Homework 8

Alexander Van Roijen

March 1, 2019

```
## Loading required package: sandwich
```

```
poisRes= glm(count1 ~ dose +sex +age +count0, family="poisson", data=cellData)
psum = summary(poisRes)
print(psum)
```

```
##
## Call:
## glm(formula = count1 ~ dose + sex + age + count0, family = "poisson",
##     data = cellData)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -26.924   -5.282   -1.411    3.957   29.779
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  4.4301424  0.0466576  94.950   <2e-16 ***
## dose         0.0277232  0.0003444  80.506   <2e-16 ***
## sex          0.1919804  0.0213115   9.008   <2e-16 ***
## age         -0.0228122  0.0006803 -33.533   <2e-16 ***
## count0       0.0052013  0.0001438  36.180   <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: 17055.9  on 39  degrees of freedom
## Residual deviance:  4018.9  on 35  degrees of freedom
## AIC: 4278.9
##
## Number of Fisher Scoring iterations: 5
```

a) The above holds the coefficient table and the summary for the poisson regression on the cell data with adjustment for the required factors

```
print(exp(psum$coefficients[2,1]))
```

```
## [1] 1.028111
```

b/c) Accordingly, dose, which has evidence of significance through its low p value, when we exponentiate the coefficient represents the ratio of the mean post cell counts per unit difference in dose. Thus the ratio of the mean post cell counts, holding everything else constant, is 1.028 per unit increase in dosage.

```
v <- vcovHC(poisRes)
robust.se <- sqrt(diag(v))
round(cbind(psum$coef,robust.se),4)
```

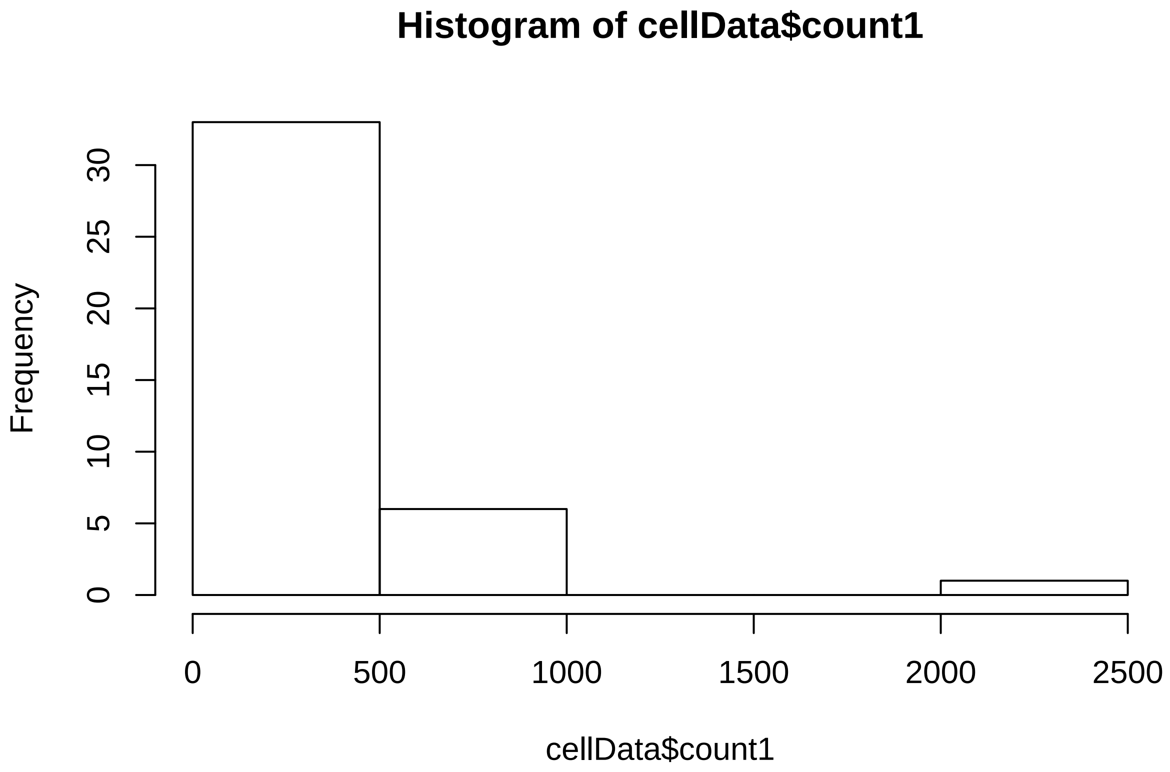
```
##             Estimate Std. Error  z value Pr(>|z|) robust.se
## (Intercept)   4.4301     0.0467  94.9501        0    0.5724
## dose          0.0277     0.0003  80.5063        0    0.0028
## sex           0.1920     0.0213   9.0083        0     0.4806
## age          -0.0228     0.0007 -33.5333        0     0.0157
## count0        0.0052     0.0001  36.1797        0     0.0022
```

d) Now looking at the the robust standard errors, we can see there are quite significant departures, showing that we probably dont satisfy the assumptions of the mean variance model of poisson regression. It appears that accordingly, it may be wise to use robust SE instead of our normal SEs. However, as we will see below, there is a small sample size, which does not bode well for robust SEs. This makes it a bit of a treacherous path. Also note that they are quite larger than our previously calculated standard errors. Now lets examine the assumptions

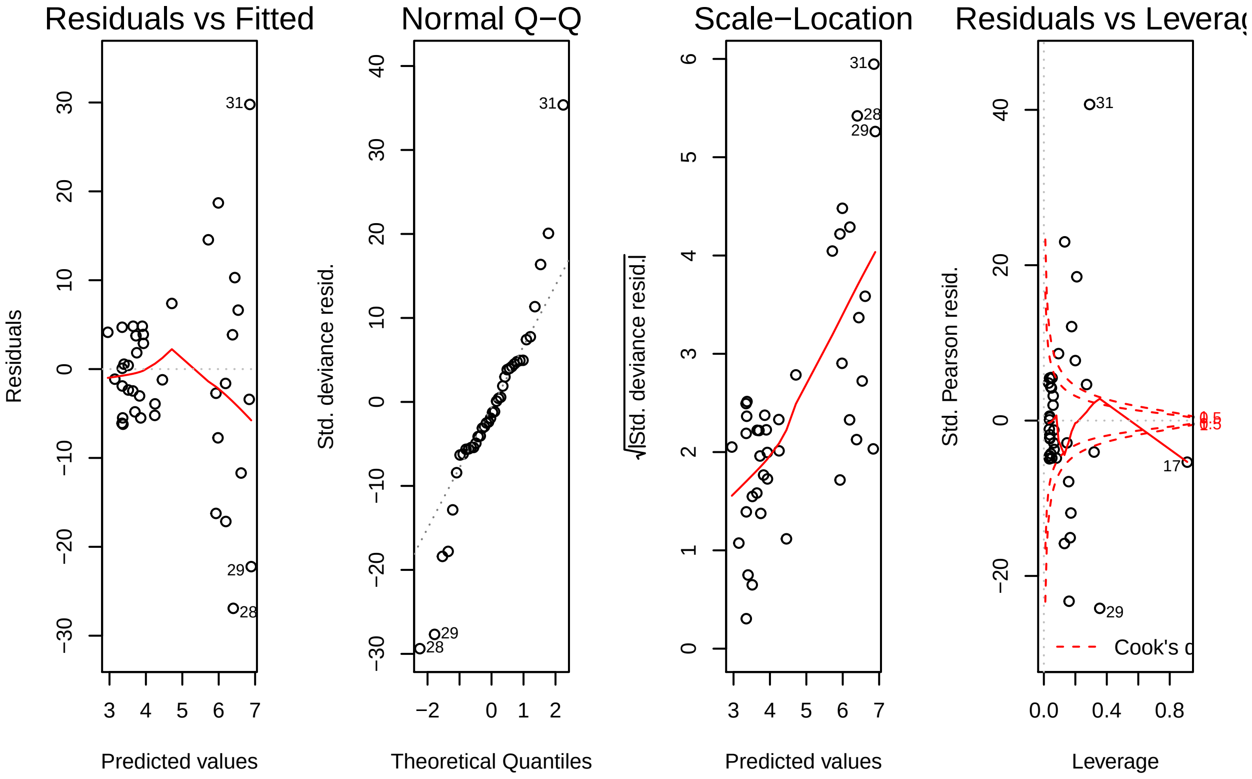
```
print(length(cellData$id))
```

```
## [1] 40
```

```
hist(cellData$count1)
```



```
par(mfrow=c(1,4))
plot(poisRes)
```



```r
print(c(mean(cellData$count1),var(cellData$count1)))
```

```
## [1]    251.05 156884.56
```

e/f) We will assume that the independence assumption is met, however this isnt always guaranteed. Now, we can see that our residuals show a non-constant variance, which agrees with the usage of robust SEs. Comparing to our results from homework 7, our other predictors became significant along with dose. However, the much lower than robust standard errors likely caused part of this. The reason for this may be to a lack of good fit of poisson regression to this data. First, we note a rather small sample size of 40, along with the fact that the $Var(Y) \neq \mu$. It is noteworthy that they did both state that dosage had a positive relationship with the post treatment cell count.

```r
poisRes= glm(count1 ~ factor(dose) +sex +age +count0, family="poisson", data=cellData)
psum = summary(poisRes)
print(psum)
```

```
##
## Call:
## glm(formula = count1 ~ factor(dose) + sex + age + count0, family = "poisson",
##     data = cellData)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -26.845   -4.809   -1.107    3.594   29.722
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)      4.518128   0.054452  82.974   <2e-16 ***
## factor(dose)10   0.100033   0.059844   1.672   0.0946 .
## factor(dose)100  2.678908   0.045693  58.628   <2e-16 ***
## sex              0.193511   0.021322   9.076   <2e-16 ***
## age             -0.022820   0.000681 -33.507   <2e-16 ***
## count0           0.005406   0.000161  33.572   <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: 17055.9  on 39  degrees of freedom
## Residual deviance:  4010.2  on 34  degrees of freedom
## AIC: 4272.2
##
## Number of Fisher Scoring iterations: 5
```

```
print(exp(psum$coefficients[1:3,1]))
```

```
##     (Intercept)  factor(dose)10 factor(dose)100
##       91.663807        1.105208       14.569178
```

a/b)We can see that the factor of 0 is our reference point, and the coefficients for factor of 10 and 100 are calculated with respect to it. Now interpreting the exponentiated coefficients, holding everything else constant, the baseline for factor(0) is a 91 ratio of mean post cell count. Relative to it, the exponentiated coefficients for factor(dose) 10 and 100 have a multiplicative factor compared to factor(0) of 0.1 and 14.57, which represents the ratio of mean post treatment cell count per unit increase in that respective level of dosage. However, factor(dose) 10 doesn't appear to be significant in this model.

```
copyTeeth = teethData
copyTeeth$newExtr = as.integer(as.logical(teethData$EXTR))
teethResB = (summary(glm(newExtr~PDALL+AGE+GENDER, data=copyTeeth, family=binomial)))
print(teethResB)
```

```
##
## Call:
## glm(formula = newExtr ~ PDALL + AGE + GENDER, family = binomial,
##     data = copyTeeth)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6678  -0.5769  -0.4611  -0.3570   2.5168
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -8.36031    1.03959  -8.042 8.84e-16 ***
## PDALL        1.45168    0.16710   8.687  < 2e-16 ***
## AGE          0.03035    0.01460   2.079   0.0376 *
## GENDERM     -0.33090    0.20589  -1.607   0.1080
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 724.76  on 795  degrees of freedom
## Residual deviance: 636.30  on 792  degrees of freedom
##   (1 observation deleted due to missingness)
## AIC: 644.3
##
```

```
## Number of Fisher Scoring iterations: 5
```

```
teethResP = (summary(glm(newExtr~PDALL+AGE+GENDER, data=copyTeeth, family=poisson)))
print(teethResP)
```

```
## 
## Call:
## glm(formula = newExtr ~ PDALL + AGE + GENDER, family = poisson, 
##     data = copyTeeth)
## 
## Deviance Residuals: 
##     Min       1Q   Median       3Q      Max  
## -1.3132  -0.5486  -0.4700  -0.3984   1.9304  
## 
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)    
## (Intercept) -6.11233    0.77177  -7.920 2.38e-15 ***
## PDALL        0.91241    0.10350   8.816  < 2e-16 ***
## AGE          0.01990    0.01235   1.612    0.107    
## GENDERM     -0.23356    0.17464  -1.337    0.181    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for poisson family taken to be 1)
## 
##     Null deviance: 479.07  on 795  degrees of freedom
## Residual deviance: 415.91  on 792  degrees of freedom
##   (1 observation deleted due to missingness)
## AIC: 693.91
## 
## Number of Fisher Scoring iterations: 6
```

```
pdallL = (teethResB$coefficients[2,1])
pdExp = exp(pdallL)
print(pdExp)
```

```
## [1] 4.270297
```

```
ageCoef = (teethResB$coefficients[3,1])
expACoef = exp(ageCoef)
print(expACoef)
```

```
## [1] 1.030816
```

```
poisData = (teethResP$coefficients[1:2,1])
pdpExp = exp(poisData)
print(pdpExp)
```

```
## (Intercept)       PDALL 
##  0.00221539  2.49030950
```

a)Looking at the results of our logistic regression model, we see that PDALL and AGE are significant and we reject the null hypothesis that they have no impact on whether or not a tooth is extracted.

b)Interpreting the exponentiated coefficients for the model, we see for the logistic regression, the ratio of the odds of a patient having a tooth extracted is 4.27 per unit increase in PDALL assuming all else held constant. Meanwhile, for age, the exponentiated coefficient indicates the ratio of the odds of a patient having a tooth extracted as 1.031 per unit increase in age assuming all else held constant.

c)Looking at the results of the Poisson regression, we can see that only PDALL is significant in this model.

d) Finally for the Poisson model, the exponentiated coefficient of 2.5 for PDALL indicates a rate ratio of 2.5 per unit increase in the PDALL term assuming all other terms held constant. Note the baseline ratio of 0.0022. I do not bother interpreting the other parameters as they are not significant.

e) Overall, the models give similar results. They both indicate a positive relationship between the odds of a patient having a tooth extracted and the PDALL term. However, they disagree on the significance of the age relationship, but agree on its signage.

Problem 4, bonus

```
minXs = rep(10,25)
maxXs = rep(40,25)
allData = c(minXs,maxXs)
meanX = mean(allData)
print(meanX)
```

```
## [1] 25
```

```
print(sum((allData-meanX)**2))
```

```
## [1] 11250
```

Now, we know that our assumptions of linearity, constant variance, and independence are met.

We further understand that the formula for $SE(\hat{\beta}) = \frac{\hat{\sigma}}{\sqrt{\sum_{i=1}^{n=50}(X_i-\bar{X})^2}}$

Knowing this, along with out assumptions, we can see that our $\hat{\sigma}$ and n are constant. Thus, the only thing that can change in our calculation is the value of $\sqrt{\sum_{i=1}^{n=50}(X_i-\bar{X})^2}$. Further, given the range $10 \leq X_i \leq 40$, we must ensure our values satisfy this requirement.

We can see intuitively, that we must maximize that square root of a sum to maximize the denominator and ultimately minimize the SE.

Thus, lets examine what happens to the sum of squared differences (ignoring the square root as it is meaningless to our goal) starting with all X equal, which means $\bar{X} = X_i$.

It is obvious that the derivative of $\frac{d}{dX_i}X_i = 1$, and $\frac{d}{dX_i}\bar{X} = \frac{1}{n}$. Now, consider moving all X_i in our sample. We will thus get for every one unit increase in all X_i a one unit increase in $\bar{X}$. This obviously doesn't help, as our sum of squared differences is still zero! Effectively, we want the differences between the sum of the respective slopes to be maximized, which can be represented by the following difference $i - \frac{i^2}{n} = \frac{i(n-i)}{n}$. Where i is the number of points we are moving, and n is the sample size.

Taking the derivative with respect to i we get $\frac{(n-i)}{n} + \frac{-i}{n} = \frac{n-2i}{n}$.

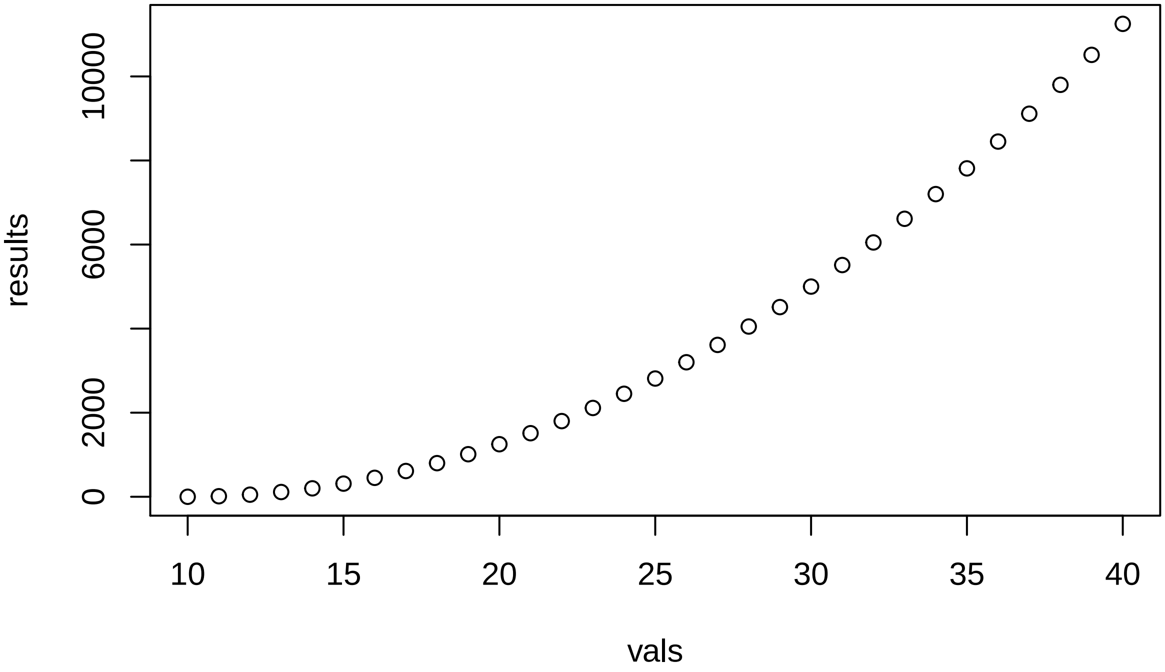
Taking the second derivative we get $\frac{-2}{n}$ which implies our function is concave down, meaning we will find a maximum at the critical point.

Finding the critical point is done by setting our first derivative to zero and solving. So we get $\frac{n-2i}{n} = 0 => 2i = n > i = \frac{n}{2}$

This means that to maximize the difference, we should move half the points by one unit, which means to maximize our overall sum of squared differences, we want to move half of the points as much as possible. so, thus in our example, the minimized standard error of the least squares estimate is achieved by setting half of our units at the minimum, 10, and the other half at maximum, 40.

The graphics below demonstrate the curvature, and reflect the results we achieved.

```r
startMax = 40
results = numeric()
vals = numeric()
counter = 1
while(startMax>=10)
{
  minXs = rep(10,25)
  maxXs = rep(startMax,25)
  allData = c(minXs,maxXs)
  meanX = mean(allData)
  results[counter] = sum((allData-meanX)**2)
  vals[counter] = startMax
  counter = counter +1
  startMax= startMax-1
}
plot(x=vals,y=results)
```



This would normally continue on if there were no bounds to either end of our range.