Modelling Count Variables with R

Poisson Regression

- Poisson regression is used to model count variables.
- Poisson regression has a number of extensions useful for count models.

Examples of Poisson regression

- The number of persons killed by mule or horse kicks in the Prussian army per year.
- Ladislaus Bortkiewicz collected data from 20 volumes of Preussischen Statistik.
- These data were collected on 10 corps of the Prussian army in the late 1800s over the course of 20 years.

Examples of Poisson regression

- The number of people in line in front of you at the grocery store.
- Predictors may include the number of items currently offered at a special discounted price and whether a special event (e.g., a holiday, a big sporting event) is three or fewer days away.

Poisson Regression with R

Examples of Poisson regression

- The number of awards earned by students at one high school.
- Predictors of the number of awards earned include the type of program in which the student was enrolled (e.g., vocational, general or academic) and the score on their final exam in math.

Conventional OLS regression

- Count outcome variables are sometimes log-transformed and analyzed using OLS regression.
- Many issues arise with this approach, including loss of data due to undefined values generated by taking the log of zero (which is undefined) and biased estimates.

Description of the data

- For the purpose of illustration, we have simulated a data set for the last example.
- The data set is called *poissonreg.csv*
- In this example, **num_awards** is the outcome variable and indicates the number of awards earned by students at a high school in a year

Predictor Variables

- **math** is a continuous predictor variable and represents students' scores on their math final exam,
- **prog** is a categorical predictor variable with three levels indicating the type of program in which the students were enrolled.
- **prog** is coded as 1 = "*General*", 2 = "*Academic*" and 3 = "*Vocational*".

Poisson Regression with R

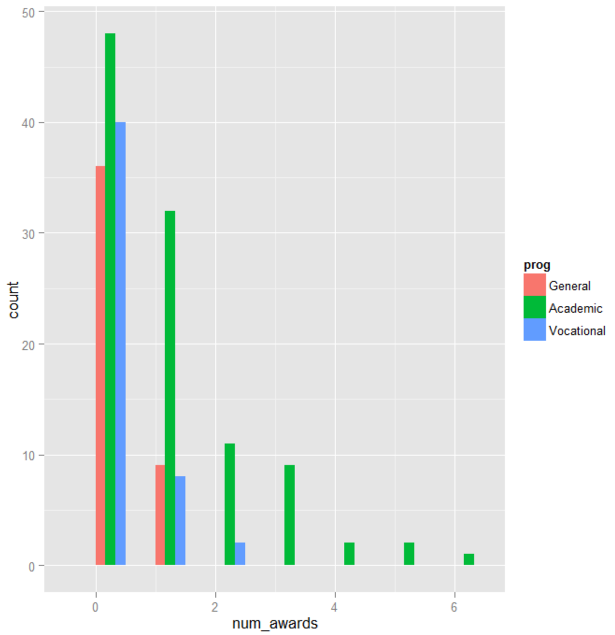
```
       id        num_awards             prog          math      
 1      :  1   Min.   :0.00   General   : 45   Min.   :33.0  
 2      :  1   1st Qu.:0.00   Academic  :105   1st Qu.:45.0  
 3      :  1   Median :0.00   Vocational: 50   Median :52.0  
 4      :  1   Mean   :0.63                    Mean   :52.6  
 5      :  1   3rd Qu.:1.00                    3rd Qu.:59.0  
 6      :  1   Max.   :6.00                    Max.   :75.0  
 (Other):194                                                 
```

Poisson Regression with R

- Each variable has 200 valid observations and their distributions seem quite reasonable.
- The unconditional mean and variance of our outcome variable are not extremely different.
- Our model assumes that these values, conditioned on the predictor variables, will be equal (or at least roughly so).

Poisson Regression with R

- Additionally, the means and variances within each level of prog–*the conditional means and variances*–are similar.
- A conditional histogram separated out by program type is plotted to show the distribution.



50
40
30
20
10
0
count
0
2
4
6
num_awards
prog
General
Academic
Vocational

Figure:

Poisson regression

- At this point, we are ready to perform our Poisson model analysis using the `glm` function.
- We fit the model and store it in the object `m1` and get a summary of the model.

Poisson Regression with R

```
m1 <- glm(num_awards ~ prog + math,
  family="poisson", data=p)

summary(m1)
```

Poisson Regression with R

```
Call:
glm(formula = num_awards ~ prog + math, family = "po
 Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.204   -0.844   -0.511    0.256    2.680
```

Poisson Regression with R

```
Coefficients:
               Estimate Std. Error z value Pr(>|z|)
(Intercept)     -5.2471     0.6585   -7.97  1.6e-15
progAcademic     1.0839     0.3583    3.03   0.0025
progVocational   0.3698     0.4411    0.84   0.4018
math             0.0702     0.0106    6.62  3.6e-11
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.'
```

Poisson Regression with R

```
(Dispersion parameter for poisson family taken to be

    Null deviance: 287.67  on 199  degrees of freedo
Residual deviance: 189.45  on 196  degrees of freedo
AIC: 373.5

Number of Fisher Scoring iterations: 6
```

Poisson Regression with R

- It is recommended using robust standard errors for the parameter estimates to control for mild violation of the distribution assumption that the variance equals the mean.
- The R package **sandwich** can be used to obtain the robust standard errors and calculated the p-values accordingly.
- Together with the p-values, we have also calculated the 95% confidence interval using the parameter estimates and their robust standard errors.

Poisson Regression with R

sandwich R Package

- Robust Covariance Matrix Estimators
- Model-robust standard error estimators for cross-sectional, time series, and longitudinal data.

Poisson Regression with R

Robust Standard Errors

```
cov.m1 <- vcovHC(m1, type="HC0")
std.err <- sqrt(diag(cov.m1))

r.est <- cbind(Estimate= coef(m1),
  "Robust SE" = std.err,
  "Pr(>|z|)" = 2 * pnorm(abs(coef(m1)/std.err),
  lower.tail=FALSE),
LL = coef(m1) - 1.96 * std.err,
UL = coef(m1) + 1.96 * std.err)
```

Poisson Regression with R

```
r.est

                 Estimate Robust SE  Pr(>|z|)      LL
 (Intercept)     -5.24712   0.64600 4.567e-16 -6.5133
 progAcademic     1.08386   0.32105 7.355e-04  0.4546
 progVocational   0.36981   0.40042 3.557e-01 -0.4150
 math             0.07015   0.01044 1.784e-11  0.0497
```

Poisson Regression with R

glm function output

- The output begins with echoing the function call. The information on deviance residuals is displayed next.
- Deviance residuals are approximately normally distributed if the model is specified correctly.
- Here it shows a little bit of skeweness since median is not quite zero.

glm function output

- The Poisson regression coefficients for each of the variables along with the standard errors, z-scores, p-values and 95% confidence intervals for the coefficients.
- The coefficient for math is 0.07.
- This means that the expected log count for a one-unit increase in math is 0.07.

Poisson Regression with R

glm function output

- The indicator variable **progAcademic** compares between **prog = Academic** and **prog = "General"** , the expected log count for **prog = Academic** increases by about 1.1.
- The indicator variable **prog.Vocational** is the expected difference in log count (≈ 0.37) between **prog = "Vocational"** and the reference group (**prog = "General"**).

Deviance

- In statistics, deviance is a quality of fit statistic for a model that is often used for statistical hypothesis testing.
- It is a generalization of the idea of using the sum of squares of residuals in ordinary least squares to cases where model-fitting is achieved by maximum likelihood.

Poisson Regression with R

glm function output

- The information on deviance is also provided.
- We can use the residual deviance to perform a goodness of fit test for the overall model.
- The residual deviance is the difference between the deviance of the current model and the maximum deviance of the ideal model where the predicted values are identical to the observed.

Poisson Regression with R

glm function output

- Therefore, if the residual difference is small enough, the goodness of fit test will not be significant, indicating that the model fits the data. We conclude that the model fits reasonably well because the goodness-of-fit chi-squared test is not statistically significant.
- If the test had been statistically significant, it would indicate that the data do not fit the model well.
- We could try to determine if there are omitted predictor variables, if our linearity assumption holds and/or if there is an issue of over-dispersion

Poisson Regression with R

```
with(m1, cbind(res.deviance = deviance,
df = df.residual,
        p = pchisq(deviance, df.residual,
            lower.tail=FALSE)))

      res.deviance  df      p
 [1,]        189.4 196 0.6182
```

Poisson Regression with R

- We can also test the overall effect of prog by comparing the deviance of the full model with the deviance of the model excluding prog.
- The two degree-of-freedom chi-square test indicates that prog, taken together, is a statistically significant predictor of num_awards.

Poisson Regression with R

```
# update m1 model dropping prog
m2 <- update(m1, . ~ . - prog)
# test model differences with chi square test
anova(m2, m1, test="Chisq")
```

Poisson Regression with R

```
Analysis of Deviance Table

Model 1: num_awards ~ math
Model 2: num_awards ~ prog + math
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1       198        204
2       196        189  2     14.6  0.00069 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.'
```

Incident Rate Ratios

- Sometimes, we might want to present the regression results as **incident rate ratios** (IRRs) and their standard errors, together with the confidence interval.
- To compute the standard error for the incident rate ratios, we will use the **Delta method** (Numerical Computation Method).
- To this end, we make use the function `deltamethod` implemented in R package **msm**.

Incident Rate Ratios

A rate ratio (sometimes called an incidence density ratio) in epidemiology, is a relative difference measure used to compare the incidence rates of events occurring at any given point in time. A common application for this measure in analytic epidemiologic studies is in the search for a causal association between a certain risk factor and an outcome.[1]

$$\text{Incidence Rate Ratio} = \frac{\text{Incidence Rate 1}}{\text{Incidence Rate 2}}$$

Incident Rate Ratios

Incidence rate is the occurrence of an event over person-time, for example person-years.

$$\text{Incidence Rate} = \frac{\text{events}}{\text{Person Time}}$$

Note: the same time intervals must be used for both incidence rates.

Poisson Regression with R

```
s <- deltamethod(list(~ exp(x1), ~ exp(x2),
#exponentiate old estimates dropping the p v
rexp.est <- exp(r.est[, -3])

# replace SEs with estimates for exponentiat
rexp.est[, "Robust SE"] <- s
```

Poisson Regression with R

```
rexp.est

                Estimate Robust SE       LL      UL
 (Intercept)    0.005263   0.00340 0.001484 0.01867
 progAcademic   2.956065   0.94904 1.575551 5.54620
 progVocational 1.447458   0.57959 0.660335 3.17284
 math           1.072672   0.01119 1.050955 1.09484
```

Poisson Regression with R

- The output above indicates that the incident rate for prog = "Academic" is 2.96 times the incident rate for the reference group (prog = "General").
- Likewise, the incident rate for prog = "Vocational" is 1.45 times the incident rate for the reference group holding the other variables at constant.

Poisson Regression with R

- The percent change in the incident rate of num_awards is by 7% for every unit increase in math.

Poisson Regression with R

- Sometimes, we might want to look at the expected marginal means.
- For example, what are the expected counts for each program type holding math score at its overall mean?
- To answer this question, we can make use of the predict function.
- First off, we will make a small data set to apply the predict function to it.

Poisson Regression with R

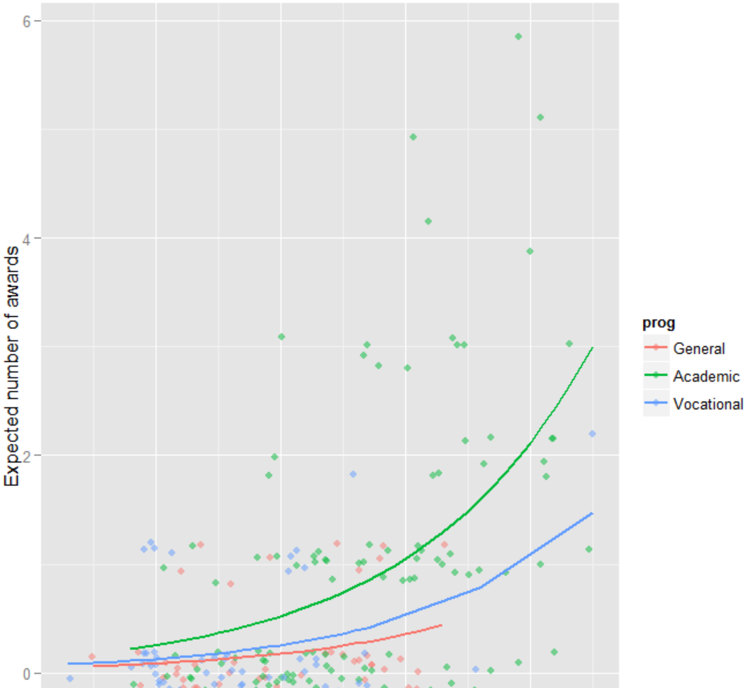
```
(s1 <- data.frame(math = mean(p$math),
  prog = factor(1:3, levels = 1:3,
  labels = levels(p$prog))))

   math       prog
1 52.65    General
2 52.65   Academic
3 52.65 Vocational
```

Poisson Regression with R

```
predict(m1, s1, type="response", se.fit=TRUE)

 $fit
      1      2      3
 0.2114 0.6249 0.3060

 $se.fit
       1       2       3
 0.07050 0.08628 0.08834

 $residual.scale
 [1] 1
```

Poisson Regression with R

- In the output above, we see that the predicted number of events for level 1 of prog is about 0.21, holding math at its mean.
- The predicted number of events for level 2 of prog is higher at 0.62, and the predicted number of events for level 3 of prog is about .31.
- The ratios of these predicted counts ($\frac{0.625}{0.211} = 2.96$, $\frac{0.306}{0.211} = 1.45$) match what we saw looking at the IRR.


Expected number of awards
0
2
4
6
prog
General
Academic
Vocational

Poisson Regression with R

- We can also graph the predicted number of events with the commands below.
- The graph indicates that the most awards are predicted for those in the academic program (prog = 2), especially if the student has a high math score.
- The lowest number of predicted awards is for those students in the general program (prog = 1).
- The graph overlays the lines of expected values onto the actual points, although a small amount of random noise was added vertically to lessen overplotting.

Poisson Regression with R

```
# Calculate and store predicted values
p$phat <- predict(m1, type="response")

# order by program and then by math
p <- p[with(p, order(prog, math)), ]
```

Poisson Regression with R

```
ggplot(p, aes(x = math, y = phat, colour = prog)) +
  geom_point(aes(y = num_awards), alpha=.5, position=
  geom_line(size = 1) +
  labs(x = "Math Score", y = "Expected number of awar
```

Poisson Regression with R

Over-Dispersion Overdispersion is the presence of greater variability (statistical dispersion) in a data set than would be expected based on a given simple statistical model.

Over-Dispersion

- When there seems to be an issue of dispersion, we should first check if our model is appropriately specified, such as omitted variables and functional forms.
- For example, if we omitted the predictor variable prog in the example above, our model would seem to have a problem with over-dispersion.
- In other words, a misspecified model could present a symptom like an over-dispersion problem.

Poisson Regression with R

- Assuming that the model is correctly specified, the assumption that the conditional variance is equal to the conditional mean should be checked.
- There are several tests including the likelihood ratio test of over-dispersion parameter alpha by running the same model using negative binomial distribution.
- The R package pscl (Political Science Computational Laboratory, Stanford University) provides many functions for binomial and count data including odTest for testing over-dispersion.

Poisson Regression with R

- One common cause of over-dispersion is excess zeros, which in turn are generated by an additional data generating process.
- In this situation, zero-inflated model should be considered.
- If the data generating process does not allow for any 0s (such as the number of days spent in the hospital), then a zero-truncated model may be more appropriate.

Poisson Regression with R

- Count data often have an exposure variable, which indicates the number of times the event could have happened.
- This variable should be incorporated into a Poisson model with the use of the offset option.
- The outcome variable in a Poisson regression cannot have negative numbers, and the exposure cannot have 0s.

Poisson Regression with R

- Many different measures of pseudo-R-squared exist. They all attempt to provide information similar to that provided by R-squared in OLS regression, even though none of them can be interpreted exactly as R-squared in OLS regression is interpreted.
- Poisson regression is estimated via maximum likelihood estimation. It usually requires a large sample size.

Poisson Regression "Exposure" and offset

- Poisson regression may also be appropriate for rate data, where the rate is a count of events occurring to a particular unit of observation, divided by some measure of that unit's exposure.
- For example, biologists may count the number of tree species in a forest, and the rate would be the number of species per square kilometre.
- Demographers may model death rates in geographic areas as the count of deaths divided by personyears.
- More generally, event rates can be calculated as events per unit time, which allows the observation window to vary for each unit.

Poisson Regression : Exposure and Offset

In these examples, exposure is respectively unit area, personyears and unit time. In Poisson regression this is handled as an offset, where the exposure variable enters on the right-hand side of the equation, but with a parameter estimate (for log(exposure)) constrained to 1.

$$\log\left(\mathrm{E}(Y \mid x)\right) = \log\left(\text{exposure}\right) + \theta' x$$

which implies

$$\log\left(\mathrm{E}(Y \mid x)\right) - \log\left(\text{exposure}\right) = \log\left(\frac{\mathrm{E}(Y \mid x)}{\text{exposure}}\right) = \theta' x$$

Poisson Regression : Exposure and Offset

Offset in the case of a GLM in R can be achieved using the offset() function:

```
glm(y ~ offset(log(exposure)) + x, family=po
```