add a lowess smooth of the plotted points
`lsopts(`*lowess_options*`)`	affect rendition of the lowess smooth
`mspline`	add median spline of the plotted points
`msopts(`*mspline_options*`)`	affect rendition of the spline
Add plots	
`addplot(`*plot*`)`	add other plots to the generated graph
Y axis, X axis, Titles, Legend, Overall	
twoway_options	any options other than `by()` documented in [G] ***twoway_options***

Menu

Statistics > Linear models and related > Regression diagnostics > Augmented component-plus-residual plot

Options for acprplot

Plot

marker_options affect the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G] ***marker_options***.

marker_label_options specify if and how the markers are to be labeled; see [G] ***marker_label_options***.

Reference line

`rlopts(`*cline_options*`)` affects the rendition of the reference line. See [G] ***cline_options***.

Options

`lowess` adds a lowess smooth of the plotted points to assist in detecting nonlinearities.

`lsopts(`*lowess_options*`)` affects the rendition of the lowess smooth. For an explanation of these options, especially the `bwidth()` option, see [R] **lowess**. Specifying `lsopts()` implies the `lowess` option.

`mspline` adds a median spline of the plotted points to assist in detecting nonlinearities.

`msopts(`*mspline_options*`)` affects the rendition of the spline. For an explanation of these options, especially the `bands()` option, see [G] **graph twoway mspline**. Specifying `msopts()` implies the `mspline` option.

Add plots

`addplot(`*plot*`)` provides a way to add other plots to the generated graph. See [G] ***addplot_option***.

Y axis, X axis, Titles, Legend, Overall

twoway_options are any of the options documented in [G] ***twoway_options***, excluding `by()`. These include options for titling the graph (see [G] ***title_options***) and for saving the graph to disk (see [G] ***saving_option***).

Syntax for avplot

`avplot` *indepvar* [`,` *avplot_options*]

avplot_options	description
Plot	
marker_options	change look of markers (color, size, etc.)
marker_label_options	add marker labels; change look or position
Reference line	
`rlopts(`*cline_options*`)`	affect rendition of the reference line
Add plots	
`addplot(`*plot*`)`	add other plots to the generated graph
Y axis, X axis, Titles, Legend, Overall	
twoway_options	any options other than `by()` documented in [G] ***twoway_options***

Menu

Statistics > Linear models and related > Regression diagnostics > Added-variable plot

Options for avplot

Plot

marker_options affect the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G] ***marker_options***.

marker_label_options specify if and how the markers are to be labeled; see [G] ***marker_label_options***.

Reference line

`rlopts(`*cline_options*`)` affects the rendition of the reference line. See [G] ***cline_options***.

Add plots

`addplot(`*plot*`)` provides a way to add other plots to the generated graph. See [G] ***addplot_option***.

Y axis, X axis, Titles, Legend, Overall

twoway_options are any of the options documented in [G] ***twoway_options***, excluding `by()`. These include options for titling the graph (see [G] ***title_options***) and for saving the graph to disk (see [G] ***saving_option***).

Syntax for avplots

`avplots` [`,` *avplots_options*]

avplots_options	description
Plot	
marker_options	change look of markers (color, size, etc.)
marker_label_options	add marker labels; change look or position
combine_options	any of the options documented in [G] **graph combine**
Reference line	
`rlopts(`*cline_options*`)`	affect rendition of the reference line
Y axis, X axis, Titles, Legend, Overall	
twoway_options	any options other than `by()` documented in [G] ***twoway_options***

Menu

Statistics > Linear models and related > Regression diagnostics > Added-variable plot

Options for avplots

Plot

marker_options affect the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G] ***marker_options***.

marker_label_options specify if and how the markers are to be labeled; see [G] ***marker_label_options***.

combine_options are any of the options documented in [G] **graph combine**. These include options for titling the graph (see [G] ***title_options***) and for saving the graph to disk (see [G] ***saving_option***).

Reference line

`rlopts(`*cline_options*`)` affects the rendition of the reference line. See [G] ***cline_options***.

Y axis, X axis, Titles, Legend, Overall

twoway_options are any of the options documented in [G] ***twoway_options***, excluding `by()`. These include options for titling the graph (see [G] ***title_options***) and for saving the graph to disk (see [G] ***saving_option***).

Syntax for cprplot

`cprplot` *indepvar* [, *cprplot_options*]

cprplot_options	description
Plot	
marker_options	change look of markers (color, size, etc.)
marker_label_options	add marker labels; change look or position
Reference line	
`rlopts(`*cline_options*`)`	affect rendition of the reference line
Options	
`lowess`	add a lowess smooth of the plotted points
`lsopts(`*lowess_options*`)`	affect rendition of the lowess smooth
`mspline`	add median spline of the plotted points
`msopts(`*mspline_options*`)`	affect rendition of the spline
Add plots	
`addplot(`*plot*`)`	add other plots to the generated graph
Y axis, X axis, Titles, Legend, Overall	
twoway_options	any options other than `by()` documented in [G] ***twoway_options***

Menu

Statistics > Linear models and related > Regression diagnostics > Component-plus-residual plot

Options for cprplot

Plot

marker_options affect the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G] ***marker_options***.

marker_label_options specify if and how the markers are to be labeled; see [G] ***marker_label_options***.

Reference line

`rlopts(`*cline_options*`)` affects the rendition of the reference line. See [G] ***cline_options***.

Options

`lowess` adds a lowess smooth of the plotted points to assist in detecting nonlinearities.

`lsopts(`*lowess_options*`)` affects the rendition of the lowess smooth. For an explanation of these options, especially the `bwidth()` option, see [R] **lowess**. Specifying `lsopts()` implies the `lowess` option.

`mspline` adds a median spline of the plotted points to assist in detecting nonlinearities.

`msopts(`*mspline_options*`)` affects the rendition of the spline. For an explanation of these options, especially the `bands()` option, see [G] **graph twoway mspline**. Specifying `msopts()` implies the `mspline` option.

Add plots

`addplot(`*plot*`)` provides a way to add other plots to the generated graph. See [G] ***addplot_option***.

Y axis, X axis, Titles, Legend, Overall

twoway_options are any of the options documented in [G] ***twoway_options***, excluding `by()`. These include options for titling the graph (see [G] ***title_options***) and for saving the graph to disk (see [G] ***saving_option***).

Syntax for lvr2plot

`lvr2plot` [, *lvr2plot_options*]

lvr2plot_options	description
Plot	
marker_options	change look of markers (color, size, etc.)
marker_label_options	add marker labels; change look or position
Add plots	
`addplot(`*plot*`)`	add other plots to the generated graph
Y axis, X axis, Titles, Legend, Overall	
twoway_options	any options other than `by()` documented in [G] ***twoway_options***

Menu

Statistics > Linear models and related > Regression diagnostics > Leverage-versus-squared-residual plot

Options for lvr2plot

Plot

marker_options affect the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G] ***marker_options***.

marker_label_options specify if and how the markers are to be labeled; see [G] ***marker_label_options***.

Add plots

`addplot(`*plot*`)` provides a way to add other plots to the generated graph; see [G] ***addplot_option***.

Y axis, X axis, Titles, Legend, Overall

twoway_options are any of the options documented in [G] ***twoway_options***, excluding `by()`. These include options for titling the graph (see [G] ***title_options***) and for saving the graph to disk (see [G] ***saving_option***).

Syntax for rvfplot

`rvfplot` [, *rvfplot_options*]

rvfplot_options	description
Plot	
marker_options	change look of markers (color, size, etc.)
marker_label_options	add marker labels; change look or position
Add plots	
`addplot(`*plot*`)`	add plots to the generated graph
Y axis, X axis, Titles, Legend, Overall	
twoway_options	any options other than `by()` documented in [G] ***twoway_options***

Menu

Statistics > Linear models and related > Regression diagnostics > Residual-versus-fitted plot

Options for rvfplot

Plot

marker_options affect the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G] ***marker_options***.

marker_label_options specify if and how the markers are to be labeled; see [G] ***marker_label_options***.

Add plots

`addplot(`*plot*`)` provides a way to add plots to the generated graph. See [G] ***addplot_option***.

Y axis, X axis, Titles, Legend, Overall

twoway_options are any of the options documented in [G] ***twoway_options***, excluding `by()`. These include options for titling the graph (see [G] ***title_options***) and for saving the graph to disk (see [G] ***saving_option***).

Syntax for rvpplot

`rvpplot` *indepvar* [, *rvpplot_options*]

rvpplot_options	description
Plot	
marker_options	change look of markers (color, size, etc.)
marker_label_options	add marker labels; change look or position
Add plots	
`addplot(`*plot*`)`	add other plots to the generated graph
Y axis, X axis, Titles, Legend, Overall	
twoway_options	any options other than `by()` documented in [G] ***twoway_options***

Menu

Statistics > Linear models and related > Regression diagnostics > Residual-versus-predictor plot

Options for rvpplot

Plot

marker_options affect the rendition of markers drawn at the plotted points, including their shape, size, color, and outline; see [G] ***marker_options***.

marker_label_options specify if and how the markers are to be labeled; see [G] ***marker_label_options***.

Add plots

`addplot(`*plot*`)` provides a way to add other plots to the generated graph; see [G] ***addplot_option***.

Y axis, X axis, Titles, Legend, Overall

twoway_options are any of the options documented in [G] ***twoway_options***, excluding `by()`. These include options for titling the graph (see [G] ***title_options***) and for saving the graph to disk (see [G] ***saving_option***).

Remarks

Remarks are presented under the following headings:

Many of these commands concern identifying influential data in linear regression. This is, unfortunately, a field that is dominated by jargon, codified and partially begun by Belsley, Kuh, and Welsch (1980). In the words of Chatterjee and Hadi (1986, 416), "Belsley, Kuh, and Welsch's book, *Regression Diagnostics*, was a very valuable contribution to the statistical literature, but it unleashed on an unsuspecting statistical community a computer speak (à la Orwell), the likes of which we have never seen." Things have only gotten worse since then. Chatterjee and Hadi's (1986, 1988) own attempts to clean up the jargon did not improve matters (see Hoaglin and Kempthorne [1986], Velleman [1986], and Welsch [1986]). We apologize for the jargon, and for our contribution to the jargon in the form of inelegant command names, we apologize most of all.

Model *sensitivity* refers to how estimates are affected by subsets of our data. Imagine data on y and x, and assume that the data are to be fit by the regression $y_i = \alpha + \beta x_i + \epsilon_i$. The regression estimates of α and β are a and b, respectively. Now imagine that the estimated a and b would be different if a small portion of the dataset, perhaps even one observation, were deleted. As a data analyst, you would like to think that you are summarizing tendencies that apply to all the data, but you have just been told that the model you fit is unduly influenced by one point or just a few points and that, as a matter of fact, there is another model that applies to the rest of the data—a model that you have ignored. The search for subsets of the data that, if deleted, would change the results markedly is a predominant theme of this entry.

There are three key issues in identifying model sensitivity to individual observations, which go by the names *residuals*, *leverage*, and *influence*. In our $y_i = a + bx_i + e_i$ regression, the residuals are, of course, e_i—they reveal how much our fitted value $\widehat{y}_i = a + bx_i$ differs from the observed y_i. A point (x_i, y_i) with a corresponding large residual is called an outlier. Say that you are interested in outliers because you somehow think that such points will exert undue influence on your estimates. Your feelings are generally right, but there are exceptions. A point might have a huge residual and yet not affect the estimated b at all. Nevertheless, studying observations with large residuals almost always pays off.

(x_i, y_i) can be an outlier in another way—just as y_i can be far from $\widehat{y}_i$, x_i can be far from the center of mass of the other x's. Such an "outlier" should interest you just as much as the more traditional outliers. Picture a scatterplot of y against x with thousands of points in some sort of mass at the lower left of the graph and one point at the upper right of the graph. Now run a regression line through the points—the regression line will come close to the point at the upper right of the graph and may in fact, go through it. That is, this isolated point will not appear as an outlier as measured by residuals because its residual will be small. Yet this point might have a dramatic effect on our resulting estimates in the sense that, were you to delete the point, the estimates would change markedly. Such a point is said to have high leverage. Just as with traditional outliers, a high leverage point does not necessarily have an undue effect on regression estimates, but if it does not, it is more the exception than the rule.

Now all this is a most unsatisfactory state of affairs. Points with large residuals may, but need not, have a large effect on our results, and points with small residuals may still have a large effect. Points with high leverage may, but need not, have a large effect on our results, and points with low leverage may still have a large effect. Can you not identify the influential points and simply have the computer list them for you? You can, but you will have to define what you mean by "influential".

"Influential" is defined with respect to some statistic. For instance, you might ask which points in your data have a large effect on your estimated a, which points have a large effect on your estimated b, which points have a large effect on your estimated standard error of b, and so on, but do not be surprised when the answers to these questions are different. In any case, obtaining such measures is not difficult—all you have to do is fit the regression excluding each observation one at a time and record the statistic of interest which, in the day of the modern computer, is not too onerous. Moreover, you can save considerable computer time by doing algebra ahead of time and working out formulas that will calculate the same answers as if you ran each of the regressions. (Ignore the question of pairs of observations that, together, exert undue influence, and triples, and so on, which remains largely unsolved and for which the brute force fit-every-possible-regression procedure is not a viable alternative.)

Fitted values and residuals

Typing `predict` *newvar* with no options creates *newvar* containing the fitted values. Typing `predict` *newvar*`, resid` creates *newvar* containing the residuals.

▷ Example 1

Continuing with example 1 from [R] **regress**, we wish to fit the following model:

$$\texttt{mpg} = \beta_0 + \beta_1\texttt{weight} + \beta_2\texttt{weight}^2 + \beta_3\texttt{foreign} + \epsilon$$

```
. use http://www.stata-press.com/data/r11/auto
(1978 Automobile Data)

. regress mpg weight c.weight#c.weight foreign

      Source |       SS       df       MS              Number of obs =      74
-------------+------------------------------           F(  3,    70) =   52.25
       Model |  1689.15372     3  563.05124            Prob > F      =  0.0000
    Residual |   754.30574    70  10.7757963           R-squared     =  0.6913
-------------+------------------------------           Adj R-squared =  0.6781
       Total |  2443.45946    73  33.4720474           Root MSE      =  3.2827

------------------------------------------------------------------------------
         mpg |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      weight |  -.0165729   .0039692    -4.18   0.000    -.0244892   -.0086567
             |
   c.weight#|
    c.weight |   1.59e-06   6.25e-07     2.55   0.013     3.45e-07    2.84e-06
             |
     foreign |    -2.2035   1.059246    -2.08   0.041      -4.3161   -.0909002
       _cons |   56.53884   6.197383     9.12   0.000     44.17855    68.89913
------------------------------------------------------------------------------
```

That done, we can now obtain the predicted values from the regression. We will store them in a new variable called pmpg by typing `predict pmpg`. Because `predict` produces no output, we will follow that by summarizing our predicted and observed values.

```
. predict pmpg
(option xb assumed; fitted values)

. summarize pmpg mpg

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
        pmpg |        74     21.2973    4.810311   13.59953   31.86288
         mpg |        74     21.2973    5.785503         12         41
```

◁

▷ Example 2: Out-of-sample predictions

We can just as easily obtain predicted values from the model by using a wholly different dataset from the one on which the model was fit. The only requirement is that the data have the necessary variables, which here are `weight` and `foreign`.

Using the data on two new cars (the Pontiac Sunbird and the Volvo 260) from the `newautos.dta` dataset, we can obtain out-of-sample predictions (or forecasts) by typing

```
. use http://www.stata-press.com/data/r11/newautos, clear
(New Automobile Models)

. predict pmpg
(option xb assumed; fitted values)

. list, divider

     +-------------------------------------------+
     |         make | weight |  foreign |   pmpg |
     |--------------+--------+----------+--------|
  1. | Pont. Sunbird |   2690 | Domestic | 23.47137 |
  2. |    Volvo 260 |   3170 |  Foreign | 17.78846 |
     +-------------------------------------------+
```

The Pontiac Sunbird has a predicted mileage rating of 23.5 mpg, whereas the Volvo 260 has a predicted rating of 17.8 mpg. In comparison, the actual mileage ratings are 24 for the Pontiac and 17 for the Volvo.

◁

Prediction standard errors

`predict` can calculate the standard error of the forecast (`stdf` option), the standard error of the prediction (`stdp` option), and the standard error of the residual (`stdr` option). It is easy to confuse `stdf` and `stdp` because both are often called the prediction error. Consider the prediction $\widehat{y}_j = \mathbf{x}_j\mathbf{b}$, where $\mathbf{b}$ is the estimated coefficient (column) vector and $\mathbf{x}_j$ is a (row) vector of independent variables for which you want the prediction. First, $\widehat{y}_j$ has a variance due to the variance of the estimated coefficient vector $\mathbf{b}$,

$$\mathrm{Var}(\widehat{y}_j) = \mathrm{Var}(\mathbf{x}_j\mathbf{b}) = s^2 h_j$$

where $h_j = \mathbf{x}_j(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x}_j'$ and s^2 is the mean squared error of the regression. Do not panic over the algebra—just remember that $\mathrm{Var}(\widehat{y}_j) = s^2 h_j$, whatever s^2 and h_j are. `stdp` calculates this quantity. This is the error in the prediction due to the uncertainty about $\mathbf{b}$.

If you are about to hand this number out as your forecast, however, there is another error. According to your model, the true value of y_j is given by

$$y_j = \mathbf{x}_j\mathbf{b} + \epsilon_j = \widehat{y}_j + \epsilon_j$$

and thus the $\mathrm{Var}(y_j) = \mathrm{Var}(\widehat{y}_j) + \mathrm{Var}(\epsilon_j) = s^2h_j + s^2$, which is the square of `stdf`. `stdf`, then, is the sum of the error in the prediction plus the residual error.

`stdr` has to do with an analysis-of-variance decomposition of s^2, the estimated variance of y. The standard error of the prediction is s^2h_j, and therefore $s^2h_j + s^2(1-h_j) = s^2$ decomposes s^2 into the prediction and residual variances.

▷ Example 3: standard error of the forecast

Returning to our model of `mpg` on `weight`, `weight`^{2}, and `foreign`, we previously predicted the mileage rating for the Pontiac Sunbird and Volvo 260 as 23.5 and 17.8 mpg, respectively. We now want to put a standard error around our forecast. Remember, the data for these two cars were in `newautos.dta`:

```
. use http://www.stata-press.com/data/r11/newautos, clear
(New Automobile Models)
. predict pmpg
(option xb assumed; fitted values)
. predict se_pmpg, stdf
. list, divider
```

	make	weight	foreign	pmpg	se_pmpg
1.	Pont. Sunbird	2690	Domestic	23.47137	3.341823
2.	Volvo 260	3170	Foreign	17.78846	3.438714

Thus an approximate 95% confidence interval for the mileage rating of the Volvo 260 is $17.8 \pm 2 \cdot 3.44 = [\,10.92, 24.68\,]$.

◁

Prediction with weighted data

`predict` can be used after frequency-weighted (`fweight`) estimation, just as it is used after unweighted estimation. The technical note below concerns the use of `predict` after analytically weighted (`aweight`) estimation.

❑ Technical note

After analytically weighted estimation, `predict` is willing to calculate only the prediction (no options), residual (`residual` option), standard error of the prediction (`stdp` option), and diagonal elements of the projection matrix (`hat` option). Moreover, the results produced by `hat` need to be adjusted, as will be described. For analytically weighted estimation, the standard error of the forecast and residuals, the standardized and studentized residuals, and Cook's D are not statistically well-defined concepts.

To obtain the correct values of the diagonal elements of the hat matrix, you can use `predict` with the `hat` option to make a first, partially adjusted calculation, and then follow that by completing the adjustment. Assume that you are fitting a linear regression model weighting the data with the variable `w` (`[aweight=w]`). Begin by creating a new variable, `w0`:

```
. predict resid if e(sample), resid
. summarize w if resid < . & e(sample)
. gen w0=w/r(mean)
```

Some caution is necessary at this step—the `summarize w` must be performed on the same sample that was used to fit the model, which means that you must include `if e(sample)` to restrict the prediction to the estimation sample. You created the residual and then included the modifier '`if resid < .`' so that if the dependent variable or any of the independent variables is missing, the corresponding observations will be excluded from the calculation of the average value of the original weight.

To correct `predict`'s `hat` calculation, multiply the result by `w0`:

```
. predict myhat, hat
. replace myhat = w0 * myhat
```

❑

Residual-versus-fitted plots

▷ Example 4: rvfplot

Using the automobile dataset described in [U] **1.2.2 Example datasets**, we will use `regress` to fit a model of `price` on `weight`, `mpg`, `foreign`, and the interaction of `foreign` with `mpg`. We specify `foreign##c.mpg` to obtain the interaction of `foreign` with `mpg`; see [U] **11.4.3 Factor variables**.

```
. use http://www.stata-press.com/data/r11/auto, clear
(1978 Automobile Data)

. regress price weight foreign##c.mpg
```

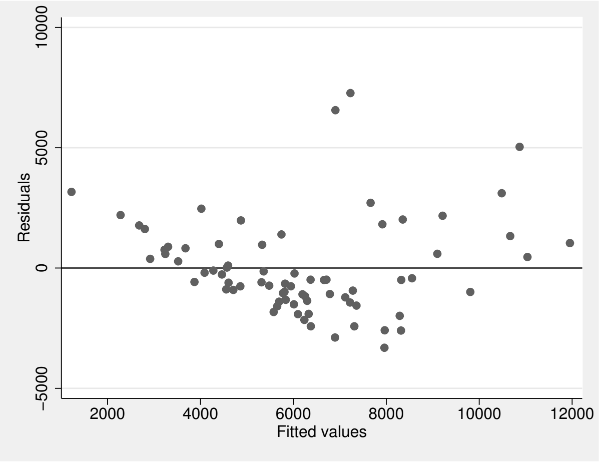
Source	SS	df	MS
Model	350319665	4	87579916.3
Residual	284745731	69	4126749.72
Total	635065396	73	8699525.97

Number of obs =	74
F(4, 69) =	21.22
Prob > F =	0.0000
R-squared =	0.5516
Adj R-squared =	0.5256
Root MSE =	2031.4

price	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
weight	4.613589	.7254961	6.36	0.000	3.166263	6.060914
1.foreign	11240.33	2751.681	4.08	0.000	5750.878	16729.78
mpg	263.1875	110.7961	2.38	0.020	42.15527	484.2197
foreign# c.mpg						
1	-307.2166	108.5307	-2.83	0.006	-523.7294	-90.70368
_cons	-14449.58	4425.72	-3.26	0.002	-23278.65	-5620.51

Once we have fit a model, we may use any of the regression diagnostics commands. `rvfplot` (read residual-versus-fitted plot) graphs the residuals against the fitted values:

```
. rvfplot, yline(0)
```



All the diagnostic plot commands allow the `graph twoway` and `graph twoway scatter` options; we specified a `yline(0)` to draw a line across the graph at $y = 0$; see [G] **graph twoway scatter**.

In a well-fitted model, there should be no pattern to the residuals plotted against the fitted values—something not true of our model. Ignoring the two outliers at the top center of the graph, we see curvature in the pattern of the residuals, suggesting a violation of the assumption that `price` is linear in our independent variables. We might also have seen increasing or decreasing variation in the residuals—heteroskedasticity. Any pattern whatsoever indicates a violation of the least-squares assumptions.

◁

Added-variable plots

▷ Example 5: avplot

We continue with our price model, and another diagnostic graph is provided by `avplot` (read added-variable plot, also known as the partial-regression leverage plot).

One of the wonderful features of one-regressor regressions (regressions of y on one x) is that we can graph the data and the regression line. There is no easier way to understand the regression than to examine such a graph. Unfortunately, we cannot do this when we have more than one regressor. With two regressors, it is still theoretically possible—the graph must be drawn in three dimensions, but with three or more regressors no graph is possible.

The added-variable plot is an attempt to project multidimensional data back to the two-dimensional world for each of the original regressors. This is, of course, impossible without making some concessions. Call the coordinates on an added-variable plot y and x. The added-variable plot has the following properties:

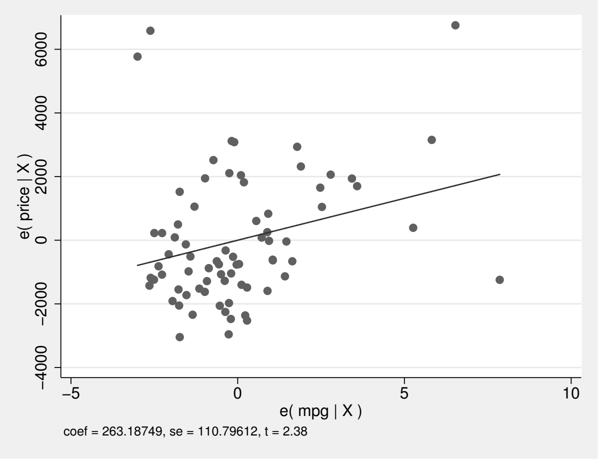
- There is a one-to-one correspondence between (x_i, y_i) and the ith observation used in the original regression.
- A regression of y on x has the same coefficient and standard error (up to a degree-of-freedom adjustment) as the estimated coefficient and standard error for the regressor in the original regression.

- The "outlierness" of each observation in determining the slope is in some sense preserved.

It is equally important to note the properties that are not listed. The y and x coordinates of the added-variable plot cannot be used to identify functional form, or, at least, not well (see Mallows [1986]). In the construction of the added-variable plot, the relationship between y and x is forced to be linear.

Let's examine the added-variable plot for `mpg` in our regression of `price` on `weight` and `foreign##c.mpg`:

```
. avplot mpg
```



This graph suggests a problem in determining the coefficient on `mpg`. Were this a one-regressor regression, the two points at the top-left corner and the one at the top right would cause us concern, and so it does in our more complicated multiple-regressor case. To identify the problem points, we retyped our command, modifying it to read `avplot mpg, mlabel(make)`, and discovered that the two cars at the top left are the Cadillac Eldorado and the Lincoln Versailles; the point at the top right is the Cadillac Seville. These three cars account for 100% of the luxury cars in our data, suggesting that our model is misspecified. By the way, the point at the lower right of the graph, also cause for concern, is the Plymouth Arrow, our data-entry error.

◁

❑ Technical note

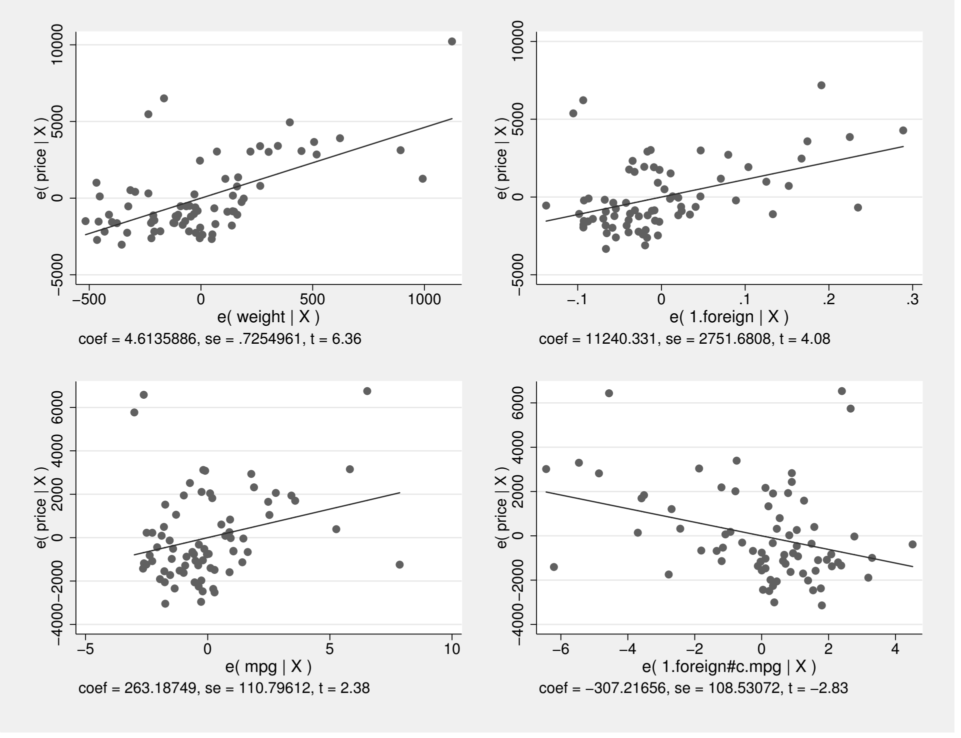
Stata's `avplot` command can be used with regressors already in the model, as we just did, or with potential regressors not yet in the model. In either case, `avplot` will produce the correct graph. The name "added-variable plot" is unfortunate in the case when the variable is already among the list of regressors but is, we think, still preferable to the name "partial-regression leverage plot" assigned by Belsley, Kuh, and Welsch (1980, 30) and more in the spirit of the original use of such plots by Mosteller and Tukey (1977, 271–279). Welsch (1986, 403), however, disagrees: "I am sorry to see that Chatterjee and Hadi [1986] endorse the term 'added-variable plot' when X_j is part of the original model" and goes on to suggest the name "adjusted partial residual plot".

❑

▷ Example 6: avplots

Added-variable plots are so useful that we should look at them for every regressor in the data. `avplots` makes this easy:

```
. avplots
```



◁

Component-plus-residual plots

Added-variable plots are successful at identifying outliers, but they cannot be used to identify functional form. The component-plus-residual plot (Ezekiel 1924; Larsen and McCleary 1972) is another attempt at projecting multidimensional data into a two-dimensional form, but with different properties. Although the added-variable plot can identify outliers, the component-plus-residual plot cannot. It can, however, be used to examine the functional-form assumptions of the model. Both plots have the property that a regression line through the coordinates has a slope equal to the estimated coefficient in the regression model.

▷ Example 7: cprplot and acprplot

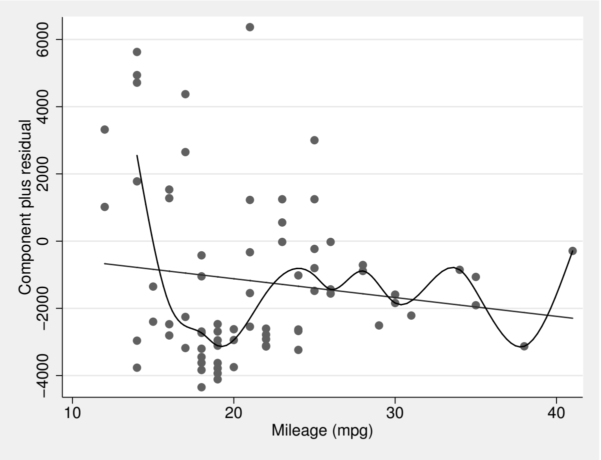
To illustrate these plots, we begin with a different model:

```
. use http://www.stata-press.com/data/r11/auto1, clear
(Automobile Models)

. regress price mpg weight

      Source |       SS       df       MS              Number of obs =      74
-------------+------------------------------           F(  2,    71) =   14.90
       Model |   187716578     2    93858289           Prob > F      =  0.0000
    Residual |   447348818    71  6300687.58           R-squared     =  0.2956
-------------+------------------------------           Adj R-squared =  0.2757
       Total |   635065396    73  8699525.97           Root MSE      =  2510.1

------------------------------------------------------------------------------
       price |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         mpg |   -55.9393   75.24136    -0.74   0.460    -205.9663    94.08771
      weight |   1.710992   .5861682     2.92   0.005     .5422063    2.879779
       _cons |     2197.9   3190.768     0.69   0.493    -4164.311     8560.11
------------------------------------------------------------------------------
```

In fact, we know that the effects of mpg in this model are nonlinear—if we added mpg squared to the model, its coefficient would have a t statistic of 2.38, the t statistic on mpg would become -2.48, and weight's effect would become about one-third of its current value and become statistically insignificant. Pretend that we do not know this.

The component-plus-residual plot for mpg is

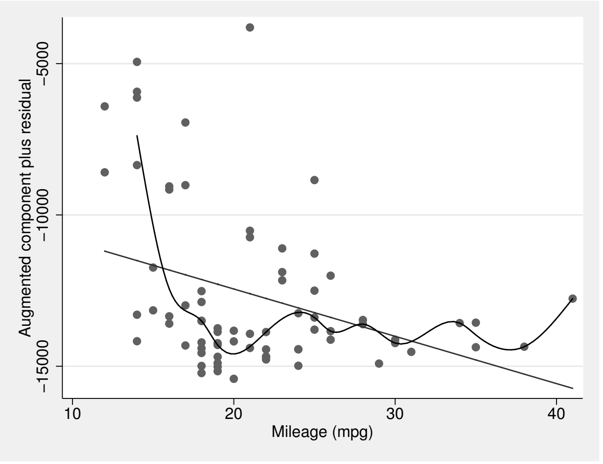
```
. cprplot mpg, mspline msopts(bands(13))
```



We are supposed to examine the above graph for nonlinearities or, equivalently, ask if the regression line, which has slope equal to the estimated effect of mpg in the original model, fits the data adequately. To assist our eyes, we added a median spline. Perhaps some people may detect nonlinearity from this graph, but we assert that if we had not previously revealed the nonlinearity of mpg and if we had not added the median spline, the graph would not overly bother us.

Mallows (1986) proposed an augmented component-plus-residual plot that is often more sensitive to detecting nonlinearity:

```
. acprplot mpg, mspline msopts(bands(13))
```



It does do somewhat better.

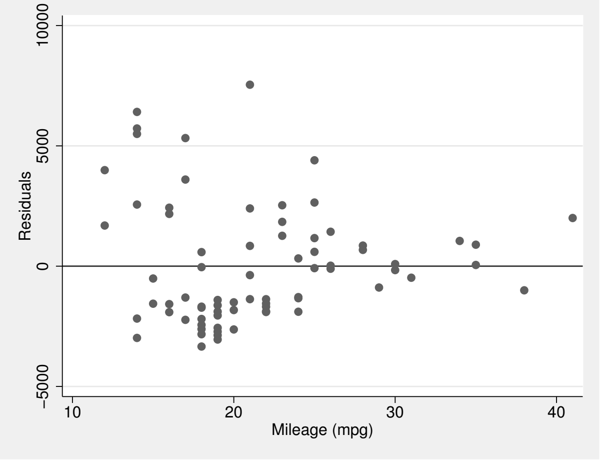
◁

Residual-versus-predictor plots

▷ Example 8: rvpplot

The residual-versus-predictor plot is a simple way to look for violations of the regression assumptions. If the assumptions are correct, there should be no pattern in the graph. Using our `price` on `mpg` and `weight` model, we type

```
. rvpplot mpg, yline(0)
```



Remember, any pattern counts as a problem, and in this graph, we see that the variation in the residuals decreases as `mpg` increases.

◁

Leverage statistics

In addition to providing fitted values and the associated standard errors, the `predict` command can also be used to generate various statistics used to detect the influence of individual observations. This section provides a brief introduction to leverage (hat) statistics, and some of the following subsections discuss other influence statistics produced by `predict`.

▷ Example 9: diagonal elements of projection matrix

The diagonal elements of the projection matrix, obtained by the `hat` option, are a measure of distance in explanatory variable space. `leverage` is a synonym for `hat`.

```
. use http://www.stata-press.com/data/r11/auto
(1978 Automobile Data)
. regress mpg weight c.weight#c.weight foreign
  (output omitted )
. predict xdist, hat
. summarize xdist, detail
                          Leverage
-------------------------------------------------------------
      Percentiles      Smallest
 1%     .0251334       .0251334
 5%     .0255623       .0251334
10%     .0259213       .0253883       Obs                  74
25%     .0278442       .0255623       Sum of Wgt.          74

50%       .04103                      Mean           .0540541
                        Largest       Std. Dev.      .0459218
75%     .0631279       .1593606
90%     .0854584       .1593606       Variance       .0021088
95%     .1593606       .2326124       Skewness       3.440809
99%     .3075759       .3075759       Kurtosis       16.95135
```

Some 5% of our sample has an `xdist` measure in excess of 0.15. Let's force them to reveal their identities:

```
. list foreign make mpg if xdist>.15, divider
```

	foreign	make	mpg
24.	Domestic	Ford Fiesta	28
26.	Domestic	Linc. Continental	12
27.	Domestic	Linc. Mark V	12
43.	Domestic	Plym. Champ	34

To understand why these cars are on this list, we must remember that the explanatory variables in our model are `weight` and `foreign` and that `xdist` measures distance in this metric. The Ford Fiesta and the Plymouth Champ are the two lightest domestic cars in our data. The Lincolns are the two heaviest domestic cars.

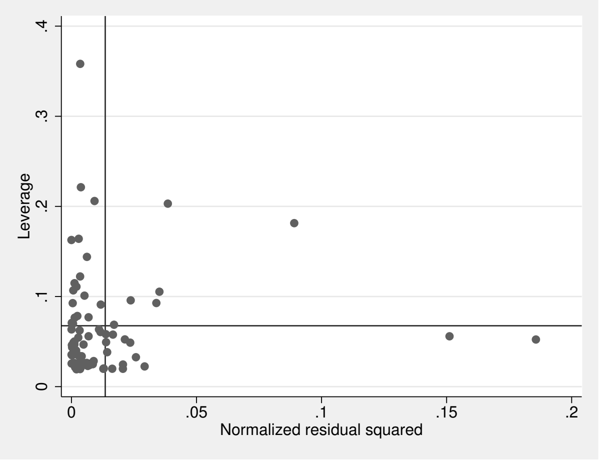
◁

L-R plots

▷ Example 10: lvr2plot

One of the most useful diagnostic graphs is provided by `lvr2plot` (leverage-versus-residual-squared plot), a graph of leverage against the (normalized) residuals squared.

```
. use http://www.stata-press.com/data/r11/auto, clear
(1978 Automobile Data)
. regress price weight foreign##c.mpg
  (output omitted)
. lvr2plot
```

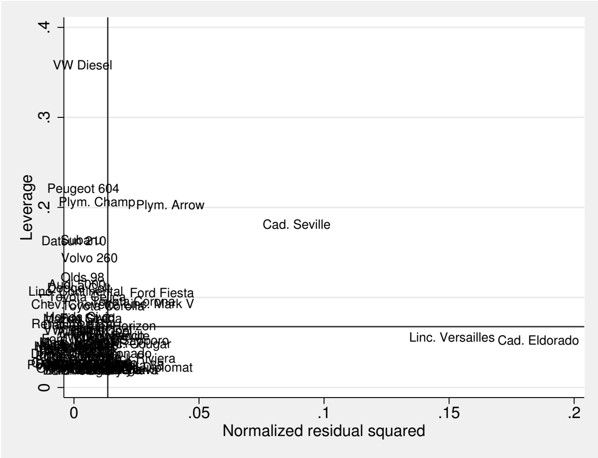


The lines on the chart show the average values of leverage and the (normalized) residuals squared. Points above the horizontal line have higher-than-average leverage; points to the right of the vertical line have larger-than-average residuals.

One point immediately catches our eye, and four more make us pause. The point at the top of the graph has high leverage and a smaller-than-average residual. The other points that bother us all have higher-than-average leverage, two with smaller-than-average residuals and two with larger-than-average residuals.

A less pretty but more useful version of the above graph specifies that `make` be used as the symbol (see [G] ***marker_label_options***):

```
. lvr2plot, mlabel(make) mlabp(0) m(none) mlabsize(small)
```



The VW Diesel, Plymouth Champ, Plymouth Arrow, and Peugeot 604 are the points that cause us the most concern. When we further examine our data, we discover that the VW Diesel is the only diesel in our data and that the data for the Plymouth Arrow were entered incorrectly into the computer. No such simple explanations were found for the Plymouth Champ and Peugeot 604.

◁

Standardized and studentized residuals

The terms standardized and studentized residuals have meant different things to different authors. In Stata, `predict` defines the standardized residual as $\widehat{e}_i = e_i/(s\sqrt{1-h_i})$ and the studentized residual as $r_i = e_i/(s_{(i)}\sqrt{1-h_i})$, where $s_{(i)}$ is the root mean squared error of a regression with the ith observation removed. Stata's definition of the studentized residual is the same as the one given in Bollen and Jackman (1990, 264) and is what Chatterjee and Hadi (1988, 74) call the "externally studentized" residual. Stata's "standardized" residual is the same as what Chatterjee and Hadi (1988, 74) call the "internally studentized" residual.

Standardized and studentized residuals are attempts to adjust residuals for their standard errors. Although the ϵ_i theoretical residuals are homoskedastic by assumption (i.e., they all have the same variance), the calculated e_i are not. In fact,

$$\mathrm{Var}(e_i) = \sigma^2(1-h_i)$$

where h_i are the leverage measures obtained from the diagonal elements of hat matrix. Thus observations with the greatest leverage have corresponding residuals with the smallest variance.

Standardized residuals use the root mean squared error of the regression for σ. Studentized residuals use the root mean squared error of a regression omitting the observation in question for σ. In general, studentized residuals are preferable to standardized residuals for purposes of outlier identification. Studentized residuals can be interpreted as the t statistic for testing the significance of a dummy variable equal to 1 in the observation in question and 0 elsewhere (Belsley, Kuh, and Welsch 1980). Such a dummy variable would effectively absorb the observation and so remove its influence in determining the other coefficients in the model. Caution must be exercised here, however, because of the simultaneous testing problem. You cannot simply list the residuals that would be individually significant at the 5% level—their joint significance would be far less (their joint significance *level* would be far greater).

▷ Example 11: standardized and studentized residuals

In the opening remarks for this entry, we distinguished residuals from leverage and speculated on the impact of an observation with a small residual but large leverage. If we had adjusted the residuals for their standard errors, however, the adjusted residual would have been (relatively) larger and perhaps large enough so that we could simply examine the adjusted residuals. Taking our `price` on `weight` and `foreign##c.mpg` model, we can obtain the in-sample standardized and studentized residuals by typing

```
. predict esta if e(sample), rstandard
. predict estu if e(sample), rstudent
```

In *L-R plots*, we discovered that the VW Diesel has the highest leverage in our data, but a corresponding small residual. The standardized and studentized residuals for the VW Diesel are

```
. list make price esta estu if make=="VW Diesel"
```

	make	price	esta	estu
74.	VW Diesel	5,397	.6142691	.6114758

The studentized residual of 0.611 can be interpreted as the t statistic for including a dummy variable for VW Diesel in our regression. Such a variable would not be significant.

◁

DFITS, Cook's Distance, and Welsch Distance

DFITS (Welsch and Kuh 1977), Cook's Distance (Cook 1977), and Welsch Distance (Welsch 1982) are three attempts to summarize the information in the leverage versus residual-squared plot into one statistic. That is, the goal is to create an index that is affected by the size of the residuals—outliers—and the size of h_i—leverage. Viewed mechanically, one way to write DFITS (Bollen and Jackman 1990, 265) is

$$\text{DFITS}_i = r_i\sqrt{\frac{h_i}{1-h_i}}$$

where r_i are the studentized residuals. Thus large residuals increase the value of DFITS, as do large values of h_i. Viewed more traditionally, DFITS is a scaled difference between predicted values for the ith case when the regression is fit with and without the ith observation, hence the name.

The mechanical relationship between DFITS and Cook's Distance, D_i (Bollen and Jackman 1990, 266), is

$$D_i = \frac{1}{k}\frac{s_{(i)}^2}{s^2}\text{DFITS}_i^2$$

where k is the number of variables (including the constant) in the regression, s is the root mean squared error of the regression, and $s_{(i)}$ is the root mean squared error when the ith observation is omitted. Viewed more traditionally, D_i is a scaled measure of the distance between the coefficient vectors when the ith observation is omitted.

The mechanical relationship between DFITS and Welsch's Distance, W_i (Chatterjee and Hadi 1988, 123), is

$$W_i = \text{DFITS}_i\sqrt{\frac{n-1}{1-h_i}}$$

The interpretation of W_i is more difficult, as it is based on the empirical influence curve. Although DFITS and Cook's distance are similar, the Welsch distance measure includes another normalization by leverage.

Belsley, Kuh, and Welsch (1980, 28) suggest that DFITS values greater than $2\sqrt{k/n}$ deserve more investigation, and so values of Cook's distance greater than $4/n$ should also be examined (Bollen and Jackman 1990, 265–266). Through similar logic, the cutoff for Welsch distance is approximately $3\sqrt{k}$ (Chatterjee and Hadi 1988, 124).

▷ Example 12: DFITS influence measure

Using our model of `price` on `weight` and `foreign##c.mpg`, we can obtain the DFITS influence measure:

```
. use http://www.stata-press.com/data/r11/auto, clear
(1978 Automobile Data)
. regress price weight foreign##c.mpg
  (output omitted )
. predict e if e(sample), resid
. predict dfits, dfits
```

We did not specify `if e(sample)` in computing the DFITS statistic. DFITS is available only over the estimation sample, so specifying `if e(sample)` would have been redundant. It would have done no harm, but it would not have changed the results.

Our model has $k = 5$ independent variables (k includes the constant) and $n = 74$ observations; following the $2\sqrt{k/n}$ cutoff advice, we type

```
. list make price e dfits if abs(dfits) > 2*sqrt(5/74), divider
```

	make	price	e	dfits
12.	Cad. Eldorado	14,500	7271.96	.9564455
13.	Cad. Seville	15,906	5036.348	1.356619
24.	Ford Fiesta	4,389	3164.872	.5724172
27.	Linc. Mark V	13,594	3109.193	.5200413
28.	Linc. Versailles	13,466	6560.912	.8760136
42.	Plym. Arrow	4,647	-3312.968	-.9384231

We calculate Cook's distance and list the observations greater than the suggested $4/n$ cutoff:

(*Continued on next page*)

```
. predict cooksd if e(sample), cooksd
. list make price e cooksd if cooksd > 4/74, divider
```

	make	price	e	cooksd
12.	Cad. Eldorado	14,500	7271.96	.1492676
13.	Cad. Seville	15,906	5036.348	.3328515
24.	Ford Fiesta	4,389	3164.872	.0638815
28.	Linc. Versailles	13,466	6560.912	.1308004
42.	Plym. Arrow	4,647	-3312.968	.1700736

Here we used `if e(sample)` because Cook's distance is not restricted to the estimation sample by default. It is worth comparing this list with the preceding one.

Finally, we use Welsch distance and the suggested $3\sqrt{k}$ cutoff:

```
. predict wd, welsch
. list make price e wd if abs(wd) > 3*sqrt(5), divider
```

	make	price	e	wd
12.	Cad. Eldorado	14,500	7271.96	8.394372
13.	Cad. Seville	15,906	5036.348	12.81125
28.	Linc. Versailles	13,466	6560.912	7.703005
42.	Plym. Arrow	4,647	-3312.968	-8.981481

Here we did not need to specify `if e(sample)` because `welsch` automatically restricts the prediction to the estimation sample.

◁

COVRATIO

COVRATIO (Belsley, Kuh, and Welsch 1980) measures the influence of the ith observation by considering the effect on the variance–covariance matrix of the estimates. The measure is the ratio of the determinants of the covariances matrix, with and without the ith observation. The resulting formula is

$$\text{COVRATIO}_i = \frac{1}{1-h_i}\left(\frac{n-k-\widehat{e}_i^2}{n-k-1}\right)^k$$

where $\widehat{e}_i$ is the standardized residual.

For noninfluential observations, the value of COVRATIO is approximately 1. Large values of the residuals or large values of leverage will cause deviations from 1, although if both are large, COVRATIO may tend back toward 1 and therefore not identify such observations (Chatterjee and Hadi 1988, 139).

Belsley, Kuh, and Welsch (1980) suggest that observations for which

$$|\text{COVRATIO}_i - 1| \geq \frac{3k}{n}$$

are worthy of further examination.

▷ Example 13: COVRATIO influence measure

Using our model of `price` on `weight` and `foreign##c.mpg`, we can obtain the COVRATIO measure and list the observations outside the suggested cutoff by typing

```
. predict covr, covratio
. list make price e covr if abs(covr-1) >= 3*5/74, divider
```

	make	price	e	covr
12.	Cad. Eldorado	14,500	7271.96	.3814242
13.	Cad. Seville	15,906	5036.348	.7386969
28.	Linc. Versailles	13,466	6560.912	.4761695
43.	Plym. Champ	4,425	1621.747	1.27782
53.	Audi 5000	9,690	591.2883	1.206842
57.	Datsun 210	4,589	19.81829	1.284801
64.	Peugeot 604	12,990	1037.184	1.348219
66.	Subaru	3,798	-909.5894	1.264677
71.	VW Diesel	5,397	999.7209	1.630653
74.	Volvo 260	11,995	1327.668	1.211888

The `covratio` option automatically restricts the prediction to the estimation sample.

◁

DFBETAs

DFBETAs are perhaps the most direct influence measure of interest to model builders. DFBETAs focus on one coefficient and measure the difference between the regression coefficient when the ith observation is included and excluded, the difference being scaled by the estimated standard error of the coefficient. Belsley, Kuh, and Welsch (1980, 28) suggest observations with $|\text{DFBETA}_i| > 2/\sqrt{n}$ as deserving special attention, but it is also common practice to use 1 (Bollen and Jackman 1990, 267), meaning that the observation shifted the estimate at least one standard error.

▷ Example 14: DFBETAs influence measure; the dfbeta() option

Using our model of `price` on `weight` and `foreign##c.mpg`, let's first ask which observations have the greatest impact on the determination of the coefficient on `1.foreign`. We will use the suggested $2/\sqrt{n}$ cutoff:

```
. sort foreign make
. predict dfor, dfbeta(1.foreign)
. list make price foreign dfor if abs(dfor) > 2/sqrt(74), divider
```

	make	price	foreign	dfor
12.	Cad. Eldorado	14,500	Domestic	-.5290519
13.	Cad. Seville	15,906	Domestic	.8243419
28.	Linc. Versailles	13,466	Domestic	-.5283729
42.	Plym. Arrow	4,647	Domestic	-.6622424
43.	Plym. Champ	4,425	Domestic	.2371104
64.	Peugeot 604	12,990	Foreign	.2552032
69.	Toyota Corona	5,719	Foreign	-.256431

The Cadillac Seville shifted the coefficient on `1.foreign` 0.82 standard deviations!

Now let us ask which observations have the greatest effect on the mpg coefficient:

```
. predict dmpg, dfbeta(mpg)
. list make price mpg dmpg if abs(dmpg) > 2/sqrt(74), divider
```

	make	price	mpg	dmpg
12.	Cad. Eldorado	14,500	14	-.5970351
13.	Cad. Seville	15,906	21	1.134269
28.	Linc. Versailles	13,466	14	-.6069287
42.	Plym. Arrow	4,647	28	-.8925859
43.	Plym. Champ	4,425	34	.3186909

Once again we see the Cadillac Seville heading the list, indicating that our regression results may be dominated by this one car.

◁

▷ Example 15: DFBETAs influence measure; the dfbeta command

We can use `predict, dfbeta()` or the `dfbeta` command to generate the DFBETAs. `dfbeta` makes up names for the new variables automatically and, without arguments, generates the DFBETAs for all the variables in the regression:

```
. dfbeta
                          _dfbeta_1: dfbeta(weight)
                          _dfbeta_2: dfbeta(1.foreign)
                          _dfbeta_3: dfbeta(mpg)
                          _dfbeta_4: dfbeta(1.foreign#c.mpg)
```

`dfbeta` created four new variables in our dataset: `_dfbeta_1`, containing the DFBETAs for `weight`; `_dfbeta_2`, containing the DFBETAs for `mpg`; and so on. Had we wanted only the DFBETAs for `mpg` and `weight`, we might have typed

```
. dfbeta mpg weight
                          _dfbeta_5: dfbeta(weight)
                          _dfbeta_6: dfbeta(mpg)
```

In the example above, we typed `dfbeta mpg weight` instead of `dfbeta`; if we had typed `dfbeta` followed by `dfbeta mpg weight`, here is what would have happened:

```
. dfbeta
                          _dfbeta_7: dfbeta(weight)
                          _dfbeta_8: dfbeta(1.foreign)
                          _dfbeta_9: dfbeta(mpg)
                         _dfbeta_10: dfbeta(1.foreign#c.mpg)
. dfbeta mpg weight
                         _dfbeta_11: dfbeta(weight)
                         _dfbeta_12: dfbeta(mpg)
```

`dfbeta` would have made up different names for the new variables. `dfbeta` never replaces existing variables—it instead makes up a different name, so we need to pay attention to `dfbeta`'s output.

◁

Formal tests for violations of assumptions

This section introduces some regression diagnostic commands that are designed to test for certain violations that `rvfplot` less formally attempts to detect. `estat ovtest` provides Ramsey's test for omitted variables—a pattern in the residuals. `estat hettest` provides a test for heteroskedasticity—the increasing or decreasing variation in the residuals with fitted values, with respect to the explanatory variables, or with respect to yet other variables. The score test implemented in `estat hettest` (Breusch and Pagan 1979; Cook and Weisberg 1983) performs a score test of the null hypothesis that $b = 0$ against the alternative hypothesis of multiplicative heteroskedasticity. `estat szroeter` provides a rank test for heteroskedasticity, which is an alternative to the score test computed by `estat hettest`. Finally, `estat imtest` computes an information matrix test, including an orthogonal decomposition into tests for heteroskedasticity, skewness, and kurtosis (Cameron and Trivedi 1990). The heteroskedasticity test computed by `estat imtest` is similar to the general test for heteroskedasticity that was proposed by White (1980). Cameron and Trivedi (2009, chap. 3) discuss most of these tests and provides more examples.

▷ Example 16: estat ovtest, estat hettest, estat szroeter, and estat imtest

We run these commands just mentioned on our model:

```
. estat ovtest

Ramsey RESET test using powers of the fitted values of price
       Ho:  model has no omitted variables
                  F(3, 66) =      7.77
                  Prob > F =      0.0002

. estat hettest

Breusch-Pagan / Cook-Weisberg tests for heteroskedasticity
         Ho: Constant variance
         variables: fitted values of price

         chi2(1)      =     6.50
         Prob > chi2  =   0.0108
```

Testing for heteroskedasticity in the right-hand-side variables is requested by specifying the `rhs` option. By specifying the `mtest(bonferroni)` option, we request that tests be conducted for each of the variables, with a Bonferroni adjustment for the p-values to accommodate our testing multiple hypotheses.

```
. estat hettest, rhs mtest(bonf)

Breusch-Pagan / Cook-Weisberg test for heteroskedasticity
         Ho: Constant variance
```

Variable	chi2	df	p
weight	15.24	1	0.0004 #
1.foreign	6.15	1	0.0525 #
mpg	9.04	1	0.0106 #
foreign# c.mpg 1	6.02	1	0.0566 #
simultaneous	15.60	4	0.0036

 # Bonferroni adjusted p-values

```
. estat szroeter, rhs mtest(holm)
Szroeter's test for homoskedasticity

    Ho: variance constant
    Ha: variance monotonic in variable

    Variable      chi2   df      p

      weight     17.07    1   0.0001 #
   1.foreign      6.15    1   0.0131 #
         mpg     11.45    1   0.0021 #
    foreign#
       c.mpg
           1      6.17    1   0.0260 #

               # Holm adjusted p-values
```

Finally, we request the information matrix test, which is a conditional moments test with second-, third-, and fourth-order moment conditions.

```
. estat imtest
Cameron & Trivedi's decomposition of IM-test

              Source       chi2     df       p

  Heteroskedasticity      18.86     10   0.0420
            Skewness      11.69      4   0.0198
            Kurtosis       2.33      1   0.1273

               Total      32.87     15   0.0049
```

We find evidence for omitted variables, heteroskedasticity, and nonnormal skewness.

So, why bother with the various graphical commands when the tests seem so much easier to interpret? In part, it is a matter of taste: both are designed to uncover the same problem, and both are, in fact, going about it in similar ways. One is based on a formal calculation, whereas the other is based on personal judgment in evaluating a graph. On the other hand, the tests are seeking evidence of specific problems, whereas judgment is more general. The careful analyst will use both.

We performed the omitted-variable test first. Omitted variables are a more serious problem than heteroskedasticity or the violations of higher moment conditions tested by `estat imtest`. If this were not a manual, having found evidence of omitted variables, we would never have run the `estat hettest`, `estat szroeter`, and `estat imtest` commands, at least not until we solved the omitted-variable problem.

◁

❑ Technical note

`estat ovtest` and `estat hettest` both perform two flavors of their respective tests. By default, `estat ovtest` looks for evidence of omitted variables by fitting the original model augmented by $\widehat{y}^2$, $\widehat{y}^3$, and $\widehat{y}^4$, which are the fitted values from the original model. Under the assumption of no misspecification, the coefficients on the powers of the fitted values will be zero. With the `rhs` option, `estat ovtest` instead augments the original model with powers (second through fourth) of the explanatory variables (except for dummy variables).

estat hettest, by default, looks for heteroskedasticity by modeling the variance as a function of the fitted values. If, however, we specify a variable or variables, the variance will be modeled as a function of the specified variables. In our example, if we had, a priori, some reason to suspect heteroskedasticity and that the heteroskedasticity is a function of a car's weight, then using a test that focuses on weight would be more powerful than the more general tests such as White's test or the first term in the Cameron–Trivedi decomposition test.

estat hettest, by default, computes the original Breusch–Pagan/Cook–Weisberg test, which includes the assumption of normally distributed errors. Koenker (1981) derived an $N * R^2$ version of this test that drops the normality assumption. Wooldridge (2009) gives an F statistic version that does not require the normality assumption.

❑

Variance inflation factors

Problems arise in regression when the predictors are highly correlated. In this situation, there may be a significant change in the regression coefficients if you add or delete an independent variable. The estimated standard errors of the fitted coefficients are inflated, or the estimated coefficients may not be statistically significant even though a statistical relation exists between the dependent and independent variables.

Data analysts rely on these facts to check informally for the presence of multicollinearity. estat vif, another command for use after regress, calculates the variance inflation factors and tolerances for each of the independent variables.

The output shows the variance inflation factors together with their reciprocals. Some analysts compare the reciprocals with a predetermined tolerance. In the comparison, if the reciprocal of the VIF is smaller than the tolerance, the associated predictor variable is removed from the regression model. However, most analysts rely on informal rules of thumb applied to the VIF; see Chatterjee and Hadi (2006). According to these rules, there is evidence of multicollinearity if

1. The largest VIF is greater than 10 (some choose a more conservative threshold value of 30).
2. The mean of all the VIFs is considerably larger than 1.

▷ Example 17: estat vif

We examine a regression model fit using the ubiquitous automobile dataset:

```
. regress price mpg rep78 trunk headroom length turn displ gear_ratio
      Source |       SS       df       MS              Number of obs =      69
-------------+------------------------------           F(  8,    60) =    6.33
       Model |   264102049     8  33012756.2           Prob > F      =  0.0000
    Residual |   312694909    60  5211581.82           R-squared     =  0.4579
-------------+------------------------------           Adj R-squared =  0.3856
       Total |   576796959    68  8482308.22           Root MSE      =  2282.9

------------------------------------------------------------------------------
       price |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         mpg |    -144.84   82.12751    -1.76   0.083    -309.1195    19.43948
       rep78 |   727.5783   337.6107     2.16   0.035     52.25638      1402.9
       trunk |   44.02061    108.141     0.41   0.685    -172.2935    260.3347
    headroom |  -807.0996   435.5802    -1.85   0.069     -1678.39    64.19061
      length |  -8.688914   34.89848    -0.25   0.804    -78.49626    61.11843
        turn |  -177.9064   137.3455    -1.30   0.200    -452.6383    96.82551
displacement |   30.73146   7.576952     4.06   0.000      15.5753    45.88762
  gear_ratio |   1500.119   1110.959     1.35   0.182    -722.1303    3722.368
       _cons |   6691.976   7457.906     0.90   0.373    -8226.057    21610.01
------------------------------------------------------------------------------
```

```
. estat vif

    Variable |       VIF       1/VIF  
-------------+----------------------
      length |      8.22    0.121614
displacement |      6.50    0.153860
        turn |      4.85    0.205997
  gear_ratio |      3.45    0.290068
         mpg |      3.03    0.330171
       trunk |      2.88    0.347444
    headroom |      1.80    0.554917
       rep78 |      1.46    0.686147
-------------+----------------------
    Mean VIF |      4.02
```

The results are mixed. Although we have no VIFs greater than 10, the mean VIF is greater than 1, though not considerably so. We could continue the investigation of collinearity, but given that other authors advise that collinearity is a problem only when VIFs exist that are greater than 30 (contradicting our rule above), we will not do so here.

◁

▷ Example 18: estat vif, with strong evidence of multicollinearity

This example comes from a dataset described in Kutner, Nachtsheim, and Neter (2004, 257) that examines body fat as modeled by caliper measurements on the triceps, midarm, and thigh.

```
. use http://www.stata-press.com/data/r11/bodyfat, clear
(Body Fat)

. regress bodyfat tricep thigh midarm

      Source |       SS       df       MS              Number of obs =      20
-------------+------------------------------           F(  3,    16) =   21.52
       Model |  396.984607     3  132.328202           Prob > F      =  0.0000
    Residual |  98.4049068    16  6.15030667           R-squared     =  0.8014
-------------+------------------------------           Adj R-squared =  0.7641
       Total |  495.389513    19  26.0731323           Root MSE      =    2.48

------------------------------------------------------------------------------
     bodyfat |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     triceps |   4.334085   3.015511     1.44   0.170    -2.058512    10.72668
       thigh |  -2.856842   2.582015    -1.11   0.285    -8.330468    2.616785
      midarm |  -2.186056   1.595499    -1.37   0.190    -5.568362     1.19625
       _cons |   117.0844   99.78238     1.17   0.258    -94.44474    328.6136
------------------------------------------------------------------------------

. estat vif

    Variable |       VIF       1/VIF  
-------------+----------------------
     triceps |    708.84    0.001411
       thigh |    564.34    0.001772
      midarm |    104.61    0.009560
-------------+----------------------
    Mean VIF |    459.26
```

Here we see strong evidence of multicollinearity in our model. More investigation reveals that the measurements on the thigh and the triceps are highly correlated:

```
. corr triceps thigh midarm
(obs=20)

             |  triceps    thigh   midarm
-------------+---------------------------
     triceps |   1.0000
       thigh |   0.9238   1.0000
      midarm |   0.4578   0.0847   1.0000
```

If we remove the predictor `tricep` from the model (because it had the highest VIF), we get

```
. regress bodyfat thigh midarm

      Source |       SS       df       MS              Number of obs =      20
-------------+------------------------------           F(  2,    17) =   29.40
       Model |  384.279748     2  192.139874           Prob > F      =  0.0000
    Residual |  111.109765    17  6.53586854           R-squared     =  0.7757
-------------+------------------------------           Adj R-squared =  0.7493
       Total |  495.389513    19  26.0731323           Root MSE      =  2.5565

------------------------------------------------------------------------------
     bodyfat |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
       thigh |   .8508818   .1124482     7.57   0.000     .6136367    1.088127
      midarm |   .0960295   .1613927     0.60   0.560    -.2444792    .4365383
       _cons |  -25.99696    6.99732    -3.72   0.002    -40.76001    -11.2339
------------------------------------------------------------------------------

. estat vif

    Variable |       VIF       1/VIF
-------------+----------------------
      midarm |      1.01    0.992831
       thigh |      1.01    0.992831
-------------+----------------------
    Mean VIF |      1.01
```

Note how the coefficients change and how the estimated standard errors for each of the regression coefficients become much smaller. The calculated value of R^2 for the overall regression for the subset model does not appreciably decline when we remove the correlated predictor. Removing an independent variable from the model is one way to deal with multicollinearity. Other methods include ridge regression, weighted least squares, and restricting the use of the fitted model to data that follow the same pattern of multicollinearity. In economic studies, it is sometimes possible to estimate the regression coefficients from different subsets of the data by using cross-section and time series.

◁

All examples above demonstrated the use of centered VIFs. As pointed out by Belsley (1991), the centered VIFs may fail to discover collinearity involving the constant term. One solution is to use the uncentered VIFs instead. According to the definition of the uncentered VIFs, the constant is viewed as a legitimate explanatory variable in a regression model, which allows one to obtain the VIF value for the constant term.

▷ Example 19: estat vif, with strong evidence of collinearity with the constant term

Consider the extreme example in which one of the regressors is highly correlated with the constant. We simulate the data and examine both centered and uncentered VIF diagnostics after fitted regression model as follows.

```
. use http://www.stata-press.com/data/r11/extreme_collin
. summarize
  (output omitted )
. regress y one x z
      Source |       SS       df       MS              Number of obs =     100
-------------+------------------------------           F(  3,    96) = 2710.27
       Model |  223801.985     3  74600.6617           Prob > F      =  0.0000
    Residual |  2642.42124    96  27.5252213           R-squared     =  0.9883
-------------+------------------------------           Adj R-squared =  0.9880
       Total |  226444.406    99  2287.31723           Root MSE      =  5.2464

------------------------------------------------------------------------------
           y |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         one |  -3.278582    10.5621    -0.31   0.757    -24.24419    17.68702
           x |   2.038696   .0242673    84.01   0.000     1.990526    2.086866
           z |   4.863137   .2681036    18.14   0.000     4.330956    5.395319
       _cons |   9.760075   10.50935     0.93   0.355    -11.10082    30.62097
------------------------------------------------------------------------------

. estat vif
    Variable |       VIF       1/VIF
-------------+----------------------
           z |      1.03    0.968488
           x |      1.03    0.971307
         one |      1.00    0.995425
-------------+----------------------
    Mean VIF |      1.02
. estat vif, uncentered
    Variable |       VIF       1/VIF
-------------+----------------------
         one |    402.94    0.002482
   intercept |    401.26    0.002492
           z |      2.93    0.341609
           x |      1.13    0.888705
-------------+----------------------
    Mean VIF |    202.06
```

According to the values of the centered VIFs (1.03, 1.03, 1.00), no harmful collinearity is detected in the model. However, by the construction of these simulated data, we know that `one` is highly collinear with the constant term. As such, the large values of uncentered VIFs for `one` (402.94) and `intercept` (401.26) reveal high collinearity of the variable `one` with the constant term.

◁

Saved results

`estat hettest` saves the following results for the (multivariate) score test in `r()`:

Scalars

`r(chi2)`	χ^2 test statistic
`r(df)`	#df for the asymptotic χ^2 distribution under H_0
`r(p)`	p-value

`estat hettest, fstat` saves results for the (multivariate) score test in `r()`:

Scalars

`r(F)`	test statistic
`r(df_m)`	#df of the test for the F distribution under H_0
`r(df_r)`	#df of the residuals for the F distribution under H_0
`r(p)`	p-value

`estat hettest` (if `mtest` is specified) and `estat szroeter` save the following in `r()`:

Matrices

`r(mtest)` a matrix of test results, with rows corresponding to the univariate tests

`mtest[.,1]`	χ^2 test statistic
`mtest[.,2]`	#df
`mtest[.,3]`	unadjusted p-value
`mtest[.,4]`	adjusted p-value (if an `mtest()` adjustment method is specified)

Macros

`r(mtmethod)` adjustment method for p-values

`estat imtest` saves the following in `r()`:

Scalars

`r(chi2_t)`	IM-test statistic $\big(=$ `r(chi2_h)` $+$ `r(chi2_s)` $+$ `r(chi2_k)`$\big)$
`r(df_t)`	df for limiting χ^2 distribution under H_0 $\big(=$ `r(df_h)` $+$ `r(df_s)` $+$ `r(df_k)`$\big)$
`r(chi2_h)`	heteroskedasticity test statistic
`r(df_h)`	df for limiting χ^2 distribution under H_0
`r(chi2_s)`	skewness test statistic
`r(df_s)`	df for limiting χ^2 distribution under H_0
`r(chi2_k)`	kurtosis test statistic
`r(df_k)`	df for limiting χ^2 distribution under H_0
`r(chi2_w)`	White's heteroskedasticity test (if `white` specified)
`r(df_w)`	df for limiting χ^2 distribution under H_0

`estat ovtest` saves the following in `r()`:

Scalars

`r(p)`	two-sided p-value	`r(df)`	degrees of freedom
`r(F)`	F statistic	`r(df_r)`	residual degrees of freedom

Methods and formulas

All regression fit and diagnostic commands are implemented as ado-files.

See Hamilton (2009, chap. 7), Kohler and Kreuter (2009, sec. 8.3), or Baum (2006, chap. 5) for an overview of using Stata to perform regression diagnostics. See Peracchi (2001, chap. 8) for a mathematically rigorous discussion of diagnostics.

Methods and formulas for predict

Assume that you have already fit the regression model

$$\mathbf{y} = \mathbf{X}\mathbf{b} + \mathbf{e}$$

where $\mathbf{X}$ is $n \times k$.

Denote the previously estimated coefficient vector by $\mathbf{b}$ and its estimated variance matrix by $\mathbf{V}$. `predict` works by recalling various aspects of the model, such as $\mathbf{b}$, and combining that information with the data currently in memory. Let $\mathbf{x}_j$ be the jth observation currently in memory, and let s^2 be the mean squared error of the regression.

Let $\mathbf{V} = s^2(\mathbf{X}'\mathbf{X})^{-1}$. Let k be the number of independent variables including the intercept, if any, and let y_j be the observed value of the dependent variable.

The *predicted value* (`xb` option) is defined as $\widehat{y}_j = \mathbf{x}_j\mathbf{b}$.

Let ℓ_j represent a lower bound for an observation j and u_j represent an upper bound. The probability that $y_j|\mathbf{x}_j$ would be observed in the interval (ℓ_j, u_j)—the `pr(`ℓ`,` u`)` option—is

$$P(\ell_j, u_j) = \Pr(\ell_j < \mathbf{x}_j\mathbf{b} + e_j < u_j) = \Phi\left(\frac{u_j - \widehat{y}_j}{s}\right) - \Phi\left(\frac{\ell_j - \widehat{y}_j}{s}\right)$$

where for the `pr(`ℓ`,` u`)`, `e(`ℓ`,` u`)`, and `ystar(`ℓ`,` u`)` options, ℓ_j and u_j can be anywhere in the range $(-\infty, +\infty)$.

The option `e(`ℓ`,` u`)` computes the expected value of $y_j|\mathbf{x}_j$ conditional on $y_j|\mathbf{x}_j$ being in the interval (ℓ_j, u_j), that is, when $y_j|\mathbf{x}_j$ is censored. It can be expressed as

$$E(\ell_j, u_j) = E(\mathbf{x}_j\mathbf{b} + e_j \mid \ell_j < \mathbf{x}_j\mathbf{b} + e_j < u_j) = \widehat{y}_j - s\frac{\phi\left(\frac{u_j - \widehat{y}_j}{s}\right) - \phi\left(\frac{\ell_j - \widehat{y}_j}{s}\right)}{\Phi\left(\frac{u_j - \widehat{y}_j}{s}\right) - \Phi\left(\frac{\ell_j - \widehat{y}_j}{s}\right)}$$

where ϕ is the normal density and Φ is the cumulative normal.

You can also compute `ystar(`ℓ, u`)`—the expected value of $y_j|\mathbf{x}_j$, where y_j is assumed truncated at ℓ_j and u_j:

$$y_j^* = \begin{cases} \ell_j & \text{if } \mathbf{x}_j\mathbf{b} + e_j \le \ell_j \\ \mathbf{x}_j\mathbf{b} + u & \text{if } \ell_j < \mathbf{x}_j\mathbf{b} + e_j < u_j \\ u_j & \text{if } \mathbf{x}_j\mathbf{b} + e_j \ge u_j \end{cases}$$

This computation can be expressed in several ways, but the most intuitive formulation involves a combination of the two statistics just defined:

$$y_j^* = P(-\infty, \ell_j)\ell_j + P(\ell_j, u_j)E(\ell_j, u_j) + P(u_j, +\infty)u_j$$

A diagonal element of the projection matrix (`hat`) or (`leverage`) is given by

$$h_j = \mathbf{x}_j(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x}_j'$$

The *standard error of the prediction* (the `stdp` option) is defined as $s_{p_j} = \sqrt{\mathbf{x}_j\mathbf{V}\mathbf{x}_j'}$ and can also be written as $s_{p_j} = s\sqrt{h_j}$.

The *standard error of the forecast* (`stdf`) is defined as $s_{f_j} = s\sqrt{1+h_j}$.

The *standard error of the residual* (`stdr`) is defined as $s_{r_j} = s\sqrt{1-h_j}$.

The *residuals* (`residuals`) are defined as $\widehat{e}_j = y_j - \widehat{y}_j$.

The *standardized residuals* (`rstandard`) are defined as $\widehat{e}_{s_j} = \widehat{e}_j / s_{r_j}$.

The *studentized residuals* (`rstudent`) are defined as

$$r_j = \frac{\widehat{e}_j}{s_{(j)}\sqrt{1-h_j}}$$

where $s_{(j)}$ represents the root mean squared error with the jth observation removed, which is given by

$$s_{(j)}^2 = \frac{s^2(T-k)}{T-k-1} - \frac{\widehat{e}_j^2}{(T-k-1)(1-h_j)}$$

Cook's D (`cooksd`) is given by

$$D_j = \frac{\widehat{e}_{s_j}^2 (s_{p_j}/s_{r_j})^2}{k} = \frac{h_j \widehat{e}_j^2}{ks^2(1-h_j)^2}$$

DFITS (`dfits`) is given by

$$\text{DFITS}_j = r_j \sqrt{\frac{h_j}{1-h_j}}$$

Welsch distance (`welsch`) is given by

$$W_j = \frac{r_j \sqrt{h_j(n-1)}}{1-h_j}$$

COVRATIO (`covratio`) is given by

$$\text{COVRATIO}_j = \frac{1}{1-h_j}\left(\frac{n-k-\widehat{e}_j^2}{n-k-1}\right)^k$$

The DFBETAs (`dfbeta`) for a particular regressor x_i are given by

$$\text{DFBETA}_j = \frac{r_j u_j}{\sqrt{U^2(1-h_j)}}$$

where u_j are the residuals obtained from a regression of x_j on the remaining x's and $U^2 = \sum_j u_j^2$.

The omitted-variable test (Ramsey 1969) reported by `ovtest` fits the regression $y_i = \mathbf{x}_i\mathbf{b} + \mathbf{z}_i\mathbf{t} + u_i$ and then performs a standard F test of $\mathbf{t} = \mathbf{0}$. The default test uses $\mathbf{z}_i = (\widehat{y}_i^2, \widehat{y}_i^3, \widehat{y}_i^4)$. If `rhs` is specified, $\mathbf{z}_i = (x_{1i}^2, x_{1i}^3, x_{1i}^4, x_{2i}^2, \ldots, x_{mi}^4)$. In either case, the variables are normalized to have minimum 0 and maximum 1 before powers are calculated.

The test for heteroskedasticity (Breusch and Pagan 1979; Cook and Weisberg 1983) models $\mathrm{Var}(e_i) = \sigma^2 \exp(\mathbf{zt})$, where $\mathbf{z}$ is a variable list specified by the user, the list of right-hand-side variables, or the fitted values $\mathbf{x}\widehat{\beta}$. The test is of $\mathbf{t} = \mathbf{0}$. Mechanically, `estat hettest` fits the augmented regression $\widehat{e}_i^2/\widehat{\sigma}^2 = a + \mathbf{z}_i\mathbf{t} + v_i$.

The original Breusch–Pagan/Cook–Weisberg version of the test assumes that the e_i are normally distributed under the null hypothesis which implies that the score test statistic S is equal to the model sum of squares from the augmented regression divided by 2. Under the null hypothesis, S has the χ^2 distribution with m degrees of freedom, where m is the number of columns of $\mathbf{z}$.

Koenker (1981) derived a score test of the null hypothesis that $\mathbf{t} = \mathbf{0}$ under the assumption that the e_i are independent and identically distributed (i.i.d.). Koenker showed that $S = N * R^2$ has a large-sample χ^2 distribution with m degrees of freedom, where N is the number of observations and R^2 is the R-squared in the augmented regression and m is the number of columns of $\mathbf{z}$. `estat hettest, iid` produces this version of the test.

Wooldridge (2009) showed that an F test of $\mathbf{t} = \mathbf{0}$ in the augmented regression can also be used under the assumption that the e_i are i.i.d. `estat hettest, fstat` produces this version of the test.

Szroeter's class of tests for homoskedasticity against the alternative that the residual variance increases in some variable x is defined in terms of

$$H = \frac{\sum_{i=1}^n h(x_i)e_i^2}{\sum_{i=1}^n e_i^2}$$

where $h(x)$ is some weight function that increases in x (Szroeter 1978). H is a weighted average of the $h(x)$, with the squared residuals serving as weights. Under homoskedasticity, H should be approximately equal to the unweighted average of $h(x)$. Large values of H suggest that e_i^2 tends to be large where $h(x)$ is large; i.e., the variance indeed increases in x, whereas small values of H suggest that the variance actually decreases in x. `estat szroeter` uses $h(x_i) = \mathrm{rank}(x_i \text{ in } x_1 \ldots x_n)$; see Judge et al. [1985, 452] for details. `estat szroeter` displays a normalized version of H,

$$Q = \sqrt{\frac{6n}{n^2-1}}H$$

which is approximately $N(0,1)$ distributed under the null (homoskedasticity).

`estat hettest` and `estat szroeter` provide adjustments of p-values for multiple testing. The supported methods are described in [R] **test**.

`estat imtest` performs the information matrix test for the regression model, as well as an orthogonal decomposition into tests for heteroskedasticity δ_1, nonnormal skewness δ_2, and nonnormal kurtosis δ_3 (Cameron and Trivedi 1990; Long and Trivedi 1992). The decomposition is obtained via three auxiliary regressions. Let e be the regression residuals, $\widehat{\sigma}^2$ be the maximum likelihood estimate of σ^2 in the regression, n be the number of observations, X be the set of k variables specified with `estat imtest`, and R^2_{un} be the uncentered R^2 from a regression. δ_1 is obtained as nR^2_{un} from a regression of $e^2 - \widehat{\sigma}^2$ on the cross-products of the variables in X. δ_2 is computed as nR^2_{un} from a regression of $e^3 - 3\widehat{\sigma}^2 e$ on X. Finally, δ_3 is obtained as nR^2_{un} from a regression of $e^4 - 6\widehat{\sigma}^2 e^2 - 3\widehat{\sigma}^4$ on X. δ_1, δ_2, and δ_3 are asymptotically χ^2 distributed with $1/2k(k+1)$, K, and 1 degree of freedom. The information test statistic $\delta = \delta_1 + \delta_2 + \delta_3$ is asymptotically χ^2 distributed with $1/2k(k+3)$ degrees of freedom. White's test for heteroskedasticity is computed as nR^2 from a regression of $\widehat{u}^2$ on X and the cross-products of the variables in X. This test statistic is usually close to δ_1.

`estat vif` calculates the centered variance inflation factor (VIF_c) (Chatterjee and Hadi 2006, 235–239) for x_j, given by

$$\mathrm{VIF}_c(x_j) = \frac{1}{1 - \widehat{R}_j^2}$$

where $\widehat{R}_j^2$ is the square of the centered multiple correlation coefficient that results when x_j is regressed with intercept against all the other explanatory variables.

The uncentered variance inflation factor (VIF_{uc}) (Belsley 1991, 28–29) for x_j is given by

$$\mathrm{VIF}_{uc}(x_j) = \frac{1}{1 - \widetilde{R}_j^2}$$

where $\widetilde{R}_j^2$ is the square of the uncentered multiple correlation coefficient that results when x_j is regressed without intercept against all the other explanatory variables including the constant term.

Acknowledgments

`estat ovtest` and `estat hettest` are based on programs originally written by Richard Goldstein (1991, 1992). `estat imtest`, `estat szroeter`, and the current version of `estat hettest` were written by Jeroen Weesie, Department of Sociology, Utrecht University, The Netherlands; `estat imtest` is based in part on code written by J. Scott Long, Department of Sociology, Indiana University.

References

Adkins, L. C., and R. C. Hill. 2008. *Using Stata for Principles of Econometrics*. 3rd ed. Hoboken, NJ: Wiley.

Baum, C. F. 2006. *An Introduction to Modern Econometrics Using Stata*. College Station, TX: Stata Press.

Baum, C. F., N. J. Cox, and V. L. Wiggins. 2000. sg137: Tests for heteroskedasticity in regression error distribution. *Stata Technical Bulletin* 55: 15–17. Reprinted in *Stata Technical Bulletin Reprints*, vol. 10, pp. 147–149. College Station, TX: Stata Press.

Baum, C. F., and V. L. Wiggins. 2000a. sg135: Test for autoregressive conditional heteroskedasticity in regression error distribution. *Stata Technical Bulletin* 55: 13–14. Reprinted in *Stata Technical Bulletin Reprints*, vol. 10, pp. 143–144. College Station, TX: Stata Press.

——. 2000b. sg136: Tests for serial correlation in regression error distribution. *Stata Technical Bulletin* 55: 14–15. Reprinted in *Stata Technical Bulletin Reprints*, vol. 10, pp. 145–147. College Station, TX: Stata Press.

Belsley, D. A. 1991. *Conditional Diagnostics: Collinearity and Weak Data in Regression*. New York: Wiley.

Belsley, D. A., E. Kuh, and R. E. Welsch. 1980. *Regression Diagnostics: Identifying Influential Data and Sources of Collinearity*. New York: Wiley.

Bollen, K. A., and R. W. Jackman. 1990. Regression diagnostics: An expository treatment of outliers and influential cases. In *Modern Methods of Data Analysis*, ed. J. Fox and J. S. Long, 257–291. Newbury Park, CA: Sage.

Breusch, T. S., and A. R. Pagan. 1979. A simple test for heteroscedasticity and random coefficient variation. *Econometrica* 47: 1287–1294.

Cameron, A. C., and P. K. Trivedi. 1990. The information matrix test and its applied alternative hypotheses. Working paper 372, University of California–Davis, Institute of Governmental Affairs.

——. 2009. *Microeconometrics Using Stata*. College Station, TX: Stata Press.

Chatterjee, S., and A. S. Hadi. 1986. Influential observations, high leverage points, and outliers in linear regression. *Statistical Science* 1: 379–393.

——. 1988. *Sensitivity Analysis in Linear Regression*. New York: Wiley.

——. 2006. *Regression Analysis by Example*. 4th ed. New York: Wiley.

Cook, R. D. 1977. Detection of influential observation in linear regression. *Technometrics* 19: 15–18.

Cook, R. D., and S. Weisberg. 1982. *Residuals and Influence in Regression*. New York: Chapman & Hall/CRC.

———. 1983. Diagnostics for heteroscedasticity in regression. *Biometrika* 70: 1–10.

Cox, N. J. 2004. Speaking Stata: Graphing model diagnostics. *Stata Journal* 4: 449–475.

DeMaris, A. 2004. *Regression with Social Data: Modeling Continuous and Limited Response Variables*. Hoboken, NJ: Wiley.

Ezekiel, M. 1924. A method of handling curvilinear correlation for any number of variables. *Journal of the American Statistical Association* 19: 431–453.

Garrett, J. M. 2000. sg157: Predicted values calculated from linear or logistic regression models. *Stata Technical Bulletin* 58: 27–30. Reprinted in *Stata Technical Bulletin Reprints*, vol. 10, pp. 258–261. College Station, TX: Stata Press.

Goldstein, R. 1991. srd5: Ramsey test for heteroscedasticity and omitted variables. *Stata Technical Bulletin* 2: 27. Reprinted in *Stata Technical Bulletin Reprints*, vol. 1, p. 177. College Station, TX: Stata Press.

———. 1992. srd14: Cook–Weisberg test of heteroscedasticity. *Stata Technical Bulletin* 10: 27–28. Reprinted in *Stata Technical Bulletin Reprints*, vol. 2, pp. 183–184. College Station, TX: Stata Press.

Hamilton, L. C. 1992. *Regression with Graphics: A Second Course in Applied Statistics*. Belmont, CA: Duxbury.

———. 2009. *Statistics with Stata (Updated for Version 10)*. Belmont, CA: Brooks/Cole.

Hardin, J. W. 1995. sg32: Variance inflation factors and variance-decomposition proportions. *Stata Technical Bulletin* 24: 17–22. Reprinted in *Stata Technical Bulletin Reprints*, vol. 4, pp. 154–160. College Station, TX: Stata Press.

Hill, R. C., W. E. Griffiths, and G. C. Lim. 2008. *Principles of Econometrics*. 3rd ed. Hoboken, NJ: Wiley.

Hoaglin, D. C., and P. J. Kempthorne. 1986. Comment [on Chatterjee and Hadi 1986]. *Statistical Science* 1: 408–412.

Hoaglin, D. C., and R. E. Welsch. 1978. The hat matrix in regression and ANOVA. *American Statistician* 32: 17–22.

Judge, G. G., W. E. Griffiths, R. C. Hill, H. Lütkepohl, and T.-C. Lee. 1985. *The Theory and Practice of Econometrics*. 2nd ed. New York: Wiley.

Koenker, R. 1981. A note on studentizing a test for heteroskedasticity. *Journal of Econometrics* 17: 107–112.

Kohler, U., and F. Kreuter. 2009. *Data Analysis Using Stata*. 2nd ed. College Station, TX: Stata Press.

Kutner, M. H., C. J. Nachtsheim, and J. Neter. 2004. *Applied Linear Regression Models*. 4th ed. New York: McGraw–Hill/Irwin.

Larsen, W. A., and S. J. McCleary. 1972. The use of partial residual plots in regression analysis. *Technometrics* 14: 781–790.

Long, J. S., and J. Freese. 2000. sg145: Scalar measures of fit for regression models. *Stata Technical Bulletin* 56: 34–40. Reprinted in *Stata Technical Bulletin Reprints*, vol. 10, pp. 197–205. College Station, TX: Stata Press.

Long, J. S., and P. K. Trivedi. 1992. Some specification tests for the linear regression model. *Sociological Methods and Research* 21: 161–204. Reprinted in *Testing Structural Equation Models*, ed. K. A. Bollen and J. S. Long, pp. 66–110. Newbury Park, CA: Sage.

Mallows, C. L. 1986. Augmented partial residuals. *Technometrics* 28: 313–319.

Mosteller, F., and J. W. Tukey. 1977. *Data Analysis and Regression: A Second Course in Statistics*. Reading, MA: Addison–Wesley.

Peracchi, F. 2001. *Econometrics*. Chichester, UK: Wiley.

Ramsey, J. B. 1969. Tests for specification errors in classical linear least-squares regression analysis. *Journal of the Royal Statistical Society, Series B* 31: 350–371.

Ramsey, J. B., and P. Schmidt. 1976. Some further results on the use of OLS and BLUS residuals in specification error tests. *Journal of the American Statistical Association* 71: 389–390.

Rousseeuw, P. J., and A. M. Leroy. 1987. *Robust Regression and Outlier Detection*. New York: Wiley.

Szroeter, J. 1978. A class of parametric tests for heteroscedasticity in linear econometric models. *Econometrica* 46: 1311–1327.

Velleman, P. F. 1986. Comment [on Chatterjee and Hadi 1986]. *Statistical Science* 1: 412–413.

Velleman, P. F., and R. E. Welsch. 1981. Efficient computing of regression diagnostics. *American Statistician* 35: 234–242.

Weesie, J. 2001. sg161: Analysis of the turning point of a quadratic specification. *Stata Technical Bulletin* 60: 18–20. Reprinted in *Stata Technical Bulletin Reprints*, vol. 10, pp. 273–277. College Station, TX: Stata Press.

Weisberg, S. 2005. *Applied Linear Regression*. 3rd ed. New York: Wiley.

Welsch, R. E. 1982. Influence functions and regression diagnostics. In *Modern Data Analysis*, ed. R. L. Launer and A. F. Siegel, 149–169. New York: Academic Press.

——. 1986. Comment [on Chatterjee and Hadi 1986]. *Statistical Science* 1: 403–405.

Welsch, R. E., and E. Kuh. 1977. Linear Regression Diagnostics. Technical Report 923-77, Massachusetts Institute of Technology, Cambridge, MA.

White, H. 1980. A heteroskedasticity-consistent covariance matrix estimator and a direct test for heteroskedasticity. *Econometrica* 48: 817–838.

Wooldridge, J. M. 2009. *Introductory Econometrics: A Modern Approach*. 4th ed. Cincinnati, OH: South-Western.

Also see

[R] **regress** — Linear regression

[R] **regress postestimation time series** — Postestimation tools for regress with time series

[U] **20 Estimation and postestimation commands**