Lab2Block2

Obaid, Sridhar, Naveen

12 December 2018

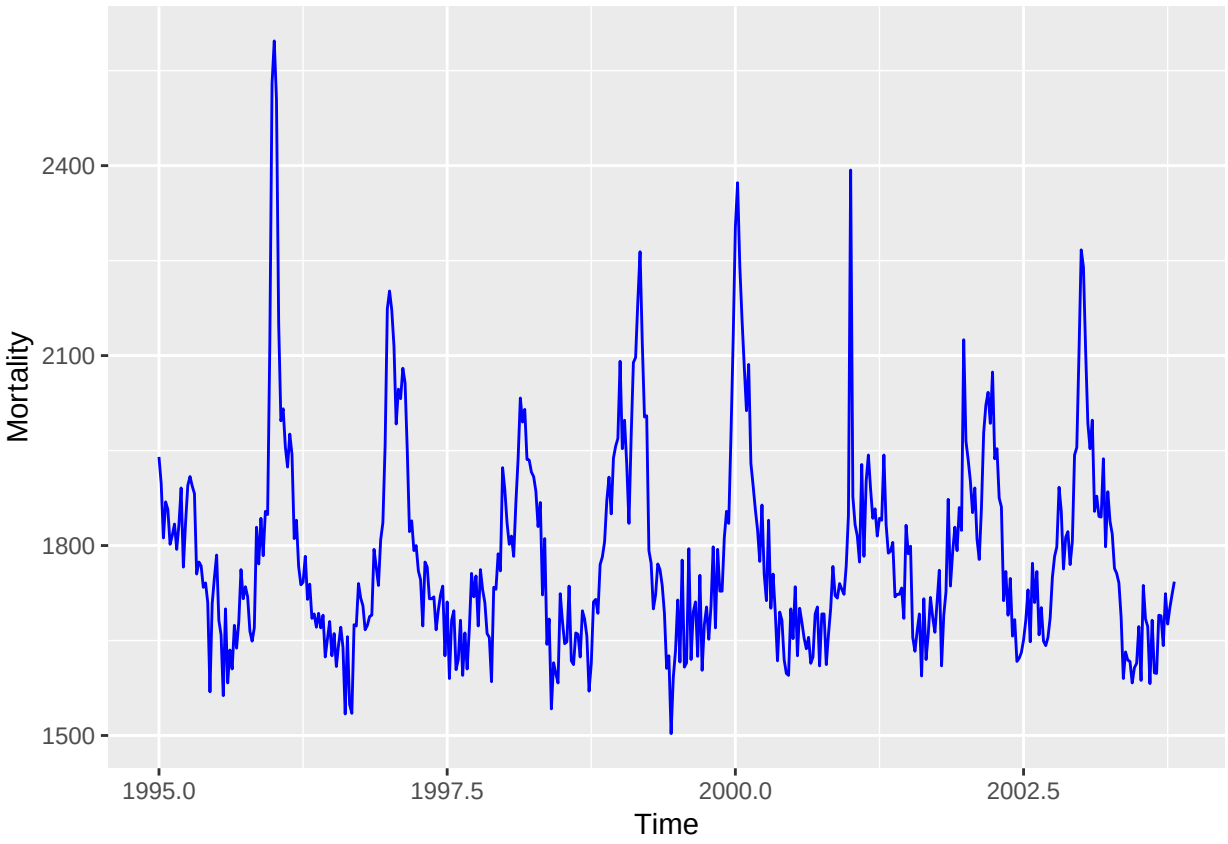
Contents

Assignment 1: Using GAM and GLM to examine the mortality rates . . . 2
- Part 1: Relation between Influenza and Mortality . . . 2
- Part 2: Fitting GAM Model . . . 3
- Part 3: Analysis on GAM Model . . . 4
  - A) Plot of original and predicted Mortality . . . 4
  - B) Significant Terms in the model . . . 4
  - C) Mortality Trend throughout the years . . . 5
  - D) Examining spline component . . . 6
- Part 4: Examining penalty factor of spline function . . . 6
  - A) How Penalty Factor influence the deviance . . . 6
  - B) Predicted and observed Mortality for cases of very high and very low penalty factor 6
  - C) Relation between penalty factor and deviance . . . 7
  - D) Relation between penalty factor and degrees of freedom . . . 8
- Part 5: Examining relation between Influenza and GAM Residuals . . . 10
- Part 6: GAM model in R in which mortality is be modelled as an additive function of the spline functions of year, week, and the number of confirmed Influenza cases. . . . 10
  - A) Examining the updated model . . . 10
  - B) Plot of predicted and observed Mortality using the updated model . . . 11
  - C) Influence of Week, Year and Influenza on Mortality . . . 11

Assignment 2: High-dimensional methods . . . 12
- Part 1: Nearest Shrunken Centroid Classification . . . 12
  - A) Fitting nearest shrunken centroid model using pamr package . . . 12
  - B) Feature selected . . . 14
  - C) Is it reasonable to that feature selected have strong effect on target? . . . 14
  - D) Test Error . . . 15
- Part 2: Computing test error using Elastic Net and SVM . . . 15
  - A) Elastic Net . . . 15
    - 1) Training Elastic Net using glmnet package . . . 15
    - 2) Test Elastic Net . . . 15
  - B) Support Vector Machine Using Kernel Vannilldot . . . 16
    - 1) Training support vector machine using kernlab package . . . 16
    - 2) Test SVM . . . 16
  - C) Comparison between all models . . . 16
- Part 3: Implementing Benjamini-Hochberg method . . . 16
- Appendix . . . 17

Assignment 1: Using GAM and GLM to examine the mortality rates

Part 1: Relation between Influenza and Mortality



The above plots clearly shows that the Influenza and Mortality are related as both have the peaks at the same time, though the scale is different. Every time there is a slight peaking in Influenza the Mortality also increases.

Part 2: Fitting GAM Model

```
##
## 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
```

Underlying Probablistic Model: $y = w_o + w_1 x_1 + s(x_2) + e$

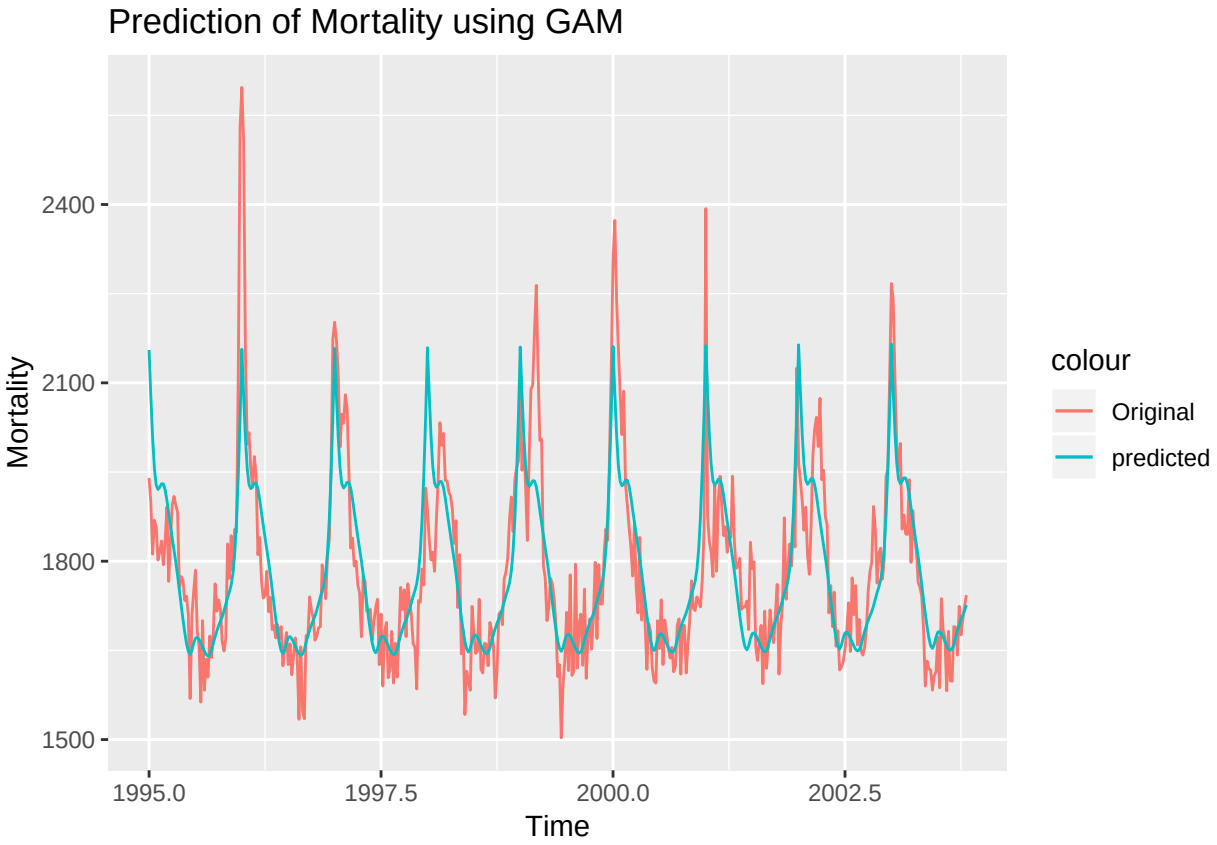
Mortality : $Mortality \sim N(\mu, \sigma^2)$

Epsilon : $\epsilon \sim N(0, \sigma^2)$

Probilistic model: $Mortality = -680.589 + 1.233 * Year + s(Week) + \epsilon$

Part 3: Analysis on GAM Model

A) Plot of original and predicted Mortality



The model is a good approximation of the data but it is too smooth for the data. Rate of change in mortality is very high and we need to fit a more complex model to be able to capture that. The model is not able to capture the trend in mortality at the start but later it goes on to capture almost all the highs and lows. There are many points in the plot ie Mortalities 1700 and higher than 2200 are not captured by the model effectively. This is not the best model for this data, if we increase the complexcity of the model by introducing more spline functions of variables that are significant in the calculation of Mortality we could get a better model.

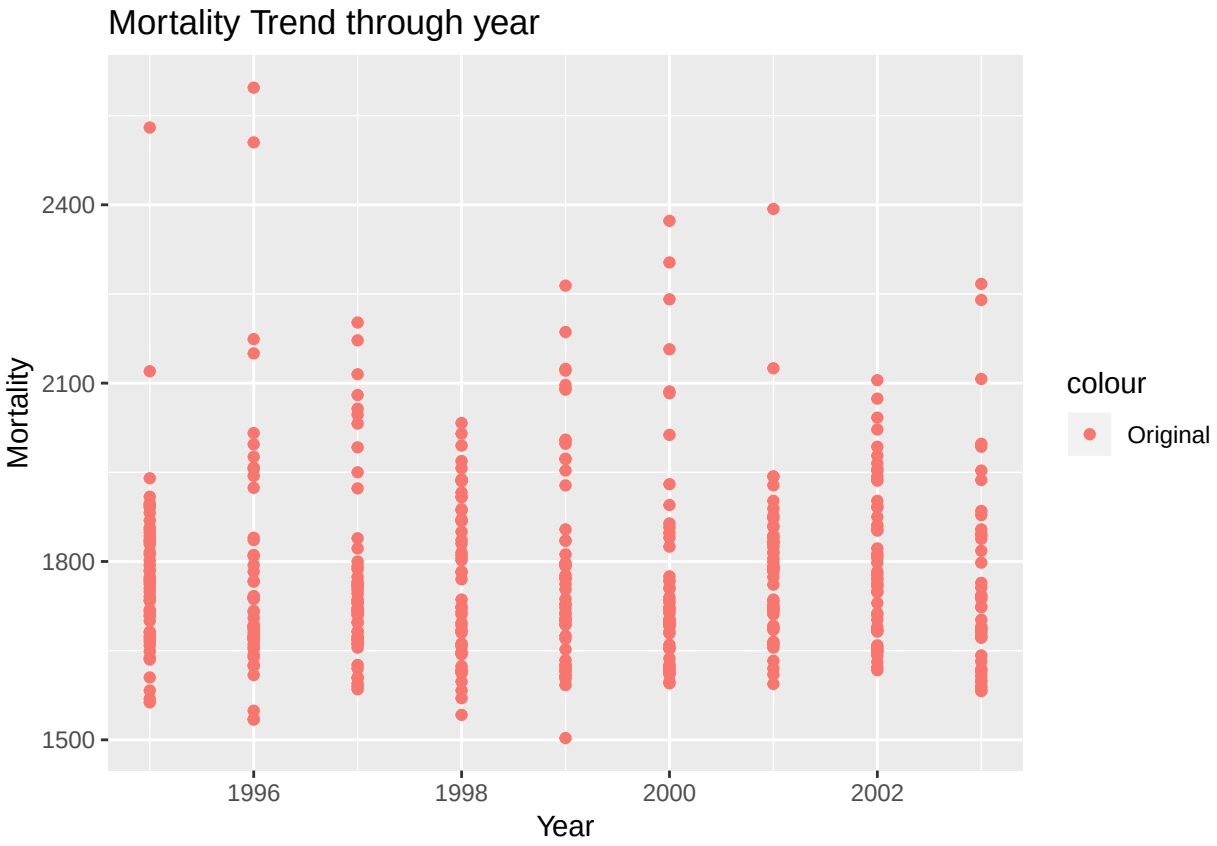
B) Significant Terms in the model

```
##
## 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
```

A small p-value indicates that it is unlikely we will observe a relationship between the predictor and response variables due to chance. Typically, a p-value of 5% or less is a good cut-off point. In our model, the p-values for week is close to zero but p-value of year is greater tha 0.05 which indicates the Year predictor as insignificant. This could be because in our model we asked GAM function to find a liner trend of mortality with year and we found that year is not linearly related to mortality when we ploted it. If we fit a spline function of Year may be we could get a better model and Year would be significant in it.

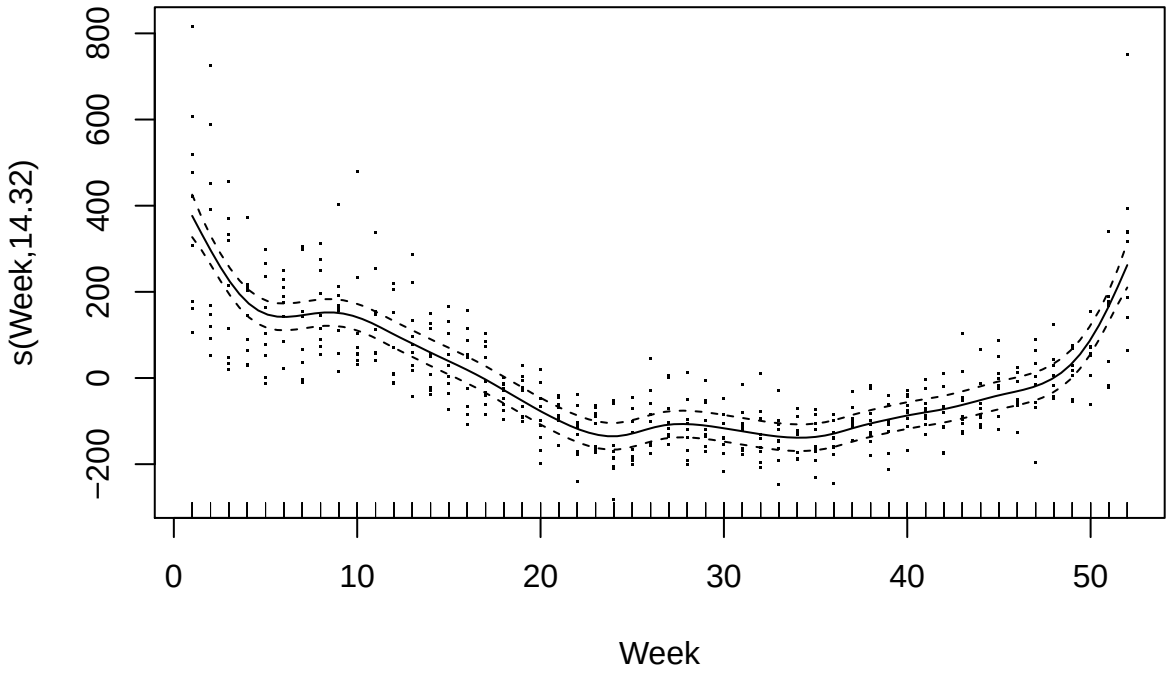
C) Mortality Trend throughout the years



This plot shows the spread of Mortality across each week of each year. Each year has 51 points in them correnponding to the mortality value in those weeks in that year. As seen from the above plot, the range

of Mortality values was low in initial years and then it kept increasing with the years. It had the largest range in the year 2000 and 2001. This trend cannot be captured by a linear realtion, this is the reason Year was marked as insignificant in the previous model. If we fit a spline function of Year also then the models performance might get better.

D) Examining spline component



This plot clearly shows that the rate of mortality is higher in the initial weeks of the year and then is low during the middle of the year and increases back again be the end of the year. We thing that this indicated the influenza rate increases during the winters due to the dry climate and this may be the reason for the highs during that period. It is high in the winters and goes down during summer.

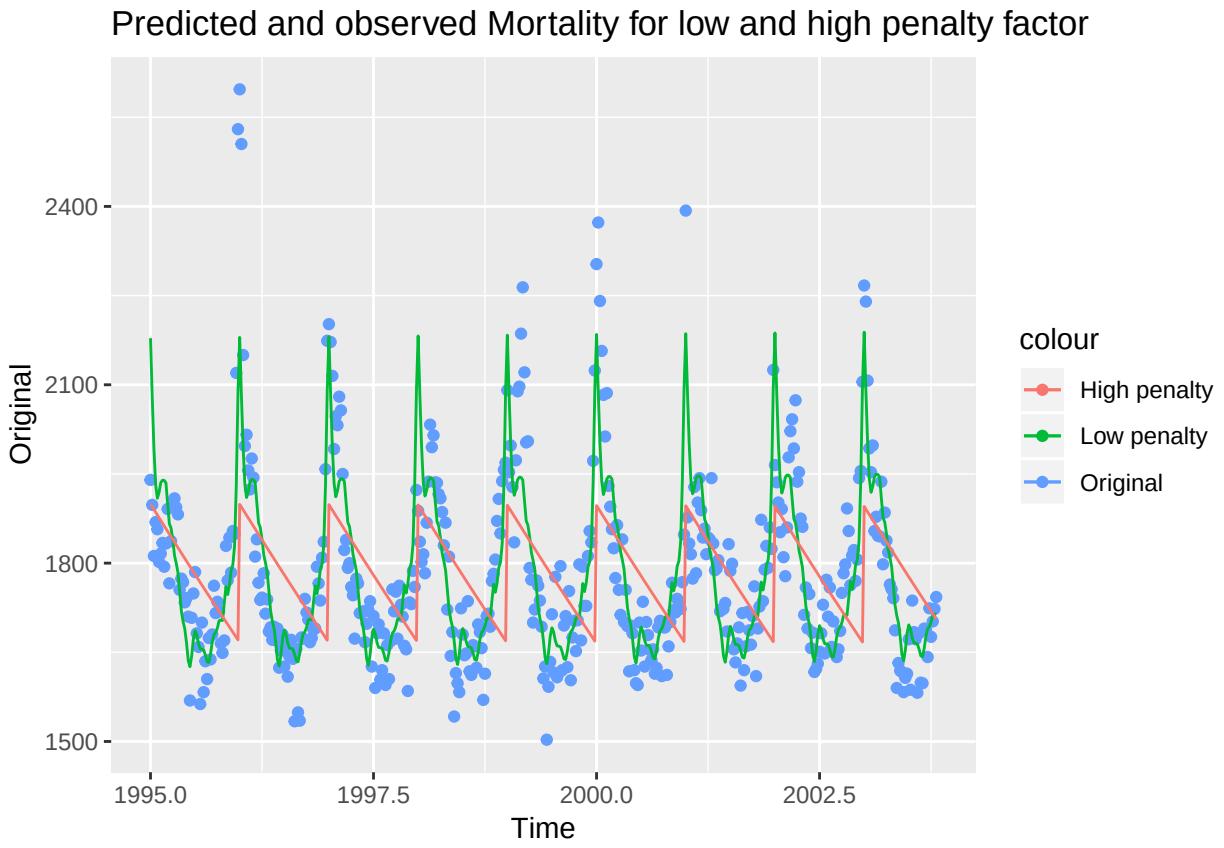
Part 4: Examining penalty factor of spline function

A) How Penalty Factor influence the deviance

```
##
## At Penalty Factor:  0.000113193  explained deviance is:  3718012
```

This is the optimal penalty factor selected by our previous model and the corresponding deviance value.

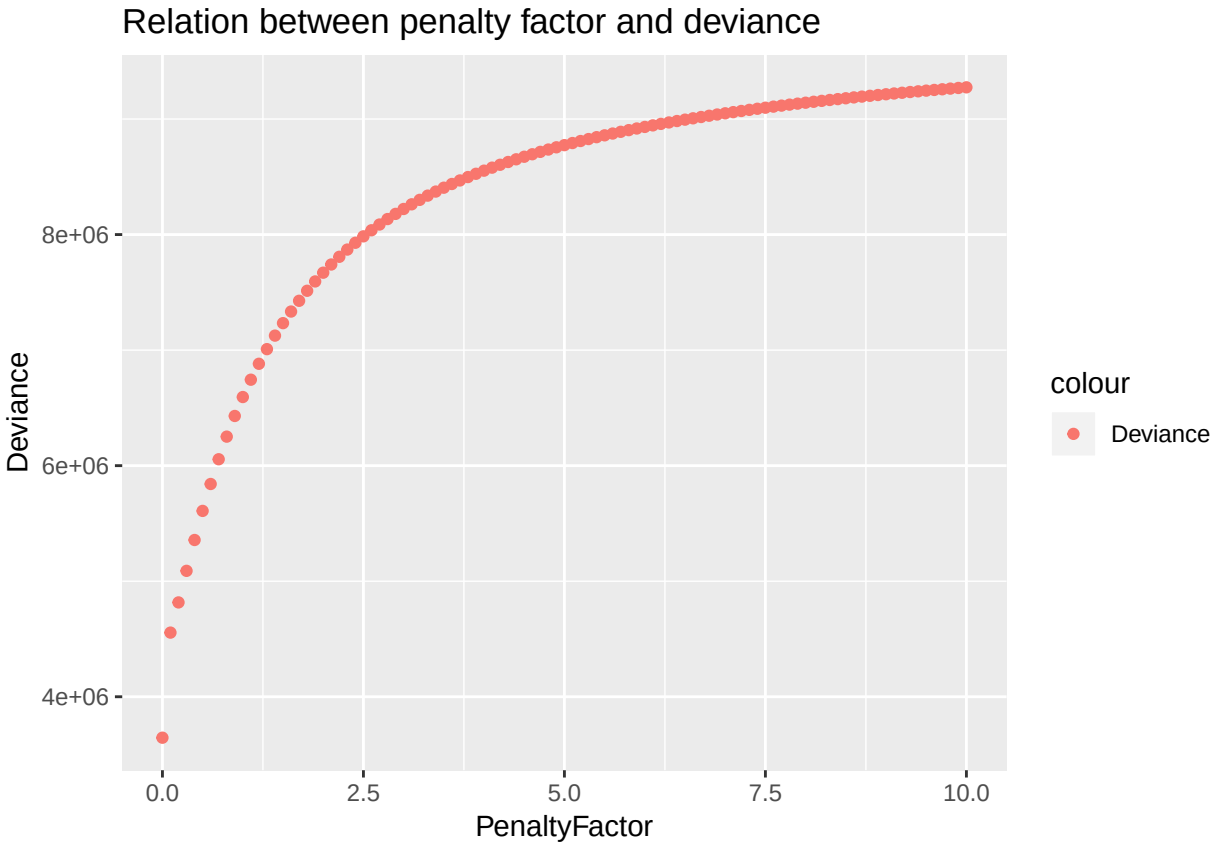
B) Predicted and observed Mortality for cases of very high and very low penalty factor



Passing in a very low value of penalty factor could result in a overfitted model. As we can see the green line fit coresponding to the Low penalty factor(0) is too trying too much to capture all the points. It is not much evident in this case as we had a relatively simple formula in GAM with just one spline function of week. No penalty factor or very low penalty factor could result in an overfitted model.

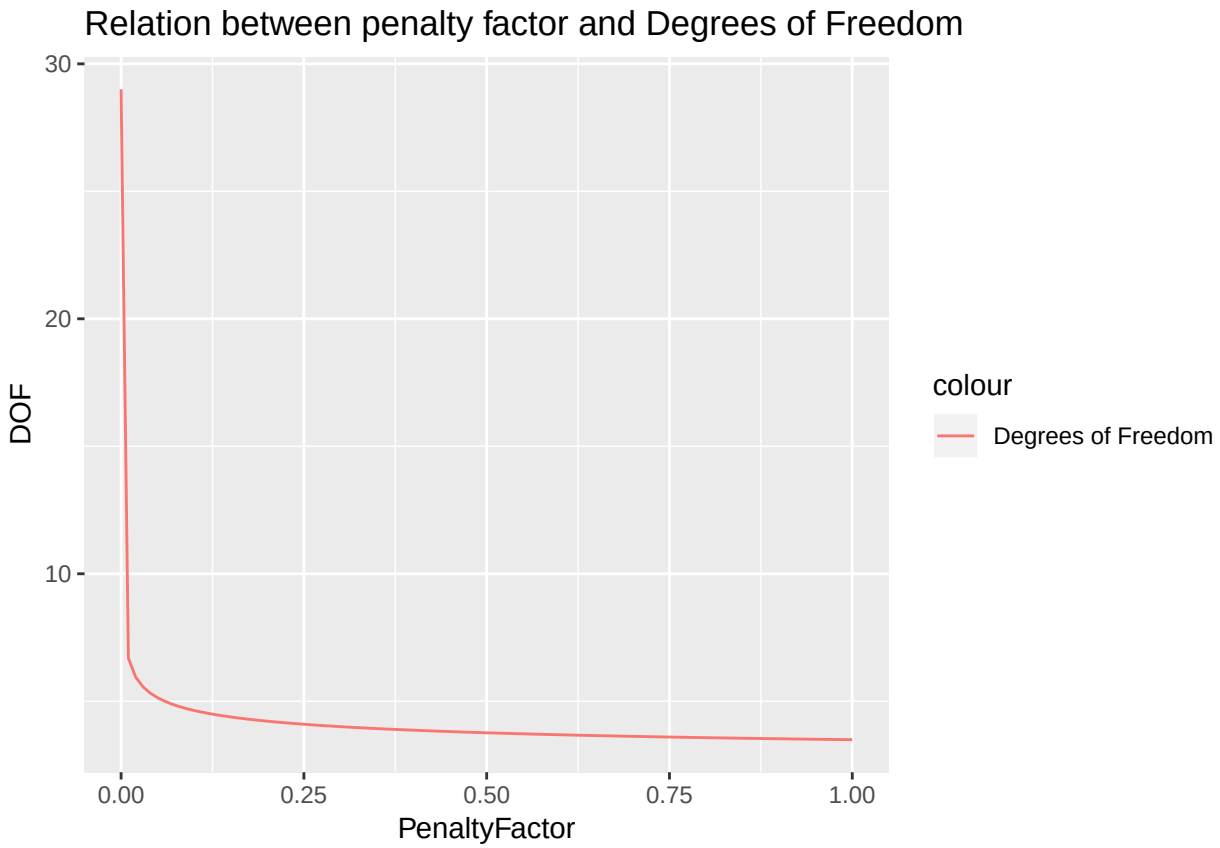
The red line corresponding to very high penalty factor(100) gives too smooth a function which underfits the data. Too high a penalty factor results in a very simple fit to the data which underfits the data. For a very high penalty factor as 100 we get a straight line fit to the data.

C) Relation between penalty factor and deviance



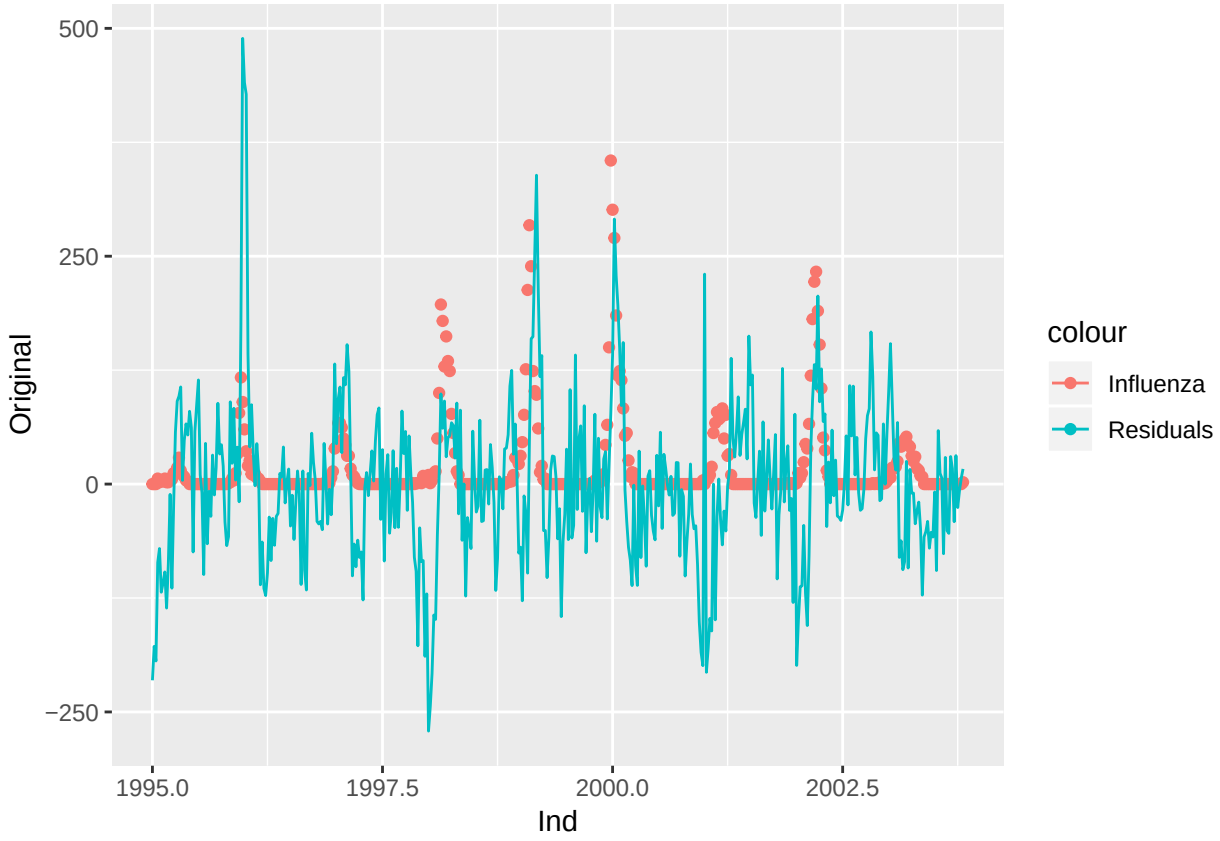
From above plot, we see that as the penalty factor increases the deviance increases. This is reasonable as, the increase in Penalty Factor means a simpler model is being fit to the data so that would increase the deviance as a simpler model wont be able to predict properly.

D) Relation between penalty factor and degrees of freedom



On the other hand the increase in penalty factor decreases the degrees of freedom of the model. This is also a reasonable thing as, the increase in penalty means a simple model being fit to the data and a simple model will have low degrees of freedom.

Part 5: Examining relation between Influenza and GAM Residuals



From above plot, it is eveident that there is a correlation between Influenza and Residuals, whenvever there is a increase in influenza case there is a increase in the residuals. This is because the prediction we got from our model was not able to capture all the peaks in the mortality, so the corresponding residuals at the point is high. This is the reason the residuals and Influenza are correlated. The residuals are very noisy but whenever there is a increase in the influenza the corresponding residual value also increases.

Part 6: GAM model in R in which mortality is be modelled as an additive function of the spline functions of year, week, and the number of confirmed Influenza cases.

A) Examining the updated model

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## Mortality ~ s(Year, k = allYears) + s(Week, k = allWeeks) + s(Influenza,
##     k = allInf)
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1783.8        3.2   557.5   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

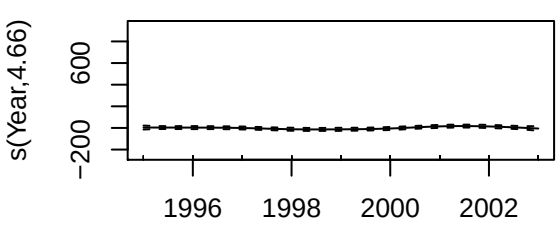
```
## Approximate significance of smooth terms:
##                  edf Ref.df      F p-value
## s(Year)        4.663  5.677  1.487   0.181
## s(Week)       14.641 18.248 18.533  <2e-16 ***
## s(Influenza) 69.740 72.833  5.600  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Rank: 134/144
## R-sq.(adj) =  0.819   Deviance explained = 85.4%
## GCV = 5846.7  Scale est. = 4699.8    n = 459
```

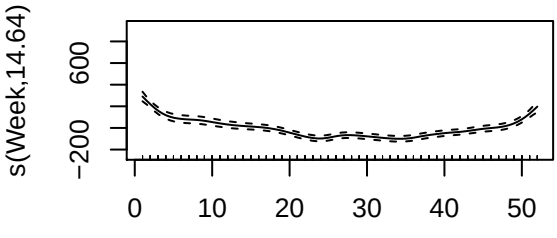
A small p-value indicates that it is unlikely we will observe a relationship between the predictor and response variables due to chance. From above results, wee see that there is significant relationship between Mortality and Influenza. Influenza also has higher degrees of freedom which indicates that it has a complex significant relationship with Mortality in the fit.

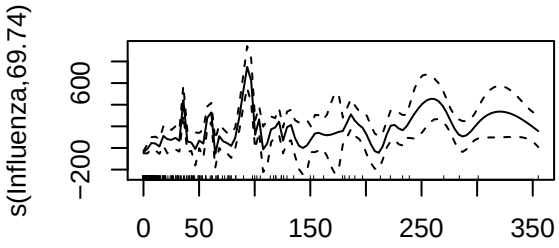
B) Plot of predicted and observed Mortality using the updated model

It looks like the prediction accuracy has increased in the updated model, as it now covers the peaks much better as compared to the older model, thus we can say there is significant relation between Mortality and Influenza.

C) Influence of Week, Year and Influenza on Mortality







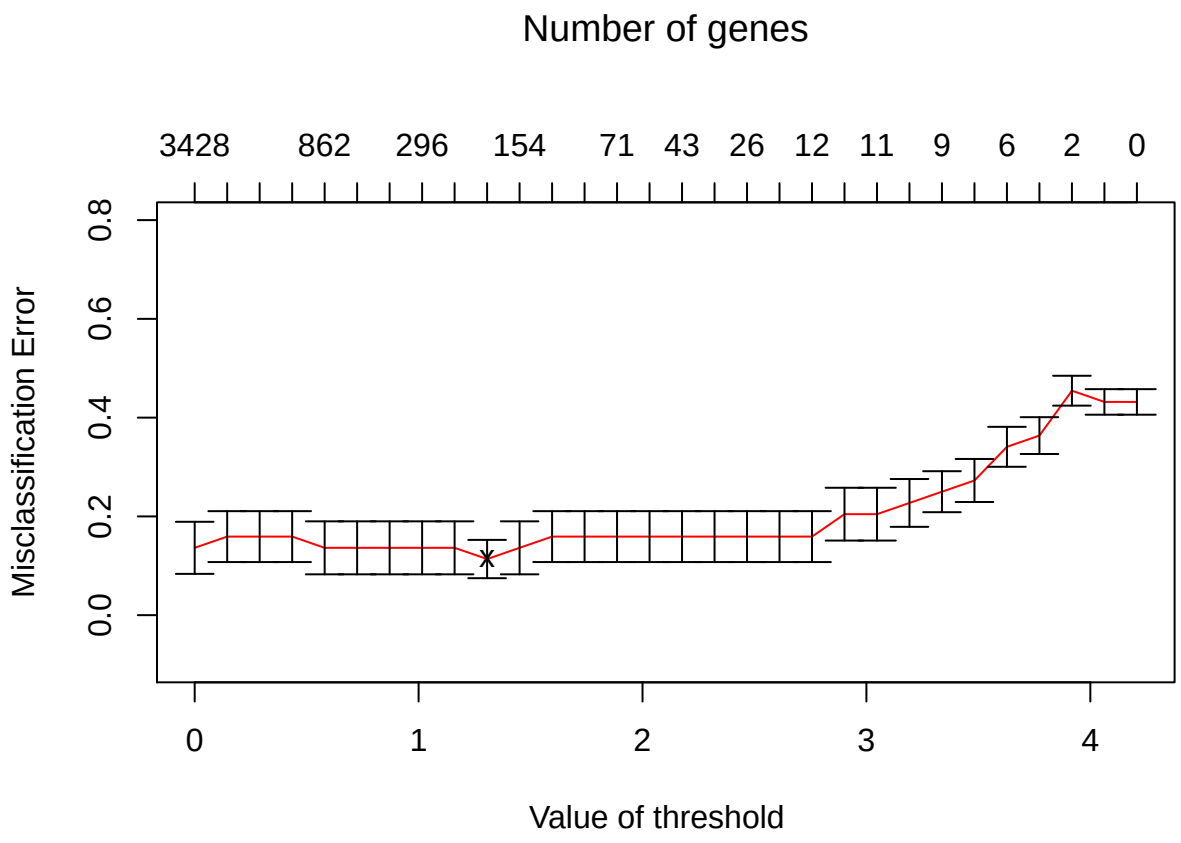
As we can see from above plots, the Mortality is not effected by Year, somwhat effected by Week but has highly significant relation with Influenza outbreak.

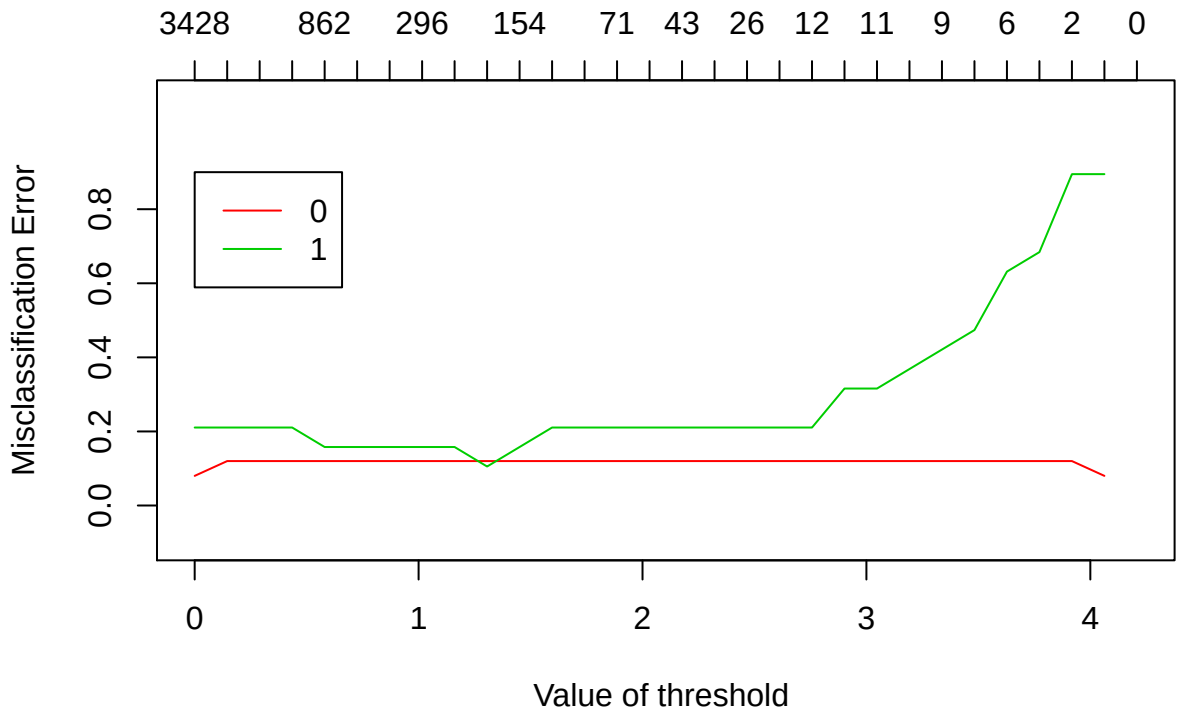
Assignment 2: High-dimensional methods

Part 1: Nearest Shrunken Centroid Classification

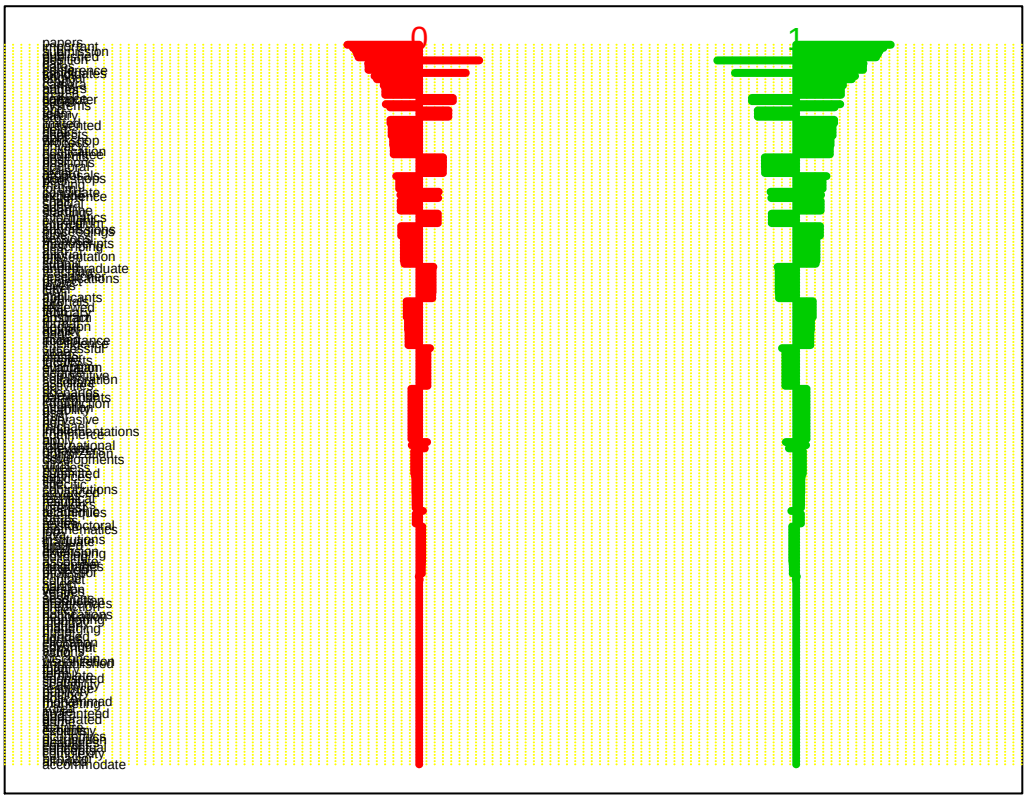
A) Fitting nearest shrunken centroid model using pamr package

```
## Train dimension:  44 4703
## Test dimension:  20 4703

## 1234567891011121314151617181920212223242526272829 30

## 12Fold 1 :1234567891011121314151617181920212223242526272829 30
## Fold 2 :1234567891011121314151617181920212223242526272829 30
## Fold 3 :1234567891011121314151617181920212223242526272829 30
## Fold 4 :1234567891011121314151617181920212223242526272829 30
## Fold 5 :1234567891011121314151617181920212223242526272829 30
## Fold 6 :1234567891011121314151617181920212223242526272829 30
## Fold 7 :1234567891011121314151617181920212223242526272829 30
## Fold 8 :1234567891011121314151617181920212223242526272829 30
## Fold 9 :1234567891011121314151617181920212223242526272829 30
## Fold 10 :1234567891011121314151617181920212223242526272829 30
```


Number of genes
3428
862
296
154
71
43
26
12
11
9
6
2
0
Misclassification Error
0.0
0.2
0.4
0.6
0.8
0
1
2
3
4
Value of threshold


3428
862
296
154
71
43
26
12
11
9
6
2
0
Misclassification Error
0.0
0.2
0.4
0.6
0.8
0
1
0
1
2
3
4
Value of threshold

```
## 1
```



The plot above represents the features that have significant relation with the target variable. In the plot, 1 represents announcements of conference and 0 represents evverything else. It is eveident from the plot, that half of the features in upper section are more important for this classification and the features in the lower half do not contribute much.

B) Feature selected

```
##
## Total Features Selected:  231
```

```
##
## 10 Most contributing features are:
##  papers important submission due published position call conference dates candidates
```

C) Is it reasonable to that feature selected have strong effect on target?

```
##       name          0-score   1-score
##  [1,] "papers"     "-0.3814" "0.5019"
##  [2,] "important"  "-0.3519" "0.4631"
##  [3,] "submission" "-0.3368" "0.4431"
##  [4,] "due"        "-0.3301" "0.4344"
##  [5,] "published"  "-0.3223" "0.4241"
##  [6,] "position"   "0.318"   "-0.4184"
##  [7,] "call"       "-0.2717" "0.3575"
##  [8,] "conference" "-0.2698" "0.355"
##  [9,] "dates"      "-0.2698" "0.355"
## [10,] "candidates" "0.2468"  "-0.3247"
```

Above are the top 10 most significant features along with score for 0 (Everything else) and 1 (Conference announcement) As we can see the name column, there is very high probability that every conference announcement email having these words. So it is reasonable to believe that these features contribute the most to classify the target variable.

D) Test Error

```
## [1] "Performance on Test set: "
```

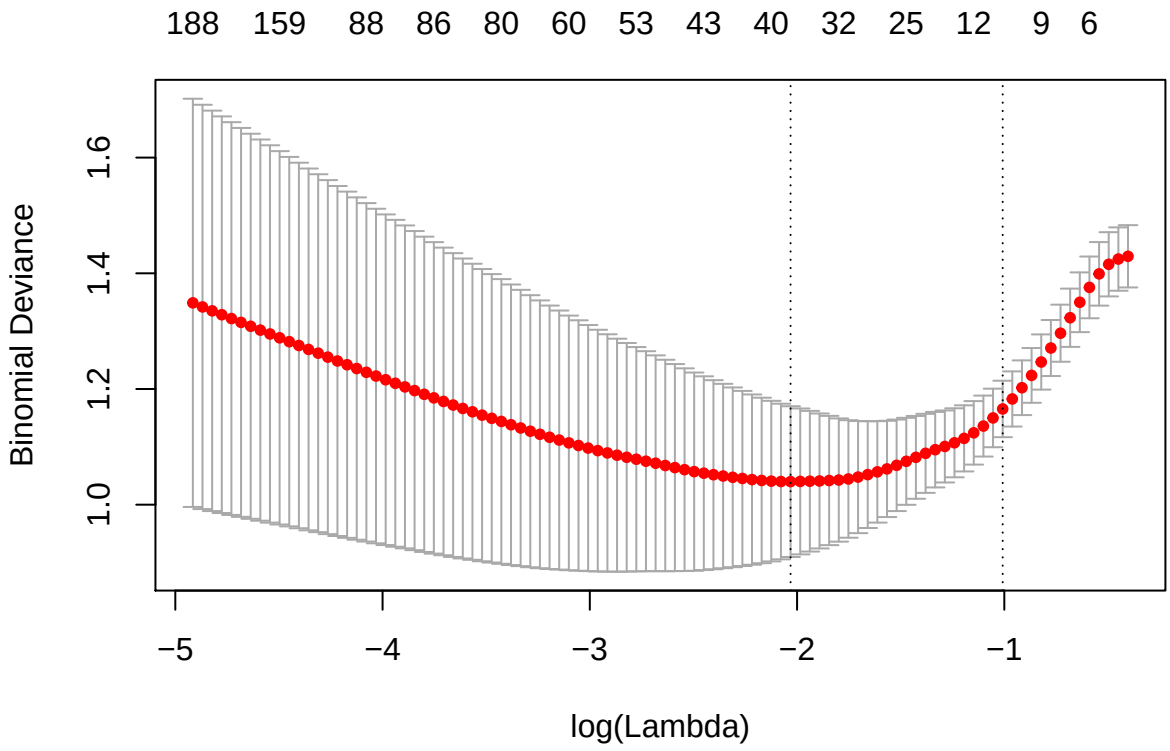
```
##      preds
## testy  0  1
##     0 10  0
##     1  2  8
```

```
##
## Misclassification rate on test:  0.1
```

Part 2: Computing test error using Elastic Net and SVM

A) Elastic Net

1) Training Elastic Net using glmnet package



lambda.min is the value of λ that gives minimum mean cross-validated error. The other λ saved is lambda.1se, which gives the most regularized model such that error is within one standard deviation of the minimum. We are going to use lambda.min in further steps.

2) Test Elastic Net

```
##      preds
## testy  0  1
##     0 10  0
##     1  2  8
```

```
##
## Misclassification rate on test Using GLMNET:  0.1
```

```
##
## Number of features selected:  100
```

B) Support Vector Machine Using Kernel Vannilldot

1) Training support vector machine using kernlab package

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

```
## Support Vector Machine object of class "ksvm"
##
## SV type: C-svc  (classification)
##  parameter : cost C = 1
##
## Linear (vanilla) kernel function.
##
## Number of Support Vectors : 43
##
## Objective Function Value : -2.0817
## Training error : 0.022727
```

2) Test SVM

```
##       Predicted
## Actual  0  1
##      0 10  0
##      1  1  9
```

```
##
## Misclassification rate on test Using SVM with Vanilladot Kernel:  0.05
```

```
##
## Number of feature selected:  43
```

C) Comparison between all models

```
##                       Model MisClassificationRates FeaturesSelected
## 1 Nearest Shrunken Centroid                   0.10               10
## 2               Elastic Net                   0.10              100
## 3    Support Vector Machine                   0.05               43
```

As seen from the above results, the misclassification rate for Support Vector Machine with Vanilladot Kernel is the least among all the models. Therefore, the optimal model to fit this data is SVM.

Part 3: Implementing Benjamini-Hochberg method

```
##   [1] "abstract"        "academic"        "acceptance"      "accepted"        "access"          "acm
```

```
##  [28] "bio"             "call"            "calls"           "camera"          "canada"          "can
##  [55] "contributions"   "copyright"       "covering"        "cross"           "curriculum"      "dat
##  [82] "expected"        "experience"      "extension"       "feature"         "february"        "fig
## [109] "include"         "included"        "india"           "infrastructures" "initially"       "ins
## [136] "letter"          "levels"          "limited"         "liu"             "looking"         "mad
## [163] "ontologies"      "opportunity"     "optimization"    "org"             "organizers"      "org
## [190] "privacy"         "proceedings"     "process"         "professor"       "proficiency"     "pro
## [217] "scalability"     "scenarios"       "science"         "scope"           "security"        "ser
## [244] "taiwan"          "takes"           "tasks"           "teaching"        "team"            "tec
## [271] "versions"        "vienna"          "visualization"   "vitae"           "wang"            "wir

## Total Features Selected are :  281
```

The above features are the ones that were retained be the hypothesis and have significant relation with the conference announcement.

Appendix

```
knitr::opts_chunk$set(
    echo = FALSE,
    message = FALSE,
    warning = FALSE
)
library(readxl)
library(ggplot2)
library(reshape2)
library(mgcv)
library(pamr)
library(glmnet)
library(kernlab)
#library(IHW)
library(FSA)
#A1
#1
influenza = read_xlsx("Influenza.xlsx")
#head(influenza)
ggplot(influenza) + geom_line(aes(Time, Influenza), col = "red")
ggplot(influenza) + geom_line(aes(Time, Mortality), col = "blue")
#ggplot(influenza) + geom_line(aes(Time, Influenza, col="red")) + geom_line(aes(Time, Mortality, col="b
#Part 2
model = gam(Mortality~Year+s(Week, k= length(unique(influenza$Week))), data = influenza, family = gaussi
#plot(model, pages = 1, residuals = T)
summary(model)
# Part 3
preds <- predict(model,influenza)
results <- data.frame(Time=influenza$Time,
                      Mortality=influenza$Mortality,
                      Predicted=preds,
                      Week=influenza$Week,
                      Year=influenza$Year)
ggplot(results)+
  geom_line(aes(x=Time,y=Mortality,color="Original"))+
  geom_line(aes(x=Time,y=Predicted,color="predicted"))+
  ggtitle("Prediction of Mortality using GAM")
```

```r
summary(model)
ggplot(results)+
  geom_point(aes(x=Year,y=Mortality,color="Original"))+
  ggtitle("Mortality Trend through year")
# Spline component
plot.gam(model,residuals = TRUE)
model2 <- gam(Mortality~Year+s(Week, k=52, sp=model$sp),data=influenza,family = "gaussian")
cat("\nAt Penalty Factor: ",model$sp," explained deviance is: ",model2$deviance)
modelL <-  gam(Mortality~Year+s(Week, k=52, sp=0),data=influenza,family = "gaussian")
modelH <-  gam(Mortality~Year+s(Week, k=52, sp=100),data=influenza,family = "gaussian")

predL <- predict(modelL,influenza)
predH <- predict(modelH,influenza)

results <- data.frame(Ind =influenza$Time,Original=influenza$Mortality,PredictedLow=predL,PredictedHigh=

ggplot(results)+
  geom_point(aes(x=Ind,y=Original,color="Original"))+
  geom_line(aes(x=Ind,y=PredictedLow,color="Low penalty"))+
  geom_line(aes(x=Ind,y=PredictedHigh,color="High penalty"))+
  xlab("Time")+
  ggtitle("Predicted and observed Mortality for low and high penalty factor")
penFacs <- seq(0,10,0.1)
getDev<- function(sp){
  model <- gam(Mortality~Year+s(Week, k=52, sp=sp),data=influenza,family = "gaussian")
  return(model$deviance)
}
devcs <- sapply(penFacs,getDev)
results<-data.frame(PenaltyFactor=penFacs,Deviance=devcs)
ggplot(results)+
  geom_point(aes(x=PenaltyFactor,y=Deviance,color="Deviance"))+
  ggtitle("Relation between penalty factor and deviance")
q = seq(0,1,0.01)
res = matrix(0, nrow = 0, ncol = 2)
for( i in q){
  model_i = gam(Mortality~Year+s(Week, k= length(unique(influenza$Week)), sp=i),
                data = influenza, family = gaussian)
  mdf = sum(model_i$edf)
  res = rbind(res, c(i, mdf))
}
res = as.data.frame(res)
colnames(res) = c("PenaltyFactor", "DOF")
ggplot(res)+
  geom_line(aes(x=PenaltyFactor,y=DOF,color="Degrees of Freedom"))+
  ggtitle("Relation between penalty factor and Degrees of Freedom")

# Part 5
results <- data.frame(Ind=influenza$Time,Original=influenza$Influenza,Residuals=as.data.frame(model$resi
colnames(results)[3] <-c("Residuals")
ggplot(results)+
  geom_point(aes(x=Ind,y=Original,color="Influenza"))+
  geom_line(aes(x=Ind,y=Residuals,color="Residuals"))
# Part 6
```

```r
allYears <- length(unique(influenza$Year))
allWeeks <- length(unique(influenza$Week))
allInf <- length(unique(influenza$Influenza))

modelFinal <-gam(Mortality~s(Year,k=allYears)
                 +s(Week,k=allWeeks)
                 +s(Influenza,k=allInf),data=influenza,
                   family = "gaussian",method="GCV.Cp")
summary(modelFinal)
preds <- predict(modelFinal,influenza)

results <- data.frame(Ind=influenza$Time,Original=influenza$Mortality,Predicted=preds)
ggplot(results)+
  geom_point(aes(x=Ind,y=Original,color="Original"))+
  geom_line(aes(x=Ind,y=Predicted,color="Predicted"))
par(mfrow=c(2,2))
plot.gam(modelFinal)
#Part1
set.seed(12345)
data <- read.csv("data.csv",sep = ";",check.names = FALSE ,encoding = "latin1")
data$Conference <- as.factor(data$Conference)

n=dim(data)[1]
id=sample(1:n, floor(n*0.70))
train=data[id,]
test=data[-id,]
cat("Train dimension: ",dim(train),"\nTest dimension: ",dim(test))

#Organize data
rownames(train) <- 1:nrow(train)
trainx <- t(as.matrix(train[,-4703]))
trainy <- as.matrix(train[,4703])

newTrain <- list(x=trainx,y=trainy,geneid=as.character(1:nrow(trainx)),genenames=rownames(trainx))

# fit model
model1 <- pamr.train(newTrain)

## CV model
model.cv <- pamr.cv(model1,newTrain,nfold = 10)

#plot of cv
pamr.plotcv(model.cv)


minThresh <- model.cv$threshold[which.min(model.cv$error)]

#Minimum Error Model
model2 <- pamr.train(newTrain, threshold = minThresh)

# Centroid Plot
pamr.plotcen(model1, newTrain, threshold =minThresh)
featSelected <- pamr.listgenes(model1, newTrain, threshold = minThresh, genenames=TRUE)
```

```r
#Total features selected
cat("\nTotal Features Selected: ",dim(featSelected)[1])

# Most contributing features (Top 10)
topFeat <- featSelected[1:10,"name"]
cat("\n10 Most contributing features are: \n",topFeat)
print(featSelected[1:10,2:4])
#Test error
print("Performance on Test set: ")
testx <- t(as.matrix(test[,-4703]))
testy <- as.matrix(test[,4703])

preds <- pamr.predict(model1,newx=testx,threshold = minThresh,type="class")

confMat <- table(testy,preds)
print(confMat)
misRate1 <- 1- sum(diag(confMat))/sum(confMat)
cat("\nMisclassification rate on test: ",misRate1)
#Part 2
set.seed(12345)

trainx <- as.matrix(train[,-4703])
trainy <- as.matrix(train$Conference)

model2 <- glmnet(x=trainx,y=trainy,family = "binomial",alpha = 0.5)

#model.cv <- cv.glmnet(model)
model.cv <- cv.glmnet(x=trainx,y=trainy,family = "binomial",alpha = 0.5)
plot(model.cv)
# test Elastic Net
set.seed(12345)
testx <- as.matrix(test[,-4703])
testy <- as.matrix(test$Conference)
preds <- predict(model2,testx,s = model.cv$lambda.min, type="class")

confMat <- table(testy,preds)
print(confMat)
misRate2 <- 1- sum(diag(confMat))/sum(confMat)
cat("\nMisclassification rate on test Using GLMNET: ",misRate2)
cat("\nNumber of features selected: ",dim(coef(model2))[2])
set.seed(12345)
model3 <- ksvm(Conference~.,data=train,kernel="vanilladot",scaled=FALSE)
print(model3)
preds <- predict(model3,test,type="response")
confMat <- table(Actual=test$Conference,Predicted=preds)
print(confMat)
misRate3 <- 1- sum(diag(confMat))/sum(confMat)
cat("\nMisclassification rate on test Using SVM with Vanilladot Kernel: ",misRate3)
cat("\nNumber of feature selected: ",length(model3@coef[[1]]))
misRates <-c(misRate1,misRate2,misRate3)
models <- c("Nearest Shrunken Centroid","Elastic Net","Support Vector Machine")
featuresSelected <- c(length(topFeat),dim(coef(model2))[2],length(model3@coef[[1]]))
results <- data.frame(Model=models,MisClassificationRates=misRates,FeaturesSelected=featuresSelected)
```

```r
print(results)
p_value <- c()
for (i in 1:4702){
  x <- data[,i]
  p_value[i] <- t.test(x ~ Conference, data = data, alternative = "two.sided")$p.value
}
p_value <- as.data.frame(p_value)
p_value$reject_flag <- as.factor(ifelse(p_value$p_value <0.05, "Retain", "Drop"))
p_value$column_index <- row.names(p_value)
keep <- na.omit(ifelse(p_value$reject_flag == "Retain", as.numeric(p_value$column_index), NA))
colnames(data[,keep])
cat("Total Features Selected are : ", length(keep), "\n")
```