Lab2Block2Group10A

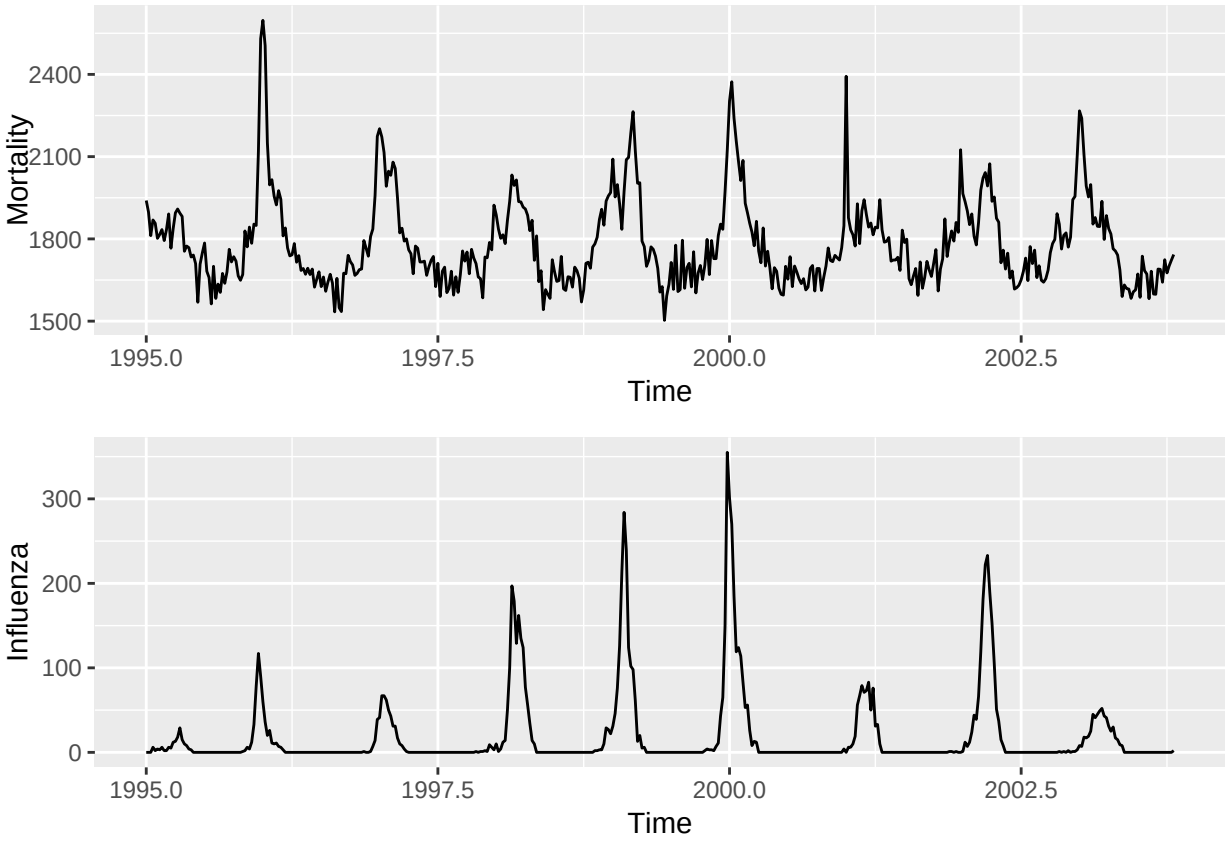
Jasleen Mann, Maria Treesa Sebastian, Prudhvi Peddmallu

12/17/2018

Assignment 1

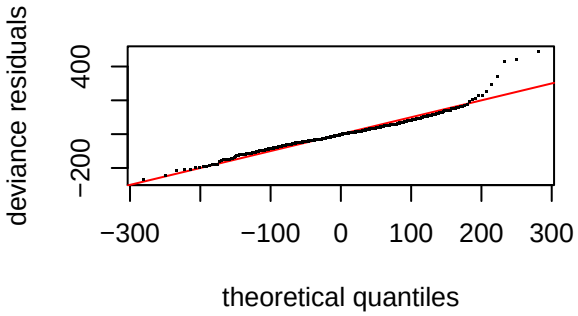
1. Time series plots

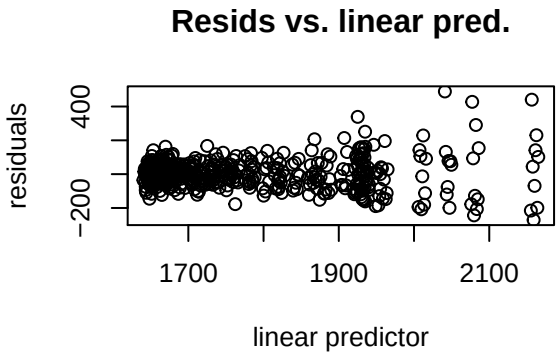
Whenever there is increase in influenza cases there is increase in mortality till 2000. After that influenza seems to be having lesser impact on mortality.

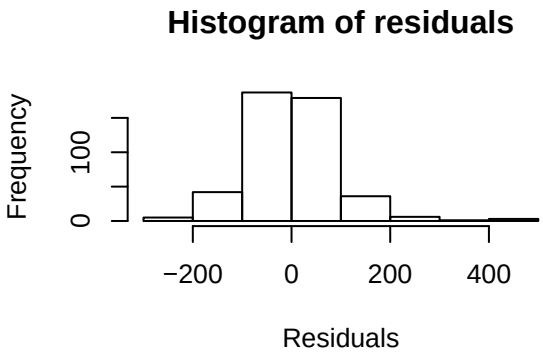


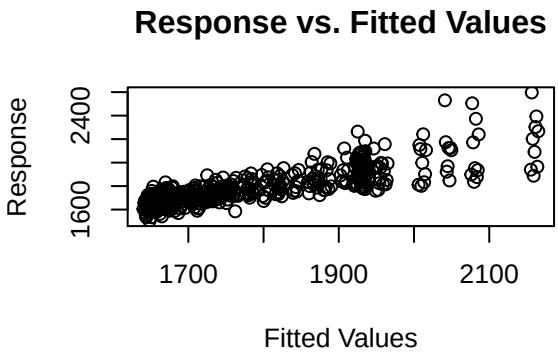
2. Underlying probabilistic model

```
res = gam(Mortality ~ Year +
          s(Week, k=length(unique(influenza$Week))),
          method="GCV.Cp", data=influenza)
gam.check(res)
```





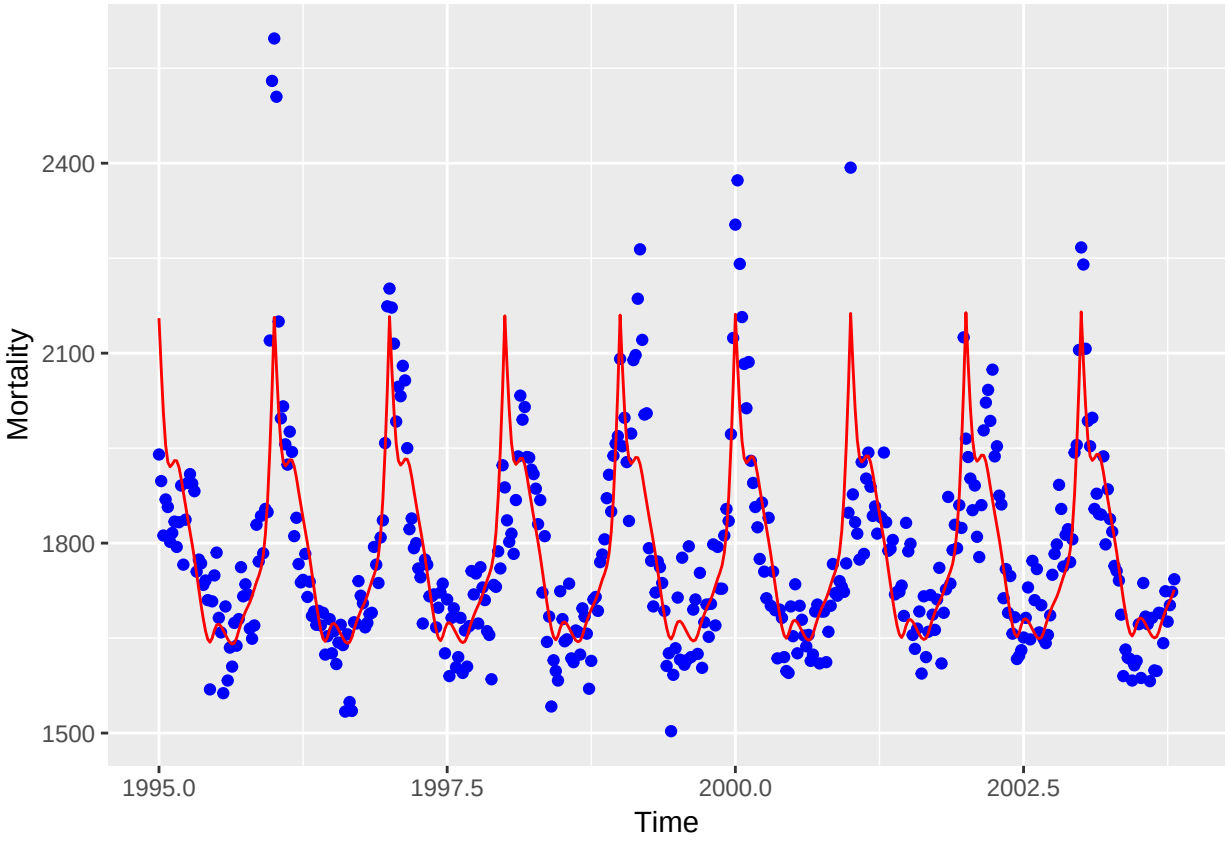





```
##
## Method: GCV   Optimizer: magic
## Smoothing parameter selection converged after 9 iterations.
## The RMS GCV score gradient at convergence was 0.00106719 .
## The Hessian was positive definite.
## Model rank =  52 / 53
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##           k'  edf k-index p-value
## s(Week) 51.0 14.3    1.09    0.95
```

From the plots above we can infer that underlying probabilistic model is normal. The residuals are normally distributed except for an outlier towards the left, still we can assume that it is normal distribution.

3. Predicted and Observed Mortality against Time

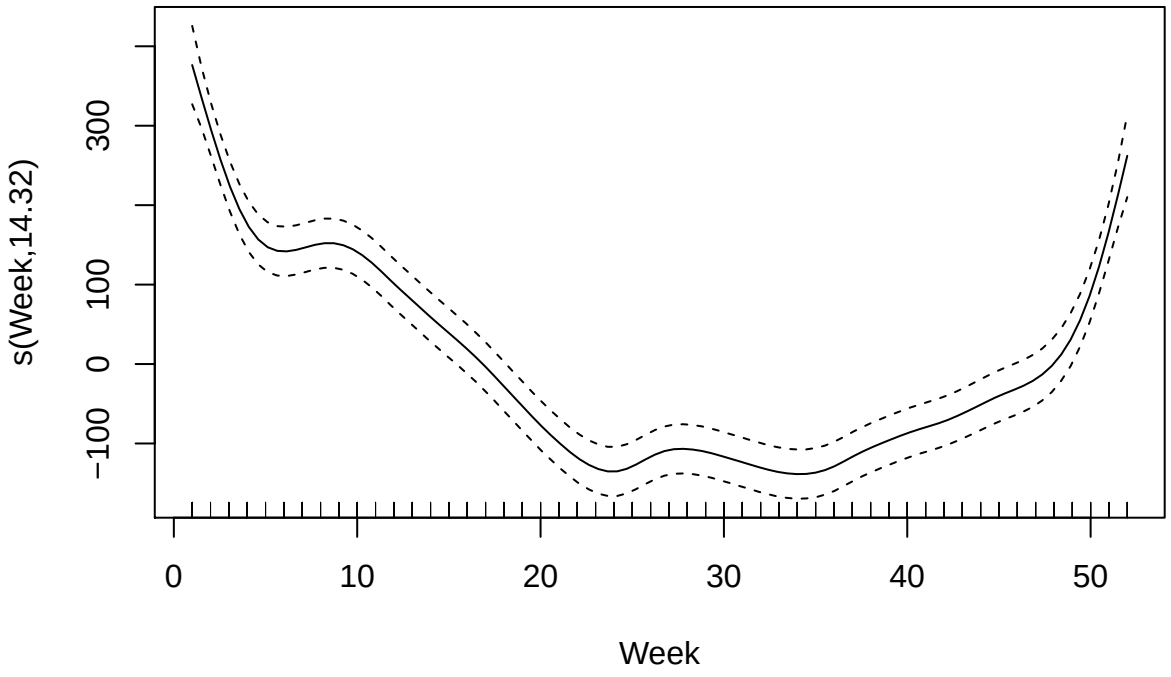



```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## Mortality ~ Year + s(Week, k = length(unique(influenza$Week)))
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -680.598   3367.760  -0.202    0.840
## Year           1.233      1.685   0.732    0.465
##
## Approximate significance of smooth terms:
##           edf Ref.df     F p-value
## s(Week) 14.32  17.87 53.86  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Rank: 52/53
## R-sq.(adj) =  0.677   Deviance explained = 68.8%
## GCV = 8708.6  Scale est. = 8398.9    n = 459
```

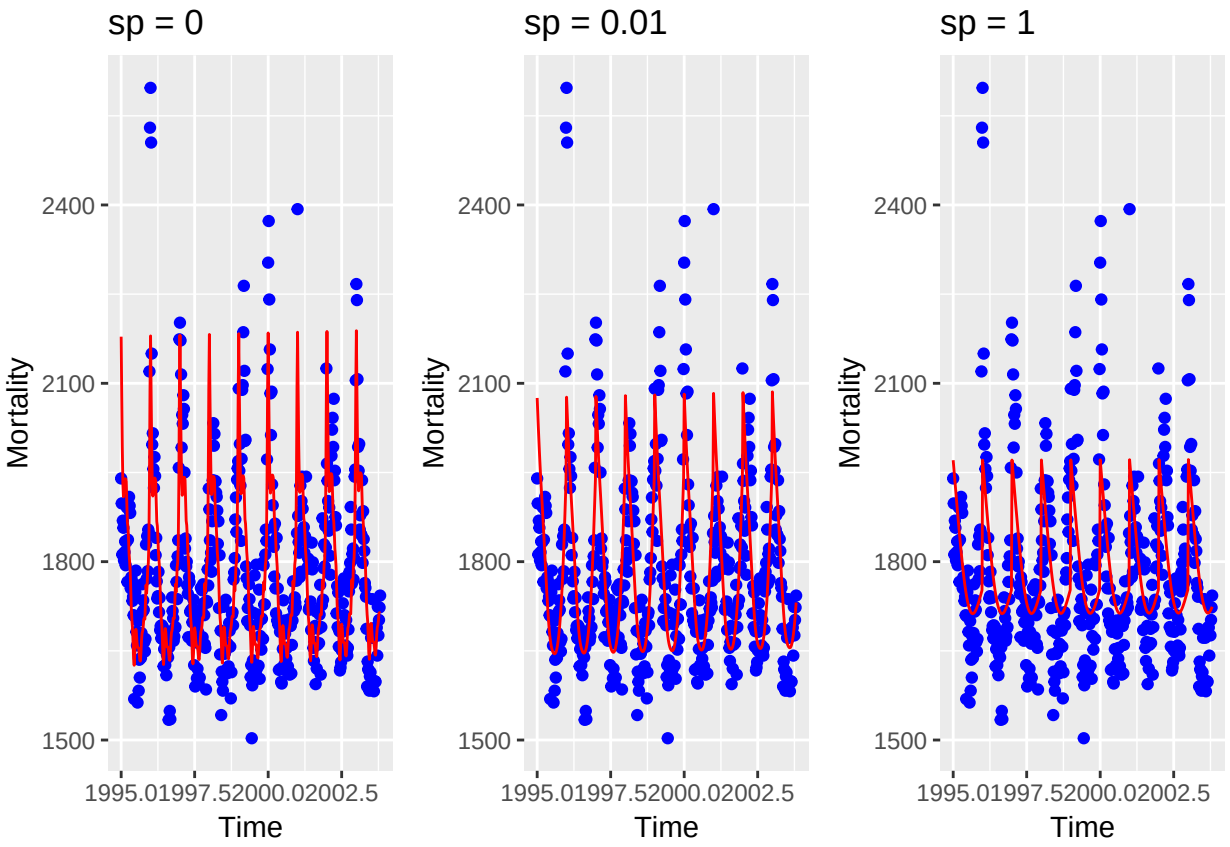
Significant terms in model - According to the p values in the summary, Spline function of week is the most sognificant term.

Trend in mortality change from one year to another - It seems to be following a cyclic trend and seems to have decreased with the years.

Plot of spline component



4. Penalty factor of the spline

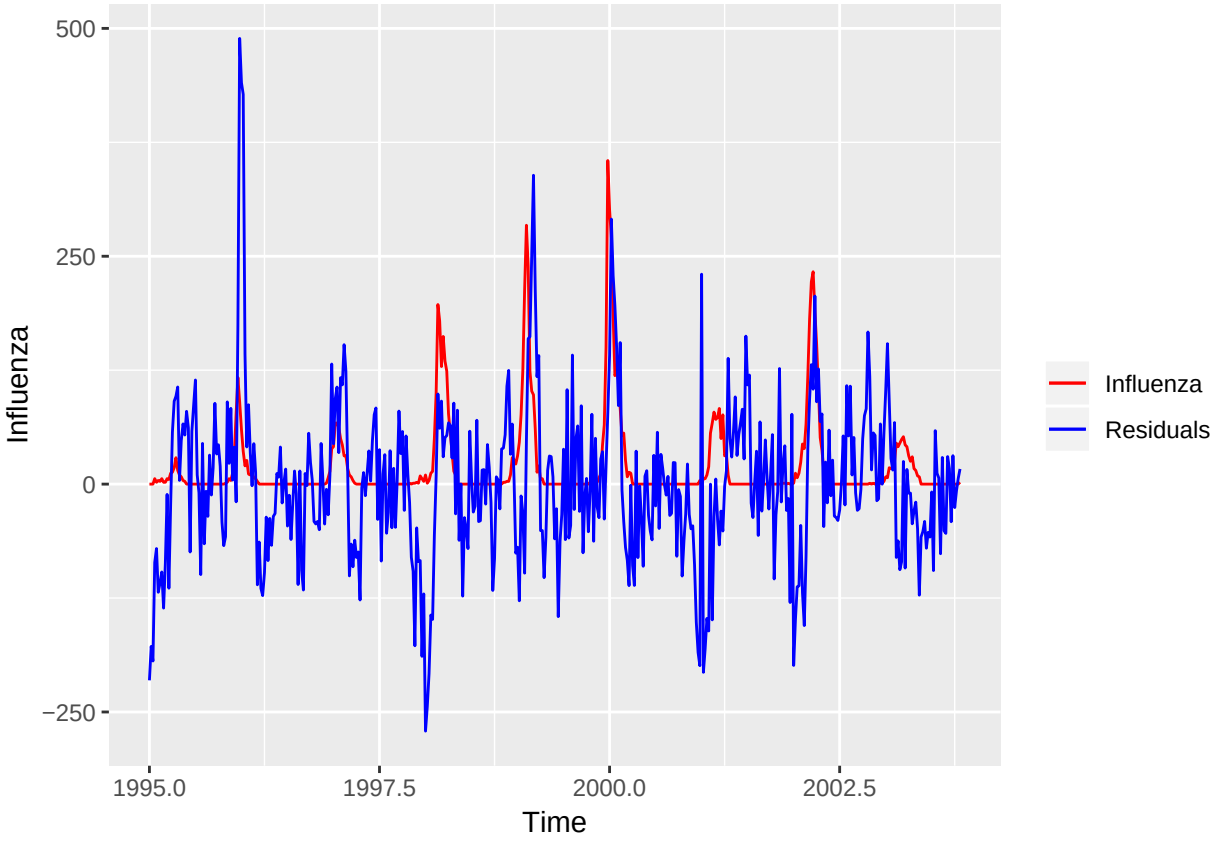



```
##       sp       edf
## 1   0.00 29.000000
## 2   0.01  6.689771
## 3   1.00  3.491752
## 4   1.50  3.365887
## 5   2.00  3.291922
## 6   3.00  3.208210
## 7   6.00  3.112130
## 8   9.00  3.076759
## 9 100.00  3.007268
```

With increase in Penalty factor the degrees of freedom should decrease. Our results also confirm this relationship. Initially the decline is steep.

5. Residuals and Influenza against Time

Pattern in residuals is related to the influenza outbreak. Rise in influenza cases seems to be causing steep rise in residuals. In case of 0 influenza cases the residuals are decreasing.

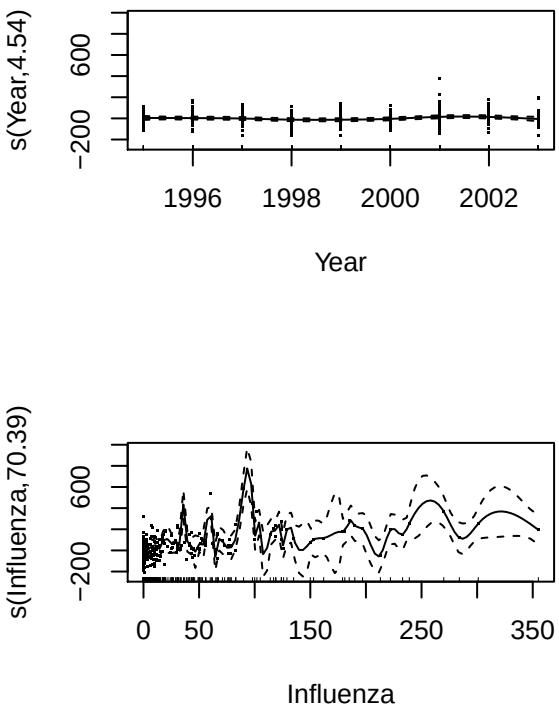


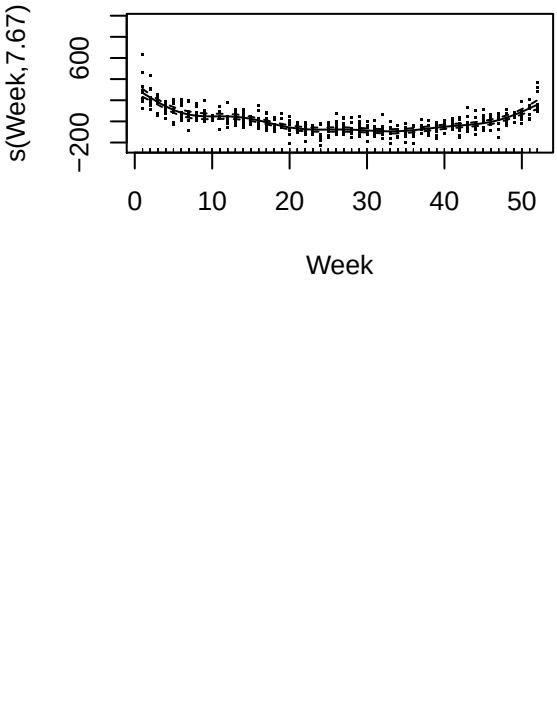
6. GAM with spline functions of year, week and influenza cases

From the p values, we can conclude that Influenza cases have high impact on mortality.

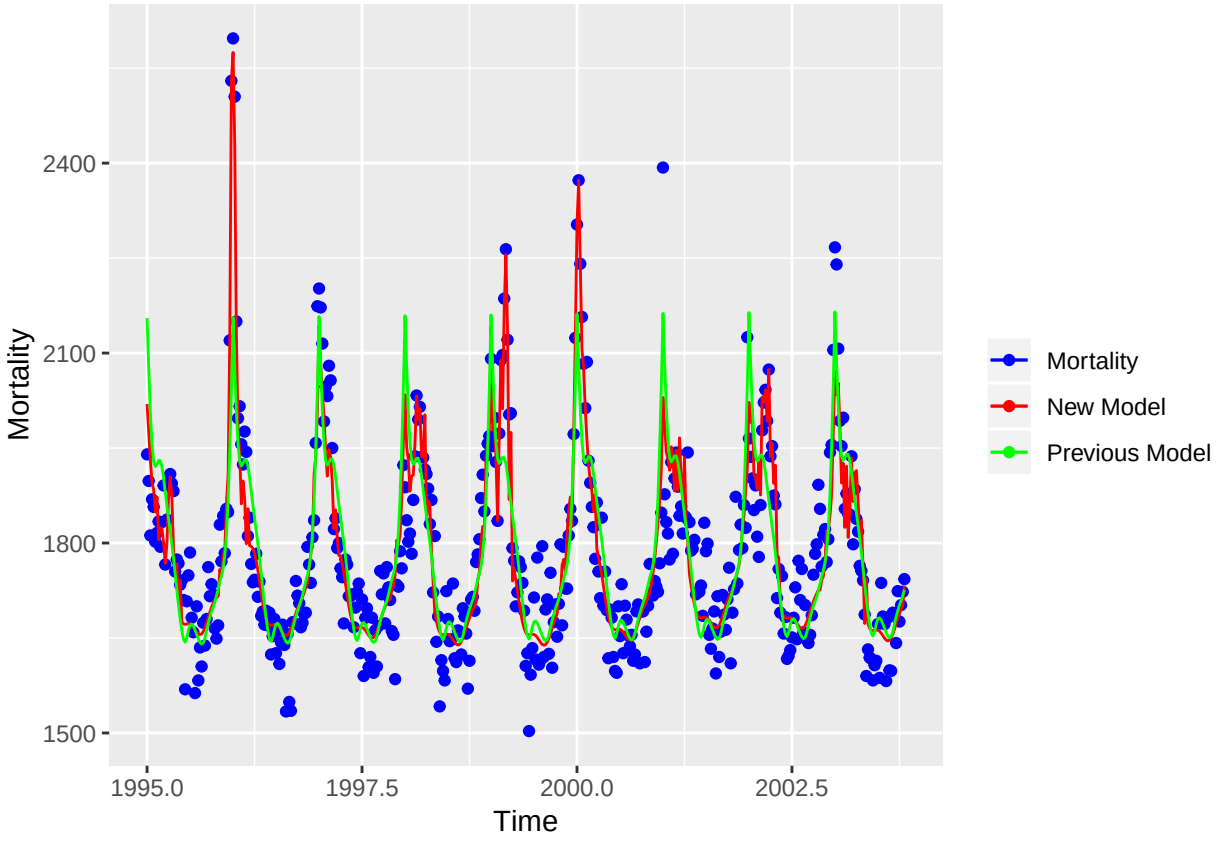
```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## Mortality ~ s(Year, k = length(unique(influenza$Year))) + s(Week,
##     k = length(unique(influenza$Year))) + s(Influenza, k = length(unique(influenza$Influenza)))
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1783.765      3.274   544.8   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Approximate significance of smooth terms:
##                  edf Ref.df      F p-value
## s(Year)        4.538  5.538  1.501   0.177
## s(Week)        7.668  7.961 38.437  <2e-16 ***
## s(Influenza) 70.389 73.772  5.594  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Rank: 92/101
## R-sq.(adj) =  0.811   Deviance explained = 84.5%
## GCV = 6016.5  Scale est. = 4920.6    n = 459
```



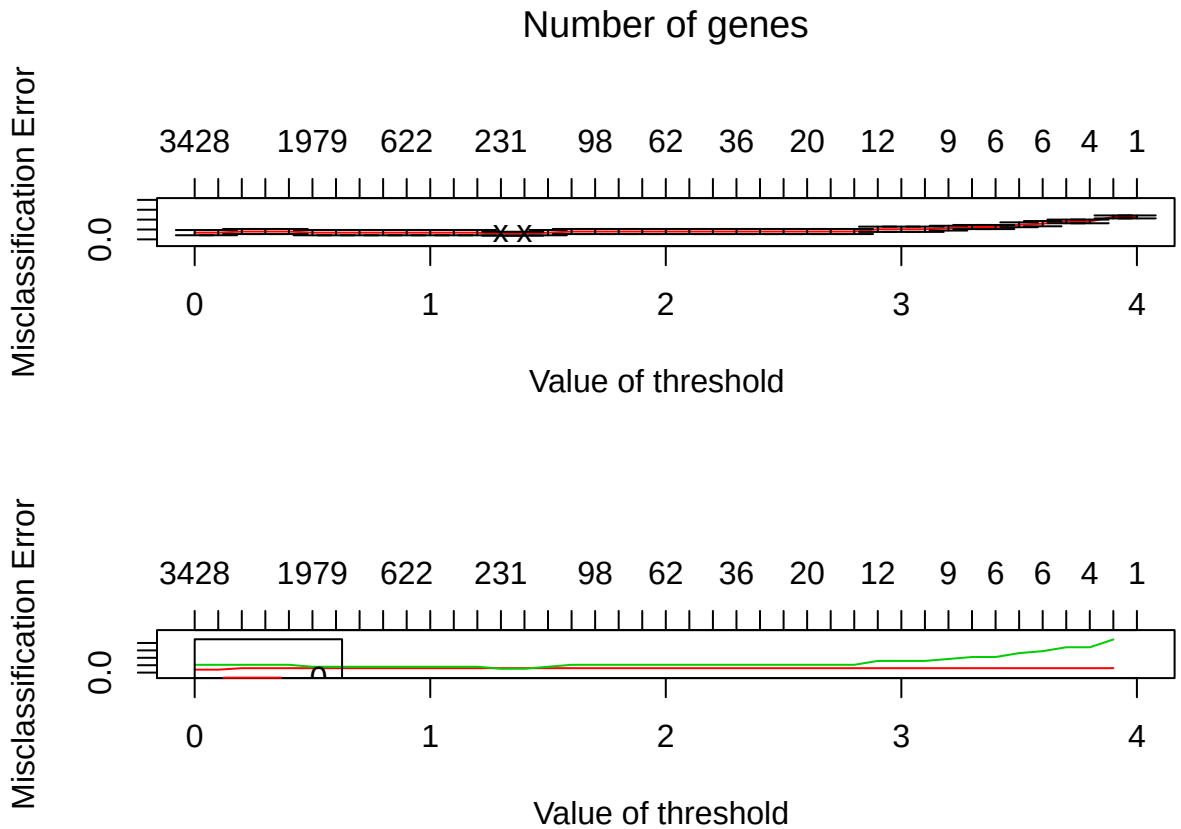


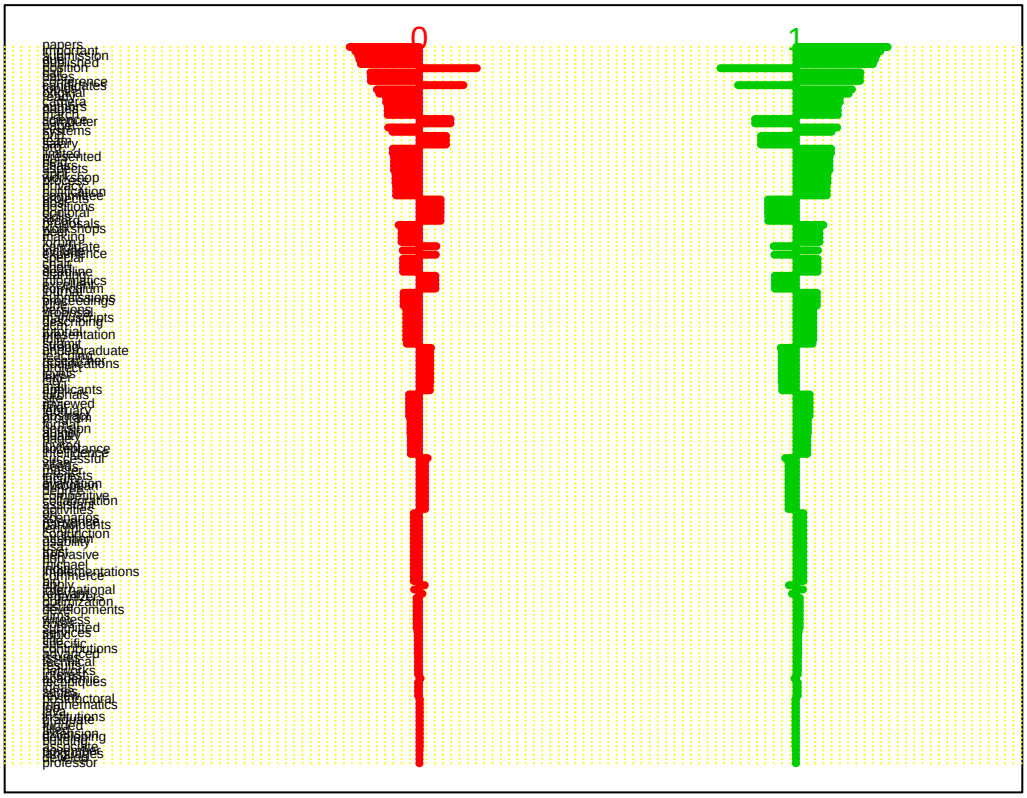
New model is better than previous model.



Assignment 2

1. Nearest shrunken centroid classification





```
## papers
## important
## submission
## due
## published
## position
## call
## conference
## dates
## candidates
```

From plot of cross validation model, which shows the depandance of misclassification error vs the threshold value, we can see that the misclassification error is least when the threshold is 1.3 and 1.4. So for this analysis, threshold value of 1.4 is chosen.

Nearest centroid method computes a standardized centroid for each class. The nearest shrunken centroid classification "shrinks" the class centroids towards zero by threshold, setting it equal to zero if it hits zero. After the shrinkage, the new sample is classifies using the centroid rule. So this will make the classifier more accurate by reducing the effecte of unimportant features so feature selection happens here.

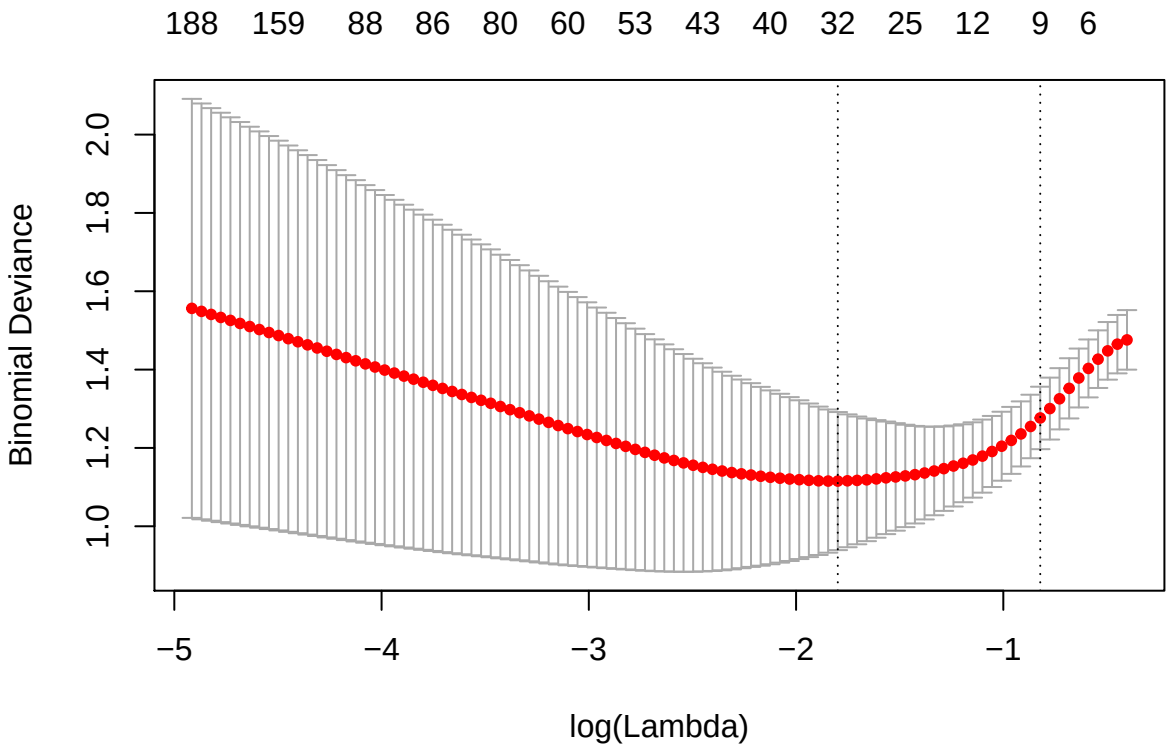
Here the centroid plot shows the shrunken centroids for each of the variables which has non zero differance to the shrunken centroids of two classes.

The method selected 231 features. The 10 most contributing features are: **paper, important, submissions, due, published, position, call, conferance, dates, candidates**

All the names and features mentioned above are significant in classification of emails as conferance emails or not.

Test Error : 0.1 - 10% test error and 90% accuracy.

2.



```
##  Setting default kernel parameters
```

1. **Elastic model** * Number of Contributing features - 40 * Test Error - 0.1

1. **SVM** * Number of Contributing features - 41 * Test Error -.05

Table 1: Model Comparison

Model	Features	Error
Nearest Shrunken Centroid Model	231	0.10
ElasticNet Model	40	0.10
SVM Model	41	0.05

On comparing all three models, Nearest Shrunken Centeroid model and Elastic model has same rate of test error which is 0.1 (10%) where centeroid model takes 231 parameters as contributing parameters whereas Elastic net model makes use of less number of features to do the prediction as the contributing features.

In case of SVM model, it uses 41 features as contributing features and the error rate is 0.05 or 5% which is less than the above two models and it uses 41 features as contributing features which is not significantly differant compared to the elastic model. So based on the above two factors, SVM model is better for classification of the data because of the accuracy that the model provides with less number of features.

3.

We consideres null hypothesis as the feature is not significant for analysing if the email is a conferance email and alternate hopothesis: The feature is significant in classifying coferance email.

From the above analysis, we can see that the model selected 39 features as significant features by rejecting the null hypothesis(the features with p values less than 0.05).

The features corresponding to rejected hypothesis are:

```
##  [1] apply         authors       call          camera        candidate     candidates    chairs
## [36] submission    team          topics        workshop
## 4702 Levels: a4soa aaai aachen aalborg aamas aarhus aaron aau abbadi abdalla abdallah abductive abil
```

The above features are the significant features as selected by the model. From the above feature list, the words team, important, topics, presented, proceedings, helg, org, international, call, papers, phd, published etc seems very relevant in classification of a conferance email

APPENDIX

```
knitr::opts_chunk$set(echo = TRUE)
library(mgcv)
library(akima)
library(plotly)
library(readxl)
library(dplyr)
library(grid)
library(ggplot2)
influenza=read_excel("influenza.xlsx")

plot1 <- ggplot(influenza, aes(Time,Mortality)) + geom_line() + xlab("Time") + ylab("Mortality")
plot2 <- ggplot(influenza, aes(Time,Influenza)) + geom_line() + xlab("Time") + ylab("Influenza")
```

```
grid.newpage()
grid.draw(rbind(ggplotGrob(plot1), ggplotGrob(plot2), size = "last"))
res = gam(Mortality ~ Year +
          s(Week, k=length(unique(influenza$Week))),
          method="GCV.Cp", data=influenza)
gam.check(res)
predicted <- predict(res, newdata = influenza, type='response')
ggplot(data =influenza) +
geom_point(aes(Time,Mortality), colour = "Blue") +
geom_line(aes(Time,predicted), colour = "Red")

summary(res)
plot(res)
res2 = gam(Mortality ~ Year +
          s(Week, k=length(unique(influenza$Week)), sp=0),
          method="GCV.Cp", data=influenza)

predicted2 <- predict(res2, newdata = influenza, type='response')
p1 <- ggplot(data =influenza) +
  geom_point(aes(Time,Mortality),colour = "Blue") +
  geom_line(aes(Time,predicted2),colour = "Red") +
  ggtitle("sp = 0")

res3 = gam(Mortality ~ Year +
          s(Week, k=length(unique(influenza$Week)), sp=0.01),
          method="GCV.Cp", data=influenza)

predicted3 <- predict(res3, newdata = influenza, type='response')
p2 <- ggplot(data =influenza) +
  geom_point(aes(Time,Mortality),colour = "Blue") +
  geom_line(aes(Time,predicted3),colour = "Red") +
  ggtitle("sp = 0.01")

res4 = gam(Mortality ~ Year +
          s(Week, k=length(unique(influenza$Week)), sp=1),
          method="GCV.Cp", data=influenza)
predicted4 <- predict(res4, newdata = influenza, type='response')
p3 <- ggplot(data =influenza) +
  geom_point(aes(Time,Mortality),colour = "Blue") +
  geom_line(aes(Time,predicted4),colour = "Red") +
  ggtitle("sp = 1")

grid.newpage()
grid.draw(cbind(ggplotGrob(p1), ggplotGrob(p2), ggplotGrob(p3), size = "last"))

#anova(res2)
#some more models with different sp values
res5 = gam(Mortality ~ Year +
          s(Week, k=length(unique(influenza$Week)), sp=1.5),
          method="GCV.Cp", data=influenza)

res6 = gam(Mortality ~ Year +
          s(Week, k=length(unique(influenza$Week)), sp=2),
```

```
          method="GCV.Cp", data=influenza)

res7 = gam(Mortality ~ Year +
          s(Week, k=length(unique(influenza$Week)), sp=3),
          method="GCV.Cp", data=influenza)

res8 = gam(Mortality ~ Year +
          s(Week, k=length(unique(influenza$Week)), sp=6),
          method="GCV.Cp", data=influenza)


res9 = gam(Mortality ~ Year +
          s(Week, k=length(unique(influenza$Week)), sp=9),
          method="GCV.Cp", data=influenza)


res10 = gam(Mortality ~ Year +
          s(Week, k=length(unique(influenza$Week)), sp=100),
          method="GCV.Cp", data=influenza)

#edf vs sp
x <- c(0, 0.01, 1, 1.5, 2, 3, 6, 9, 100)
y <- c(sum(res2$edf), sum(res3$edf), sum(res4$edf), sum(res5$edf), sum(res6$edf),
       sum(res7$edf), sum(res8$edf), sum(res9$edf), sum(res10$edf))
data.frame(cbind(sp=x,edf=y) )
ggplot(data = influenza, aes(x = Time)) +
  geom_line(aes(y = Influenza,colour = "Influenza")) +
  geom_line(aes(y = res$residuals,colour = "Residuals")) +
  scale_colour_manual("", breaks = c("Influenza", "Residuals"),
                      values = c("Red", "Blue"))
new_res=gam(Mortality~s(Year, k=length(unique(influenza$Year)))+s(Week, k=length(unique(influenza$Year)))
summary(new_res)
plot(new_res, residuals=TRUE, page=1)
new_predicted <- predict(new_res, newdata = influenza, type='response')
ggplot(data = influenza, aes(x = Time)) +
  geom_point(aes(y = Mortality, colour = "Mortality")) +
  geom_line(aes(y = new_predicted,colour = "New Model")) +
  geom_line(aes(y = predicted,colour = "Previous Model")) +
  scale_colour_manual("", breaks = c("Mortality","New Model", "Previous Model"),
                      values = c("Blue", "Red", "Green"))
library(pamr)
library(glmnet)
library(dplyr)
library(kernlab)
# dividing data into train and test set
data <- read.csv(file = "data.csv", sep = ";", header = TRUE, fileEncoding = "Latin1")
data$Conference=as.factor(data$Conference)
rownames(data)=1:nrow(data)
n=dim(data)[1]
set.seed(12345)
id=sample(1:n, floor(n*0.7))
train=data[id,]
test=data[-id,]
```

```
#Perform  of training data in which the threshold is chosen
# by cross-validation
x = t(train[,-4703])
y = train[[4703]]
x_test = t(test[,-4703])
y_test = test[[4703]]
mydata=list(x=x,y=as.factor(y),geneid=as.character(1:nrow(x)), genenames=rownames(x))
model = pamr.train(mydata,threshold=seq(0,4, 0.1))
cvmodel=pamr.cv(model,mydata)
pamr.plotcv(cvmodel)
# from cv model,when the threshold is 1.3 and 1.4, the error is least. Hence selecting the threshold as

#centroid plot
pamr.plotcen(model, mydata, threshold=1.4)
# The method selected 231 features
contri_genes=pamr.listgenes(model,mydata,threshold=1.4)
# 10 most contributing features
imp = as.numeric(contri_genes[1:10,1])
cat( paste( colnames(data[imp]), collapse='\n' ) )
#test error
pred_test <- pamr.predict(model, newx = x_test, threshold=1.4)
misclass_table <- table(y_test,pred_test)
test_error <- 1 - sum(diag(misclass_table))/sum(misclass_table)
#Elastic net
x = train[,-4703]%>% as.matrix()
y = train[[4703]]
x_test = test[,-4703]%>% as.matrix()
y_test = test[[4703]]
cvfit <- cv.glmnet(x, y, family = "binomial", alpha = 0.5)
plot(cvfit)
predict_elastic <- predict.cv.glmnet(cvfit, newx = x_test, s="lambda.min",type = "class")
coeffs <- coef(cvfit,s="lambda.min")
e_variables = as.data.frame(coeffs[which(coeffs[,1]!=0),])
cmatrix_elastic <- table(y_test, predict_elastic)
testerror_elastic <- 1 -sum(diag(cmatrix_elastic))/sum(cmatrix_elastic)
#SVM
svm_model <- ksvm(x, y, type = 'C-svc',  kernel = "vanilladot", scale = F)
predict_svm <- predict(svm_model,newdata = x_test, type = "response")
mtable_svm <- table(y_test,predict_svm)
testerror_svm <- 1 -sum(diag(mtable_svm))/sum(mtable_svm)

#  table
table_result <- data.frame("Model" = c("Nearest Shrunken Centroid Model",
                           "ElasticNet Model", "SVM Model"), "Features" = c(231,40,41),
                           "Error" = c(test_error, testerror_elastic,testerror_svm ))

knitr::kable(table_result, caption = "Model Comparison")
pvalues <- c()
x <- data[,-4703]
#y <- as.vector(data[,4703])

for(i in 1:(ncol(data)-1)){
  t_res <- t.test(x[,i]~Conference, data = data)
```

```
  pvalues[i] <- t_res$p.value
}

p_adj <- p.adjust(pvalues, method = "BH", n = length(pvalues))
p_adj_df <- data.frame("feature" = colnames(data[,-4703]), "pvals" = p_adj)
p_adj_df <- p_adj_df[which(p_adj_df[,2] <= 0.05), ]
num_features <- nrow(p_adj_df)
p_adj_df[c(1:39),1]
```