R Code And Tasks Chapter 5 (MAS 6003)

Witold Wolski

December 27, 2016

Chapter 5 Poisson regression

5.1 Introduction

pdf of poisson

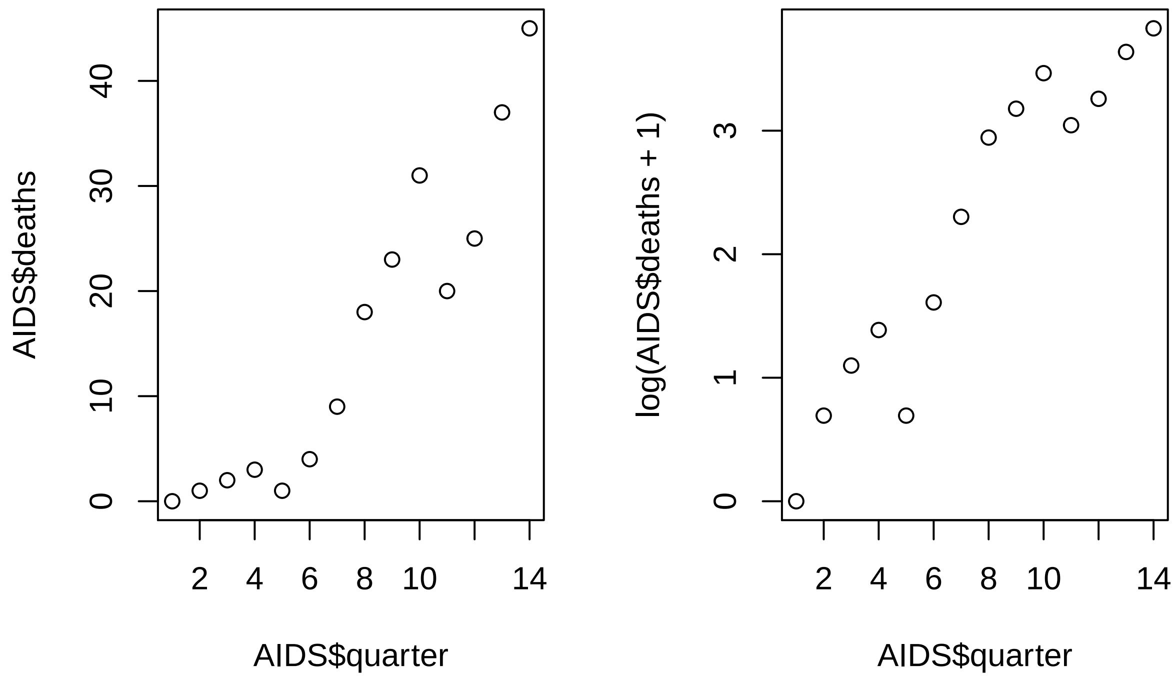
$$\frac{\lambda^k e^{-\lambda}}{k!}$$

5.2.1 Example : AIDS deaths over time (Task 15)

1 plot :

```
rm(list=ls())
load("data/MAS367-GLMs.RData", envir = e <- new.env())

AIDS <- e$AIDS
par(mfrow=c(1,2))
plot(AIDS$quarter, AIDS$deaths)
plot(AIDS$quarter, log(AIDS$deaths+1))
```



2 fit poisson with log link

```
glm.lin <- glm(deaths ~ quarter, data=AIDS, family=poisson(link='log'))
qchisq(0.95,glm.lin$df.residual)
```

```
## [1] 21.02607
```

3 adding a quadratic term

```
glm.quad <- glm(deaths ~ quarter + I(quarter^2), data=AIDS, family=poisson(link='log'))
summary(glm.quad)
```

```
##
## Call:
## glm(formula = deaths ~ quarter + I(quarter^2), family = poisson(link = "log"),
##     data = AIDS)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7708  -0.9385   0.1304   0.8190   1.4421
##
```

```
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -1.713375   0.733108  -2.337 0.019432 *
## quarter       0.746031   0.153391   4.864 1.15e-06 ***
## I(quarter^2) -0.025836   0.007751  -3.333 0.000859 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 207.272  on 13  degrees of freedom
## Residual deviance:  16.371  on 11  degrees of freedom
## AIC: 75.298
##
## Number of Fisher Scoring iterations: 4
```

```
qchisq(0.95,glm.lin$df.residual)
```

```
## [1] 21.02607
```

4 a line predictor on log(x)

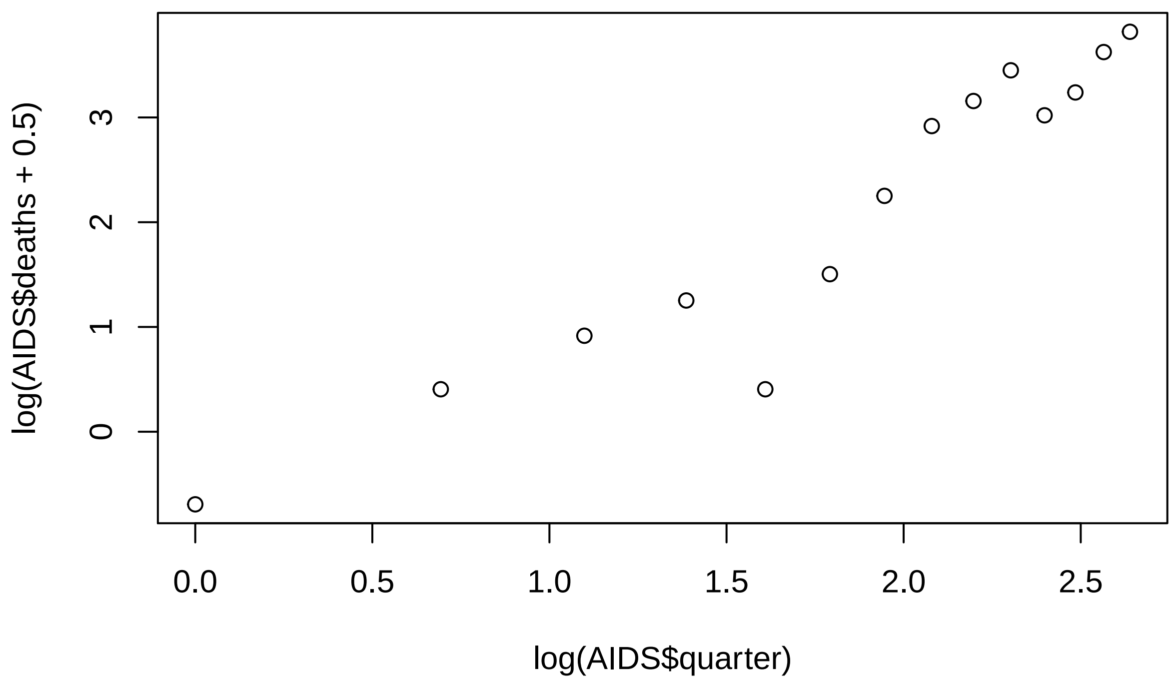
```
glm.logline <- glm(deaths ~ I(log(quarter)), data=AIDS, family=poisson(link='log'))
summary(glm.logline)
```

```
##
## Call:
## glm(formula = deaths ~ I(log(quarter)), family = poisson(link = "log"),
##     data = AIDS)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.08992  -1.07141  -0.04657   0.38956   1.94311
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -1.9442     0.5116   -3.80 0.000145 ***
## I(log(quarter))   2.1748     0.2150   10.11  < 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: 207.272  on 13  degrees of freedom
## Residual deviance:  17.092  on 12  degrees of freedom
## AIC: 74.019
##
## Number of Fisher Scoring iterations: 4
```

```r
qchisq(0.95,glm.logline$df.residual)
```

```
## [1] 21.02607
```

```r
plot(log(AIDS$quarter), log(AIDS$deaths+0.5))
```



5

Thus possible simple models are a line in logx or a quadratic in x, but there are reservations about both.

5.3 Adjusting for exposure : offset (Task 16)

An explanation of offset which is brief and clear can be found here (but not in the lecture notes of MAS 6003).

5.3.1 Example: Smoking and heart disease

1,2,3,4 death rates

```r
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.3.2
```

```r
library(gridExtra)
```

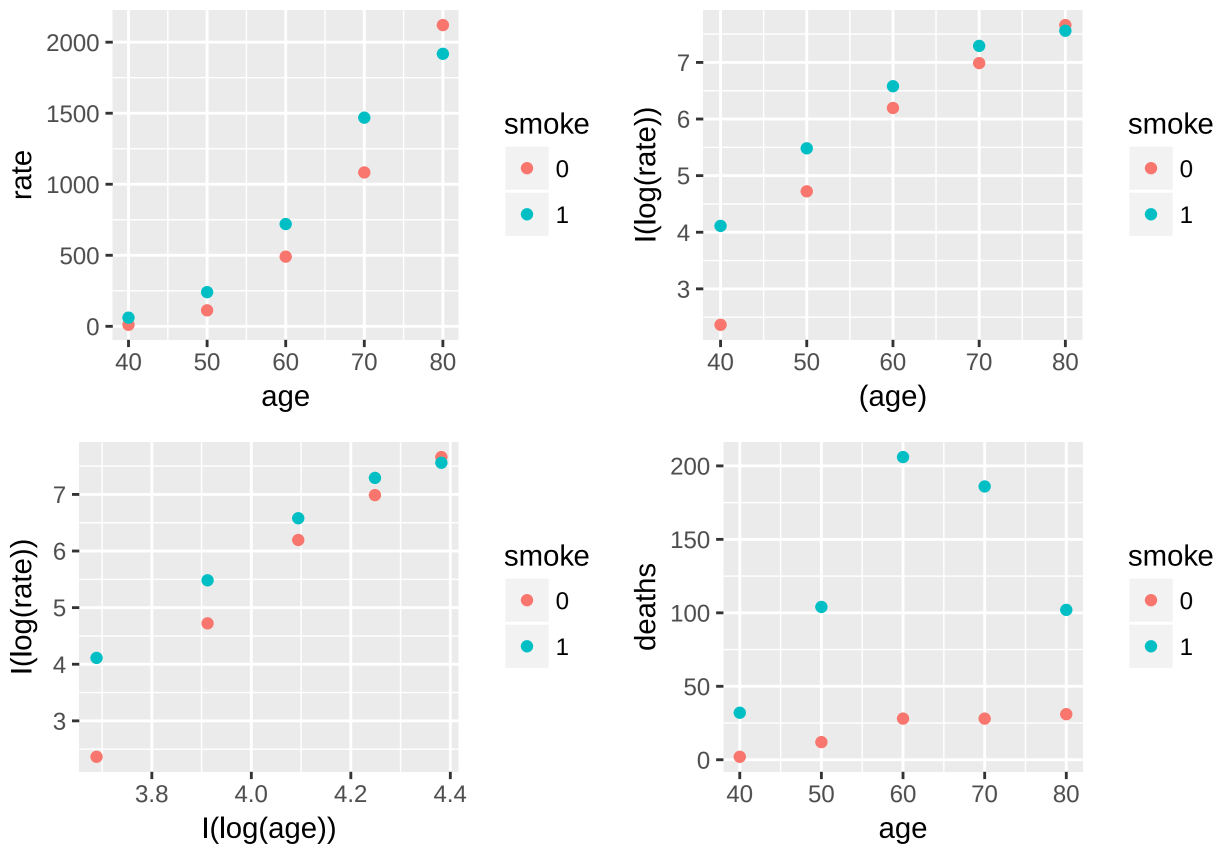
```
## Warning: package 'gridExtra' was built under R version 3.3.2
```

```r
smoking <- e$smoking
smoking$rate <- smoking$deaths/smoking$person.years * 1e5
lapply(smoking,class)
```

```
## $age
## [1] "integer"
##
## $smoke
## [1] "integer"
##
## $deaths
## [1] "integer"
##
## $person.years
## [1] "integer"
##
## $rate
## [1] "numeric"
```

```r
smoking$smoke <- as.factor(smoking$smoke)
p1 <- ggplot(smoking, aes(age, rate, colour=smoke)) + geom_point()
p2 <- ggplot(smoking, aes((age), I(log(rate)), colour=smoke)) + geom_point()
p3 <- ggplot(smoking, aes(I(log(age)), I(log(rate)), colour=smoke)) + geom_point()
p4 <- ggplot(smoking, aes(age, deaths, colour=smoke)) + geom_point()


grid.arrange(p1,p2,p3,p4, ncol = 2)
```



5 The model

```
mod.offset <- glm(deaths~ offset(log(person.years)) + smoke * age + I(age^2), family = poisson, data=sm
summary(mod.offset)
```

```
##
## Call:
## glm(formula = deaths ~ offset(log(person.years)) + smoke * age +
##     I(age^2), family = poisson, data = smoking)
##
## Deviance Residuals:
##        1         2         3         4         5         6         7
## -0.83049   0.43820   0.13404  -0.27329   0.64107  -0.15265  -0.41058
##        8         9        10
##  0.23393  -0.01275  -0.05700
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.970e+01  1.253e+00 -15.717  < 2e-16 ***
## smoke1       2.364e+00  6.562e-01   3.602 0.000316 ***
## age          3.563e-01  3.632e-02   9.810  < 2e-16 ***
## I(age^2)    -1.977e-03  2.737e-04  -7.223 5.08e-13 ***
## smoke1:age  -3.075e-02  9.704e-03  -3.169 0.001528 **
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 935.0673  on 9  degrees of freedom
## Residual deviance:   1.6354  on 5  degrees of freedom
## AIC: 66.703
##
## Number of Fisher Scoring iterations: 4
```

With smokers = 1 and 0 for nonsmokers: for non-smokers:

$-19.7 + 0.36x^2 - 0.02x^2$

for smokers:

$-17.34 + 0.33x2 - 0.02x2$

5.4 Non negative data with variance $\propto$ means (Task 17)

Compare the output from fitting a Poisson with log link and a line predictor on x to the data in Example 5.2.1 with that obtained using the the log link and assuming that the variance is proportional to the mean.

```
glm.lin <- glm(deaths ~ quarter, data=AIDS, family=poisson(link='log'))
summary(glm.lin)
```

```
##
## Call:
## glm(formula = deaths ~ quarter, family = poisson(link = "log"),
##     data = AIDS)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.21008  -1.02032  -0.69704   0.04028   2.70758
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.33963    0.25119   1.352    0.176
## quarter      0.25652    0.02204  11.639   <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: 207.272  on 13  degrees of freedom
## Residual deviance:  29.654  on 12  degrees of freedom
## AIC: 86.581
##
## Number of Fisher Scoring iterations: 5
```

```
glm.quasi <- glm(deaths ~ quarter, data=AIDS, family=quasi(variance = "mu", link='log') )
summary(glm.quasi)
```

```
##
## Call:
## glm(formula = deaths ~ quarter, family = quasi(variance = "mu",
##     link = "log"), data = AIDS)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.21008  -1.02032  -0.69704   0.04028   2.70758
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.33963    0.38946   0.872      0.4
## quarter      0.25652    0.03417   7.507 7.17e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for quasi family taken to be 2.403942)
##
##     Null deviance: 207.272  on 13  degrees of freedom
## Residual deviance:  29.654  on 12  degrees of freedom
## AIC: NA
##
## Number of Fisher Scoring iterations: 5
```

The only difference is in the "Std. Error" (see summary output above).

5.5 Further statistical modelling of count data

A train operator is reviewing the capacity of its trains between certain English cities. Their particular interest is in modelling the number of train passengers starting their train journey between 16:00 and 18:00 on weekdays between English cities.

- i starting station
- j destination
- n_{ij} number of people living within 5 miles of either station i or j
- r_{ij} number of passengers starting journey between 4 and 6 pm between station i and j

$$Po(\mu_{ij}) = \frac{\exp(-\mu_{ij})\mu^{r_{ij}}}{r_{ij}!}$$

$$\mu_{ij} = \alpha_i \exp(\beta n_{ij})$$

Models the probablity of having r_{ij} passengers starting journey from station i to station j.

- S - starting station (same as i)
- P - number of people living within 5 miles of the starting or destination stations
- C - number of passengers starting their train journey between 4 and 6 pm at station S.

```r
train <- e$train
train.glm <- glm(C~ factor(S) + P, family=poisson("log"),data=train)
summary(train.glm)
```

```
##
## Call:
## glm(formula = C ~ factor(S) + P, family = poisson("log"), data = train)
##
## Deviance Residuals:
##       1        2        3        4        5        6        7        8
##   2.022   -6.832    5.354   14.764   -5.759  -10.361   -2.667    9.397
##       9
##  -8.576
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  7.48476    0.04216  177.53   <2e-16 ***
## factor(S)2  -0.75522    0.01495  -50.53   <2e-16 ***
## factor(S)3  -1.13123    0.02228  -50.78   <2e-16 ***
## P            1.77882    0.05496   32.37   <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: 14258.07  on 8  degrees of freedom
## Residual deviance:   606.91  on 5  degrees of freedom
## AIC: 702.84
##
## Number of Fisher Scoring iterations: 4
```

We want to estimate α_i and β from the data to estimate μ_{ij} given n_{ij}.

- $\hat{\alpha_1} = \exp \gamma_1$
- $\hat{\alpha_i} = \exp \gamma_1 + \gamma_i$, for $i \geq 1$
- $\beta = \delta$

Task 18

Use the output to determine α_1. Hence, show that the parameter estimates α_i and β satisfy the first of the mle equations

$$\sum_{j \in D(i)} [r_{ij}/\alpha_i - \exp(\beta n_{ij})] = 0$$