

logistic postestimation — Postestimation tools for logistic

[Postestimation commands](#)
[Remarks and examples](#)
[Also see](#)

[predict](#)
[Methods and formulas](#)

[margins](#)
[References](#)

Postestimation commands

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

Command	Description
<code>estat classification</code>	report various summary statistics, including the classification table
<code>estat gof</code>	Pearson or Hosmer–Lemeshow goodness-of-fit test
<code>lroc</code>	compute area under ROC curve and graph the curve
<code>lsens</code>	graph sensitivity and specificity versus probability cutoff

These commands are not appropriate with `svy` estimation results.

The following standard postestimation commands are also available:

Command	Description
<code>contrast</code>	contrasts and ANOVA-style joint tests of estimates
<code>estat ic</code>	Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC)
<code>estat summarize</code>	summary statistics for the estimation sample
<code>estat vce</code>	variance–covariance matrix of the estimators (VCE)
<code>estat (svy)</code>	postestimation statistics for survey data
<code>estimates</code>	cataloging estimation results
<code>etable</code>	table of estimation results
* <code>forecast</code>	dynamic forecasts and simulations
* <code>hausman</code>	Hausman's specification test
<code>lincom</code>	point estimates, standard errors, testing, and inference for linear combinations of coefficients
<code>linktest</code>	link test for model specification
* <code>lrtest</code>	likelihood-ratio test
<code>margins</code>	marginal means, predictive margins, marginal effects, and average marginal effects
<code>marginsplot</code>	graph the results from margins (profile plots, interaction plots, etc.)
<code>nlcom</code>	point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients
<code>predict</code>	probabilities, influence statistics, residuals, etc.
<code>predictnl</code>	point estimates, standard errors, testing, and inference for generalized predictions

<code>pwcompare</code>	pairwise comparisons of estimates
<code>suest</code>	seemingly unrelated estimation
<code>test</code>	Wald tests of simple and composite linear hypotheses
<code>testnl</code>	Wald tests of nonlinear hypotheses

*`forecast`, `hausman`, and `lrtest` are not appropriate with `svy` estimation results. `forecast` is also not appropriate with `mi` estimation results.

predict

Description for predict

`predict` creates a new variable containing predictions such as probabilities, linear predictions, standard errors, influence statistics, deviance residuals, leverages, sequential numbers, Pearson residuals, and the equation-level score.

Menu for predict

Statistics > Postestimation

Syntax for predict

```
predict [type] newvar [if] [in] [, statistic nooffset rules asif]
```

<i>statistic</i>	Description
<hr/>	
Main	
<code>pr</code>	probability of a positive outcome; the default
<code>xb</code>	linear prediction
<code>stdp</code>	standard error of the prediction
* <code>dbeta</code>	Pregibon (1981) $\Delta \hat{\beta}$ influence statistic
* <code>ddeviance</code>	deviance residual
* <code>dx2</code>	Hosmer, Lemeshow, and Sturdivant (2013) $\Delta \chi^2$ influence statistic
* <code>ddeviance</code>	Hosmer, Lemeshow, and Sturdivant (2013) ΔD influence statistic
* <code>hat</code>	Pregibon (1981) leverage
* <code>number</code>	sequential number of the covariate pattern
* <code>residuals</code>	Pearson residuals; adjusted for number sharing covariate pattern
* <code>rstandard</code>	standardized Pearson residuals; adjusted for number sharing covariate pattern
<code>score</code>	first derivative of the log likelihood with respect to $x_j\beta$

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.

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

Options for predict

Main

`pr`, the default, calculates the probability of a positive outcome.

`xb` calculates the linear prediction.

`stdp` calculates the standard error of the linear prediction.

`dbeta` calculates the Pregibon (1981) $\Delta\hat{\beta}$ influence statistic, a standardized measure of the difference in the coefficient vector that is due to deletion of the observation along with all others that share the same covariate pattern. In Hosmer, Lemeshow, and Sturdivant (2013, 154–155) jargon, this statistic is *M*-asymptotic; that is, it is adjusted for the number of observations that share the same covariate pattern.

`deviance` calculates the deviance residual.

`dx2` calculates the Hosmer, Lemeshow, and Sturdivant (2013, 191) $\Delta\chi^2$ influence statistic, reflecting the decrease in the Pearson χ^2 that is due to deletion of the observation and all others that share the same covariate pattern.

`ddeviance` calculates the Hosmer, Lemeshow, and Sturdivant (2013, 191) ΔD influence statistic, which is the change in the deviance residual that is due to deletion of the observation and all others that share the same covariate pattern.

`hat` calculates the Pregibon (1981) leverage or the diagonal elements of the hat matrix adjusted for the number of observations that share the same covariate pattern.

`number` numbers the covariate patterns—observations with the same covariate pattern have the same `number`. Observations not used in estimation have `number` set to missing. The first covariate pattern is numbered 1, the second 2, and so on.

`residuals` calculates the Pearson residual as given by Hosmer, Lemeshow, and Sturdivant (2013, 155) and adjusted for the number of observations that share the same covariate pattern.

`rstandard` calculates the standardized Pearson residual as given by Hosmer, Lemeshow, and Sturdivant (2013, 191) and adjusted for the number of observations that share the same covariate pattern.

`score` calculates the equation-level score, $\partial \ln L / \partial (\mathbf{x}_j \boldsymbol{\beta})$.

Options

`nooffset` is relevant only if you specified `offset(varname)` for `logistic`. It modifies the calculations made by `predict` so that they ignore the offset variable; the linear prediction is treated as $\mathbf{x}_j \mathbf{b}$ rather than as $\mathbf{x}_j \mathbf{b} + \text{offset}_j$.

`rules` requests that Stata use any rules that were used to identify the model when making the prediction. By default, Stata calculates missing for excluded observations. See [example 1](#) in [\[R\] logit postestimation](#).

`asif` requests that Stata ignore the rules and the exclusion criteria and calculate predictions for all observations possible by using the estimated parameter from the model. See [example 1](#) in [\[R\] logit postestimation](#).

margins

Description for margins

`margins` estimates margins of response for probabilities and linear predictions.

Menu for margins

Statistics > Postestimation

Syntax for margins

```
margins [marginlist] [, options]
```

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

<i>statistic</i>	Description
<code>pr</code>	probability of a positive outcome; the default
<code>xb</code>	linear prediction
<code>stdp</code>	not allowed with <code>margins</code>
<code>dbeta</code>	not allowed with <code>margins</code>
<code>ddeviance</code>	not allowed with <code>margins</code>
<code>dx2</code>	not allowed with <code>margins</code>
<code>ddeviance</code>	not allowed with <code>margins</code>
<code>hat</code>	not allowed with <code>margins</code>
<code>number</code>	not allowed with <code>margins</code>
<code>residuals</code>	not allowed with <code>margins</code>
<code>rstandard</code>	not allowed with <code>margins</code>
<code>score</code>	not allowed with <code>margins</code>

Statistics not allowed with `margins` are functions of stochastic quantities other than $e(b)$.

For the full syntax, see [R] [margins](#).

Remarks and examples

[stata.com](http://www.stata.com)

`predict` is used after `logistic` to obtain predicted probabilities, residuals, and influence statistics for the estimation sample. The suggested diagnostic graphs below are from [Hosmer, Lemeshow, and Sturdivant \(2013\)](#), where they are more elaborately explained. Also see [Collett \(2003, 129–168\)](#) for a thorough discussion of model checking.

Remarks are presented under the following headings:

predict without options
predict with the xb and stdp options
predict with the residuals option
predict with the number option
predict with the deviance option
predict with the rstandard option
predict with the hat option
predict with the dx2 option
predict with the ddeviance option
predict with the dbeta option

predict without options

Typing `predict newvar` after estimation calculates the predicted probability of a positive outcome.

In [example 1](#) of [\[R\] logistic](#), we ran the model `logistic low age lwt i.race smoke ptl ht ui`. We obtain the predicted probabilities of a positive outcome by typing

```
. use https://www.stata-press.com/data/r18/lbw
(Hosmer & Lemeshow data)
. logistic low age lwt i.race smoke ptl ht ui
(output omitted)
. predict p
(option pr assumed; Pr(low))
. summarize p low
```

Variable	Obs	Mean	Std. dev.	Min	Max
p	189	.3121693	.1913915	.0272559	.8391283
low	189	.3121693	.4646093	0	1

predict with the xb and stdp options

`predict` with the `xb` option calculates the linear combination $x_j\mathbf{b}$, where x_j are the independent variables in the j th observation and \mathbf{b} is the estimated parameter vector. This is sometimes known as the index function because the cumulative distribution function indexed at this value is the probability of a positive outcome.

With the `stdp` option, `predict` calculates the standard error of the prediction, which is *not* adjusted for replicated covariate patterns in the data. The influence statistics described below are adjusted for replicated covariate patterns in the data.

predict with the residuals option

`predict` can calculate more than predicted probabilities. The Pearson residual is defined as the square root of the contribution of the covariate pattern to the Pearson χ^2 goodness-of-fit statistic, signed according to whether the observed number of positive responses within the covariate pattern is less than or greater than expected. For instance,

```
. predict r, residuals
. summarize r, detail
```

Pearson residual					
	Percentiles	Smallest			
1%	-1.750923	-2.283885			
5%	-1.129907	-1.750923			
10%	-.9581174	-1.636279	Obs		189
25%	-.6545911	-1.636279	Sum of wgt.		189
50%	-.3806923		Mean		-.0242299
		Largest	Std. dev.		.9970949
75%	.8162894	2.23879			
90%	1.510355	2.317558	Variance		.9941981
95%	1.747948	3.002206	Skewness		.8618271
99%	3.002206	3.126763	Kurtosis		3.038448

We notice the prevalence of a few large positive residuals:

```
. sort r
. list id r low p age race in -5/1
```

	id	r	low	p	age	race
185.	33	2.224501	1	.1681123	19	White
186.	57	2.23879	1	.166329	15	White
187.	16	2.317558	1	.1569594	27	Other
188.	77	3.002206	1	.0998678	26	White
189.	36	3.126763	1	.0927932	24	White

predict with the number option

Covariate patterns play an important role in logistic regression. Two observations are said to share the same covariate pattern if the independent variables for the two observations are identical. Although we might think of having individual observations, the statistical information in the sample can be summarized by the covariate patterns, the number of observations with that covariate pattern, and the number of positive outcomes within the pattern. Depending on the model, the number of covariate patterns can approach or be equal to the number of observations, or it can be considerably less.

Stata calculates all the residual and diagnostic statistics in terms of covariate patterns, not observations. That is, all observations with the same covariate pattern are given the same residual and diagnostic statistics. [Hosmer, Lemeshow, and Sturdivant \(2013, 154–155\)](#) argue that such “ M -asymptotic” statistics are more useful than “ N -asymptotic” statistics.

To understand the difference, think of an observed positive outcome with predicted probability of 0.8. Taking the observation in isolation, the residual must be positive—we expected 0.8 positive responses and observed 1. This may indeed be the correct residual, but not necessarily. Under the M -asymptotic definition, we ask how many successes we observed across all observations with this covariate pattern. If that number were, say, six, and there were a total of 10 observations with this covariate pattern, then the residual is negative for the covariate pattern—we expected eight positive outcomes but observed six. `predict` makes this kind of calculation and then attaches the same residual to all observations in the covariate pattern.

Occasionally, you might want to find all observations sharing a covariate pattern. `number` allows you to do this:

```
. predict pattern, number
. summarize pattern
```

Variable	Obs	Mean	Std. dev.	Min	Max
pattern	189	89.2328	53.16573	1	182

We previously fit the model `logistic low age lwt i.race smoke ptl ht ui` over 189 observations. There are 182 covariate patterns in our data.

predict with the deviance option

The deviance residual is defined as the square root of the contribution to the likelihood-ratio test statistic of a saturated model versus the fitted model. It has slightly different properties from the Pearson residual (see [Hosmer, Lemeshow, and Sturdivant \[2013, 155–157\]](#)):

```
. predict d, deviance
. summarize d, detail
```

deviance residual					
Percentiles		Smallest			
1%	-1.843472	-1.911621			
5%	-1.33477	-1.843472			
10%	-1.148316	-1.843472	Obs	189	
25%	-.8445325	-1.674869	Sum of wgt.	189	
50%	-.5202702		Mean	-.1228811	
		Largest	Std. dev.	1.049237	
75%	.9129041	1.894089			
90%	1.541558	1.924457	Variance	1.100898	
95%	1.673338	2.146583	Skewness	.6598857	
99%	2.146583	2.180542	Kurtosis	2.036938	

predict with the rstandard option

Pearson residuals do not have a standard deviation equal to 1. `rstandard` generates Pearson residuals normalized to have an *expected* standard deviation equal to 1.

```
. predict rs, rstandard
. summarize r rs
```

Variable	Obs	Mean	Std. dev.	Min	Max
r	189	-.0242299	.9970949	-2.283885	3.126763
rs	189	-.0279135	1.026406	-2.4478	3.149081

```
. correlate r rs
(obs=189)
```

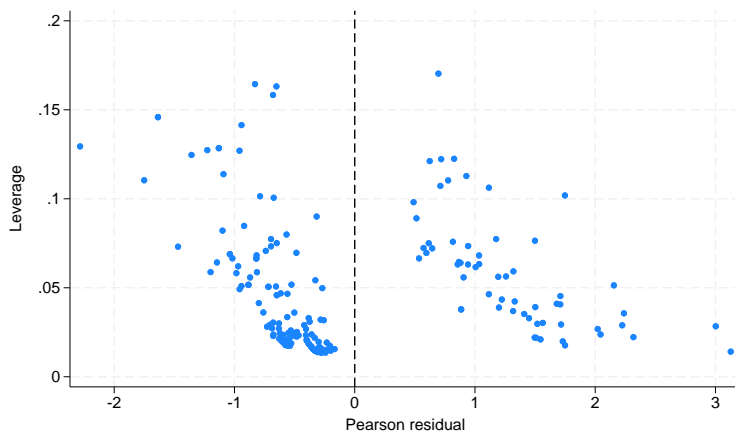
	r	rs
r	1.0000	
rs	0.9998	1.0000

Remember that we previously created `r` containing the (unstandardized) Pearson residuals. In these data, whether we use standardized or unstandardized residuals does not matter much.

predict with the hat option

`hat` calculates the leverage of a covariate pattern—a scaled measure of distance in terms of the independent variables. Large values indicate covariate patterns far from the average covariate pattern that can have a large effect on the fitted model even if the corresponding residual is small. Consider the following graph:

```
. predict h, hat
. scatter h r, xline(0) ytitle("Leverage")
```



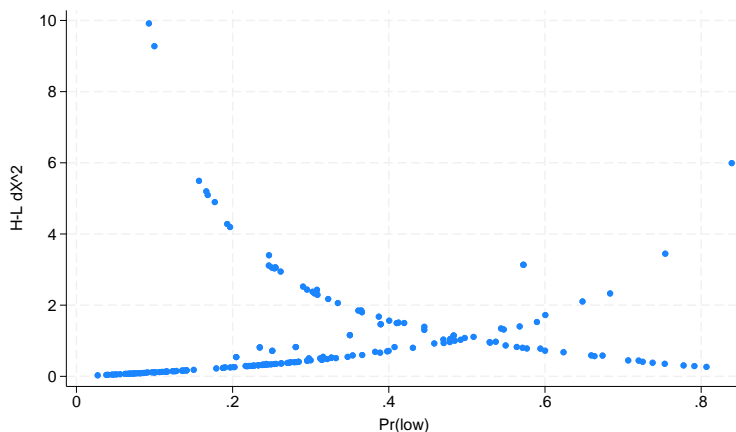
The points to the left of the vertical line are observed negative outcomes; here our data contain almost as many covariate patterns as observations, so most covariate patterns are unique. In such unique patterns, we observe either 0 or 1 success and expect p , thus forcing the sign of the residual. If we had fewer covariate patterns—if we did not have continuous variables in our model—there would be no such interpretation, and we would not have drawn the vertical line at 0.

Points on the left and right edges of the graph represent large residuals—covariate patterns that are not fit well by our model. Points at the top of our graph represent high leverage patterns. When analyzing the influence of observations on the model, we are most interested in patterns with high leverage and small residuals—patterns that might otherwise escape our attention.

predict with the dx2 option

There are many ways to measure influence, and `hat` is one example. `dx2` measures the decrease in the Pearson χ^2 goodness-of-fit statistic that would be caused by deleting an observation (and others sharing the covariate pattern):


```
. predict dx2, dx2
. scatter dx2 p
```



Paraphrasing [Hosmer, Lemeshow, and Sturdivant \(2013, 195–197\)](#), the points going from the top left to the bottom right correspond to covariate patterns with the number of positive outcomes equal to the number in the group; the points on the other curve correspond to 0 positive outcomes. In our data, most of the covariate patterns are unique, so the points tend to lie along one or the other curves; the points that are off the curves correspond to the few repeated covariate patterns in our data in which all the outcomes are not the same.

We examine this graph for large values of `dx2`—there are two at the top left.

predict with the ddeviance option

Another measure of influence is the change in the deviance residuals due to deletion of a covariate pattern:

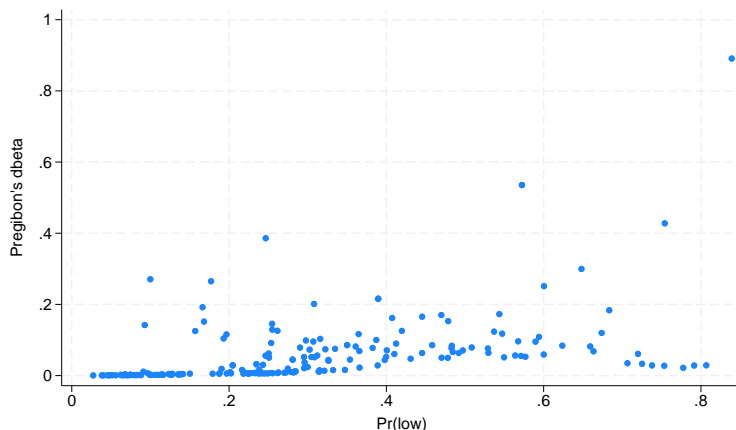
```
. predict dd, ddeviance
```

As with `dx2`, we typically graph `ddeviance` against the probability of a positive outcome. We direct you to [Hosmer, Lemeshow, and Sturdivant \(2013, 195\)](#) for an example and for the interpretation of this graph.

predict with the dbeta option

One of the more direct measures of influence of interest to model fitters is the [Pregibon \(1981\)](#) dbeta measure, a measure of the change in the coefficient vector that would be caused by deleting an observation (and all others sharing the covariate pattern):

```
. predict db, dbeta
. scatter db p
```



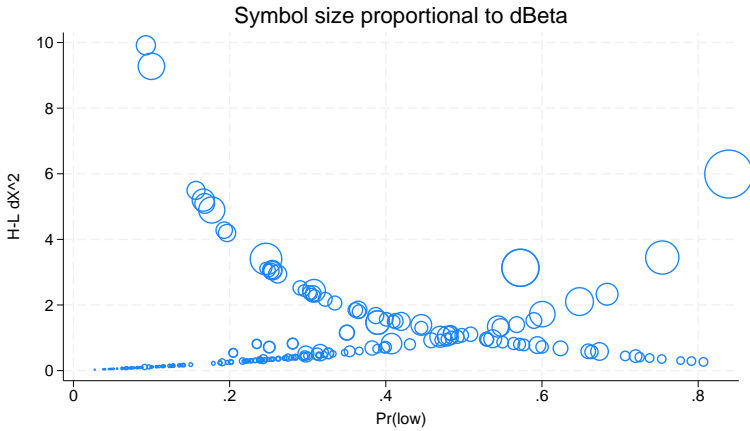
One observation has a large effect on the estimated coefficients. We can easily find this point:

```
. sort db
. list in 1
```

189.	id	low	age	lwt	race	smoke	pt1	ht	ui	ftv	bwt
	188	0	25	95	White	Smoker	3	0	1	0	3637
	p		r		pattern	d	rs		h		
	.8391283		-2.283885		117	-1.911621	-2.4478		.1294439		
	dx2				dd			db			
	5.991726				4.197658			.8909163			

Hosmer, Lemeshow, and Sturdivant (2013, 196) suggest a graph that combines two of the influence measures:

```
. scatter dx2 p [aw=db], title("Symbol size proportional to dBeta") mfcolor(none)
```



We can easily spot the most influential points by the `dbeta` and `dx2` measures.

Methods and formulas

Let j index observations. Define M_j for each observation as the total number of observations sharing j 's covariate pattern. Define Y_j as the total number of positive responses among observations sharing j 's covariate pattern.

The Pearson residual for the j th observation is defined as

$$r_j = \frac{Y_j - M_j p_j}{\sqrt{M_j p_j (1 - p_j)}}$$

For $M_j > 1$, the deviance residual d_j is defined as

$$d_j = \pm \left(2 \left[Y_j \ln \left(\frac{Y_j}{M_j p_j} \right) + (M_j - Y_j) \ln \left\{ \frac{M_j - Y_j}{M_j (1 - p_j)} \right\} \right] \right)^{1/2}$$

where the sign is the same as the sign of $(Y_j - M_j p_j)$. In the limiting cases, the deviance residual is given by

$$d_j = \begin{cases} -\sqrt{2M_j |\ln(1 - p_j)|} & \text{if } Y_j = 0 \\ \sqrt{2M_j |\ln p_j|} & \text{if } Y_j = M_j \end{cases}$$

The *unadjusted* diagonal elements of the hat matrix h_{Uj} are given by $h_{Uj} = (\mathbf{XVX}')_{jj}$, where V is the estimated covariance matrix of parameters. The adjusted diagonal elements h_j created by `hat` are then $h_j = M_j p_j (1 - p_j) h_{Uj}$.

The standardized Pearson residual r_{Sj} is $r_j / \sqrt{1 - h_j}$.

The Pregibon (1981) $\Delta\widehat{\beta}_j$ influence statistic is

$$\Delta\widehat{\beta}_j = \frac{r_j^2 h_j}{(1 - h_j)^2}$$

The corresponding change in the Pearson χ^2 is r_{Sj}^2 . The corresponding change in the deviance residual is $\Delta D_j = d_j^2 / (1 - h_j)$.

References

- Collett, D. 2003. *Modelling Binary Data*. 2nd ed. London: Chapman and Hall/CRC.
- Hosmer, D. W., Jr., S. A. Lemeshow, and R. X. Sturdivant. 2013. *Applied Logistic Regression*. 3rd ed. Hoboken, NJ: Wiley.
- Kasza, J. 2015. *Stata tip 125: Binned residual plots for assessing the fit of regression models for binary outcomes*. *Stata Journal* 15: 599–604.
- Mitchell, M. N., and X. Chen. 2005. *Visualizing main effects and interactions for binary logit models*. *Stata Journal* 5: 64–82.
- Newson, R. B. 2013. *Attributable and unattributable risks and fractions and other scenario comparisons*. *Stata Journal* 13: 672–698.
- Powers, D. A., H. Yoshioka, and M.-S. Yun. 2011. *mvdcmp: Multivariate decomposition for nonlinear response models*. *Stata Journal* 11: 556–576.
- Pregibon, D. 1981. *Logistic regression diagnostics*. *Annals of Statistics* 9: 705–724. <https://doi.org/10.1214/aos/1176345513>.
- Wang, Z. 2007. *Two postestimation commands for assessing confounding effects in epidemiological studies*. *Stata Journal* 7: 183–196.

Also see

- [R] **logistic** — Logistic regression, reporting odds ratios
- [R] **estat classification** — Classification statistics and table
- [R] **estat gof** — Pearson or Hosmer–Lemeshow goodness-of-fit test
- [R] **lroc** — Compute area under ROC curve and graph the curve
- [R] **lsens** — Graph sensitivity and specificity versus probability cutoff
- [U] **20 Estimation and postestimation commands**

Stata, Stata Press, and Mata are registered trademarks of StataCorp LLC. Stata and Stata Press are registered trademarks with the World Intellectual Property Organization of the United Nations. StataNow and NetCourseNow are trademarks of StataCorp LLC. Other brand and product names are registered trademarks or trademarks of their respective companies. Copyright © 1985–2023 StataCorp LLC, College Station, TX, USA. All rights reserved.



For suggested citations, see the FAQ on [citing Stata documentation](#).