

**tpoisson postestimation** — Postestimation tools for `tpoisson`

Postestimation commands    [predict](#)    [margins](#)    [Methods and formulas](#)

Also see

## Postestimation commands

The following postestimation commands are available after `tpoisson`:

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>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 <code>margins</code> (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.

## predict

### Description for predict

`predict` creates a new variable containing predictions such as numbers of events, incidence rates, conditional means, probabilities, conditional probabilities, linear predictions, standard errors, and equation-level scores.

### Menu for predict

Statistics > Postestimation

### Syntax for predict

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

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

Main

<code>n</code>	number of events; the default
<code>ir</code>	incidence rate
<code>cm</code>	conditional mean, $E(y_j   y_j > \tau_j)$
<code>pr(<i>n</i>)</code>	probability $\Pr(y_j = n)$
<code>pr(<i>a</i>,<i>b</i>)</code>	probability $\Pr(a \leq y_j \leq b)$
<code>cpr(<i>n</i>)</code>	conditional probability $\Pr(y_j = n   y_j > \tau_j)$
<code>cpr(<i>a</i>,<i>b</i>)</code>	conditional probability $\Pr(a \leq y_j \leq b   y_j > \tau_j)$
<code>xb</code>	linear prediction
<code>stdp</code>	standard error of the linear prediction
<code>score</code>	first derivative of the log likelihood with respect to $\mathbf{x}_j\beta$

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()` nor `exposure()` 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.

`cm` calculates the conditional mean,

$$E(y_j | y_j > \tau_j) = \frac{E(y_j, y_j > \tau_j)}{\Pr(y_j > \tau_j)}$$

where  $\tau_j$  is the truncation point found in `e(llopt)`.

`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* ≥ .) means  $+\infty$ ;

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

`pr(20,b)` calculates  $\Pr(y_j \geq 20)$  in observations for which *b* ≥ . 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)`.

`cpr(n)` calculates the conditional probability  $\Pr(y_j = n | y_j > \tau_j)$ , where  $\tau_j$  is the truncation point found in `e(11opt)`. *n* is an integer greater than the truncation point that may be specified as a number or a variable.

`cpr(a,b)` calculates the conditional probability  $\Pr(a \leq y_j \leq b | y_j > \tau_j)$ , where  $\tau_j$  is the truncation point found in `e(11opt)`. The syntax for this option is analogous to that used for `pr(a,b)` except that *a* must be greater than the truncation point.

`xb` calculates the linear prediction, which is  $\mathbf{x}_j\beta$  if neither `offset()` nor `exposure()` was specified when the model was fit;  $\mathbf{x}_j\beta + \text{offset}_j$  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.

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

`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`.

## margins

### Description for margins

`margins` estimates margins of response for numbers of events, incidence rates, conditional means, probabilities, conditional 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>n</code>	number of events; the default
<code>ir</code>	incidence rate
<code>cm</code>	conditional mean, $E(y_j   y_j > \tau_j)$
<code>pr(n)</code>	probability $\Pr(y_j = n)$
<code>pr(a,b)</code>	probability $\Pr(a \leq y_j \leq b)$
<code>cpr(n)</code>	conditional probability $\Pr(y_j = n   y_j > \tau_j)$
<code>cpr(a,b)</code>	conditional probability $\Pr(a \leq y_j \leq b   y_j > \tau_j)$
<code>xb</code>	linear prediction
<code>stdp</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](#).

### Methods and formulas

In the following formula, we use the same notation as in [R] [tpoisson](#).

The equation-level scores are given by

$$\text{score}(\mathbf{x}\beta)_j = y_j - e^{\xi_j} - \frac{e^{-e^{\xi_j}} e^{\xi_j}}{\Pr(Y > \tau_j | \xi_j)}$$

where  $\tau_j$  is the truncation point found in `e(1lopt)`.

## Also see

[R] [tpoisson](#) — Truncated Poisson regression

[U] [20 Estimation and postestimation commands](#)