Title

stata.com

nl postestimation — Postestimation tools for nl

Description

The following postestimation commands are available after `nl`:

Command	Description
`estat ic`	Akaike's and Schwarz's Bayesian information criteria (AIC and BIC)
`estat summarize`	summary statistics for the estimation sample
`estat vce`	variance–covariance matrix of the estimators (VCE)
`estat` (svy)	postestimation statistics for survey data
`estimates`	cataloging estimation results
`forecast`[1]	dynamic forecasts and simulations
`lincom`	point estimates, standard errors, testing, and inference for linear combinations of coefficients
`lrtest`[2]	likelihood-ratio test
`margins`[3]	marginal means, predictive margins, marginal effects, and average marginal effects
`marginsplot`	graph the results from margins (profile plots, interaction plots, etc.)
`nlcom`	point estimates, standard errors, testing, and inference for nonlinear combinations of coefficients
`predict`	predictions and residuals
`predictnl`	point estimates, standard errors, testing, and inference for generalized predictions
`test`	Wald tests of simple and composite linear hypotheses
`testnl`	Wald tests of nonlinear hypotheses

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

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

[3] You must specify the `variables()` option with `nl`.

Syntax for predict

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

predict [type] { stub* | newvar_1 ... newvar_k } [if] [in] , scores
```

where k is the number of parameters in the model.

statistic	Description
Main	
yhat	fitted values; the default
residuals	residuals
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*)	$E(y_j^*)$, $y_j^* = \max\{a, \min(y_j, b)\}$

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

Menu for predict

Statistics > Postestimation > Predictions, residuals, etc.

Options for predict

Main

`yhat`, the default, calculates the fitted values.

`residuals` calculates the residuals.

`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 truncated. *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 censored. *a* and *b* are specified as they are for `pr()`.

`scores` calculates the scores. The jth new variable created will contain the score for the jth parameter in `e(b)`.

Remarks and examples

stata.com

▷ Example 1

Obtaining predictions after fitting a nonlinear regression model with `nl` is no more difficult than obtaining predictions after fitting a linear regression model with `regress`. Here we fit a model of `mpg` on `weight`, allowing for a nonlinear relationship:

```
. use http://www.stata-press.com/data/r13/auto
(1978 Automobile Data)
. nl (mpg = {b0} + {b1}*weight^{gamma=-.5}), variables(weight) nolog
(obs = 74)
```

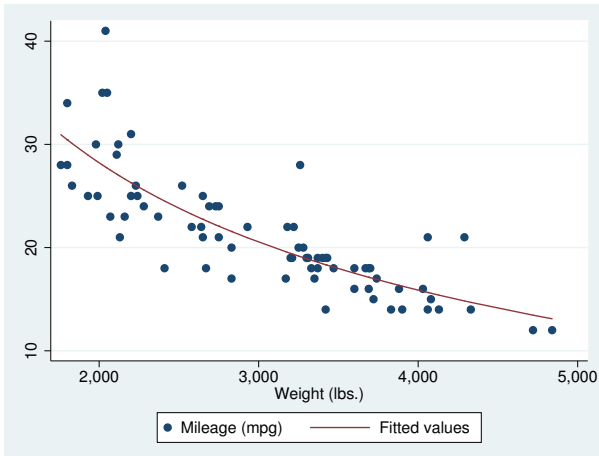
Source	SS	df	MS		
Model	1646.43761	2	823.218806	Number of obs	= 74
Residual	797.021847	71	11.2256598	R-squared	= 0.6738
				Adj R-squared	= 0.6646
Total	2443.45946	73	33.4720474	Root MSE	= 3.350472
				Res. dev.	= 385.8874

mpg	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
/b0	-18.17583	60.61713	-0.30	0.765	-139.0429	102.6913
/b1	1377.267	5292.443	0.26	0.795	-9175.564	11930.1
/gamma	-.4460916	.6763643	-0.66	0.512	-1.794724	.9025405

```
  Parameter b0 taken as constant term in model & ANOVA table
```

Now we obtain the predicted values of `mpg` and plot them in a graph along with the observed values:

```
. predict mpghat
(option yhat assumed; fitted values)
. scatter mpg weight || line mpghat weight, sort
```



Suppose we wanted to know how sensitive mpg is to changes in weight for cars that weigh 3,000 pounds. We can use margins to find out:

```
. margins, eyex(weight) at(weight = 3000)
Warning: cannot perform check for estimable functions.
Conditional marginal effects                      Number of obs   =         74
Model VCE    : GNR

Expression   : Fitted values, predict()
ey/ex w.r.t. : weight
at           : weight          =        3000
```

	ey/ex	Delta-method Std. Err.	z	P>|z|	[95% Conf.	Interval]
weight	-.8408119	.0804379	-10.45	0.000	-.9984673	-.6831565

With the eyex() option, margins reports elasticities. These results show that if we increase weight by 1%, then mpg decreases by about 0.84%.

◁

❑ Technical note

Observant readers will notice that margins issued a warning message stating that it could not perform its usual check for estimable functions. In the case of nl, as long as you do not specify the predict() option of margins or specify the default predict(yhat), you can safely ignore that message. The predicted values that nl produces are suitable for use with margins. However, if you specify any predict() options other than yhat, then the output from margins after using nl will not be correct.

❑

Also see

[R] **nl** — Nonlinear least-squares estimation

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