

## Models for Count Outcomes

Richard Williams, University of Notre Dame, <https://www3.nd.edu/~rwilliam/>  
Last revised April 12, 2022

These notes borrow heavily (sometimes verbatim) from Long 1997, *Regression Models for Categorical and Limited Dependent Variables*, and Long & Freese, 2003 *Regression Models for Categorical Dependent Variables Using Stata, Revised Edition*, and also the 2014 3<sup>rd</sup> edition of Long & Freese. For `rpoisson`, see *Right-censored Poisson regression model*, *Stata Journal* 2011, 11(1) pp. 95-105. Materials prepared by my former teaching assistant, the late Jamie Przybysz, are also incorporated in these notes.

Variables that count the # of times something happens are common in the Social Sciences.

- Hausman looked at effect of R & D expenditures on # of patents received by US companies
- Grogger examined deterrent effects of capital punishment on daily homicides
- King examined effect of # of alliances on the # of nations at war
- Long looked at # of publications of scientists

Count variables are often treated as though they are continuous and the linear regression model is applied; but this can result in inefficient, inconsistent and biased estimates. Fortunately, there are many models that deal explicitly with count outcomes.

- The most basic is the *Poisson Regression Model* (PRM). In the PRM the probability of a count is determined by a Poisson distribution, where the mean of the distribution is a function of the IVs. The conditional mean of the outcome is equal to the conditional variance.
- In practice, however, the conditional variance often exceeds the conditional mean. The *Negative Binomial Regression Model* (NBRM) deals with this problem by allowing the variance to exceed the mean.
- The PRM and NBRM are covered in the main handout. Other more advanced models are briefly covered in the appendix.
  - Many count variables are only observed after the first count occurs. For example, we may not have a list of every Sociologist; we only have a list of those who have published at least one article. This requires a *Truncated Count Model*.
  - Another problem is that the # of 0's in a sample often exceeds the # predicted by either the PRM or the NBRM. *Zero Modified Count Models* explicitly model the # of predicted 0s, and also allow the variance to differ from the mean. For example, sometimes the process that generates zeros is believed to be different than the process that generates higher counts. *Hurdle Models* and *Zero-Inflated Models* are among the ways to deal with this.

The Poisson Distribution.

Let  $y$  be a random variable indicating the # of times an event has occurred during an interval of time.  $y$  has a Poisson distribution with parameter  $\mu > 0$  if

$$\Pr(y | \mu) = \frac{\exp(-\mu)\mu^y}{y!} \quad \text{for } y = 0, 1, 2, \dots$$

# of occurrences	$\Pr(y=\# \text{ of occurrences}   \mu)$
0	$\text{Exp}(-\mu)$
1	$\text{Exp}(-\mu) \mu$
2	$\text{Exp}(-\mu) \mu^2/2$
3	$\text{Exp}(-\mu) \mu^3/6$
4	$\text{Exp}(-\mu) \mu^4/24$

So, for example, with 50 events occurring to 100 units, we find the following:

Prop(0) =  $[(.5^0)*(e^{-.5})/1] = .61$  (61 of the 100 units will experience no events)

Prop(1) =  $[(.5^1)*(e^{-.5})/1] = .30$  (30 of the 100 units will experience 1 event)

Prop(2) =  $[(.5^2)*(e^{-.5})/(2*1)] = .08$  (8 of the 100 units will experience 2 events)

Prop(3) =  $[(.5^3)*(e^{-.5})/(3*2*1)] = .01$  (1 of the 100 units will experience 3 events)

Prop(4) =  $[(.5^4)*(e^{-.5})/(4*3*2*1)] = .002$  (not substantively meaningful here, as it is too small,)

Prop(5) =  $[(.5^5)*(e^{-.5})/(5*4*3*2*1)] = .0002$  (but presented to show the example calculations )

This figure shows what the Poisson distribution looks like for different values of  $\mu$

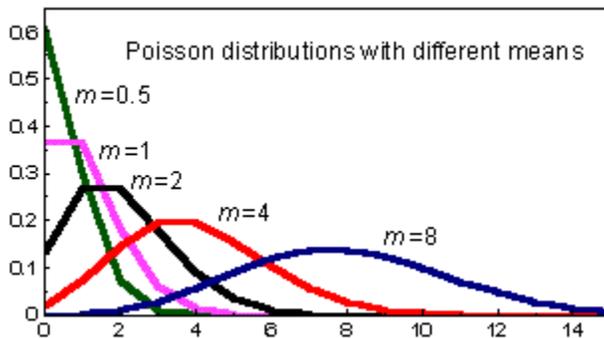


Image copied from <http://www.cmh.edu/stats/model/poiss10.htm>

Key properties of the Poisson distribution:

- As  $\mu$  increases, the mass of the distribution shifts to the right. Specifically,  $E(y) = \mu$ . The parameter  $\mu$  is known as the rate since it is the expected # of times that an event has occurred per unit of time.  $\mu$  can also be thought of as the mean or expected count.
- The variance equals the mean. The equality of the mean and the variance is known as *equidispersion*. In practice, count variables often have a variance that is greater than the mean, which is called *overdispersion*. The development of many models for count data is an attempt to account for overdispersion.
- As  $\mu$  increases, the probability of 0s decreases. For  $\mu = .8$ , the probability of a 0 is .45. For  $\mu = 1.5$ , it is .22, for  $\mu = 2.9$ , it is .05; and for  $\mu = 10.5$ , the probability is .00002. For many count variables, there are more observed 0s than predicted by the Poisson distribution.
- As  $\mu$  increases, the Poisson distribution approximates a normal distribution.

A critical assumption of a Poisson process is that events are independent; this means that when an event occurs it does not affect the probability of an event occurring in the future. For example, this implies that when a scientist publishes a paper, her rate of publication does not change. Past success in publishing does not affect future success.

As noted, the actual variance is often larger than a Poisson process would suggest. One likely explanation is that  $\mu$  differs across individuals, e.g. not all scientists are equally productive. This is known as heterogeneity. For example, suppose that for men, mean productivity =  $\mu + \delta$ , and for women it is  $\mu - \delta$ . If the number of men and women is equal, the mean productivity will be  $\mu$ , but the variance will exceed  $\mu$ . In general, failure to account for heterogeneity among individuals in the rate of a count variable leads to overdispersion. This leads to the Poisson Regression Model which introduces heterogeneity based on *observed* characteristics.

### Poisson Regression Model

In the PRM, the # of events  $y$  has a Poisson distribution with a conditional mean that depends on an individual's characteristics:

$$\mu_i = E(y_i | x_i) = \exp(x_i\beta)$$

Note the exponentiation forces the expected count to be positive. It can also be written as (and this is more consistent with the way we have written all our other models)

$$\ln(\mu_i) = x_i\beta$$

Under this model, as  $\mu$  increases, the conditional variance of  $y$  increases, the proportion of predicted 0s decreases and the distribution around the expected value becomes approximately normal.

The PRM can be thought of as a non-linear regression model with errors equal to  $\varepsilon = y - E(y|x)$ . The errors have a Poisson distribution. But, we cannot use OLS as the regression technique for data that resemble a Poisson distribution because in the Poisson, the mean ( $\mu$ ) = Variance of  $x$ . As  $\mu$  increases, so does the variance around it. (You'll recall that OLS assumes a constant variance.) The dispersion of data increases as  $\mu$  increases. Since the level of the DV affects dispersion, the errors in a Poisson regression are inherently heteroskedastic. The PRM is, in fact, another case of the Generalized Linear Model that we have been talking about and is estimated via maximum likelihood. The family is Poisson (errors have a Poisson distribution) and the link is log (the log of  $E(Y)$  is the dependent variable).

You can use the parameters to compute the probability distribution for a given level of the IVs. For a given  $x$ , the probability that  $y = m$  is

$$\hat{\Pr}(y = m | x) = \frac{\exp(-\hat{\mu})\hat{\mu}^m}{m!} \quad \text{where } \hat{\mu} = \exp(x\hat{\beta})$$

The PRM model should do better than a univariate Poisson distribution. Still, it can under predict 0s and have a variance that is greater than the conditional mean. Hence, other models have been developed which we will discuss shortly.

*Estimating the PRM in Stata.* The `poisson` command is used to estimate Poisson Regression Models. Long and Freese present an analysis of the number of publications produced by Ph.D. biochemists:

```
. use https://www3.nd.edu/~rwilliam/statafiles/couart4.dta, clear
(couart4.dta | Long data on Ph.D. biochemists | 2014-04-24)
```

```
. sum art female married kid5 mentor phd
```

Variable	Obs	Mean	Std. Dev.	Min	Max
art	915	1.692896	1.926069	0	19
female	915	.4601093	.4986788	0	1
married	915	.6622951	.473186	0	1
kid5	915	.495082	.76488	0	3
mentor	915	8.767213	9.483916	0	77
phd	915	3.103109	.9842491	.755	4.62

Note that the mean # of articles published is 1.69. Note too that the variance is  $1.926^2 = 3.71$ , which is substantially more than the mean.

We now estimate a simple model with constant-only. If this model is valid, then every academic biochemist has the same rate of productivity.

```
. poisson art, nolog
```

```
Poisson regression              Number of obs   =          915
                               LR chi2(0)         =           0.00
                               Prob > chi2          =           .
Log likelihood = -1742.5735      Pseudo R2       =          0.0000
```

	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]
art					
_cons	.5264408	.0254082	20.72	0.000	.4766416 .57624

Note that the coefficient for the constant is .52664408. Further, note that  $\exp(.52664408) = 1.693$ , the same as the mean given in the earlier descriptive statistics.

Your intuition probably tells you that this model does not make much sense – but how do you test it? You can do so with the `estat gof` post-estimation command (the older `poisgof` command also works)

```
. estat gof
```

```
Deviance goodness-of-fit = 1817.405
Prob > chi2(914)         = 0.0000

Pearson goodness-of-fit = 2002.901
Prob > chi2(914)         = 0.0000
```

This command compares the observed distribution with the distribution predicted by a Poisson distribution. The highly significant test statistic indicates that this is not a very good model. Long and Freese describe a procedure for comparing the predicted with the observed distribution. Their post-estimation command `mgen` computes the predicted rate and predicted probabilities of each count from 0 to the specified maximum for every observation.

```
. mgen, pr(0/9) meanpred stub(psn)
```

Predictions from:

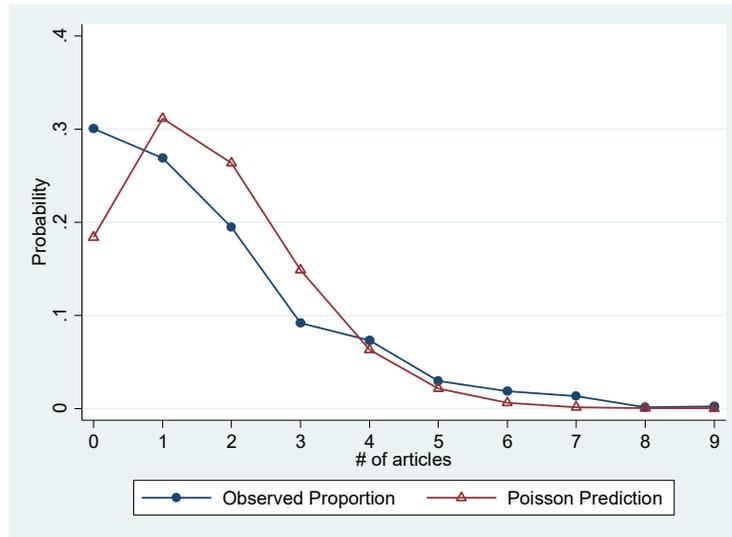
Variable	Obs	Unique	Mean	Min	Max	Label
psnval	10	10	4.5	0	9	Articles in last 3 yrs of PhD
psnobeq	10	10	.0993443	.0010929	.3005464	Observed proportion
psnoble	10	10	.8328962	.3005464	.9934427	Observed cum. proportion
psnpreq	10	10	.0999988	.0000579	.311469	Avg predicted Pr(y=#)
psnprle	10	10	.8307106	.1839859	.9999884	Avg predicted cum. Pr(y=#)
psnob_pr	10	10	-.0006546	-.0691068	.1165605	Observed - Avg Pr(y=#)

```
. label var psnobeq "Observed Proportion"
. label var psnpreq "Poisson Prediction"
. label var psnval "# of articles"
. list psnval psnobeq psnpreq in 1/10
```

```
+-----+
| psnval  psnobeq  psnpreq |
+-----+
1. |      0  .3005464  .1839859 |
2. |      1  .2688525  .311469  |
3. |      2  .1945355  .2636424 |
4. |      3  .0918033  .148773  |
5. |      4  .073224  .0629643 |
+-----+
6. |      5  .0295082  .0213184 |
7. |      6  .0185792  .006015  |
8. |      7  .0131148  .0014547 |
9. |      8  .0010929  .0003078 |
10. |      9  .0021858  .0000579 |
+-----+
```

As you can see, when the mean is 1.69, a Poisson distribution predicts that 18.39% of the cases will be zeros; but in reality more than 30% are. You also see more people than predicted in the 3+ range. If you want to graph this (and can remember the command!):

```
. graph twoway connected psnobeq psnpreq psnval, ///
> ytitle("Probability") ylabel(0(.1).4) xlabel(0/9) msym(O Th)
```



Of course, we never believed in that model anyway. Productivity may differ by gender, marital status, number of young children, prestige of the graduate program, and the number of articles written by a scientist's mentor. If so, mixing together scientists who differ in their rate of productivity can cause the univariate distribution of the articles to be overdispersed, i.e. have a variance greater than its mean. To account for these differences we add IVs to our model:

```
. poisson art i.female i.married kid5 phd mentor, 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]
female					
Female	-.2245942	.0546138	-4.11	0.000	-.3316352 -.1175532
married					
Married	.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
mentor	.0255427	.0020061	12.73	0.000	.0216109 .0294746
_cons	.3046168	.1029822	2.96	0.003	.1027755 .5064581

```
. estat gof
```

```
Deviance goodness-of-fit = 1634.371
Prob > chi2(909)        = 0.0000

Pearson goodness-of-fit  = 1662.547
Prob > chi2(909)        = 0.0000
```

Alas, the fit still isn't very good. Repeating our earlier procedure:

```
. mgen, pr(0/9) meanpred stub(psn) replace
```

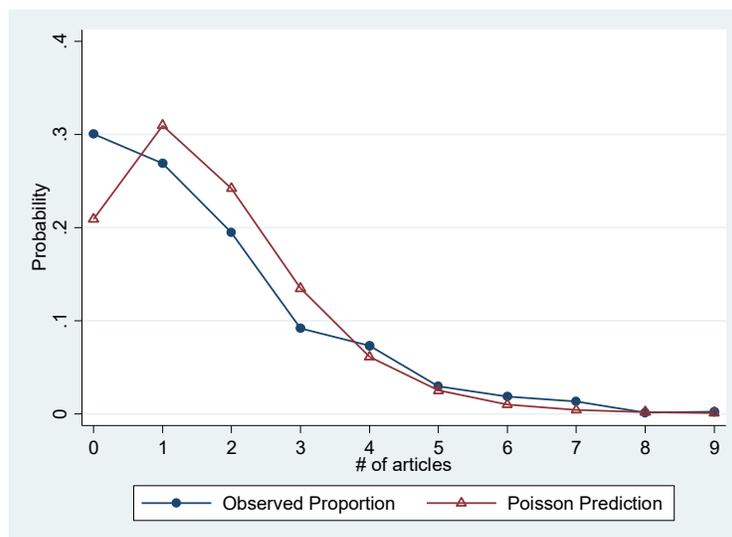
Predictions from:

Variable	Obs	Unique	Mean	Min	Max	Label
psnval	10	10	4.5	0	9	Articles in last 3 yrs of PhD
psnobeq	10	10	.0993443	.0010929	.3005464	Observed proportion
psnoble	10	10	.8328962	.3005464	.9934427	Observed cum. proportion
psnpreq	10	10	.0998819	.0009304	.3098447	Avg predicted Pr(y=#)
psnprle	10	10	.8308733	.2092071	.9988188	Avg predicted cum. Pr(y=#)
psnob_pr	10	10	-.0005376	-.0475604	.0913393	Observed - Avg Pr(y=#)

```
. label var psnobeq "Observed Proportion"
. label var psnpreq "Poisson Prediction"
. label var psnval "# of articles"
. list psnval psnobeq psnpreq in 1/10
```

	psnval	psnobeq	psnpreq
1.	0	.3005464	.2092071
2.	1	.2688525	.3098447
3.	2	.1945355	.242096
4.	3	.0918033	.1346656
5.	4	.073224	.0611696
6.	5	.0295082	.0249554
7.	6	.0185792	.0099346
8.	7	.0131148	.0041384
9.	8	.0010929	.001877
10.	9	.0021858	.0009304

```
. graph twoway connected psnobeq psnpreq psnval, ///
> ytitle("Probability") ylabel(0(.1).4) xlabel(0/9) msym(O Th)
```



Again, we see more observed zeroes than predicted zeros. We'll talk about some alternatives to this model, but first we'll talk about how to interpret the parameters we have got.

**Optional: Relationship to the Generalized Linear Model.** As noted before, Poisson Regression models are a special case of the Generalized Linear Model. Therefore they can also be estimated with the `glm` command:

```
. glm art i.female i.married kid5 phd mentor, family(poisson) link(log)

Iteration 0:  log likelihood = -1670.3221
Iteration 1:  log likelihood = -1651.1048
Iteration 2:  log likelihood = -1651.0563
Iteration 3:  log likelihood = -1651.0563

Generalized linear models              No. of obs      =          915
Optimization      : ML                 Residual df    =          909
                                                Scale parameter =           1
Deviance          = 1634.370984         (1/df) Deviance = 1.797988
Pearson           = 1662.54655          (1/df) Pearson  = 1.828984

Variance function: V(u) = u           [Poisson]
Link function     : g(u) = ln(u)       [Log]

Log likelihood    = -1651.056316      AIC             = 3.621981
                                                BIC             = -4564.031

-----
          |               OIM
          |      Coef.   Std. Err.      z    P>|z|    [95% Conf. Interval]
-----+-----
    female |
Female    |  -.2245942   .0546138   -4.11  0.000   - .3316352   -.1175532
          |
    married |
Married   |  .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
    mentor |  .0255427   .0020061   12.73  0.000   .0216109   .0294746
    _cons  |  .3046168   .1029822    2.96  0.003   .1027755   .5064581
-----
```

**Interpreting the Results of the PRM.** In their current form, the beta coefficients tell us how much a 1 unit increase in each  $X$  causes the log of  $\mu$  to increase. Since that isn't the most intuitive idea in the world, it will be useful to exponentiate the coefficients. We can do this by adding the `irr` parameter (which, mathematically, does the exact same thing as the odds ratio parameter we have used in the past; but `irr` stands for *incident rate ratio*, with the idea being that the coefficient tells you how changes in  $X$  affect the rate at which  $Y$  occurs (keeping in mind that the terms rate and mean stand for the same thing here.)

```
. poisson art i.female i.married kid5 phd mentor, nolog irr
```

```
Poisson regression                               Number of obs   =          915
                                                LR chi2(5)      =        183.03
                                                Prob > chi2     =          0.0000
Log likelihood = -1651.0563                    Pseudo R2      =          0.0525
```

art	IRR	Std. Err.	z	P> z	[95% Conf. Interval]	
female						
Female	.7988403	.0436277	-4.11	0.000	.7177491	.8890932
married						
Married	1.167942	.0716821	2.53	0.011	1.035569	1.317236
kid5	.8312018	.0333538	-4.61	0.000	.7683342	.8992134
phd	1.012905	.0267379	0.49	0.627	.9618325	1.06669
mentor	1.025872	.002058	12.73	0.000	1.021846	1.029913
_cons	1.356105	.1396546	2.96	0.003	1.108243	1.659403

These coefficients tell us that, on an all other things equal basis,

- Females publish 80% as many articles as males, i.e. are 20% less productive
- Married people are about 17% more productive than unmarried people
- Each additional child multiplies the rate of productivity by .83, e.g. somebody with one child will only produce 83% as many articles as somebody with no children.
- The prestige of the PHD institution doesn't have much effect
- For each additional article a mentor publishes, productivity gets multiplied by 1.025872, i.e. there is about a 2.6% increase per article. (But remember, you do compounding, not addition, as you figure the effect of increases in X that are greater than one.

*Optional:* Old commands used in slightly new ways. The margins command is also helpful. Note that the default asobserved is being used instead of atmeans.

```
. margins female married
```

```
Predictive margins                               Number of obs   =          915
Model VCE      : OIM
Expression     : Predicted number of events, predict()
```

	Margin	Delta-method Std. Err.	z	P> z	[95% Conf. Interval]	
female						
Male	1.863249	.062788	29.68	0.000	1.740187	1.986312
Female	1.488439	.0614126	24.24	0.000	1.368072	1.608805
married						
Single	1.526787	.0742234	20.57	0.000	1.381312	1.672263
Married	1.7832	.0576126	30.95	0.000	1.670281	1.896118

```
. margins, dydx(*)
```

```
Average marginal effects          Number of obs   =          915  
Model VCE      : OIM
```

```
Expression      : Predicted number of events, predict()  
dy/dx w.r.t.    : 1.female 1.married kid5 phd mentor
```

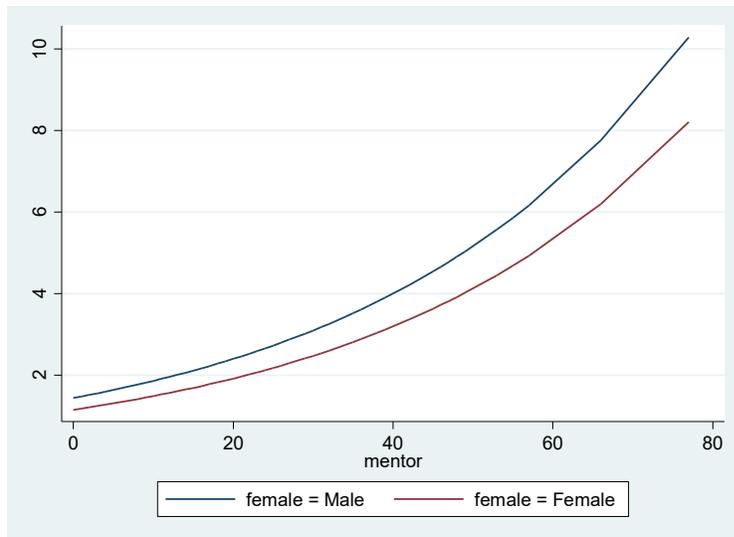
	dy/dx	Delta-method Std. Err.	z	P> z	[95% Conf. Interval]	
female						
Female	-.3748107	.0900846	-4.16	0.000	-.5513733	-.1982481
married						
Married	.256412	.0990332	2.59	0.010	.0623105	.4505135
kid5	-.3129872	.068395	-4.58	0.000	-.447039	-.1789354
phd	.0217073	.0446911	0.49	0.627	-.0658857	.1093003
mentor	.0432412	.0035694	12.11	0.000	.0362454	.0502371

Note: dy/dx for factor levels is the discrete change from the base level.

The results tell us that, after controlling for other variables, on average woman publish .375 fewer articles than men; and on average, married people publish .256 more articles.

The mcp command is also good:

```
. mcp mentor female
```



Even though there are no interaction terms in the model, we see that the expected differences between men and women are small when mentors have low productivity, and gradually become much larger.

mchange continues to be useful:

```
. mchange
```

```
poisson: Changes in mu | Number of obs = 915
```

```
Expression: Predicted number of art, predict()
```

	Change	p-value
female		
Female vs Male	-0.375	0.000
married		
Married vs Single	0.256	0.010
kid5		
+1	-0.286	0.000
+SD	-0.223	0.000
Marginal	-0.313	0.000
phd		
+1	0.022	0.629
+SD	0.022	0.629
Marginal	0.022	0.627
mentor		
+1	0.044	0.000
+SD	0.464	0.000
Marginal	0.043	0.000

```
Average prediction
```

```
1.693
```

The listcoef command can also be used here:

```
. listcoef, help
```

```
poisson (N=915): Factor change in expected count
```

```
Observed SD: 1.9261
```

	b	z	P> z	e^b	e^bStdX	SDofX
female						
Female	-0.2246	-4.112	0.000	0.799	0.894	0.499
married						
Married	0.1552	2.529	0.011	1.168	1.076	0.473
kid5						
kid5	-0.1849	-4.607	0.000	0.831	0.868	0.765
phd						
phd	0.0128	0.486	0.627	1.013	1.013	0.984
mentor						
mentor	0.0255	12.733	0.000	1.026	1.274	9.484
constant						
constant	0.3046	2.958	0.003	.	.	.

```
b = raw coefficient
```

```
z = z-score for test of b=0
```

```
P>|z| = p-value for z-test
```

```
e^b = exp(b) = factor change in expected count for unit increase in X
```

```
e^bStdX = exp(b*SD of X) = change in expected count for SD increase in X
```

```
SDofX = standard deviation of X
```

The main additional piece of information you are gaining here is the effect on productivity of a 1 standard deviation increase in X. Alternatively, we can get the percent change produced by changes in X with the following:

```
. listcoef, help percent
poisson (N=915): Percentage change in expected count

Observed SD: 1.9261
```

	b	z	P> z	%	%StdX	SDofX
female						
Female	-0.2246	-4.112	0.000	-20.1	-10.6	0.499
married						
Married	0.1552	2.529	0.011	16.8	7.6	0.473
kid5						
kid5	-0.1849	-4.607	0.000	-16.9	-13.2	0.765
phd						
phd	0.0128	0.486	0.627	1.3	1.3	0.984
mentor						
mentor	0.0255	12.733	0.000	2.6	27.4	9.484
constant						
constant	0.3046	2.958	0.003	.	.	.

```

b = raw coefficient
z = z-score for test of b=0
P>|z| = p-value for z-test
% = percent change in expected count for unit increase in X
%StdX = percent change in expected count for SD increase in X
SDofX = standard deviation of X

```

**Exposure time.** So far we have implicitly assumed that each observation was “at risk” of an event occurring for the same amount of time. This need not be true; for example, scientists may have received their Ph.D.s in different years. Amount of time in career will certainly affect the number of publications. Further, if exposure time is correlated with our variables, e.g. men have had the Ph.D.s longer than women have, we may get very misleading results.

Since the data from our example do not include exposure data, we will make some up. The variable `profage` corresponds to the scientists professional age which corresponds to the amount of time a scientist has been exposed to the risk of publishing. In the following, men have an average professional age of 30, while women have an average professional age of 15:

```
. set seed 123456
. gen profage = (10 + invnorm(uniform())) * 3 if female == 0
(421 missing values generated)
. set seed 1234567
. replace profage = (5 + invnorm(uniform())) * 3 if female == 1
(421 real changes made)
. bysort female: sum profage
```

-----  
-> female = Male

Variable	Obs	Mean	Std. Dev.	Min	Max
profage	494	29.98072	2.888995	21.81031	39.3049

-----  
-> female = Female

Variable	Obs	Mean	Std. Dev.	Min	Max
profage	421	14.73158	3.16198	6.919592	24.08156

As Long and Freese note, there are different ways to incorporate exposure time into Poisson models. The simplest may be to use the `exposure` option.

```
. poisson art i.female i.married kid5 phd ment, nolog exposure(profage) irr
```

```
Poisson regression                               Number of obs   =          915
                                                LR chi2(5)      =        239.98
                                                Prob > chi2     =         0.0000
Log likelihood = -1671.3634                    Pseudo R2      =         0.0670
```

art	IRR	Std. Err.	z	P> z	[95% Conf. Interval]
female					
Female	1.631257	.0889582	8.97	0.000	1.465896 1.815271
married					
Married	1.1572	.0709218	2.38	0.017	1.02622 1.304897
kid5	.8341539	.0333962	-4.53	0.000	.7712009 .9022459
phd	1.025738	.0270875	0.96	0.336	.9739977 1.080226
mentor	1.025974	.002049	12.84	0.000	1.021966 1.029998
_cons	.0435926	.0045095	-30.29	0.000	.0355926 .0533908
ln(profage)	1	(exposure)			

Notice how this dramatically changes our estimate of the effect of gender; once we control for exposure time, women are much more productive than men. In other words, their lower productivity is due to the fact that they haven't had their Ph.Ds as long. Hence, failing to control for exposure time could create a very misleading impression.

---

## Negative Binomial Regression Model

The PRM accounts for observed heterogeneity (i.e. observed differences among sample members) by specifying the rate  $\mu$  as a function of the observed  $X$ s. In practice, the PRM rarely fits, because of overdispersion. That is, the model underestimates the amount of dispersion in the outcome. If the mean structure from the PRM is correct, but there is overdispersion in the estimates,

- PRM estimates are consistent, but inefficient
- Standard errors will be biased downward resulting in spuriously large z-values

The NBRM adds a parameter that allows the conditional variance of  $y$  to exceed the conditional mean. In the NBRM, the mean  $\mu$  is replaced with the random variable  $\tilde{\mu}$ :

$$\tilde{\mu}_i = \exp(x_i\beta + \varepsilon_i)$$

where  $\varepsilon$  is a random error that is assumed to be uncorrelated with  $x$ . You can think of  $\varepsilon$  as either the combined effects of unobserved variables that have been omitted from the model or as another source of pure randomness.

Put another way, in the PRM, variation in  $\mu$  is introduced through *observed heterogeneity*. In the NBRM, you also have variation due to *unobserved heterogeneity*. For a given combination of  $x$ s there is a distribution of  $\mu$ s rather than a single  $\mu$ . The conditional mean is still  $\mu$ , but the variance will be greater because of the error term.

*Optional.* The relationship between  $\mu$ -squiggle and  $\mu$  is

$$\tilde{\mu}_i = \exp(x_i\beta) \exp(\varepsilon_i) = \mu_i \exp(\varepsilon_i) = \mu \delta_i$$

The NBRM is not identified without an assumption about the mean of the error term, and the most convenient assumption is that the mean is 1. (This is analogous to assuming in OLS regression that the mean of the residuals is 0). Hence,

$$\tilde{\mu}_i = \exp(x_i\beta) \exp(\varepsilon_i) = \mu_i \exp(\varepsilon_i) = \mu \delta_i = \mu_i$$

What is the distribution of  $\delta$ ? The most common assumption is that  $\delta$  has a gamma distribution with parameter  $v$ . If  $\delta$  has a gamma distribution, then  $E(\delta) = 1$  and  $\text{Var}(\delta) = 1/v$ .

The expected value of  $y$  for the Negative Binomial distribution is the same as for the Poisson distribution, but the conditional variance differs:

$$\text{Var}(y_i | x) = \mu_i \left( 1 + \frac{\mu_i}{v_i} \right) = \exp(x_i\beta) \left( 1 + \frac{\exp(x_i\beta)}{v_i} \right)$$

Since  $\mu$  and  $v$  are positive, the conditional variance of  $y$  in the NBRM must exceed the conditional mean  $\exp(x\beta)$ .

If  $v$  varies by individuals, then there are more parameters than there are observations. The most common identifying assumption is that  $v$  is the same for all individuals (again note the similarities with OLS):

$$v_i = \alpha^{-1} \quad \text{for } \alpha > 0$$

$\alpha$  is known as the *dispersion parameter* since increasing  $\alpha$  increases the conditional variance of  $y$ . Substituting back into our formula for the conditional variance of  $y$ ,

$$\text{Var}(y_i | x) = \mu_i \left( 1 + \frac{\mu_i}{\alpha} \right) = \exp(x_i \beta) \left( 1 + \frac{\exp(x_i \beta)}{v_i} \right) = \mu_i (1 + \alpha \mu_i) = \mu_i + \alpha \mu_i^2$$

Note that, if  $\alpha = 0$ , the mean and variance become one and the same, and you have a Poisson model.

The larger conditional variance in  $y$  increases the relative frequency of low and high counts. The NB distribution corrects a number of sources of poor fit that are often found when the Poisson distribution is used:

- The variance of the NB distribution exceeds the variance of the Poisson distribution for a given mean
- The increased variance in the NBRM results in substantially larger probabilities for small counts.
- There are slightly larger probabilities for larger counts in the NB distribution.

**Optional: Heterogeneity and Contagion.** Our discussion so far has motivated the NB distribution by talking about unobserved heterogeneity. An alternative derivation is based on the idea of *contagion*. Contagion occurs when individuals with a given set of  $X$ s have the same probability of an event occurring, but this probability changes as events occur. For example, suppose a scientist publishes a paper. Her rate of productivity may go up as a result of contagion from the initial publication. She might receive additional resources as a result of her success which will lead to further increases in productivity. A second scientist, who had the same initial rate of productivity, would have his rate stay the same so long as he did not publish. The process is contagious in the sense that success in publishing increases the rate of future publishing. Contagion violates the independence assumption of the Poisson distribution.

Unobserved heterogeneity and contagion can both generate the same NB distribution of observed counts. Consequently, heterogeneity is sometimes referred to as “spurious” or “apparent” contagion, as opposed to “true” contagion. *With cross-sectional data, it is impossible to determine whether the observed distribution of counts arose from true or spurious contagion.*

**Testing for overdispersion.** Remember that, with the PRM, if overdispersion is present then estimates are inefficient and standard errors are biased downward. It is therefore important to test for overdispersion. There are various ways to do this. The approaches described below take advantage of the fact that the PRM is a special case of the NBRM, when  $\alpha = 0$ . Alpha is the dispersion parameter; if it equals 0 there is no over-dispersion.

1. You can do a 1-tailed test of  $H_0: \alpha = 0$ . (The test is one-tailed, because  $\alpha$  cannot be less than zero.) Stata’s `nbreg` routine reports this for you automatically:

```
. use https://www3.nd.edu/~rwilliam/statafiles/couart4.dta, clear
(couart4.dta | Long data on Ph.D. biochemists | 2014-04-24)
```

```
. nbreg art i.female i.married kid5 phd ment, nolog
```

```
Negative binomial regression          Number of obs   =          915
                                      LR chi2(5)       =          97.96
Dispersion      = mean                Prob > chi2     =          0.0000
Log likelihood = -1560.9583           Pseudo R2      =          0.0304
```

	art	Coef.	Std. Err.	z	P> z	[95% Conf. Interval]	
	female						
	Female	-.2164184	.0726724	-2.98	0.003	-.3588537	-.0739832
	married						
	Married	.1504895	.0821063	1.83	0.067	-.0104359	.3114148
	kid5	-.1764152	.0530598	-3.32	0.001	-.2804105	-.07242
	phd	.0152712	.0360396	0.42	0.672	-.0553652	.0859075
	mentor	.0290823	.0034701	8.38	0.000	.0222811	.0358836
	_cons	.256144	.1385604	1.85	0.065	-.0154294	.5277174
	/lnalpha	-.8173044	.1199372			-1.052377	-.5822318
	alpha	.4416205	.0529667			.3491069	.5586502

```
Likelihood-ratio test of alpha=0:  chibar2(01) = 180.20 Prob>=chibar2 = 0.000
```

As we see from the last line of the printout, alpha significantly differs from 0. Incidentally, what the program actually estimates is  $\ln(\alpha)$ . This forces the estimated alpha to be positive.

2. You can do the LR chi-square test yourself by estimating both the Poisson and NBRM:

```
. quietly poisson art i.female i.married kid5 phd ment, nolog
. est store poisson
. quietly nbreg art i.female i.married kid5 phd ment, nolog
. est store nbreg
. lrtest poisson nbreg, stats force
```

```
Likelihood-ratio test          LR chi2(1) = 180.20
(Assumption: poisson nested in nbreg) Prob > chi2 = 0.0000
```

```
Akaike's information criterion and Bayesian information criterion
```

	Model	Obs	ll(null)	ll(model)	df	AIC	BIC
	poisson	915	-1742.573	-1651.056	6	3314.113	3343.026
	nbreg	915	-1609.937	-1560.958	7	3135.917	3169.649

```
Note: N=Obs used in calculating BIC; see [R] BIC note
```

3. You can do a Wald test of  $\ln(\alpha) = 1$  (which corresponds to a test of  $\alpha = 0$ ):

```
. test [lnalpha]_cons = 1
( 1) [lnalpha]_cons = 1
      chi2( 1) = 229.59
      Prob > chi2 = 0.0000
```

To confirm this:

$$\frac{-0.8173044 - 1}{0.1199372} = \frac{-1.8173044}{0.1199372} = 15.15213295$$

Square the above and you get 229.59

Clearly, overdispersion is a problem with the PRM in this case, and the NBRM should be preferred. This side by side comparison of the PRM and NBRM further illustrates the point:

```
. est table poisson nbreg, t varlabel varwidth(32) stats(alpha N) b(%9.3f)
```

	Variable	poisson	nbreg
art	female		
	Female	-0.225	-0.216
		-4.11	-2.98
	married		
	Married	0.155	0.150
		2.53	1.83
	# of kids < 6	-0.185	-0.176
		-4.61	-3.32
	PhD prestige	0.013	0.015
		0.49	0.42
	Mentor's # of articles	0.026	0.029
	12.73	8.38	
Constant	0.305	0.256	
	2.96	1.85	
lnalpha	Constant		-0.817
			-6.81
Statistics	alpha		0.442
	N	915	915

legend: b/t

As we see, the Poisson distribution consistently has higher t values than the NBREG distribution. The Poisson estimates are less precise and you are more likely to conclude that an effect differs from zero when in reality it does not.

**Optional:** Interpretation. Interpretation of the NBRM is pretty much the same as the PRM. Using the margins command,

```
. quietly nbreg art i.female i.married kid5 phd ment
. margins female married
```

```
Predictive margins                                Number of obs   =           915
Model VCE      : OIM

Expression    : Predicted number of events, predict()
```

	Margin	Delta-method Std. Err.	z	P> z	[95% Conf. Interval]	
female						
Male	1.868735	.0869613	21.49	0.000	1.698294	2.039176
Female	1.505076	.0823171	18.28	0.000	1.343737	1.666414
married						
Single	1.542236	.1002205	15.39	0.000	1.345808	1.738665
Married	1.7927	.079988	22.41	0.000	1.635926	1.949474

```
. margins, dydx(*)
```

```
Average marginal effects                          Number of obs   =           915
Model VCE      : OIM

Expression    : Predicted number of events, predict()
dy/dx w.r.t. : 1.female 1.married kid5 phd mentor
```

	dy/dx	Delta-method Std. Err.	z	P> z	[95% Conf. Interval]	
female						
Female	-.3636591	.1211958	-3.00	0.003	-.6011984	-.1261197
married						
Married	.2504638	.1337954	1.87	0.061	-.0117703	.512698
kid5	-.3007755	.0914704	-3.29	0.001	-.4800543	-.1214967
phd	.0260362	.0614472	0.42	0.672	-.0943981	.1464706
mentor	.0495833	.0065477	7.57	0.000	.0367501	.0624166

Note: dy/dx for factor levels is the discrete change from the base level.

Using the listcoef command,

```
. listcoef
```

```
nbreg (N=915): Factor change in expected count
```

```
Observed SD: 1.9261
```

	b	z	P> z	e^b	e^bStdX	SDofX
female						
Female	-0.2164	-2.978	0.003	0.805	0.898	0.499
married						
Married	0.1505	1.833	0.067	1.162	1.074	0.473
kid5	-0.1764	-3.325	0.001	0.838	0.874	0.765
phd	0.0153	0.424	0.672	1.015	1.015	0.984
mentor	0.0291	8.381	0.000	1.030	1.318	9.484
constant	0.2561	1.849	0.065	.	.	.

alpha	lnalpha	alpha
lnalpha	-0.8173	.
alpha	0.4416	.

```
LR test of alpha=0: 180.20 Prob>=LRX2 = 0.000
```

Perhaps the most helpful column is e^b (which you can also get by specifying the irr option on nbreg). If you prefer, you can get equivalent results with the percent option:

```
. listcoef, percent
```

```
nbreg (N=915): Percentage change in expected count
```

```
Observed SD: 1.9261
```

	b	z	P> z	%	%StdX	SDofX
female						
Female	-0.2164	-2.978	0.003	-19.5	-10.2	0.499
married						
Married	0.1505	1.833	0.067	16.2	7.4	0.473
kid5	-0.1764	-3.325	0.001	-16.2	-12.6	0.765
phd	0.0153	0.424	0.672	1.5	1.5	0.984
mentor	0.0291	8.381	0.000	3.0	31.8	9.484
constant	0.2561	1.849	0.065	.	.	.

alpha	lnalpha	alpha
lnalpha	-0.8173	.
alpha	0.4416	.

```
LR test of alpha=0: 180.20 Prob>=LRX2 = 0.000
```

Looking at the % column, we see that, on an all other things equal basis, women are 19.5% less productive than men; married people are 16.2% more productive; each additional child lowers productivity by 16.2% (again, remember to compound, not add, for units greater than 1, e.g. somebody with 3 kids would have a rate  $.8383^3 = 58.9\%$  as great as somebody with no children); each additional article by a mentor adds 3% productivity.

See Long and Freese (2014) for additional examples, or else try the commands yourself. There aren't many new surprises here given what we have gone over before.

## Appendix: Advanced Models for Count Outcomes (Optional)

This appendix briefly covers more advanced count models that are sometimes needed. You should read Long & Freese or other sources if you need to better understand or use these models. The models discussed are

- **Zero Truncated Models** – these are used when the sample selection procedures exclude cases with 0 counts. For example, we may not have a list of every Sociologist; we only have a list of those who have published at least one article.
- **Hurdle Models** – Sometimes you may believe that zeros are generated by a different process from that of positive counts. Zero is a “hurdle” that you have to get past before reaching positive counts (but everyone has a nonzero probability of doing so). Hurdle regression models combine a binary model (e.g. logit) to predict zeros with a zero-truncated Poisson or zero-truncated negative binomial model to predict nonzero counts.
- **Zero-Inflated Count Models** – Zero-inflated models assume that there are two latent groups. One group has no chance of going beyond zero, e.g. they might be scientists in fields or companies that do not allow publishing. We call this Group A, the Always Zero Group. Members of the other group may have a zero count, but the probability of having a positive count is nonzero, e.g. a scientist who could publish may or may not do so. We call this Group –A, the Not Always Zero Group. Zero-Inflated models allow for this possibility, thereby increasing the conditional variance and the probability of zero counts.

### Models for Truncated Counts

Sometimes observations with outcomes equal to zero are missing from the sample because of the way the data are collected. For example, we may not have a list of every Sociologist; we only have a list of those who have published at least one article. Or, a survey of how often people visit the shopping mall may be done of people who are currently at the mall. Of, if you have bought a TV, the warranty card may ask you how many other TVs you have. In each case, observations with a value of 0 are not included in the sample. Zero-truncated count models are designed for such situations.

Long & Freese (2014) go through the math on pp. 519-520. A key thing to note is that the adverse effects of over-dispersion are worse with truncated models. Estimates are biased and inefficient if there is overdispersion. You should estimate a zero-truncated negative binomial model to test for overdispersion.

The `ztp` (zero truncated Poisson) and `ztnb` (zero truncated negative binomial) commands can be used. Output is similar to the `poisson` and `nbreg` commands. If zero counts are missing from your data because of the way the data were collected, and zero counts are generated by the same process as positive counts, interpretation is also similar.

NOTE: In Stata 12, these commands were replaced by `tpoisson` and `tnbreg`. Their main advantage is that you can specify a truncation point other than zero.

NOTE: There is also a user-written command for right-censored data called `rcpoisson`. Quoting from the article that introduced the command,

For example, a researcher who is interested in alcohol consumption patterns among male college students may define binge drinking as “five or more drinks in one sitting” and may code the dependent variable as 0, 1, 2, . . . , 5 or more drinks. In this case, the number of drinks consumed will be censored at five... Applying a traditional Poisson regression model to censored data will produce biased and inconsistent estimates. Intuitively, when the data are right-censored, large values of the dependent variable are coded as small and the conditional mean of the dependent variable and the marginal effects will be attenuated.

## Hurdle Models

Sometimes you may believe that zeros are generated by a different process from that of positive counts. Zero is a “hurdle” that you have to get past before reaching positive counts (but everyone has a nonzero probability of doing so). Hurdle regression models combine a binary model (e.g. logit) to predict zeros with a zero-truncated Poisson or zero-truncated negative binomial model to predict nonzero counts.

Long & Freese (2014) show how to estimate hurdle models, even though there is no “official” Stata command for doing so. (There are some user-written commands by the late Joe Hilbe, but I am not sure they still work correctly.) Compare the following results with those reported by Long & Freese (2014) on p. 530:

```
. * Hurdle Models - Adapted from Long & Freese 3rd edition, section 9.5
. use https://www3.nd.edu/~rwilliam/statafiles/couart4, clear
(couart4.dta | Long data on Ph.D. biochemists | 2014-04-24)

. * Run logit model. Nonzero values are treated as 1s by logit command.
. quietly logit art female married kid5 phd mentor, or nolog
. est store Hlogit

. * Run Zero-truncated nbreg. Keep cases where articles are > 0
. quietly ztnb art female married kid5 phd mento if art>0, nolog irr
. est store Hztnb

. * You need suest to get the standard errors right.
```

```
. suest Hlogit Hztnb, vce(robust) eform(expB)
```

Simultaneous results for Hlogit, Hztnb

Number of obs = 915

	expB	Robust Std. Err.	z	P> z	[95% Conf. Interval]	
-----						
Hlogit_art						
female	.7779048	.1215446	-1.61	0.108	.5727031	1.056631
married	1.385739	.2475798	1.83	0.068	.9763455	1.966796
kid5	.7518272	.0831787	-2.58	0.010	.6052643	.93388
phd	1.022468	.0822485	0.28	0.782	.8733296	1.197075
mentor	1.083419	.0154716	5.61	0.000	1.053515	1.114171
_cons	1.267183	.3694752	0.81	0.417	.7155706	2.244016
-----						
Hztnb_art						
female	.7829619	.0724833	-2.64	0.008	.6530404	.9387312
married	1.108954	.1169726	0.98	0.327	.9018383	1.363636
kid5	.8579072	.0626125	-2.10	0.036	.743562	.9898365
phd	.9970707	.0504934	-0.06	0.954	.9028585	1.101114
mentor	1.024022	.0050724	4.79	0.000	1.014129	1.034012
_cons	1.426359	.27488	1.84	0.065	.9776649	2.080979
-----						
/Hztnb						
lnalpha	.5469076	.1302053			.3429761	.8720957
-----						

```
. * See Long & Freese section 9.5 for computing predictions, using margins
```

The logit equation tells you what affects the likelihood of clearing the zero “hurdle.” Women, and those with kids under 5, are less likely to clear the hurdle. Married people, those who went to more prestigious PHD institutions, and those whose mentors are more productive are more likely to clear the hurdle. (Not all effects are significant though.)

In the zero-truncated negative binomial part of the model, the coefficients indicate whether increases in the variable increase or decrease productivity. A variable can be significant in one part of the model, but not in the other part.

Long & Freese (2014) show how to run margins and get the predicted probabilities of different counts, e.g. 0 articles, one article, etc. If you want them you have to use Long and Freese’s rather lengthy but straightforward procedure.

## Zero-Inflated Count Models

Zero-inflated models assume that there are two latent groups. One group has no chance of going beyond zero, e.g. they might be scientists in fields or companies that do not allow publishing. We call this Group A, the Always Zero Group. Members of the other group may have a zero count, but the probability of having a positive count is nonzero, e.g. a scientist who could publish may or may not do so. We call this Group –A, the Not Always Zero Group. Zero-Inflated models allow for this possibility, thereby increasing the conditional variance and the probability of zero counts.

Estimating such models is a 3-step process. First, you model membership into the latent groups. Then, you model the counts for those in Group –A (Not Always Zero). Finally, you compute observed probabilities as a mixture of the probabilities for the two groups.

The commands are `zip` and `zinb`. They include an `inflate` option. The vars specified in the `inflate` option are used to predict group membership.

Long and Freese give examples and show how to make interpretation of results easier.

NOTE: Paul Allison (<http://www.statisticalhorizons.com/zero-inflated-models>) asks “Do we really need zero-inflated models?” He says

In all data sets that I’ve examined, the negative binomial model fits much better than a ZIP model, as evaluated by AIC or BIC statistics. And it’s a much simpler model to estimate and interpret. So if the choice is between ZIP and negative binomial, I’d almost always choose the latter.

But what about the zero-inflated negative binomial (ZINB) model? It’s certainly possible that a ZINB model could fit better than a conventional negative binomial regression model. But the latter is a special case of the former, so it’s easy to do a likelihood ratio test to compare them (by taking twice the positive difference in the log-likelihoods). In my experience, the difference in fit is usually trivial...

So next time you’re thinking about fitting a zero-inflated regression model, first consider whether a conventional negative binomial model might be good enough. Having a lot of zeros doesn’t necessarily mean that you need a zero-inflated model.

---

## Comparisons of Count Models

Long & Freese’s `countfit` command makes it easy to compare the results of PRM, NBRM, ZIP, and ZINB models.

```
. use http://www.indiana.edu/~jslsoc/stata/spex_data/couart4, clear
(couart4.dta | Long data on Ph.D. biochemists | 2014-04-24)
```

```
. countfit art i.female i.married kid5 phd mentor, inflate(mentor i.female) replace
```

Variable		PRM	NBRM	ZIP	ZINB
-----					
art					
	female				
	Female	0.799	0.805	0.812	0.836
		-4.11	-2.98	-3.31	-2.40
	married				
	Married	1.168	1.162	1.142	1.150
		2.53	1.83	2.01	1.72
	# of kids < 6	0.831	0.838	0.849	0.845
		-4.61	-3.32	-3.77	-3.22
	PhD prestige	1.013	1.015	0.993	1.001
		0.49	0.42	-0.24	0.04
	Mentor's # of articles	1.026	1.030	1.018	1.025
		12.73	8.38	8.09	7.07
	Constant	1.356	1.292	1.874	1.465
		2.96	1.85	5.54	2.69
-----					
lnalpha					
	Constant		0.442		0.375
			-6.81		-7.06
-----					
inflate					
	Mentor's # of articles			0.876	0.470
				-3.23	-2.55
	female				
	Female			1.120	2.868
				0.42	1.40
	Constant			0.484	0.275
				-3.15	-2.14
-----					
Statistics					
	alpha		0.442		
	N	915	915	915	915
	ll	-1651.056	-1560.958	-1605.644	-1552.034
	bic	3343.026	3169.649	3272.659	3172.257
	aic	3314.113	3135.917	3229.288	3124.068
-----					

legend: b/t

Comparison of Mean Observed and Predicted Count

Model	Maximum Difference	At Value	Mean  Diff
PRM	0.091	0	0.026
NBRM	-0.015	3	0.006
ZIP	0.052	1	0.014
ZINB	-0.019	3	0.008

PRM: Predicted and actual probabilities

Count	Actual	Predicted	Diff	Pearson
0	0.301	0.209	0.091	36.489
1	0.269	0.310	0.041	4.962
2	0.195	0.242	0.048	8.549
3	0.092	0.135	0.043	12.483
4	0.073	0.061	0.012	2.174
5	0.030	0.025	0.005	0.760
6	0.019	0.010	0.009	6.883
7	0.013	0.004	0.009	17.815
8	0.001	0.002	0.001	0.300
9	0.002	0.001	0.001	1.550
Sum	0.993	0.999	0.259	91.964

NBRM: Predicted and actual probabilities

Count	Actual	Predicted	Diff	Pearson
0	0.301	0.304	0.003	0.028
1	0.269	0.272	0.003	0.039
2	0.195	0.180	0.014	1.066
3	0.092	0.106	0.015	1.818
4	0.073	0.060	0.013	2.753
5	0.030	0.033	0.004	0.348
6	0.019	0.018	0.000	0.004
7	0.013	0.010	0.003	0.719
8	0.001	0.006	0.005	3.593
9	0.002	0.004	0.001	0.456
Sum	0.993	0.993	0.062	10.824

ZIP: Predicted and actual probabilities

Count	Actual	Predicted	Diff	Pearson
0	0.301	0.298	0.003	0.022
1	0.269	0.217	0.052	11.526
2	0.195	0.210	0.016	1.095
3	0.092	0.142	0.050	16.281
4	0.073	0.076	0.002	0.071
5	0.030	0.034	0.005	0.612
6	0.019	0.014	0.005	1.346
7	0.013	0.005	0.008	9.840
8	0.001	0.002	0.001	0.447
9	0.002	0.001	0.001	1.985
Sum	0.993	0.999	0.143	43.225

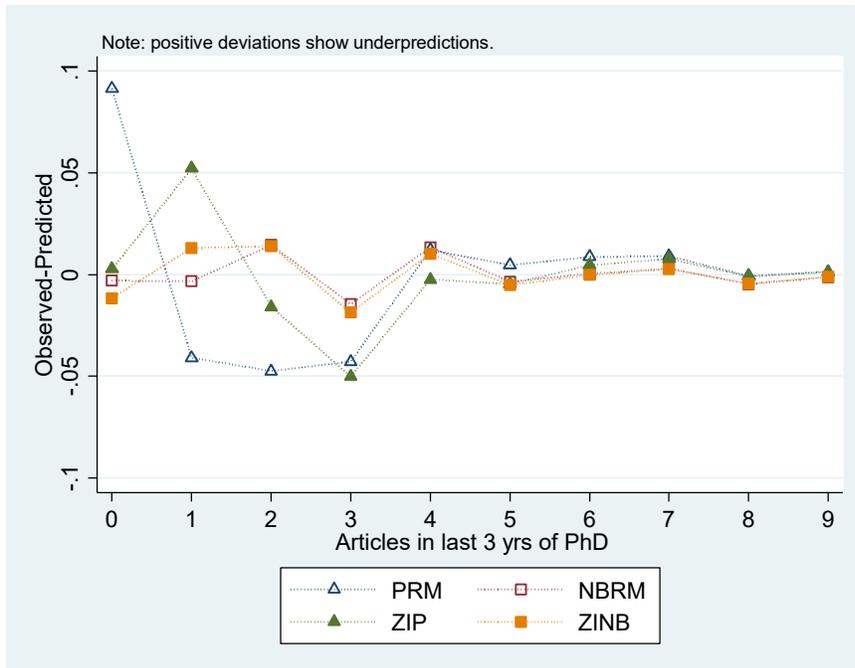
ZINB: Predicted and actual probabilities

Count	Actual	Predicted	Diff	Pearson
0	0.301	0.312	0.012	0.396
1	0.269	0.256	0.013	0.611
2	0.195	0.181	0.014	0.981
3	0.092	0.110	0.019	2.889
4	0.073	0.063	0.010	1.524
5	0.030	0.035	0.005	0.709
6	0.019	0.019	0.000	0.004
7	0.013	0.010	0.003	0.710
8	0.001	0.006	0.005	3.397
9	0.002	0.003	0.001	0.302
Sum	0.993	0.995	0.081	11.522

Tests and Fit Statistics

PRM	BIC=	3343.026	AIC=	3314.113	Prefer	Over	Evidence
vs NBRM	BIC=	3169.649	dif=	173.377	NBRM	PRM	Very strong
	AIC=	3135.917	dif=	178.196	NBRM	PRM	
	LRX2=	180.196	prob=	0.000	NBRM	PRM	p=0.000
vs ZIP	BIC=	3272.659	dif=	70.367	ZIP	PRM	Very strong
	AIC=	3229.288	dif=	84.824	ZIP	PRM	
	Vuong=	4.133	prob=	0.000	ZIP	PRM	p=0.000
vs ZINB	BIC=	3172.257	dif=	170.769	ZINB	PRM	Very strong
	AIC=	3124.068	dif=	190.045	ZINB	PRM	
NBRM	BIC=	3169.649	AIC=	3135.917	Prefer	Over	Evidence
vs ZIP	BIC=	3272.659	dif=	-103.010	NBRM	ZIP	Very strong
	AIC=	3229.288	dif=	-93.372	NBRM	ZIP	
vs ZINB	BIC=	3172.257	dif=	-2.608	NBRM	ZINB	Positive
	AIC=	3124.068	dif=	11.849	ZINB	NBRM	
	Vuong=	2.069	prob=	0.019	ZINB	NBRM	p=0.019
ZIP	BIC=	3272.659	AIC=	3229.288	Prefer	Over	Evidence
vs ZINB	BIC=	3172.257	dif=	100.402	ZINB	ZIP	Very strong
	AIC=	3124.068	dif=	105.221	ZINB	ZIP	
	LRX2=	107.221	prob=	0.000	ZINB	ZIP	p=0.000

Both the NBRM & ZINB consistently fit better than either the PRM or ZIP. BIC favors NBRM; AIC likes ZINB.



## Additional Reading

Sometimes you may want to do a regression where the dependent variable is the log of  $y$  rather than  $y$ . For example, you might want to use the log of income. In such instances, William Gould (President of Stata Corp) suggests that you may want to use `poisson` rather than `regress`. He explains why and how at

<http://blog.stata.com/2011/08/22/use-poisson-rather-than-regress-tell-a-friend/>

Paul Allison questions the need for zero-inflated models. He says that Negative Binomial Regression models often might be good enough. See

<http://statisticalhorizons.com/zero-inflated-models>

We haven't talked about fixed effects models yet. But Allison expresses concern about some of the approaches commonly used for Fixed Effects Negative Binomial Regression. See

<http://statisticalhorizons.com/fe-nbreg>