MAST30027: Modern Applied Statistics

Week 5 Lab

1. In a binomial model we assume that any given observation is from a bin(m, p) distribution, for some m and p. That is, we count the number of successes from m i.i.d. bernoulli(p) trials. There are two main ways that overdispersion can arise: the trials are not identically distributed, or the trials are not independent.

   A common way in which we can have heterogeneous trials is via clustering. Suppose that we have m trials split into h clusters of size $k = m/h$, and that the probability of success for a trial in the i-th cluster is p_i. Now suppose that p_i is random, with $\mathbb{E}p_i = p$ and $\operatorname{Var} p_i = \tau^2 p(1-p)$. Let the number of successes from cluster i be Z_i and let the total number of successes be $Y = Z_1 + \cdots + Z_h$.

   Show that

   (a)

   $$\mathbb{E}Y = mp$$

   (b)

   $$\operatorname{Var} Y = (1 + (k-1)\tau^2)mp(1-p)$$

   Hint: $\operatorname{Var} Y = \mathbb{E}\operatorname{Var}(Y|X) + \operatorname{Var}\mathbb{E}(Y|X)$.

   Thus Y is overdispersed, relative to a binomial.

   Solution:

   $$\begin{aligned} \mathbb{E}Y &= \sum_{i=1}^{h} \mathbb{E}Z_i = \sum_{i=1}^{h} \mathbb{E}\mathbb{E}(Z_i|p_i) \\ &= \sum_{i=1}^{h} \mathbb{E}kp_i = \sum_{i=1}^{h} kp = mp \end{aligned}$$

   By independence $\operatorname{Var} Y = \sum_i \operatorname{Var} Z_i$, and

   $$\begin{aligned} \operatorname{Var} Z_1 &= \mathbb{E}\operatorname{Var}(Z_1|p_1) + \operatorname{Var}\mathbb{E}(Z_1|p_1) \\ &= \mathbb{E}kp_1(1-p_1) + \operatorname{Var} kp_1 \\ &= kp - k(\tau^2 p(1-p) + p^2) + k^2\tau^2 p(1-p) \\ &= kp(1-p)(1 + (k-1)\tau^2). \end{aligned}$$

   Multiplying by h gives the result.

2. Suppose that $Y_i \sim \text{Poisson}(\lambda_i)$, where $\lambda_i \propto t_i$. For example, if we record the number of burglaries reported in different cities, the observed number will depend on the number of households in these cities. In other cases, the size variable t may be time. For example, if we record the number of customers served by sales people, we must take account of the differing amounts of time worked.

   We can model the rate *per unit time* using a log link via

   $$\log(\lambda_i/t_i) = x_i^T \beta$$

   where x_i are known predictors and β unknown parameters. That is

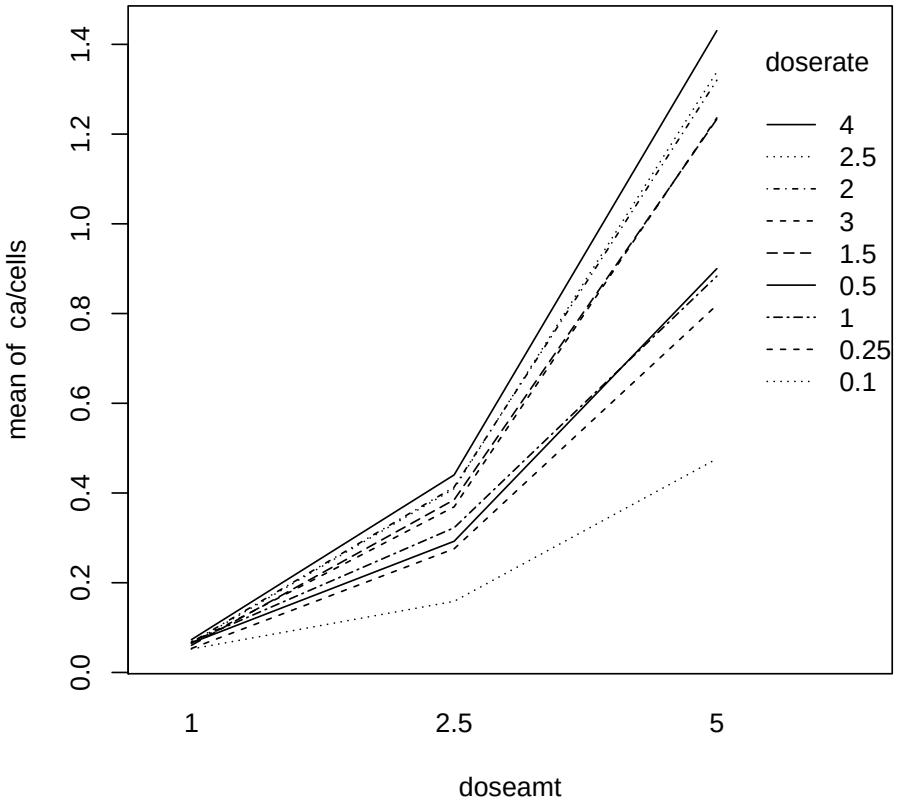
   $$\log(\lambda_i) = \log t_i + x_i^T \beta.$$

   This is of the form of a Poisson glm with log link, but where the coefficient of $\log t_i$ has been constrained to be 1. This is called a *rate model*.

In an R model description we can fix the coefficient of a variable to 1 by enclosing it in the `offset` function, viz `y ~ offset(log(t)) + x1 + x2 + ···`.

In Purott and Reeder (1976), some data is presented from an experiment conducted to determine the effect of gamma radiation on the numbers of chromosomal abnormalities (ca) observed. The number (cells), in hundreds of cells exposed in each run, differs. The dose amount (doseamt) and the rate (doserate) at which the dose is applied are the predictors of interest. We can plot the data as follows

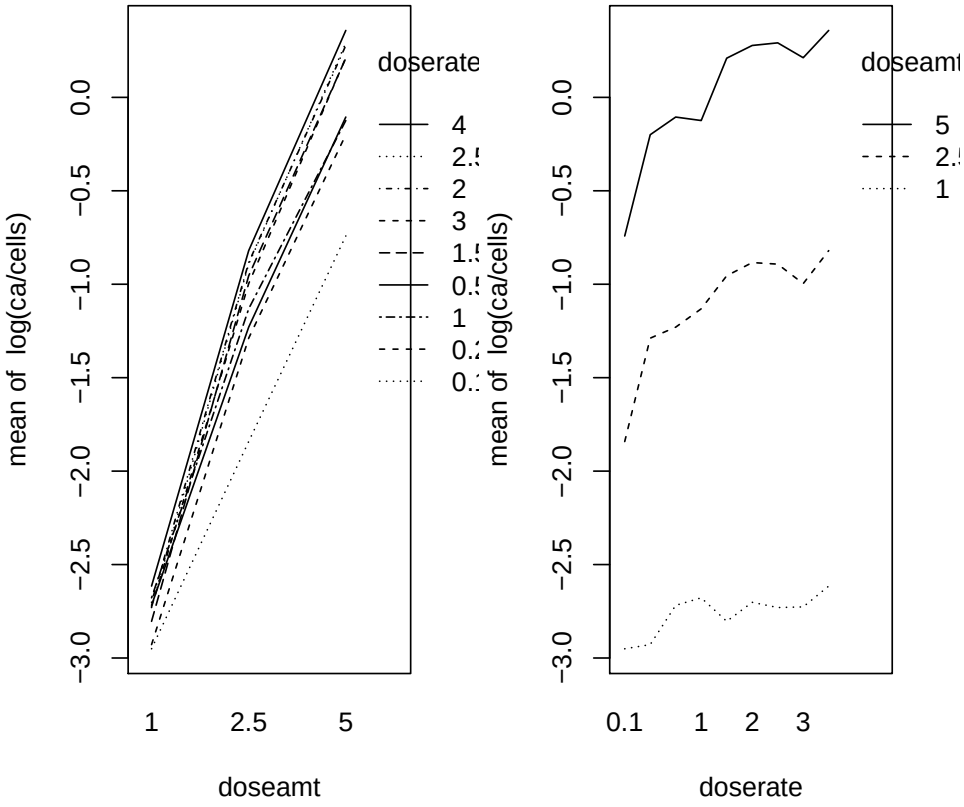
```
> library(faraway)
> data(dicentric)
> with(dicentric, interaction.plot(doseamt, doserate, ca/cells))
```



Fit a rate model to this data and perform diagnostics.

Solution: Plotting log(ca/cells) against doseamt and doserate helps us judge linearity.

```
> par(mfrow=c(1,2))
> with(dicentric, interaction.plot(doseamt, doserate, log(ca/cells)))
> with(dicentric, interaction.plot(doserate, doseamt, log(ca/cells)))
> par(mfrow=c(1,1))
```



The plots show nice linear relationships between log(ca/cells) and both doseamt and doserate. They also show a possible interaction between doseamt and doserate, since the slope of doserate vs. log(ca/cells) seems to depend on doseamt (and vice versa). We can now fit the model:

```
> model <- glm(ca ~ offset(log(cells)) + doserate*doseamt, family=poisson, data=dicentric)
> summary(model)

Call:
glm(formula = ca ~ offset(log(cells)) + doserate * doseamt, family = poisson,
    data = dicentric)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-5.7308  -2.2842  -0.6264   3.3487   5.8272

Coefficients:
                Estimate Std. Error z value Pr(>|z|)
(Intercept)     -3.29994    0.06160 -53.567  < 2e-16 ***
doserate         0.06401    0.02922   2.191 0.028476 *
doseamt          0.61224    0.01707  35.862  < 2e-16 ***
doserate:doseamt 0.02715    0.00765   3.549 0.000387 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 4753.00  on 26  degrees of freedom
Residual deviance:  270.26  on 23  degrees of freedom
AIC: 453.67
```

```
Number of Fisher Scoring iterations: 4
```

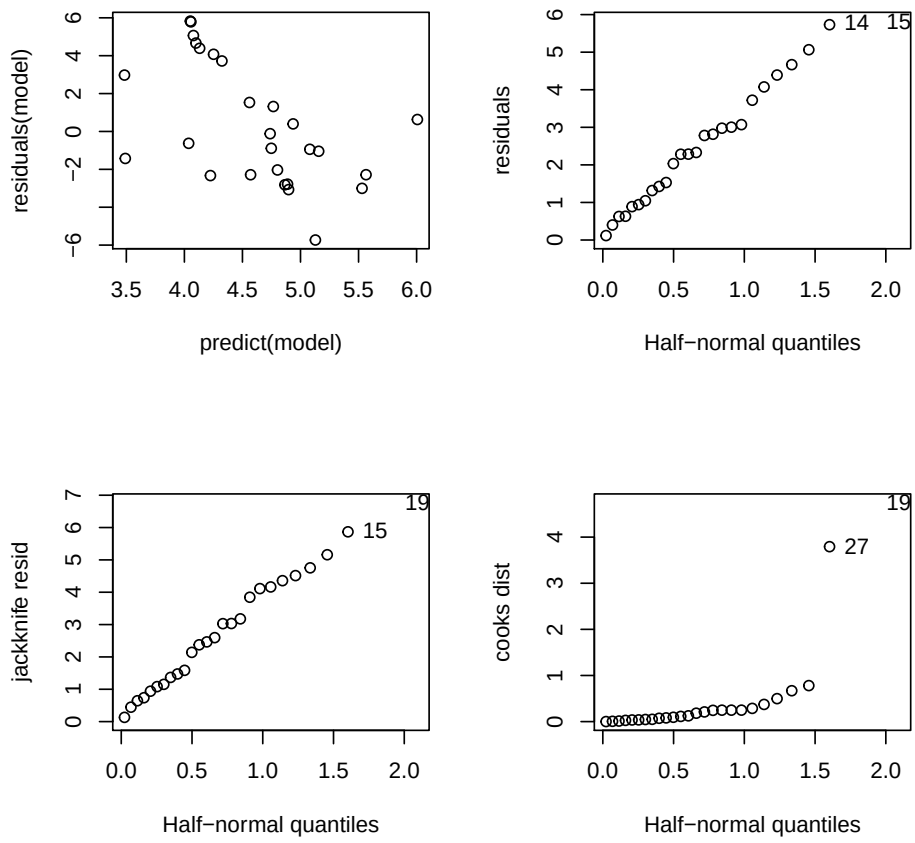
We can test the significance of the interaction using a chi-squared test. Not surprisingly, given its z-value, it appears very significant (but see below).

```
> anova(model, test="Chi")

Analysis of Deviance Table

Model: poisson, link: log

Response: ca

Terms added sequentially (first to last)


                 Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                                26     4753.0
doserate          1    231.3        25     4521.7 < 2.2e-16 ***
doseamt           1   4238.7        24      282.9 < 2.2e-16 ***
doserate:doseamt  1     12.7        23      270.3 0.0003681 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Our counts ca are reasonably large (the smallest is 25), so the deviance should look roughly chi-squared. You can check that the deviance of our fitted model is very high by testing for model adequacy. Thus something is amiss with the model.

The residuals look mostly OK. Points 19 and 27 have a large Cook's distance, but aren't distinguished otherwise. You can check that if you fit a model omitting these points, then the coefficients do not change much and the deviance is still very high.

```
> par(mfrow=c(2,2))
> plot(predict(model), residuals(model))
> halfnorm(residuals(model), ylab="residuals")
> halfnorm(rstudent(model), ylab="jackknife resid")
> halfnorm(cooks.distance(model), ylab="cooks dist")
> par(mfrow=c(1,1))
```



The resaon for the high deviance is overdispersion. We will consider this issue in the practical problems for the week 6.