

nbreg postestimation — Postestimation tools for nbreg and gnbreg

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

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

[margins](#)
[Reference](#)

Postestimation commands

The following postestimation commands are available after `nbreg` and `gnbreg`:

Command	Description
<code>contrast</code>	contrasts and ANOVA-style joint tests of estimates
<code>estat ic</code>	Akaike's and Schwarz's Bayesian information criteria (AIC 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>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>	predictions, residuals, influence statistics, and other diagnostic measures
<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 numbers of events, incidence rates, probabilities, linear predictions, standard errors, and predicted values.

Menu for predict

Statistics > Postestimation

Syntax for predict

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

```
predict [type] { stub* | newvarreg newvardisp } [if] [in] , scores
```

<i>statistic</i>	Description
------------------	-------------

Main	
<code>n</code>	number of events; the default
<code>ir</code>	incidence rate (equivalent to <code>predict ... , n nooffset</code>)
<code>pr(n)</code>	probability $\Pr(y_j = n)$
<code>pr(a,b)</code>	probability $\Pr(a \leq y_j \leq b)$
<code>xb</code>	linear prediction
<code>stdp</code>	standard error of the linear prediction

In addition, relevant only after `gnbreg` are the following:

<i>statistic</i>	Description
------------------	-------------

Main	
<code><u>alpha</u></code>	predicted values of α_j
<code><u>lnalpha</u></code>	predicted values of $\ln\alpha_j$
<code><u>stdplna</u></code>	standard error of predicted $\ln\alpha_j$

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

Options for predict

Main

`n`, the default, calculates the predicted number of events, which is $\exp(\mathbf{x}_j\beta)$ if neither `offset(varnameo)` nor `exposure(varnamee)` was specified when the model was fit; $\exp(\mathbf{x}_j\beta + \text{offset}_j)$ if `offset()` was specified; or $\exp(\mathbf{x}_j\beta) \times \text{exposure}_j$ if `exposure()` was specified.

`ir` calculates the incidence rate $\exp(\mathbf{x}_j\beta)$, which is the predicted number of events when exposure is 1. This is equivalent to specifying both the `n` and the `nooffset` options.

`pr(n)` calculates the probability $\Pr(y_j = n)$, where n is a nonnegative integer that may be specified as a number or a variable.

`pr(a,b)` calculates the probability $\Pr(a \leq y_j \leq b)$, where a and b are nonnegative integers that may be specified as numbers or variables;

b missing ($b \geq .$) means $+\infty$;

`pr(20,.)` calculates $\Pr(y_j \geq 20)$;

`pr(20,b)` calculates $\Pr(y_j \geq 20)$ in observations for which $b \geq .$ and calculates $\Pr(20 \leq y_j \leq b)$ elsewhere.

`pr(.,b)` produces a syntax error. A missing value in an observation of the variable a causes a missing value in that observation for `pr(a,b)`.

`xb` calculates the linear prediction, which is $\mathbf{x}_j\beta$ if neither `offset()` nor `exposure()` was specified; $\mathbf{x}_j\beta + \text{offset}_j$ if `offset()` was specified; or $\mathbf{x}_j\beta + \ln(\text{exposure}_j)$ if `exposure()` was specified; see `nooffset` below.

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

`alpha`, `lnalpha`, and `stdplna` are relevant after `gnbreg` estimation only; they produce the predicted values of α_j , $\ln\alpha_j$, and the standard error of the predicted $\ln\alpha_j$, respectively.

`nooffset` is relevant only if you specified `offset()` or `exposure()` when you fit the model. It modifies the calculations made by `predict` so that they ignore the offset or exposure variable; the linear prediction is treated as $\mathbf{x}_j\beta$ rather than as $\mathbf{x}_j\beta + \text{offset}_j$ or $\mathbf{x}_j\beta + \ln(\text{exposure}_j)$. Specifying `predict ... , nooffset` is equivalent to specifying `predict ... , ir`.

`scores` calculates equation-level score variables.

The first new variable will contain $\partial \ln L / \partial (\mathbf{x}_j\beta)$.

The second new variable will contain $\partial \ln L / \partial (\ln\alpha_j)$ for `dispersion(mean)` and `gnbreg`.

The second new variable will contain $\partial \ln L / \partial (\ln\delta)$ for `dispersion(constant)`.

margins

Description for margins

`margins` estimates margins of response for numbers of events, incidence rates, probabilities, linear predictions, and predicted values.

Menu for margins

Statistics > Postestimation

Syntax for margins

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

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

<i>statistic</i>	Description
<code>n</code>	number of events; the default
<code>ir</code>	incidence rate (equivalent to <code>predict ... , n nooffset</code>)
<code>pr(n)</code>	probability $\Pr(y_j = n)$
<code>pr(a,b)</code>	probability $\Pr(a \leq y_j \leq b)$
<code>xb</code>	linear prediction
<code>stdp</code>	not allowed with <code>margins</code>

In addition, relevant only after `gnbreg` are the following:

<i>statistic</i>	Description
<code><u>alpha</u></code>	predicted values of α_j
<code><u>lnalpha</u></code>	predicted values of $\ln\alpha_j$
<code><u>stdplna</u></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

After `nbreg` and `gnbreg`, `predict` returns the expected number of deaths per cohort and the probability of observing the number of deaths recorded or fewer.

```
. use https://www.stata-press.com/data/r16/rod93
. nbreg deaths i.cohort, nolog
```

```
Negative binomial regression          Number of obs   =          21
                                      LR chi2(2)       =           0.14
Dispersion = mean                    Prob > chi2      =          0.9307
Log likelihood = -108.48841           Pseudo R2       =          0.0007
```

deaths	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
cohort						
1960-1967	.0591305	.2978419	0.20	0.843	-.5246289	.64289
1968-1976	-.0538792	.2981621	-0.18	0.857	-.6382662	.5305077
_cons	4.435906	.2107213	21.05	0.000	4.0229	4.848912
/lnalpha	-1.207379	.3108622			-1.816657	-.5980999
alpha	.29898	.0929416			.1625683	.5498555

```
LR test of alpha=0: chibar2(01) = 434.62          Prob >= chibar2 = 0.000
```

```
. predict count
(option n assumed; predicted number of events)
. predict p, pr(0, deaths)
. summarize deaths count p
```

Variable	Obs	Mean	Std. Dev.	Min	Max
deaths	21	84.66667	48.84192	10	197
count	21	84.66667	4.00773	80	89.57143
p	21	.4991542	.2743702	.0070255	.9801285

The expected number of deaths ranges from 80 to 90. The probability $\Pr(y_i \leq \text{deaths})$ ranges from 0.007 to 0.98.

Methods and formulas

In the following, we use the same notation as in [R] `nbreg`.

Methods and formulas are presented under the following headings:

Mean-dispersion model
Constant-dispersion model

Mean-dispersion model

The equation-level scores are given by

$$\begin{aligned} \text{score}(\mathbf{x}\beta)_j &= p_j(y_j - \mu_j) \\ \text{score}(\tau)_j &= -m \left\{ \frac{\alpha_j(\mu_j - y_j)}{1 + \alpha_j\mu_j} - \ln(1 + \alpha_j\mu_j) + \psi(y_j + m) - \psi(m) \right\} \end{aligned}$$

where $\tau_j = \ln\alpha_j$ and $\psi(z)$ is the digamma function.

Constant-dispersion model

The equation-level scores are given by

$$\begin{aligned}\text{score}(\mathbf{x}\boldsymbol{\beta})_j &= m_j \{ \psi(y_j + m_j) - \psi(m_j) + \ln(p) \} \\ \text{score}(\tau)_j &= y_j - (y_j + m_j)(1 - p) - \text{score}(\mathbf{x}\boldsymbol{\beta})_j\end{aligned}$$

where $\tau_j = \ln\delta_j$.

Reference

Manjón, M., and O. Martínez. 2014. [The chi-squared goodness-of-fit test for count-data models](#). *Stata Journal* 14: 798–816.

Also see

[R] **nbreg** — Negative binomial regression

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