Notes on poisson regression models

1 Introduction

Lots of student project groups have count data, which can't be analysed in a normal linear regression framework because the response variable is not normally distributed and not continuous. For example, the small mammals group has numbers of ink tracks found in their plastic tubes as a measure of small mammal activity.

The poisson distribution is a discrete probability distribution. It is useful for modelling the occurrence of discrete events within a time period or over a unit area. The poisson probability distribution:

$$P(k) = \frac{\lambda^k e^{-\lambda}}{k!} \tag{1}$$

- Where k is the number of times an event occurs and is a positive integer greater than or equal to zero
- Where $P(k)$ is the probability of observing a phenomenon k times
- Where λ is the average number of occurrences (rate parameter); i.e. the expected number of occurrences
- Where $k!$ is the factorial of k
- Where the occurrence of an event doesn't affect the probability of future events (data are independent)

2 Poisson regression

The poisson regression generalised linear model assumes that the response variable has a poisson distribution and assumes that the log of the expected value can be modelled by a linear combination of explanatory variables. The model can be expressed as:

$$\log \lambda_i = \beta_0 + \beta_1 x_i \tag{2}$$

Note that compared to a linear model, there is no error term ϵ as λ determines both the mean and the variance.

The link function which relates the response variable to the linear model is:

$$\mathbf{X}\beta = \ln \lambda \tag{3}$$

In R, a poisson distributed GLM can be specified as:

```
mod <- glm(y ~ x, family = poisson(link = log))
```

3 Offset

Data collection for poisson models can be expressed in terms of "events", "exposure" and "rate". For example, when counting plant abundance, the event would be observations of the plant of interest, the exposure would be the area of ground searched, and the rate would be the number of plants per unit area. Typically, poisson regression is used to model counts, but sometimes it is more relevant to model rates if exposure varied between samples. Rates can be used to account for variation in the exposure. It is possible to build in variation in exposure to the poisson regression by treating it as an "offset". In R, this can be written as:

```
mod <- glm(y ~ offset(log(exposure)) + x, family = poisson(link = log))
```

4 Over-dispersion and zero inflation

When the observed variance is greater than the mean, this is known as overdispersion. When this occurs, a negative binomial distribution is an appropriate alternative to a poisson distribution in a generalised linear model.

The negative binomial regression is a generalisation of the poisson regression that loosens the assumption that the variance equals the mean by adding the parameter θ in addition to λ, which gives more flexibility to the model. Mathematically, the negative binomial model is like a poisson model where λ is random, following a gamma distribution.

In R, a negative binomial model can be specified with:

```
library(MASS)

mod_nb <- glm.nb(y ~ x)
```

Having excess zeroes will also invalidate the use of a poisson distributed GLM and a zero-inflated model (ZIP) or a hurdle model should be used instead.

A ZIP model has two parts, a poisson count model and a logistic model for predicting excess zeroes. Zero-inflated models hypothesise that there are two processes acting on the response variable one which determines the presence or absence of an observation (0 vs >0) and one which determines the number of observations. A zero-inflated model predicts the response variable as a mixture of a bernoulli distribution (logistic part) and a poisson distribution.

Alternatively a hurdle model can be used. While hurdle models are similar to zero-inflated models, they differ in the modelling process. Hurdle models have two distinct model parts estimating the response variable, one being a bernoulli distribution for all the data and the next being the poisson distribition for only the non-zero truncated data.

A hurdle model might be more appropriate where zeroes can only come from one source, while a zero-inflated model is more appropriate if zeroes can come from two processes. Hurdle models assume that observations are one of two types, observations of zero, and observations always greater than zero. Conversely, zero-inflated models conceptualize observations as those always with a value of zero, and observations which can be non-zero, but aren't always.

In R, a zero-inflated model can be fitted like so:

```
library(pscl)

mod_zeroinfl <- zeroinfl(y ~ x)
```

Similarly, a hurdle model:

```
library(pscl)

mod_hurdle <- hurdle(y ~ x)
```

The hurdle model function can additionally specify the distributions of the zero-count part of the model and the positive-count part of the model using the arguments `zero.dist =` and `dist =`, which are by default set to `poisson` and `binomial`, respectively.

5 Large datasets

For very large values of λ (e.g. $\lambda > 1000$), the normal distribution with mean = λ and standard deviation = $\sqrt{\lambda}$ is a good approximation of the poisson distribution. This means that theoretically, if students had a mean number of occurrences >1000, they might be able to use a normal linear regression.

6 Analysing a poisson regression

To test whether a poisson GLM is appropriate, first see whether the variance equals the mean of the response variable:

```
mean(df$y)

var(df$y)
```

If the variance is much greater than the mean, there is overdispersion and a negative binomial model might be more appropriate.

The `summary()` of a poisson GLM looks like this, in this case for a model of hospital visits (`ofp`) vs. the number of years of schooling (`school`):

```
Call:
glm(formula = ofp ~ school, family = poisson, data = df)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.6727  -2.2847  -0.8555   0.8144  17.7501

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) 1.539399   0.019091   80.63   <2e-16 ***
school      0.020517   0.001706   12.03   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 26943  on 4405  degrees of freedom
Residual deviance: 26797  on 4404  degrees of freedom
AIC: 39576

Number of Fisher Scoring iterations: 5
```

It looks similar to the output for a normal linear regression, with slope values for each coefficient (e.g. `0.020517`) and standard errors for those slopes (e.g. `0.001706`) a z-value which shows the ratio of the coefficient slope to its standard error and tests that the coefficient slope is not equal to zero (e.g. `12.03`) and the P-value showing the probability that the observed z-value is as extreme as, or more so, than that which would be observed if the null hypothesis were true.

It is important to remember in any GLM, that because of the link function which relates values of y to variation in x, the slope coefficient given is not actually showing directly how how the value of y (`ofp`) changes per unit increase in x (`school`). Instead it is showing that for an increase in x of 1, the value of $\ln(y)$ increases by $y(e^{0.020517} - 1)$. To convert so that it shows how much y increases with a unit increase in x, you must multiply y by $e^{0.020517}$. So, according to the model, if the average person spending 4 years at school visits the hospital 10 times, the average person spending 5 years at school will visit the hospital $10 \times e^{0.020517} = 10.207$ times.

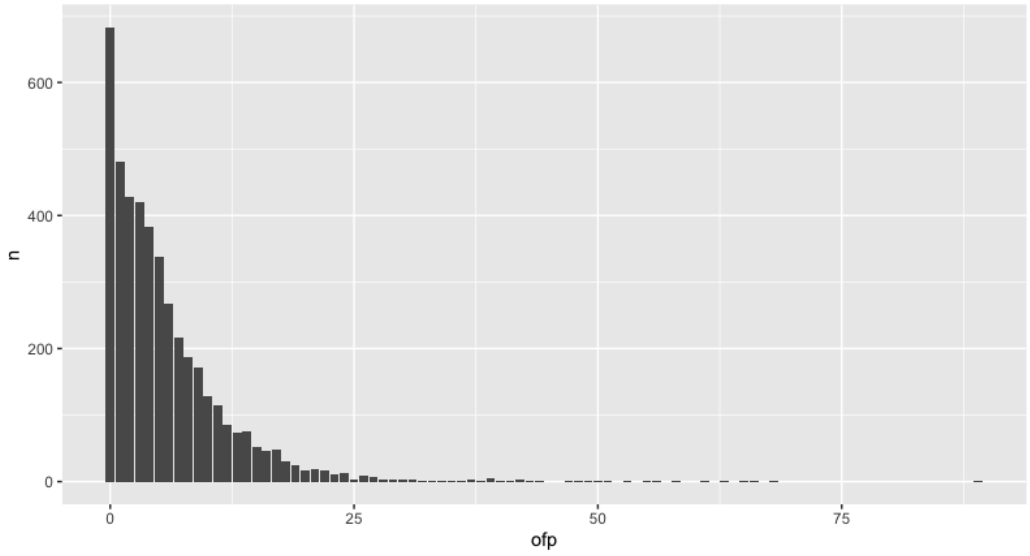
R-squared values cannot be reliably reported in poisson regressions, however a number of pseudo-r-squared methods exist. See Heinzl and Mittlböck (2003) for an indepth comparison of methods.

To visualise data used in a poisson regression, it is often useful to plot the distribution of the count data response variable y as a bar graph, with the frequency of each integer count occurrence on the y axis, with the frequency of each integer count occurrence on the y axis. See the example below:

```
library(ggplot2)
library(dplyr)

ofp_summ <- df %>%
  group_by(ofp) %>%
  tally()

ggplot(ofp_summ, aes(x = ofp, y = n)) +
  geom_bar(stat = "identity")
```



7 Non-parametric alternatives

Especially if sample size is small, a non-parametric alternative to a GLM may be suitable for count data, specifically a Spearman's Rank Correlation Coefficient if the explanatory variable is continuous, or a Kruskal-Wallis Test if the explanatory variable is categorical. Be aware though that these tests will significantly limit the power of the statistical inference you can make on the data. Neither of these tests assume a direction of influence between the independent and dependent variables, they are essentially correlations. Both tests use ranks of data rather than the raw data to account for non-normality of variance:

As a reminder, the Spearman's Rank correlation coefficient (r_s) is calculated as:

$$r_s = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)} \tag{4}$$

Where d_i^2 is the squared difference between the two ranks of each observation, and n is the number of observations.

The standard error of the coefficient σ_{r_s} is:

$$\sigma_{r_s} = \frac{0.6325}{\sqrt{n - 1}} \tag{5}$$

To run a Spearman's Rank in R:

```
cor.test(x, y, method = "spearman")
```

This will return a t value, degrees of freedom and a P-value to assess the fit of the rank correlation.

The Kruskal-Wallis Test, also known as the Kruskal-Wallis Test Statistic (H) is calculated as:

$$H = \left[\frac{12}{n(n+1)} \sum_{j=1}^{c} \frac{T_j^2}{n_j} \right] - 3(n+1) \tag{6}$$

Where n is the sum of the sample sizes across replicates, c is the number of replicates, T_j^2 is the squared sum of ranks in sample j, n_j is the size of sample j.

To run a Kruskal-Wallis Test in R:

```
kruskal.test(y ~ x)
```

8 Reading list

Heinzl, H. and Mittlböck (2003), 'Pseudo r-squared measures for poisson regression models with over- or under-dispersion', *Computational Statistics & Data Analysis* **44**, 253–271.

O'Hara, R. B. and Kotze, D. J. (2010), 'Do not log-transform count data', *Methods in Ecology and Evolution* **1**, 118–122.

Zeileis, A., Kleiber, C. and Jackman, S. (2008), 'Regression models for count data in r', *Journal of Statistical Software* **27**(8), 1–25.

Zuur, A. F., Ieno, E. N. and Elphick, C. S. (2010), 'A protocol for data exploration to avoid common statistical problems', *Methods in Ecology and Evolution* **1**, 3–14.