The Stata Journal (2001)
1, Number 1, pp. 51–57

Predicted probabilities for count models

J. Scott Long
Indiana University
jslong@indiana.edu

Jeremy Freese
University of Wisconsin–Madison
jfreese@ssc.wisc.edu

Abstract. The post-estimation command `prcounts` for generating predicted probabilities after using `poisson`, `nbreg`, `zip`, and `zinb` is introduced and illustrated.

Keywords: st0002, predicted probabilities, count models

1 Overview

Stata's `poisson` and `nbreg` commands estimate Poisson and negative binomial regression models for count outcomes. `zip` and `zinb` estimate zero-inflated Poisson and negative binomial models, which are useful when there are high frequencies of zero counts. After estimating a model using any of these four commands, our post-estimation command `prcounts` may be used to generate predicted probabilities. `prcounts` generates new variables that contain the predicted rate, the probability of each count from 0 to a user-specified maximum, and the cumulative probabilities that a count is less than or equal to each count from 0 to a user-specified maximum. When the `plot` option is specified, `prcounts` will also generate variables for the graphical comparison of observed and expected counts.

2 Syntax

`prcounts` *name* [`if` *exp*] [`in` *range*] [`, max(`*maxvalue*`) plot`]

where *name* specifies the prefix for the new variables that are created by `prcounts`. *name* cannot be the name of an existing variable.

3 Options

`max(`*maxvalue*`)` is the maximum count for which predicted probabilities should be computed. The default is 9.

`plot` specifies that variables for plotting expected counts should be generated.

Note that `if` and `in` restrict the sample for which predictions are made. By default, `prcounts` computes predicted values for all cases in the estimation sample.

4 Variables created

In the following, *name* represents the prefix specified as the argument to `prcounts`. y is the dependent count variable and each prediction is conditional on the variables included in the count regression model. Specific definitions of each predicted quantity are given in the *Methods and formulas* section below.

name`rate` is the predicted rate or count $E(y)$.

name`pr`*k* is the predicted probability $\Pr(y = k)$ for $k = 0$ to *maxvalue*. By default, *maxvalue* is 9.

name`prgt` is the predicted probability $\Pr(y > maxvalue)$.

name`cu`*k* is the predicted cumulative probability $\Pr(y \leq k)$ for $k = 0$ to *maxvalue*. By default, *maxvalue* is 9.

For `zip` and `zinb`, `prcounts` also generates

name`all0` is the predicted probability of being in the "always zero" (i.e., *inflate* = 1 group for `zip` and `zinb` models.

When the `plot` option is specified, more new variables are created with the average predicted probabilities. Note that this will include out of sample predictions if the estimation command included `if` or `in` conditions, but these conditions were not specified with `prcounts`. When these variables are generated, only the first *maxvalue* + 1 observations are nonmissing; these observations correspond to the counts 0 through *maxvalue*.

name`val` is the specific value k of the count y ranging from 0 to *maxvalue*.

name`obeq` is the observed probability $\Pr(y = k)$.

name`oble` is the observed cumulative probability $\Pr(y \leq k)$.

name`obeq` is the average *predicted* probability $\Pr(y = k)$.

name`oble` is the average *predicted* cumulative probability $\Pr(y \leq k)$.

5 Example

Using data on the scientific productivity of biochemists (Long 1997), the dependent variable `art` is the number of articles published in the three years prior to receiving the Ph.D. The independent variables are gender (`fem`), whether the scientist is married (`mar`), the number of children under age 5 (`kid5`), the prestige of the Ph.D. department ranging from .75 to 5 (`phd`), and the number of articles published by the scientist's mentor in the last three years (`ment`). We begin by estimating a Poisson regression.

```
. poisson art fem mar kid5 phd ment, nolog
Poisson regression                                Number of obs   =        915
                                                  LR chi2(5)      =     183.03
                                                  Prob > chi2     =     0.0000
Log likelihood = -1651.0563                       Pseudo R2       =     0.0525
```

art	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
fem	-.2245942	.0546138	-4.11	0.000	-.3316352	-.1175532
mar	.1552434	.0613747	2.53	0.011	.0349512	.2755356
kid5	-.1848827	.0401272	-4.61	0.000	-.2635305	-.1062349
phd	.0128226	.0263972	0.49	0.627	-.038915	.0645601
ment	.0255427	.0020061	12.73	0.000	.0216109	.0294746
_cons	.3046168	.1029822	2.96	0.003	.1027755	.5064581

Next, we run `prcounts` and then `summarize` the generated variables. Note that we have chosen the prefix `pois` to indicate that the created variables came from a Poisson regression, but any other name could have been used.

```
. prcounts pois, max(8) plot
. summarize pois*
```

Variable	Obs	Mean	Std. Dev.	Min	Max
poisrate	915	1.692896	.6685824	.8883344	9.627207
poispr0	915	.2092071	.0794247	.0000659	.4113403
poispr1	915	.3098447	.0634931	.0006345	.3678775
poispr2	915	.242096	.0311473	.0030544	.2706704
poispr3	915	.1346656	.0415861	.0098018	.2240418
poispr4	915	.0611696	.0383808	.0106732	.1951233
poispr5	915	.0249554	.0287183	.0018963	.1742638
poispr6	915	.0099346	.0201179	.0002808	.1603728
poispr7	915	.0041384	.0137756	.0000356	.1428533
poispr8	915	.001877	.0094055	3.96e-06	.1206255
poiscu0	915	.2092071	.0794247	.0000659	.4113403
poiscu1	915	.5190518	.1395755	.0007004	.7767481
poiscu2	915	.7611477	.1407294	.0037549	.9390502
poiscu3	915	.8958133	.1126566	.0135567	.9871097
poiscu4	915	.956983	.0824803	.0371477	.9977829
poiscu5	915	.9819384	.0589296	.0825709	.9996792
poiscu6	915	.991873	.0423403	.155454	.9999599
poiscu7	915	.9960114	.0310561	.2556911	.9999956
poiscu8	915	.9978884	.023188	.3763166	.9999995
poisprgt	915	.0021116	.023188	4.77e-07	.6236834
poisval	9	4	2.738613	0	8
poisobeq	9	.1101396	.1153559	.0010929	.3005464
poispreq	9	.1108765	.1174511	.001877	.3098447
poisoble	9	.8150577	.2373893	.3005464	.9912568
poisprle	9	.8122127	.2760109	.2092071	.9978884

To compare alternative count models, we can estimate each model in turn and use `prcounts` to generate predicted counts using prefixes that reflect which model was estimated. The commands are as follows:

```
. nbreg art fem mar kid5 phd ment, nolog
. prcounts nbreg, max(8) plot
. zip art fem mar kid5 phd ment, inf(fem mar kid5 phd ment) nolog
. prcounts zip, max(8) plot
. zinb art fem mar kid5 phd ment, inf(fem mar kid5 phd ment) nolog
. prcounts zinb, max(8) plot
```

By specifying the `plot` option after `prcounts`, we generate additional variables that contain the observed probability of each count from 0 to 8 (the maximum count specified by `max()`) and the average predicted probabilities of each count. Using the `list` command to display the values of the new variables created with the `plot` option for our Poisson model illustrates further what this option does.

```
. list poisval poisobeq poispreq poisoble poisprle in 1/10
      poisval   poisobeq   poispreq   poisoble   poisprle
 1.         0   .3005464   .2092071   .3005464   .2092071
 2.         1   .2688525   .3098447   .5693989   .5190518
 3.         2   .1945355    .242096   .7639344   .7611477
 4.         3   .0918033   .1346656   .8557377   .8958133
 5.         4    .073224   .0611696   .9289618    .956983
 6.         5   .0295082   .0249554   .9584699   .9819384
 7.         6   .0185792   .0099346   .9770492    .991873
 8.         7   .0131148   .0041384   .9901639   .9960114
 9.         8   .0010929    .001877   .9912568   .9978884
10.         .          .          .          .          .
```

We can then compute the difference between the observed probability of each count and the prediction from each of the four models.

```
. generate devpois = poisobeq - poispreq
(906 missing values generated)
. generate devnbreg = poisobeq - nbregpreq
(906 missing values generated)
. generate devzip = poisobeq - zippreq
(906 missing values generated)
. generate devzinb = poisobeq - zinbpreq
(906 missing values generated)
. label var devpois "poisson"
. label var devnbreg "nbreg"
. label var devzip "zip"
. label var devzinb "zinb"
. label var poisval "Count"
```

Finally, the results can be plotted:

```
. graph devpois devnbreg devzip devzinb poisval, /*
> */ c(llll) s(OSTp) xlab(0 1 to 8) ylab(-.1,-.05,0,.05,.1) /*
> */ yline(-.1,-.05,0,.05,.1) l2title("Deviation from Observed") gap(4)
```

This leads to the plot in Figure 1.

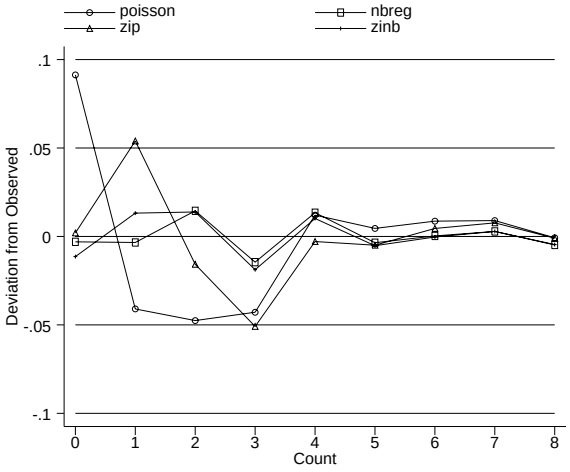

Figure 1: An example plot from `prcounts`.

This figure plots the difference between the observed proportions for each count and the mean probability from the four models. We see immediately that the major failure of the Poisson regression model is in predicting the number of zeros, with an underprediction of about 0.1. The ZIP model does much better at predicting zeros, but has poor predictions for counts one through three. The negative binomial regression model predicts the zeros very well and also has much better predictions for the counts from one to three. The ZINB model slightly overpredicts zeros and underpredicts ones, with similar predictions to the negative binomial model for other counts. Overall, the negative binomial model provides the most accurate predictions, which are slightly better than those for the ZINB model.

6 Methods and formulas

Details on these models can be found in Chapter 8 of Long (1997) or Cameron and Trivedi (1998). More information on using Stata with count outcomes can be found in Long and Freese (2001). See also the manual entries for `poisson`, `nbreg`, `zip`, and `zinb`. Here we briefly review only the calculation of predicted rates and probabilities.

6.1 The Poisson regression model

The predicted rate is calculated as

$$\mu_i = \mathrm{E}(y_i = k|x_i) = \exp(x_i\beta) \tag{1}$$

The probability of observing a specific count given x_i is computed as

$$\Pr(y_i = k|\mu_i) = \frac{e^{-\mu_i}\mu_i^k}{k!}, \qquad k = 0, 1, 2, \ldots$$

6.2 The negative binomial regression model

In this model, the mean structure remains the same as Equation (1), but the variance in the predicted counts is increased through the addition of a single parameter, generally referred to as α. The predicted rate is still calculated by Equation (1), but the predicted probabilities now have a negative binomial distribution

$$\Pr(y_i = k|x_i) = \frac{\Gamma(k+\alpha^{-1})}{k!\Gamma(\alpha^{-1})}\left(\frac{\alpha^{-1}}{\alpha^{-1}+\mu_i}\right)^{\alpha^{-1}}\left(\frac{\mu_i}{\alpha^{-1}+\mu_i}\right)^{k}, \qquad k = 0, 1, 2, \ldots$$

6.3 Zero-inflated regression models

The zero-inflated models introduce unobserved discrete heterogeneity to differentiate those who will *always* have zero counts and those who are only "at risk" of having a zero count. The ZIP model combines the Poisson regression model with a binary logit or probit model differentiating those who will always have a zero count from those who will not always have a zero count. The ZINB model combines the negative binomial regression model with a binary model.

In Stata's `zip` and `zinb` commands, the idea of inflation is used to define those in the "always zero" class. This class is defined as those for which $\mathit{inflate} = 1$. The probability of being in this class equals

$$\Pr(\mathit{always}\ 0|x_i, z_i) = \Pr(\mathit{inflate} = 1|x_i, z_i) = F(z_i\gamma) = \psi_i$$

where F is the cumulative density function (cdf) for the logistic if logit is used or the cdf for the normal if probit is used for the binary model. The predicted rate combines the results for those who are always zero with those who are not always zero, using the equation

$$\mathrm{E}(y_i|x_i, z_i) = [0 \times \psi_i] + [\mu_i \times (1-\psi_i)] = \mu_i - \mu_i\psi_i$$

To calculate the probability of observing a particular count, the results from the count equation must be adjusted according to the probability of the observation being in the always zero category. For example, for Poisson regression,

$$\begin{aligned}\Pr(y_i = 0|x_i, z_i) &= \Pr(\text{always } 0) + \Pr(0 \text{ by chance}) \\ &= \psi_i + (1-\psi_i)e^{-\mu_i}\end{aligned}$$

For non-zero counts,

$$\overline{\Pr}(y_i = k|x_i) = (1-\psi_i)\frac{e^{-\mu_i}\mu_i^k}{k!}$$

6.4 Probabilities for plotting

A useful, informal method for comparing predictions across models is to plot the mean predicted probability for each count value against the observed probability. The mean predicted probability for a given count model is defined as

$$\overline{\Pr}(y = m) = \frac{1}{N}\sum_{i=1}^{N} \Pr(y_i = m|\mu_i)$$

When comparing across several models, it is useful to subtract the predicted probability from the observed probability, as shown in our example above.

7 Acknowledgment

We thank Simon Cheng for his help in testing this command. For information on related programs and future updates to this program, please check *http://www.indiana.edu/~jsl650/spost.htm*.

8 References

Cameron, A. C. and P. K. Trivedi. 1998. *Regression Analysis of Count Data*. New York: Cambridge University Press.

Long, J. S. 1997. *Regression Models for Categorical and Limited Dependent Variables*. Thousand Oaks, CA: Sage Publications.

Long, J. S. and J. Freese. 2001. *Regression Models for Categorical Dependent Variables using Stata*. College Station, TX: Stata Press.

About the Authors

J. Scott Long is Chancellor's Professor of Sociology at Indiana University. Jeremy Freese is Assistant Professor of Sociology at the University of Wisconsin–Madison. In addition to their own projects, their recent work includes the book *Regression Models for Categorical Dependent Variables using Stata*, published by Stata Press.