poisson postestimation — Postestimation tools for poisson

Postestimation commands

The following postestimation commands are of special interest after poisson:

Command	Description
estat gof	goodness-of-fit test
lassogof	calculate goodness-of-fit predictions

estat gof is not appropriate with svy estimation results.

The following standard postestimation commands are also available:

Command	Description
contrast	contrasts and ANOVA-style joint tests of parameters
estat ic	Akaike's, consistent Akaike's, corrected Akaike's, and Schwarz's Bayesian information criteria (AIC, CAIC, AICc, and BIC, respectively)
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
etable	table of estimation results
* forecast	dynamic forecasts and simulations
* hausman	Hausman's specification test
lincom	point estimates, standard errors, testing, and inference for linear combinations of parameters
linktest	link test for model specification
* lrtest	likelihood-ratio test
margins	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 parameters
predict	number of events, incidence rates, probabilities, etc.
predictnl	point estimates, standard errors, testing, and inference for generalized predictions
pwcompare	pairwise comparisons of parameters
suest	seemingly unrelated estimation
test	Wald tests of simple and composite linear hypotheses
testnl	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 the equation-level score.

Menu for predict

Statistics > Postestimation

Syntax for predict

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

statistic	Description
Main	
`n`	number of events; the default
`ir`	incidence rate
`pr(`*n*`)`	probability $\Pr(y_j = n)$
`pr(`*a*`,`*b*`)`	probability $\Pr(a \le y_j \le b)$
`xb`	linear prediction
`stdp`	standard error of the linear prediction
`score`	first derivative of the log likelihood with respect to $\mathbf{x}_j\boldsymbol{\beta}$

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 $\exp(\mathbf{x}_j\boldsymbol{\beta})$ if neither `offset()` nor `exposure()` was specified when the model was fit; $\exp(\mathbf{x}_j\boldsymbol{\beta} + \text{offset}_j)$ if `offset()` was specified; or $\exp(\mathbf{x}_j\boldsymbol{\beta}) \times \text{exposure}_j$ if `exposure()` was specified.

`ir` calculates the incidence rate $\exp(\mathbf{x}_j\boldsymbol{\beta})$, which is the predicted number of events when exposure is 1. Specifying `ir` is equivalent to specifying `n` when neither `offset()` nor `exposure()` was specified when the model was fit.

`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 $\Pr(a \le y_j \le b)$, where *a* and *b* are nonnegative integers that may be specified as numbers or variables;

b missing ($b \ge .$) means $+\infty$;
`pr(20,.)` calculates $\Pr(y_j \ge 20)$;
`pr(20,`*b*`)` calculates $\Pr(y_j \ge 20)$ in observations for which $b \ge .$ and calculates
$\Pr(20 \le y_j \le b)$ 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 $\mathbf{x}_j\boldsymbol{\beta}$ if neither `offset()` nor `exposure()` was specified; $\mathbf{x}_j\boldsymbol{\beta} + \text{offset}_j$ if `offset()` was specified; or $\mathbf{x}_j\boldsymbol{\beta} + \ln(\text{exposure}_j)$ if `exposure()` was specified; see `nooffset` below.

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

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

`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 $\mathbf{x}_j\boldsymbol{\beta}$ rather than as $\mathbf{x}_j\boldsymbol{\beta} + \text{offset}_j$ or $\mathbf{x}_j\boldsymbol{\beta} + \ln(\text{exposure}_j)$. Specifying `predict ..., nooffset` is equivalent to specifying `predict ..., ir`.

margins

Description for margins

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

Menu for margins

Statistics > Postestimation

Syntax for margins

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

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

statistic	Description
`n`	number of events; the default
`ir`	incidence rate
`pr(`*n*`)`	probability $\Pr(y_j = n)$
`pr(`*a*`,`*b*`)`	probability $\Pr(a \leq y_j \leq b)$
`xb`	linear prediction
`stdp`	not allowed with `margins`
`score`	not allowed with `margins`

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

For the full syntax, see [R] **margins**.

estat

Description for estat

`estat gof` performs a goodness-of-fit test of the model. Both the deviance statistic and the Pearson statistic are reported. If the tests are significant, the Poisson regression model is inappropriate.

Menu for estat

Statistics > Postestimation

Syntax for estat

`estat gof`

`collect` is allowed; see **[U] 11.1.10 Prefix commands**.

Remarks and examples

▷ Example 1

Continuing with example 2 of [R] **poisson**, we use `estat gof` to determine whether the model fits the data well.

```
. use https://www.stata-press.com/data/r19/dollhill3
(Doll and Hill (1966))
. poisson deaths smokes i.agecat, exp(pyears) irr
  (output omitted)
. estat gof
         Deviance goodness-of-fit =  12.13237
         Prob > chi2(4)           =    0.0164
         Pearson goodness-of-fit  =  11.15533
         Prob > chi2(4)           =    0.0249
```

The deviance goodness-of-fit test tells us that, given the model, we can reject the hypothesis that these data are Poisson distributed at the 1.64% significance level. The Pearson goodness-of-fit test tells us that we can reject the hypothesis at the 2.49% significance level.

So let us now back up and be more careful. We can most easily obtain the incidence-rate ratios within age categories by using ir; see [R] **Epitab**:

```
. ir deaths smokes pyears, by(agecat) nohet
Stratified incidence-rate analysis
```

Age category	IRR	[95% conf.	interval]	M-H weight	
35-44	5.736638	1.463557	49.40468	1.472169	(exact)
45-54	2.138812	1.173714	4.272545	9.624747	(exact)
55-64	1.46824	.9863624	2.264107	23.34176	(exact)
65-74	1.35606	.9081925	2.096412	23.25315	(exact)
75-84	.9047304	.6000757	1.399687	24.31435	(exact)
Crude	1.719823	1.391992	2.14353		(exact)
M-H combined	1.424682	1.154703	1.757784		

We find that the mortality incidence ratios are greatly different within age category, being highest for the youngest categories and actually dropping below 1 for the oldest. (In the last case, we might argue that those who smoke and who have not died by age 75 are self-selected to be particularly robust.)

Seeing this, we will now parameterize the smoking effects separately for each category, although we will begin by constraining the smoking effects on third and fourth age categories to be equivalent:

```
. constraint 1 smokes#3.agecat = smokes#4.agecat
. poisson deaths c.smokes#agecat i.agecat, exposure(pyears) irr constraints(1)
Iteration 0:  Log likelihood =  -31.95424
Iteration 1:  Log likelihood = -27.796801
Iteration 2:  Log likelihood = -27.574177
Iteration 3:  Log likelihood = -27.572645
Iteration 4:  Log likelihood = -27.572645
Poisson regression                                  Number of obs =     10
                                                    Wald chi2(8)  = 632.14
Log likelihood = -27.572645                         Prob > chi2   = 0.0000
 ( 1)  [deaths]3.agecat#c.smokes - [deaths]4.agecat#c.smokes = 0
```

deaths	IRR	Std. err.	z	P>|z|	[95% conf.	interval]
agecat# c.smokes						
35-44	5.736637	4.181256	2.40	0.017	1.374811	23.93711
45-54	2.138812	.6520701	2.49	0.013	1.176691	3.887609
55-64	1.412229	.2017485	2.42	0.016	1.067343	1.868557
65-74	1.412229	.2017485	2.42	0.016	1.067343	1.868557
75-84	.9047304	.1855513	-0.49	0.625	.6052658	1.35236
agecat						
45-54	10.5631	8.067701	3.09	0.002	2.364153	47.19623
55-64	47.671	34.37409	5.36	0.000	11.60056	195.8978
65-74	98.22765	70.85012	6.36	0.000	23.89324	403.8244
75-84	199.2099	145.3356	7.26	0.000	47.67693	832.3648
_cons	.0001064	.0000753	-12.94	0.000	.0000266	.0004256
ln(pyears)	1	(exposure)				

Note: **_cons** estimates baseline incidence rate.

```
. estat gof

         Deviance goodness-of-fit =  .0773491
         Prob > chi2(1)           =    0.7809

         Pearson goodness-of-fit  =  .0773885
         Prob > chi2(1)           =    0.7809
```

The goodness-of-fit is now small; we are no longer running roughshod over the data. Let us now consider simplifying the model. The point estimate of the incidence-rate ratio for smoking in age category 1 is much larger than that for smoking in age category 2, but the confidence interval for `smokes#1.agecat` is similarly wide. Is the difference real?

```
. test smokes#1.agecat = smokes#2.agecat

 ( 1)  [deaths]1b.agecat#c.smokes - [deaths]2.agecat#c.smokes = 0

           chi2(  1) =    1.56
         Prob > chi2 =    0.2117
```

The point estimates of the incidence-rate ratio for smoking in the 35–44 age category is much larger than that for smoking in the 45–54 age category, but there is insufficient data, and we may be observing random differences. With that success, might we also combine the smokers in the third and fourth categories with those in the first and second categories?

```
. test smokes#2.agecat = smokes#3.agecat, accum

 ( 1)  [deaths]1b.agecat#c.smokes - [deaths]2.agecat#c.smokes = 0
 ( 2)  [deaths]2.agecat#c.smokes - [deaths]3.agecat#c.smokes = 0

           chi2(  2) =    4.73
         Prob > chi2 =    0.0938
```

Combining the first four categories may be overdoing it—the 9.38% significance level is enough to stop us, although others may disagree.

Thus, we now fit our final model:

```
. constraint 2 smokes#1.agecat = smokes#2.agecat
. poisson deaths c.smokes#agecat i.agecat, exposure(pyears) irr constraints(1/2)
Iteration 0:  Log likelihood = -31.550722
Iteration 1:  Log likelihood = -28.525057
Iteration 2:  Log likelihood = -28.514535
Iteration 3:  Log likelihood = -28.514535
Poisson regression                                      Number of obs =     10
                                                        Wald chi2(7)  = 642.25
Log likelihood = -28.514535                             Prob > chi2   = 0.0000
 ( 1)  [deaths]3.agecat#c.smokes - [deaths]4.agecat#c.smokes = 0
 ( 2)  [deaths]1b.agecat#c.smokes - [deaths]2.agecat#c.smokes = 0
------------------------------------------------------------------------------
      deaths |        IRR   Std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
      agecat#|
    c.smokes |
      35-44  |   2.636259   .7408403     3.45   0.001     1.519791    4.572907
      45-54  |   2.636259   .7408403     3.45   0.001     1.519791    4.572907
      55-64  |   1.412229   .2017485     2.42   0.016     1.067343    1.868557
      65-74  |   1.412229   .2017485     2.42   0.016     1.067343    1.868557
      75-84  |   .9047304   .1855513    -0.49   0.625     .6052658     1.35236
             |
      agecat |
      45-54  |   4.294559   .8385329     7.46   0.000     2.928987    6.296797
      55-64  |   23.42263   7.787716     9.49   0.000     12.20738    44.94164
      65-74  |   48.26309   16.06939    11.64   0.000     25.13068    92.68856
      75-84  |   97.87965   34.30881    13.08   0.000     49.24123     194.561
             |
       _cons |   .0002166   .0000652   -28.03   0.000     .0001201    .0003908
  ln(pyears) |          1  (exposure)
------------------------------------------------------------------------------
Note: _cons estimates baseline incidence rate.
```

The above strikes us as a fair representation of the data. The probabilities of observing the deaths seen in these data are estimated using the following `predict` command:

```
. predict p, pr(0, deaths)
. list deaths p

     +-------------------+
     | deaths          p |
     |-------------------|
  1. |     32   .6891766 |
  2. |    104   .4456625 |
  3. |    206   .5455328 |
  4. |    186   .4910622 |
  5. |    102   .5263011 |
     |-------------------|
  6. |      2    .227953 |
  7. |     12   .7981917 |
  8. |     28   .4772961 |
  9. |     28   .6227565 |
 10. |     31   .5475718 |
     +-------------------+
```

The probability $\Pr(y \leq \texttt{deaths})$ ranges from 0.23 to 0.80.

◁

Stored results

`estat gof` after `poisson` stores the following in `r()`:

Scalars

`r(df)`	degrees of freedom (Pearson and deviance)
`r(chi2_p)`	χ^2 (Pearson)
`r(chi2_d)`	χ^2 (deviance)
`r(p_p)`	p-value for χ^2 test (Pearson)
`r(p_d)`	p-value for χ^2 test (deviance)

Methods and formulas

In the following, we use the same notation as in [R] **poisson**.

The equation-level score is given by

$$\text{score}(\mathbf{x}\boldsymbol{\beta})_j = y_j - e^{\xi_j}$$

The deviance (D) and Pearson (P) goodness-of-fit statistics are given by

$$\begin{aligned}
\ln L_{\text{max}} &= \sum_{j=1}^{n} w_j \left[y_j \{ \ln(y_j) - 1 \} - \ln(y_j!) \right] \\
\chi^2_D &= -2\{ \ln L - \ln L_{\text{max}} \} \\
\chi^2_P &= \sum_{j=1}^{n} \frac{w_j (y_j - e^{\xi_j})^2}{e^{\xi_j}}
\end{aligned}$$

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] **poisson** — Poisson regression

[LASSO] **lassogof** — Goodness of fit after lasso for prediction

[U] 20 Estimation and postestimation commands