GLMs with a Poisson

Bob O'Hara

March 8, 2019

Last Week: The binomial distribution

What is the binomial GLM (logistic regression)?

When and why to use a binomial GLM?

Link Functions

Categorical and continuous variables

Overdispersion

This Week: log-linear models

This week you will:

- learn about log-linear models
- learn about over-dispersion
  - when there is more error than you expect

A Typical Problem: Count data

Numbers of

- fish caught
- murders
- offspring
- bacterial/fungal colonies
- deaths due to lip cancer

A Model for Counts: Fishing

Figure 1: Anglers by Raoul Dufy

A Model for Counts: Fishing

We sit by the Seine, fishing. We catch fish at a constant rate

If we catch fish for an hour, how many fish do we catch?

A Model for Counts: Fishing

If we catch fish at rate μ, the mean number we catch in time t will be $\lambda = \mu t$

The actual number will vary, and will follow a Poisson distribution:

$$Pr(N = r|\lambda) = \frac{\lambda^r e^{-\lambda}}{r!}$$

A Model for Counts

Figure 2: Siméon Denis Poisson

The Poisson distribution

Look at simulations of the Poisson distribution for different means (`plot(table(...))` is nicer than `hist()`)

What happens to the shape of the distribution when

- the mean is less than 1?
- the mean equals 1
- the mean is above 1
- the mean gets large?

```r
Pois <- rpois(1e3, 1)
plot(table(Pois), lwd=8, lend=3)
```

Is the Poisson a GLM?

The log-likelihood:

$$l(N = r|\lambda) = r \log \lambda - \lambda - log(t!)$$

GLM likelihood:

$$l(\theta|y) = \frac{y\theta - b(\theta)}{a(\phi)} - c(\phi, y)$$

Is the Poisson a GLM?

The log-likelihood:

$$l(N = r|\lambda) = r \log \lambda - \lambda - \log(t!)$$

GLM likelihood:

$$l(\theta|y) = \frac{y\theta - b(\theta)}{a(\phi)} - c(\phi, y)$$

So $a(\phi) = 1$

and $\theta = \log \lambda$

Interpretation

This is a GLM

The natural link function is a log link

- very rare something else is used

If we are counting, the process is multiplicative (double the effort, double the counts)

This is additive on the log scale.

Interpretation

The log link means that the model is multiplicative

$$\log(\lambda) = \alpha + \beta x$$
$$\lambda = e^{\alpha+\beta x} = e^{\alpha} e^{\beta x}$$

(we'll assume x is a dummy variable, i.e. 0 or 1)

e.g. if $\alpha = 0$, $e^{\alpha} = 1$. Then if β doubles the mean (i.e. $\lambda = 2e^{\alpha}$), $\beta = \log(2) = 0.69$

Some claims

If a coefficient is small, it is (approximately) the percent increase

- $e^{\alpha+\beta}$ means an increase by $e^{\beta} \approx 1 + \beta$ times (if β is small)

The coefficients are symmetrical

- a value of $+0.01$ increases the mean by $e^{0.01}$ times
- a value of -0.01 *decreases* the mean by $e^{0.01}$ times

Some claims: An Exercise

Write some questions to illustrate these claims:

- If a coefficient is small, it is (approximately) the percent increase
- The coefficients are symmetrical

e.g. "If we have a Poisson process with a mean of 1, what value of β would we need to double the mean?"

Model Fitting

Model fitting is easy:

```r
mu <- seq(1,2, length=10)
Count <- rpois(length(mu), mu)
m1 <- glm(Count ~1, family=poisson("log"))
m1a <- glm(Count ~1, family="poisson")
```

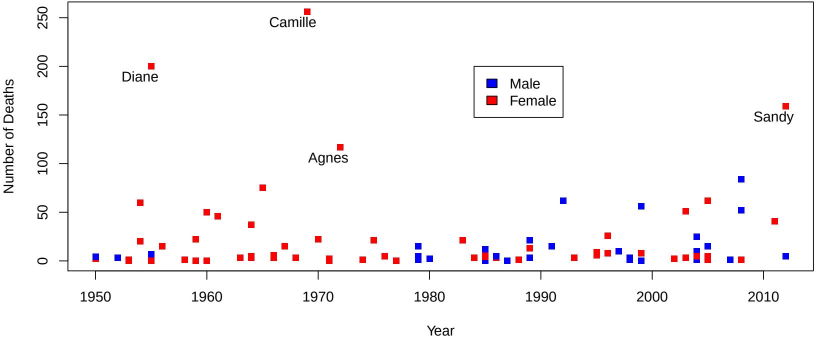
An Example: Himmicanes

A few years a go a strange paper appeared in PNAS that suggested that hurricanes in the USA with female names caused more deaths than those with male names.

```r
Stem <- "https://www.math.ntnu.no/emner/"
Fl <- "ST2304/2019v/Week13/Himmicanes.csv"
Data <- read.csv(paste0(Stem,Fl), stringsAsFactors=FALSE)
# Select hurricanes with > 100 deaths
BigH <- which(Data$alldeaths>100)
```

Himmicane Data

```
plot(Data$Year, Data$alldeaths, col=Data$ColourMF,
     type="p", pch=15, xlab="Year",
     ylab="Number of Deaths")
text(Data$Year[BigH], Data$alldeaths[BigH],
     Data$Name[BigH], adj=c(0.8,1.5))
legend(1984,200,c("Male","Female"),fill=c("blue","red"))
```



The Data

The variables in the data are:

- Year: Year
- Name: Hurricane's name
- Gender: Gender (0: Male, 1: Female)
- MasFem: A scoring of how feminine the name sounds (we won't use this here)
- Minpressure: minimum air pressure in the hurricane (a measure of stength)
- Category: Category of hurricane (larger is more severe)
- NDAM: Normalised damage (i.e. how much the hurricane cost, corrected for inflation etc.)
- alldeaths: Number of deaths

The aim is to predict the number of deaths.

The Modelling Step 1: Chose a model

Your task (to discuss in your group):

- what distribution should we use?
- what link function?
- which variables do you want to consider to explain the number of deaths?
  - should we only use Gender, or do we need other variables?

Step II: Get Estimates of the Parameters

Fit the model!

Does there seem to be an effect of gender?

Today

More with the Poisson

- overdispersion and residuals

Before we Start

We will have revision sessions on these dates:

14th, 16th and 21st May

Before then we will put up a practice exam (with solutions) as well as solutions to all of the exercises

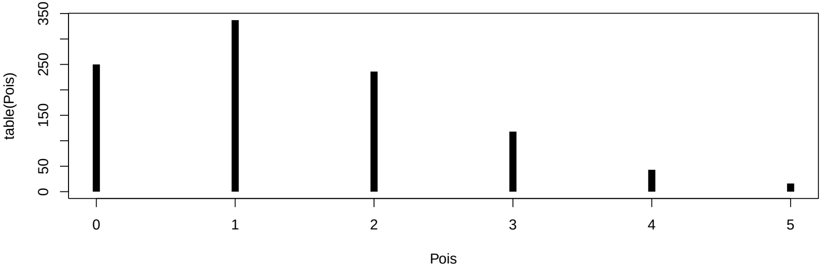
End of Course Survey: see Blackboard (under "Surveys"). Your chance to tell us what we can do better

Yesterday

We were introduced to the Poisson distribution

$$l(N = r|\lambda) = r \log \lambda - \lambda - \log(t!)$$

```r
Pois <- rpois(1e3, 1.4)
plot(table(Pois), lwd=8, lend=3)
```



Yesterday

It is natural to model counts my multiplying

- additive on the log scale

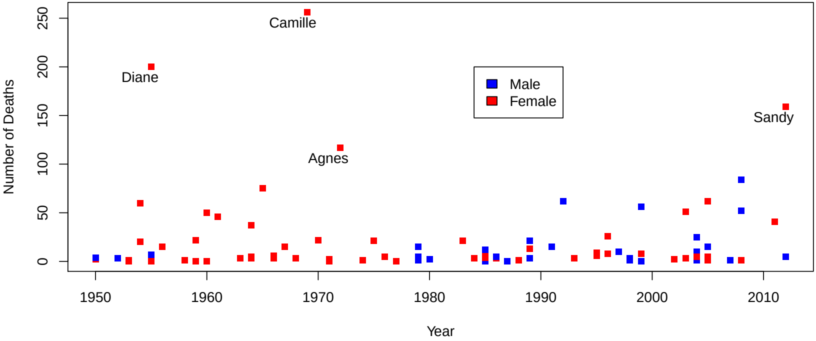
The log link is thus a natural link function

Model fitting is easy:

```r
m1 <- glm(Count ~ 1, family=poisson("log"))
```

Yesterday

We looked at the Himmicanes data



The Modelling

Step 1: Chose a model

Step II: Get Estimates of the Parameters

```
Data$Category <- factor(Data$Category)
# these all use family="poisson", data=Data
mod.poisMinP <- glm(alldeaths ~  Gender+Minpressure, famil
mod.poisCat <- glm(alldeaths ~  Gender+Category, family="po
mod.poisNDAM <- glm(alldeaths ~  Gender+NDAM, family="poiss
```

Today

Step III: Is the model any good, and can we improve it?

Step IV: Comparing Models

Step III: Model Checking

(we could also do model selection now, but as we will find a big problem, it's quicker to look at this first)

A big problem: over-dispersion

Overdispersion

Too much variation!

Mean = Variance for a Poisson

But the data may have more variation (“extra-Poisson variation”)

What happens when we have overdispersion?

Let's simulate some data without over-dispersion

```r
alpha <- 1.5;   beta <- 0.1
X <- rnorm(1e3)
lambda <- exp(alpha + beta*X)
Y <- rpois(length(lambda), lambda)
var(Y)
```

```
## [1] 4.787872
```

. . . and simulate some data with over-dispersion

```r
eps <- rnorm(length(X), 0, 0.5)
lambda2 <- exp(alpha + beta*X + eps)
Y2 <- rpois(length(lambda2), lambda2)
var(Y2)
```

```
## [1] 14.21932
```

What happens when we have overdispersion III?

Fit the model without over-dispersion

Look at the standard errors (= standard deviation of the estimators)

```r
mod.noOD <- glm(Y ~ X, family = poisson())
summary(mod.noOD)
```

. . . and with overdispersion

```r
mod.OD <- glm(Y2 ~ X, family = poisson())
summary(mod.OD)
```

What happens when we have overdispersion IV?

So, overdispersion increases the variation in the data

This *should* increase the standard errors (as more variation means we are less certain), but is doesn't.

Does this matter? We can look at what the distribution of parameters should be

What happens when we have overdispersion IV?

We can do these simulations lots of times, and look at the standard deviation

```r
SimWithOD <- function(al=1.5, be=0.1, sigma=0, N=1e3) {
  X <- rnorm(N)
  eps <- rnorm(N, 0, sigma)
  lambda <- exp(al + be*X + eps)
  Y <- rpois(length(lambda), lambda)
  mod <- glm(Y ~ X, family = poisson())
  coef(mod)["X"]
}
Repbeta0 <- replicate(1e2, SimWithOD(sigma=0))
Repbeta1 <- replicate(1e2, SimWithOD(sigma=1))
c(NoOD=sd(Repbeta0), OD=sd(Repbeta1))
```

What happens when we have overdispersion V?

Pause Here

The effects of Over-dispersion

The uncertainty in the estimates increases **but** this isn't seen in the confidence intervals

However, we can see it in the residual deviance increase

- with no overdispersion, this should (roughly) equal the residual degrees of freedom

Dealing With Overdispersion

There are a few ways to deal with overdispersion

- Correct in the likelihood, using ϕ
- Use a mixed model (not in this course, sorry)
- Use a different distribution

Correct the likelihood I

The likelihood is

$$l(\theta|y) = \frac{y\theta - b(\theta)}{a(\phi)} + c(y, \phi)$$

So we can estimate ϕ, the dispersion. We can use the deviance ratio.

Deviance/Degrees of Freedom

```r
Dispersion <- deviance(mod.OD)/df.residual(mod.OD)
```

This should equal 1: values below about 1.2 are fine

(without overdispersion, this actually follow a χ^2 distribution)

Correct the likelihood II

We can plug the dispersion estimate into the summary:

```
summary(mod.OD, dispersion = Dispersion)
```

```
## 
## Call:
## glm(formula = Y2 ~ X, family = poisson())
## 
## Deviance Residuals: 
##     Min       1Q   Median       3Q      Max  
## -3.3983  -1.3281  -0.3925   0.7541   7.6240  
## 
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)    
## (Intercept)  1.63160    0.02245  72.681   <2e-16 ***
## X            0.05605    0.02284   2.454   0.0141 *  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1
##
```

Correct the likelihood III

The effect is to increase standard errors by sqrt(Dispersion):

```
summary(mod.OD)$coefficients[,"Std. Error"]
```

```
## (Intercept)           X
##  0.01401977  0.01426281
```

```
summary(mod.OD, dispersion =
                Dispersion)$coefficients[,"Std. Error"]
```

```
## (Intercept)           X
##  0.02244896  0.02283812
```

Use a different distribution I

The Negative Binomial distribution assumes that there is over-dispersion

We fit the model in almost the same way, but with the `glm.nb()` function in the `MASS`package

```
mod.NB <- MASS::glm.nb(Y2 ~ X)

round(summary(mod.NB)$coefficients, 2)
```

```
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)     1.63       0.02   73.06     0.00
## X               0.06       0.02    2.49     0.01
```

There is an extra parameters, θ, is an estimate of the amount of overdispersion (lower = more overdispersion)

Use a different distribution: long version

Our model is $\log(\mu_i) = \sum_j X_{ij}\beta_j$. But we could add a random term, so it becomes $\log(\mu_i) = \sum_j X_{ij}\beta_j + \varepsilon_i$

If we use $\varepsilon_i \sim N(0, \sigma^2)$ this is like a regression

- need a Generalised Linear Mixed Model to estimate it

We could also use $e^{\varepsilon_i} \sim \chi^2_\nu$. This is the same as assuming a negative binomial distribution.

Back to Himmicanes

Is there evidence of over-dispersion?

What happens if you correct it?

- either using a neative binomial distribution or correcting the likelihood

More model Checking: Residuals

We can also calculate residuals

```
mod.NBndam <- MASS::glm.nb(alldeaths ~  Gender+NDAM, data=
resid(mod.NBndam)[1:5]
```

```
##          1          2          3          4          5
## -0.8795870 -0.5884542 -0.4389292 -1.1047473 -1.8174212
```

```
# plot on log scale
plot(log(fitted(mod.NBndam)), resid(mod.NBndam))
```

Any evidence for curvature? Or other effects?

Suggestions for improving the model?

Model Comparison

We have a specific hypothesis: Gender affects number of deaths.

We can ask specifically about this question by hypothesis testing: does the model with Gender fit better than the model without Gender?

Model Comparison

Your turn: test this hypothesis (see Weeks 10 & 12 for how to do this)

Summary

We have seen a log-linear model, using a Poisson distribution

- use with counts

Interpret the parameters on the log scale

- multiplicative on the scale of the data

Easy to fit the model

Overdispersion is a common problem

- can now solve in 2 ways