MA 576 HW 4

Benjamin Draves

```
#load necessary packages
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

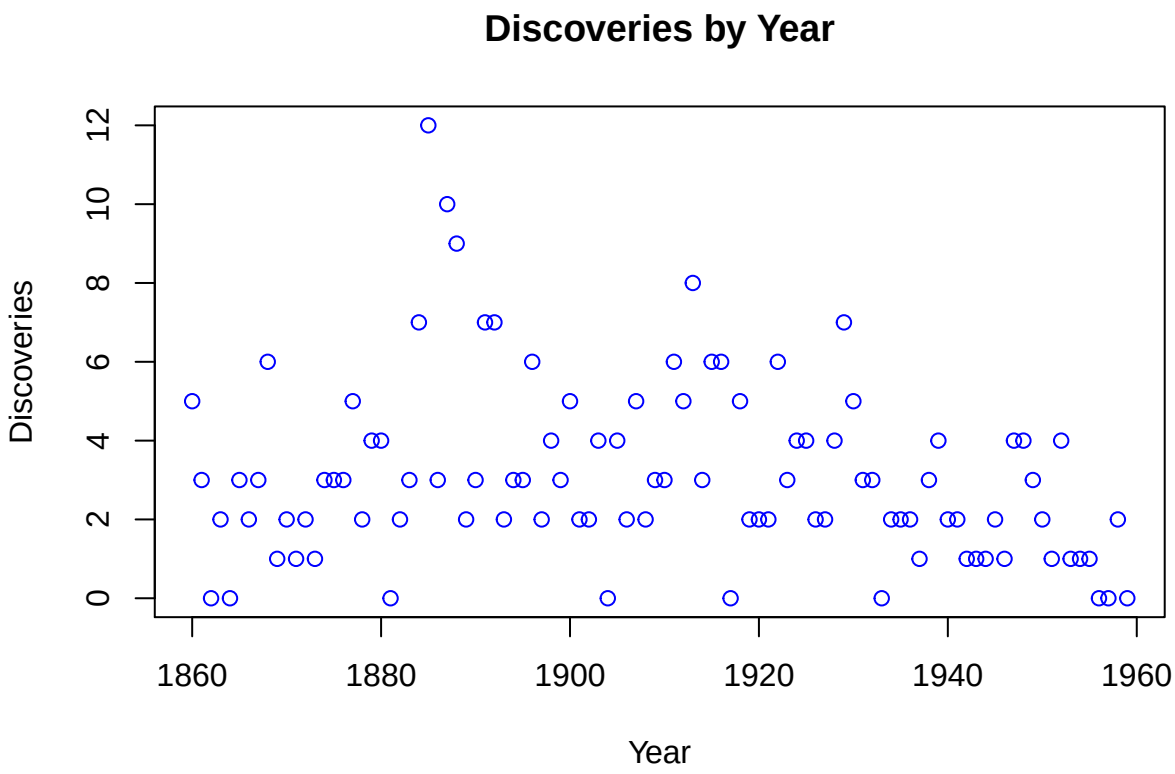
```
library(tidyr)
library(ggplot2)
```

Exercise 3

(a)

```
#read in and format data
discover = read.table("~/Desktop/Courses/MA 576/data/discoveries.dat")
discover$Year = 1860:1959
colnames(discover)[1] = "Discoveries"
discover = data.frame(discover)

#Make plot
plot(discover$Year, discover$Discoveries,
     xlab = "Year", ylab = "Discoveries",
     main = "Discoveries by Year", col = "blue")
```



The rate appears to be variable through time. Specifically around 1880 - 1900, there are several years with more than 5 discoveries per year. This rate generally decreases as a function of a time. That is, past 1940, there are no years with more than 5 major discoveries. In generally, it appears the discovery rate increases until about 1900 when it starts to decrease. We can model this rate using a Poisson GLM.

(b)

```
model = glm(Discoveries ~ poly(Year, 5), data = discover, family = poisson)
summary(model)
```

```
##
## Call:
## glm(formula = Discoveries ~ poly(Year, 5), family = poisson,
##     data = discover)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.8960  -0.8608  -0.2358   0.5944   3.1357
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)      1.06305    0.06183  17.194  < 2e-16 ***
## poly(Year, 5)1 -2.11931    0.72160  -2.937  0.00331 **
## poly(Year, 5)2 -3.28460    0.73115  -4.492 7.04e-06 ***
## poly(Year, 5)3 -0.18724    0.72058  -0.260  0.79498
```

```
## poly(Year, 5)4 -0.75876    0.67824  -1.119  0.26326
## poly(Year, 5)5 -0.91801    0.65796  -1.395  0.16294
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 164.68  on 99  degrees of freedom
## Residual deviance: 129.99  on 94  degrees of freedom
## AIC: 410.99
##
## Number of Fisher Scoring iterations: 5
```

While using the poly function, we can compare orthogonal polynomial values of Year. In this way, we can ensure that there is no inflated variance due to colinearity. This model shows that any polynomial degree past 2 describes an insignificant amount of the variability in the discovery rate. For this reason, we will only consider the linear and quadratic models.

```
model0 = glm(Discoveries ~1, data = discover, family = poisson)
model1 = glm(Discoveries ~ poly(Year, 1, raw = TRUE), data = discover, family = poisson)
model2 = glm(Discoveries ~ poly(Year, 2, raw = TRUE), data = discover, family = poisson)
model3 = glm(Discoveries ~ poly(Year, 3, raw = TRUE), data = discover, family = poisson)
anova(model0, model1, model2, model3, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: Discoveries ~ 1
## Model 2: Discoveries ~ poly(Year, 1, raw = TRUE)
## Model 3: Discoveries ~ poly(Year, 2, raw = TRUE)
## Model 4: Discoveries ~ poly(Year, 3, raw = TRUE)
##   Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
## 1        99     164.69
## 2        98     157.32  1   7.3688  0.006637 **
## 3        97     132.84  1  24.4774 7.519e-07 ***
## 4        96     132.73  1   0.1112  0.738801
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Here we see that indeed the addition of a quadratic year term results in a significant reduction in the deviance. While the linear term does the same, this test statistic is not nearly as strong as that of the quadratic term. For this reason we consider the quadratic only model and will compare model fit criterion to choose the most parsimonious model.

```
model3 = glm(Discoveries ~ I(Year^2), data = discover, family = poisson)
summary(model3)
```

```
##
## Call:
## glm(formula = Discoveries ~ I(Year^2), family = poisson, data = discover)
##
```

```
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.8125  -0.9510  -0.3522   0.6636   3.5462
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  6.304e+00  1.887e+00   3.340 0.000838 ***
## I(Year^2)   -1.422e-06  5.197e-07  -2.735 0.006231 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 164.68  on 99  degrees of freedom
## Residual deviance: 157.14  on 98  degrees of freedom
## AIC: 430.15
##
## Number of Fisher Scoring iterations: 5
```

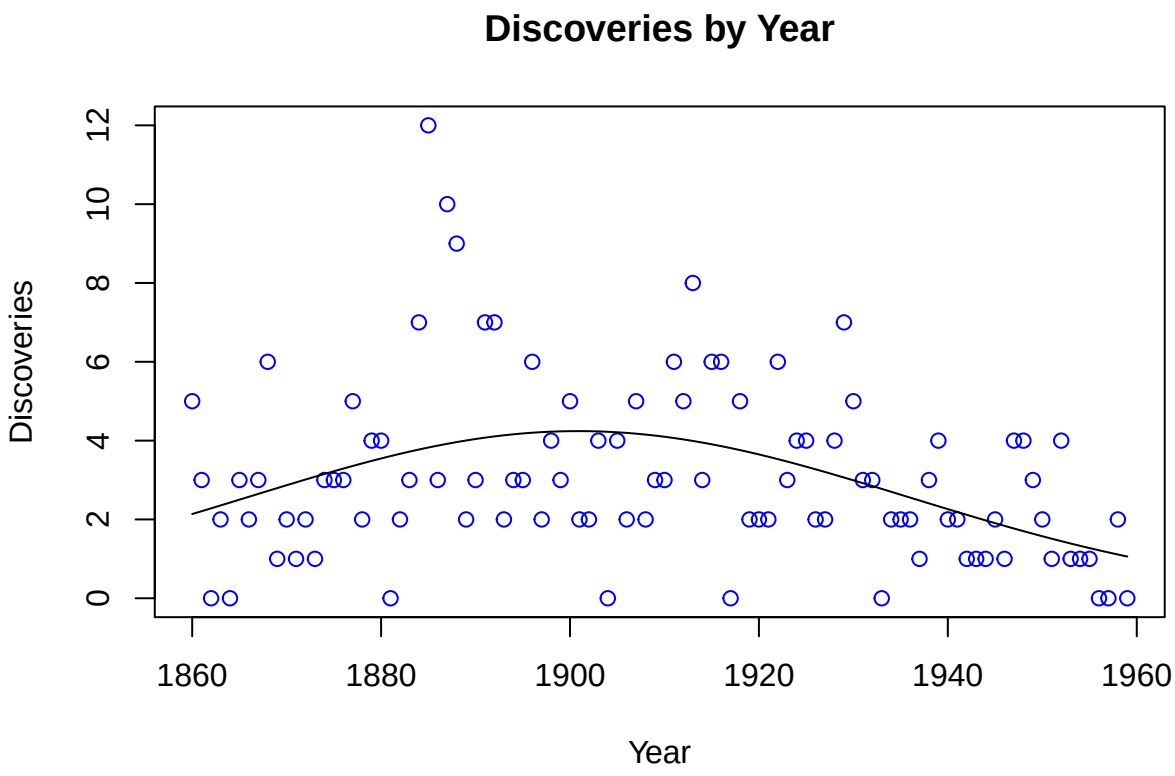
Here we see that the single term model, with $Year^2$ as the only covariate, still suggests that this term explains a sizable portion of the deviance in the discovery rate. This model, however, does not explain the data as well as that of the model including the linear term. One can see this by the increase in AIC values from 407.85 to 430.15. Hence, we choose the two term model including Year and $Year^2$ for our working model.

The estimated coefficient values are given by $\beta = (-1482, 1.561, -0.0004106)$. We can interpret these values as follows. In year 0, we expect to see $exp(-1482) \approx 0$ discoveries. We can interpret the covariate values as follows. As

$$Discoveries = e^{\beta_0} e^{\beta_1 Year + \beta_2 Year^2}$$

we see that for every additional year, we expect to see the baseline number of discoveries to be modulated by the value of $e^{\hat{\beta}_1 Year + \hat{\beta}_2 Year^2}$. Seeing that β_1 is positive, we expect as number of years increases, then we expect the discovery rate to increase by a factor of 4.763582. But with this growth, we also expect each additional year^2 we expect the discovery rate to decrease by a factor of 0.9995895 (due to the quadratic term). Jointly, the effect of year on discovery rate is a concave function in Year. We expect to see a peak number of discoveries at around the year 1900. Moreover, with any additional year before (or after) 1900, we expect a reduction in the number of discoveries. This trend is evident in the following plot.

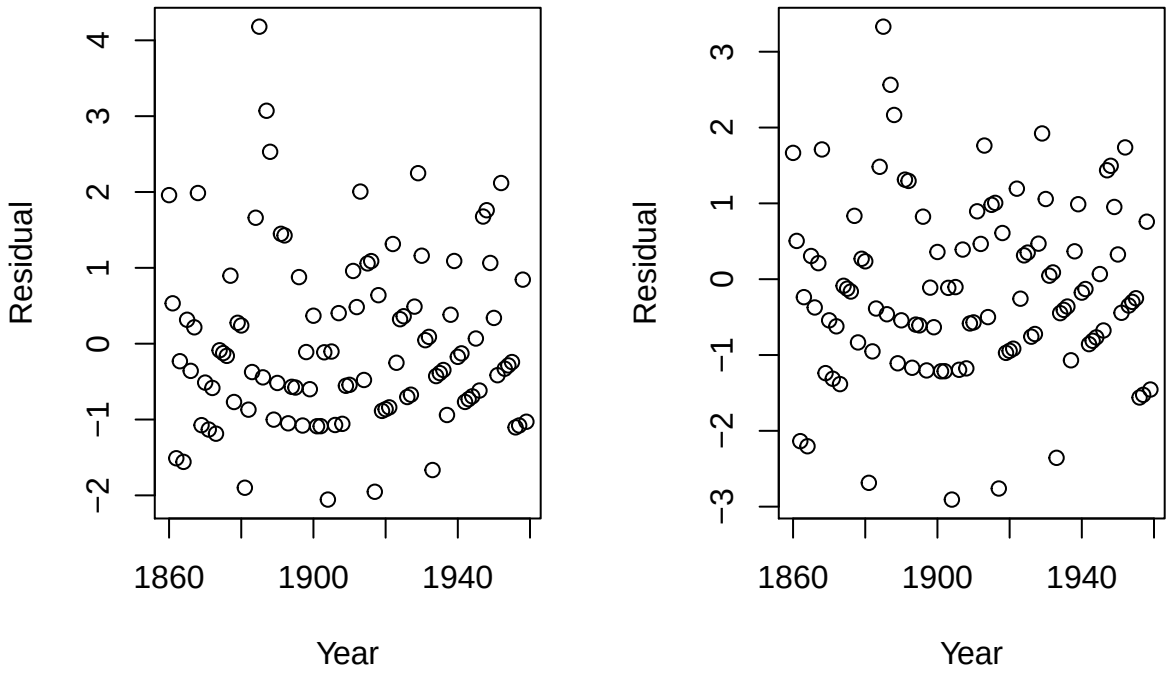
```
plot(discover$Year, discover$Discoveries,
      xlab = "Year", ylab = "Discoveries",
      main = "Discoveries by Year", col = "blue")
points(1860:1959, model2$fitted.values, type = "l")
```



(c)

```
dev.res = resid(model2,type='pearson')
pear.res = resid(model2,type='deviance')

par(mfrow = c(1,2))
plot(discover$Year, dev.res,xlab = "Year", ylab = "Residual")
plot(discover$Year, pear.res,xlab = "Year", ylab = "Residual")
```



The residuals appear to have clear banding issues. That is there is a clear quadratic patter in both the Pearson and deviance residuals. This suggests that we could be missing some quadratic variance in the model due to another outstanding covariate. Moreover, we see that several values lay outside the range of plus or minus 2 in the Pearson residuals. As we expect that the majority of these residuals should asymptotically lay within ± 2 we could have a few outliers/leverage points. Lastly, we see that out residual deviance is 132.84 on 97 degrees of freedom suggesting some over dispersion in this model. We test this with the following chi-square test

```
#find rejection region
sigma2 <- sum(residuals(model2,type="pearson")^2)/(model2$df.resid)
#Rejection region: (cutoff, \infty)
cutoff = qchisq(0.95, model2$df.residual)
#test statistic
ts = sigma2*model2$df.residual

#test hypothesis
cutoff < ts
```

```
## [1] TRUE
```

Here we see that we have significant evidence to suggest that $\hat{\sigma^2} \not\leq 1$. Therefore, we see we have a significant amount of overdispersion in this model.

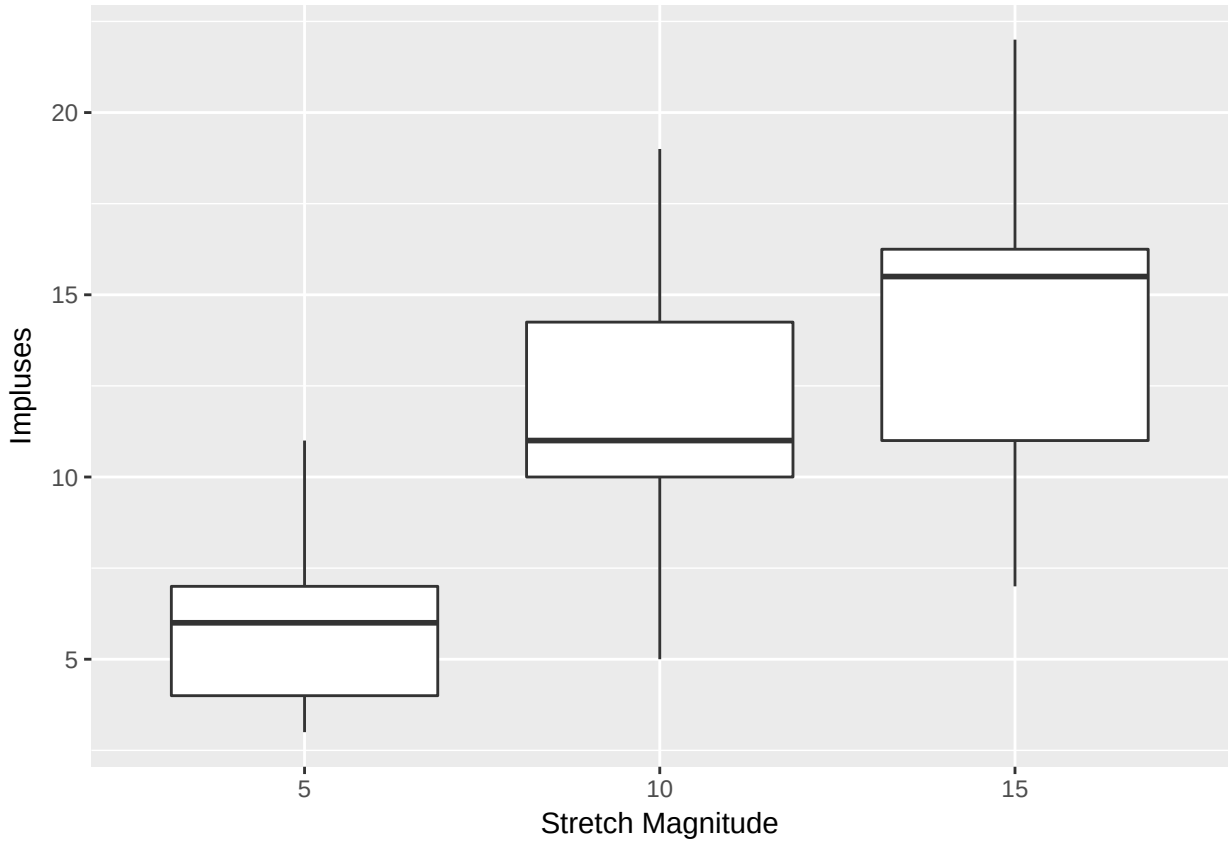
4

(a)

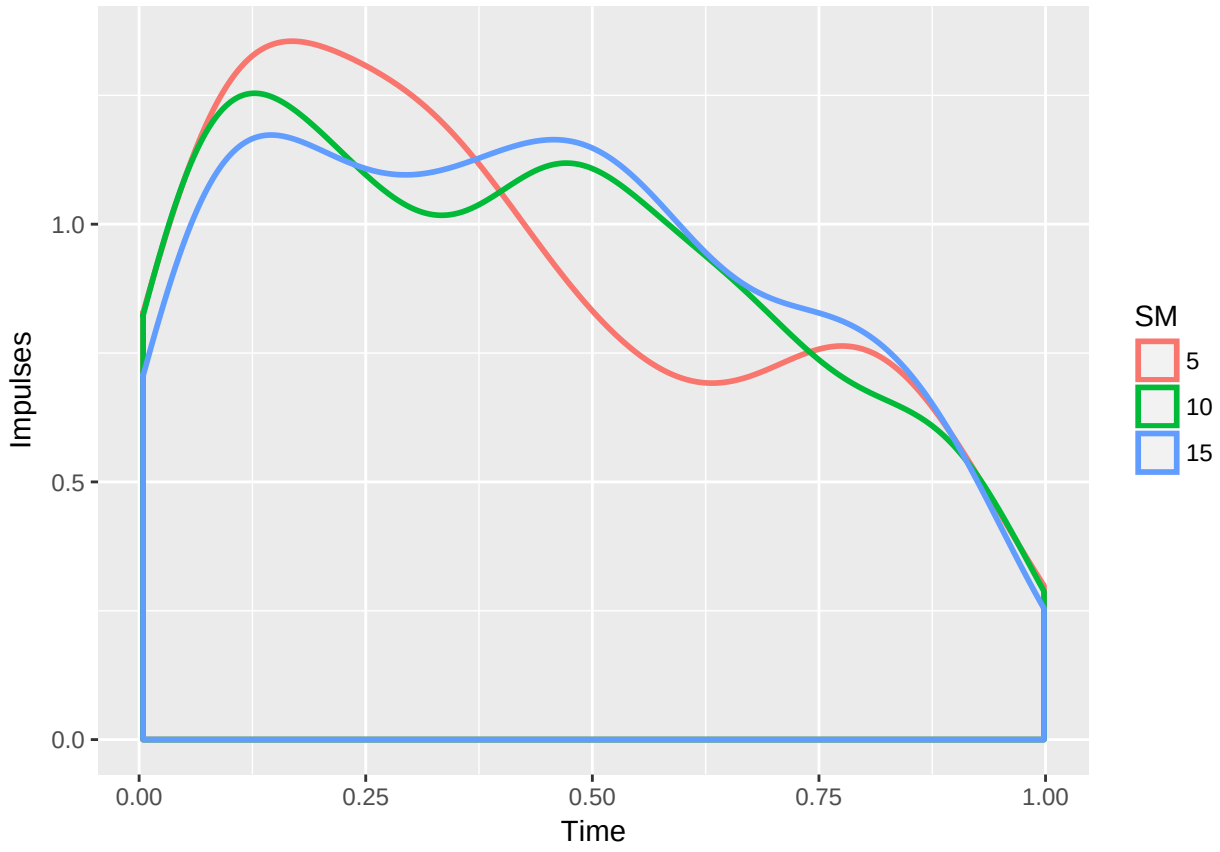
```
#read in data
frogs = discover = read.table("~/Desktop/Courses/MA 576/data/stretch.dat",
                              header = TRUE)
```

```r
# format data
nfrogs = group_by(frogs, Trial, StretchMagnitude)
impulses = data.frame(summarize(nfrogs, nimpulses = n()))

#visualize data
ggplot(impulses, aes(x = as.factor(StretchMagnitude), y = nimpulses)) +
  geom_boxplot() +
  labs(x = "Stretch Magnitude", y = "Impluses")
```




```r
groupfrogs = frogs
groupfrogs$SM= as.factor(frogs$StretchMagnitude)
ggplot(groupfrogs, aes(x = SpikeTimes, color = SM))+
  geom_density(position = "identity", fill = NA, size = 1) +
  labs(x = "Time", y = "Impulses")
```



(b)

```
model = glm(nimpulses ~ StretchMagnitude, data = impulses, family = poisson)
summary(model)
```

```
##
## Call:
## glm(formula = nimpulses ~ StretchMagnitude, family = poisson,
##     data = impulses)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.36645  -0.91061  -0.04635   0.44639   2.47577
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)       1.50228    0.12055  12.462  < 2e-16 ***
## StretchMagnitude  0.08149    0.01006   8.099 5.53e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
```

```
##     Null deviance: 135.919  on 59  degrees of freedom
## Residual deviance:  67.589  on 58  degrees of freedom
## AIC: 318.29
##
## Number of Fisher Scoring iterations: 4
```

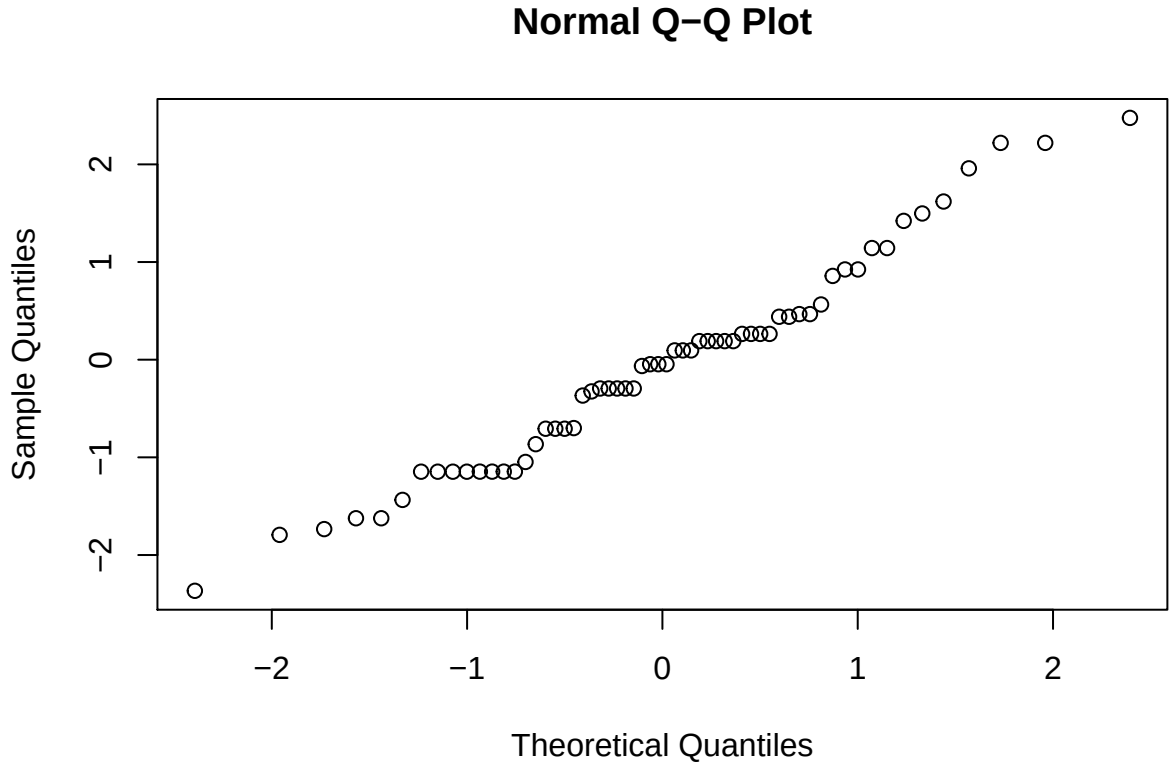
I argue that this value should be considered a continuous value. A muscle can be stretched in a number of ways that does not naturally lay on a discrete scale. That is, these values,(5,10,15), all represent an underlying continuous scale of muscle stretching. Therefore we should model is as a continuous quantity.

The model coefficients are given by $\hat{\beta} = (1.50228, 0.08149)$. These values can be interpreted as follows. For a muscle under no tension ($StretchMagnitude = 0$), we expect to observe $e^{1.50228} = 4.491919$ electrical impulses in a minute. In addition with each unit increase in Stretch Magnitude, we expect that the number of observed electrical impulses per minute to increase by a factor of $e^{0.08149} = 1.084902$ compared to the baseline rate. Moreover, we see that the StretchMagnitude is quite significant and explains a significant portion of deviance in this model. Moreover, we see that the residual deviance is 67.589 on 58 degrees of freedom. We will test to see if this significant evidence to for over dispersion.

```r
#Chisquare test for sigma^2
sigma2 <- sum(residuals(model,type="pearson")^2)/(model$df.resid)
1-pchisq(sigma2*model$df.residual,model$df.residual)
```

```
## [1] 0.15798
```

```r
#check our residuals
qqnorm(residuals(model,type="deviance"))
```

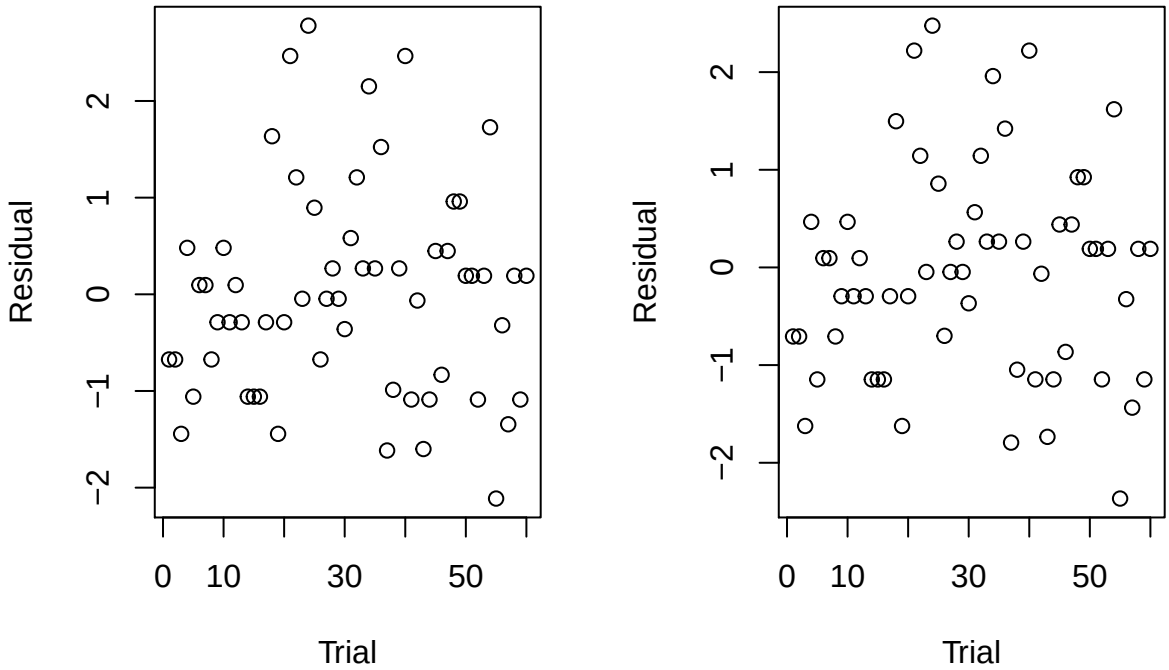


```r
anova(model,test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model: poisson, link: log
##
## Response: nimpulses
##
## Terms added sequentially (first to last)
##
##
##                   Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                                     59    135.919
## StretchMagnitude  1   68.329        58     67.589 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
dev.res = resid(model,type='pearson')
pear.res = resid(model,type='deviance')

par(mfrow = c(1,2))
plot(dev.res,xlab = "Trial", ylab = "Residual")
plot(pear.res,xlab = "Trial", ylab = "Residual")
```



As our chi-square test shows, we do not have evidence for over dispersion in this model. Moreover, the residuals in this model appear quite normal. Lastly, they appear to have constant variance and lay within the ± 2 threshold as we expect asymptotically. While there may be a few issues with outliers, this model fits quite well.

(c)

```
#get counts by ms using hist function
df = matrix(NA, ncol = 4)
for(i in 1:60){

    #use hist to get counts per bin
  tmp = frogs[frogs$Trial==i,]
  x = hist(tmp$SpikeTimes, breaks = seq(0,1,.001), plot = FALSE)$counts
  dfi = cbind(rep(i, 1000), rep(frogs[frogs$Trial == i,2][1], 1000),seq(0,1,.001)[-1],x)
  dim(dfi)

  #put in the df
  df = rbind(df,dfi)
}

#clean up df
df = df[-1,]
colnames(df) = c("Trial", "StretchMagnitude","msBin","Count")
df = data.frame(df)
```

```
model = glm(Count ~ StretchMagnitude + msBin, data = df, family = poisson)
summary(model)
```

```
##
## Call:
## glm(formula = Count ~ StretchMagnitude + msBin, family = poisson,
##     data = df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.2172  -0.1638  -0.1393  -0.1182   3.0025
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -4.96796    0.13423 -37.011  < 2e-16 ***
## StretchMagnitude  0.08149    0.01006   8.099 5.53e-16 ***
## msBin            -0.94850    0.13968  -6.791 1.12e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 5833.2  on 59999  degrees of freedom
## Residual deviance: 5717.8  on 59997  degrees of freedom
## AIC: 7009.8
##
## Number of Fisher Scoring iterations: 7
```

Without considering the deviance of this model, there is evidence to suggest that time plays a roll in this process as the msBin is a significant variable. But we do note that as the deviance is 5717.8 on 5717.8 we see that this model is quite underdispersed so making any statement on confidence levels is inappropriate.

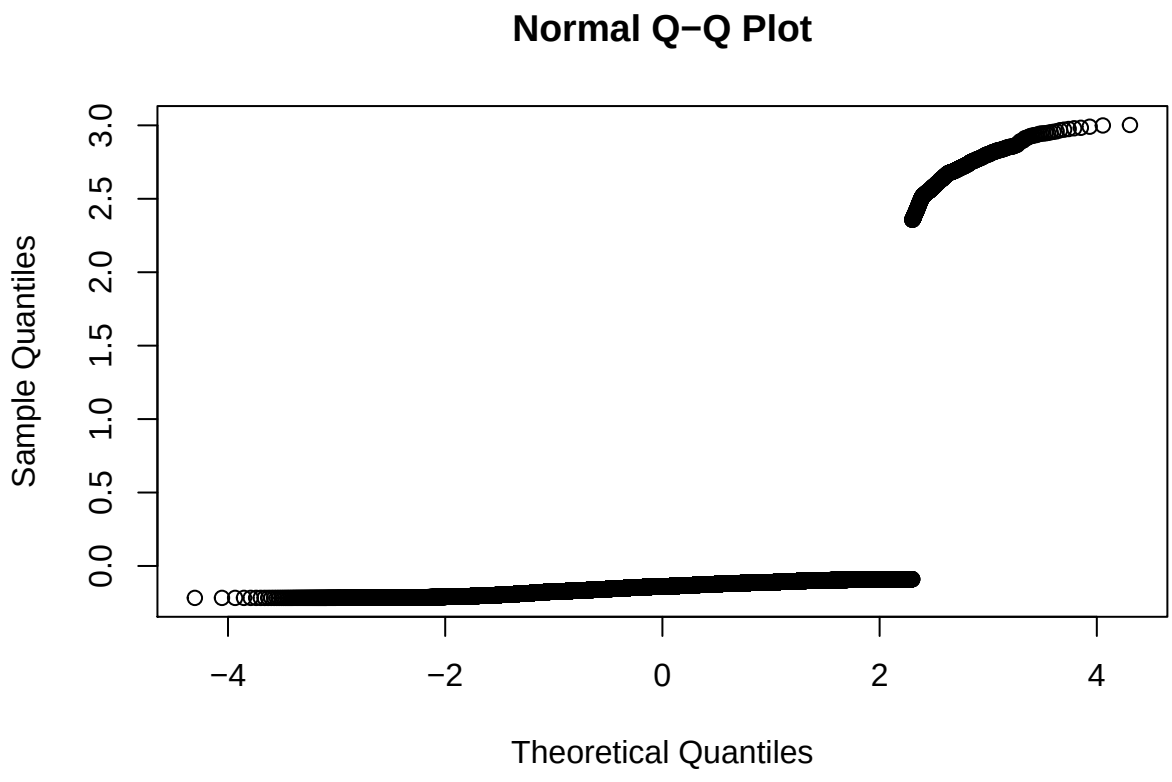
As above, for $t = 0$ and no stretching of the muscle, we expect to observe $e^{-4.96796} = 0.006957327$ electrical impulses. Moreover, at $t = 0$, we expect this baseline rate to increase by a factor of $e^{0.08149} = 1.084902$ electrical impulses for a unit increase in Stretch Magnitude. Lastly, under no stretching, we expect to see a decrease in the number of impulses by a factor of $e^{-0.94850} = 0.3873216$ for every passing millisecond.

We know evaluate the fit of this model.

```
#Chisquare test for sigma^2
sigma2 <- sum(residuals(model,type="pearson")^2)/(model$df.resid)
1-pchisq(sigma2*model$df.residual,model$df.residual)
```

```
## [1] 0.9998137
```

```
#check our residuals
qqnorm(residuals(model,type="deviance"))
```



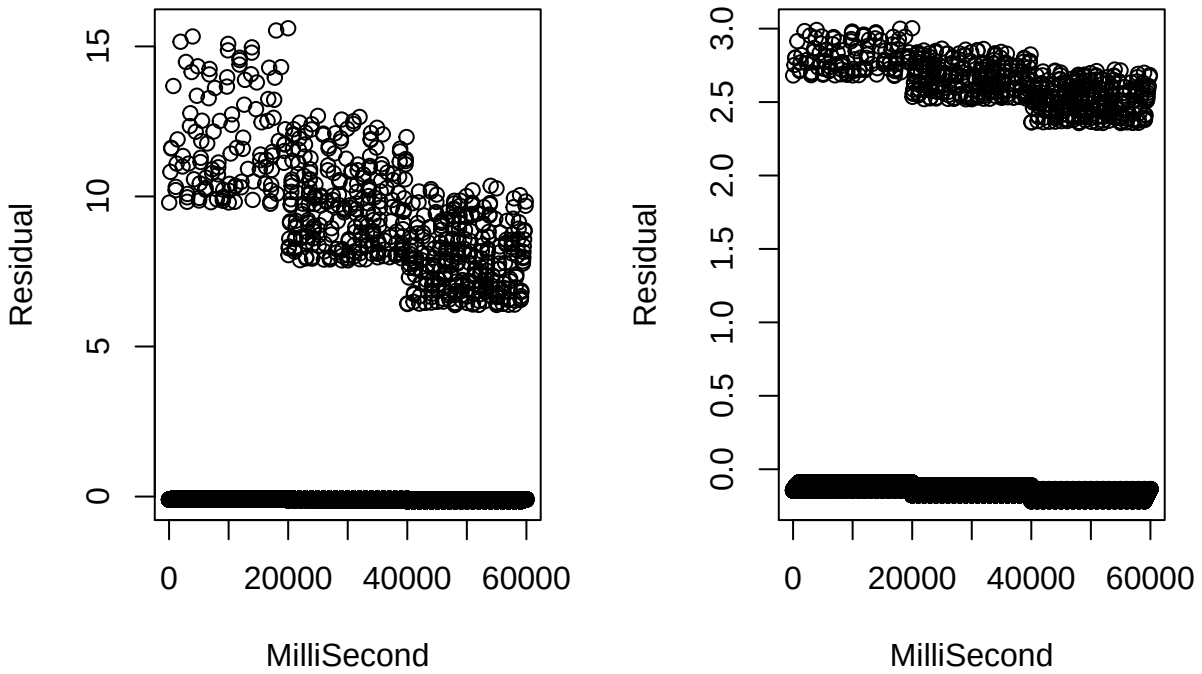

```
anova(model,test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model: poisson, link: log
##
## Response: Count
```

```
##
## Terms added sequentially (first to last)
##
##
##                   Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                                59999     5833.2
## StretchMagnitude  1   68.329     59998     5764.9 < 2.2e-16 ***
## msBin             1   47.147     59997     5717.8 6.586e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
dev.res = resid(model,type='pearson')
pear.res = resid(model,type='deviance')

par(mfrow = c(1,2))
plot(dev.res,xlab = "MilliSecond", ylab = "Residual")
plot(pear.res,xlab = "MilliSecond", ylab = "Residual")
```



While we do not have statistical evidence to suggest that we have underdispersion, the residuals clearly show that this model is overfit. The residuals that are greater than zero correspond to a count greater than one. But as there are so many zero counts, the model's optimal solution is to simply estimate ~0 for all points. With so many time covariates, this model is practically the saturated model which explains the small values of deviance.

(d)

To determine if we have a history dependency in this dataset, we will include previous counts as covariates to the GLM. In this way we can model any autocorrelation structure in the response. We will consider lags up to 10 milliseconds.

```r
n = 10
df2 = df[,-1]
df2 = cbind(df2, matrix(NA, nrow = 60000, ncol = n))
models = list()
take.aways = 60000:(60000-10)

for(i in 1:n){

  lag = c(rep(0,i), df$Count[-take.aways[1:i]])
  #take care of boundaries
  for(j in 1:i){
    lag[seq(j,60000, 1000)] = 0 # start of every trial
  }

  df2[,(3 + i)] = lag
  model = glm(Count ~., data = df2[,1:(3+i)], family = poisson)
  models[[i]] = model
}

model0 = glm(Count ~ StretchMagnitude + msBin, data = df, family = poisson)
anova(model0, models[[1]],models[[2]],models[[3]],models[[4]],test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: Count ~ StretchMagnitude + msBin
## Model 2: Count ~ StretchMagnitude + msBin + `1`
## Model 3: Count ~ StretchMagnitude + msBin + `1` + `2`
## Model 4: Count ~ StretchMagnitude + msBin + `1` + `2` + `3`
## Model 5: Count ~ StretchMagnitude + msBin + `1` + `2` + `3` + `4`
##   Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
## 1     59997     5717.8
## 2     59996     5701.4  1  16.3331 5.313e-05 ***
## 3     59995     5684.9  1  16.5341 4.778e-05 ***
## 4     59994     5680.3  1   4.5681   0.03257 *
## 5     59993     5680.3  1   0.0149   0.90289
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
summary(models[[3]])
```

```
##
## Call:
## glm(formula = Count ~ ., family = poisson, data = df2[, 1:(3 +
##     i)])
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.2232  -0.1652  -0.1398  -0.1169   3.1475
```

```
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)       -4.94734    0.13409 -36.897  < 2e-16 ***
## StretchMagnitude   0.08371    0.01005   8.327  < 2e-16 ***
## msBin             -0.97651    0.13968  -6.991 2.73e-12 ***
## `1`              -13.94924  223.43930  -0.062   0.9502
## `2`              -13.94869  223.61228  -0.062   0.9503
## `3`               -1.02928    0.57887  -1.778   0.0754 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 5833.2  on 59999  degrees of freedom
## Residual deviance: 5680.3  on 59994  degrees of freedom
## AIC: 6978.3
##
## Number of Fisher Scoring iterations: 16
```

Using an analysis of deviance table, we see that including lags up to order 3 play a significant role in predicting the current time's electrical implus count. Again, however, by including the time variable these models in essence are approaching the saturated model so any assessment of term significance is inappropriate in this setting.

For interpretation, the Intercept, Stretch Magnitude and msBin covariates do not change interpretation from part c. We even see that several of the estimated coefficients are similar. For the lag covariates, we can interpret these values as follows; given that we observe an electrical impulses at time $t-1$, we expect to see the baseline rate of impulses to reduce by a factor of $e^{-13.94924} \approx 0$. That if we see an impulse at time $t-1$, we expect zero impulses at this time. The same interpretation holds for lag $t-2$ with a reduction $e^{-13.94869} \approx 0$. Note for the $t-3$ lag however, we only see a reduction of $e^{-0.3572641} = 0.3572641$ from the baseline rate. With these estimates, one reasonable interpretation electrical impulses can only occur in $3ms$ time frames.

As discussed above, these models are practically the saturated models given the time variable $msBins$. Therefore, we model more variance than we anticipate given the number of covariates in the model which corresponds to underdispersion. This can be seen by the deviance 5680.3 on 59994 degrees of freedom.