Lab2_block2

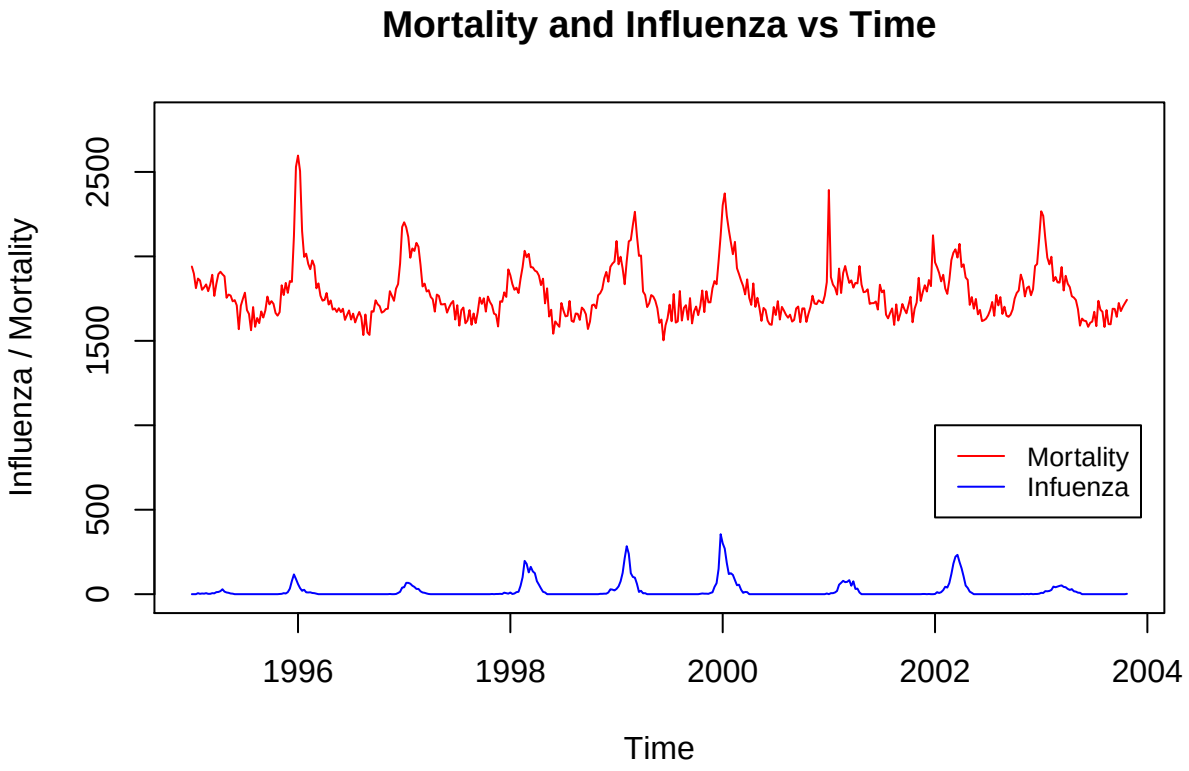
Andreas

14 Dec 2018

Contents

Assignment 1 -Mortality Rates

1

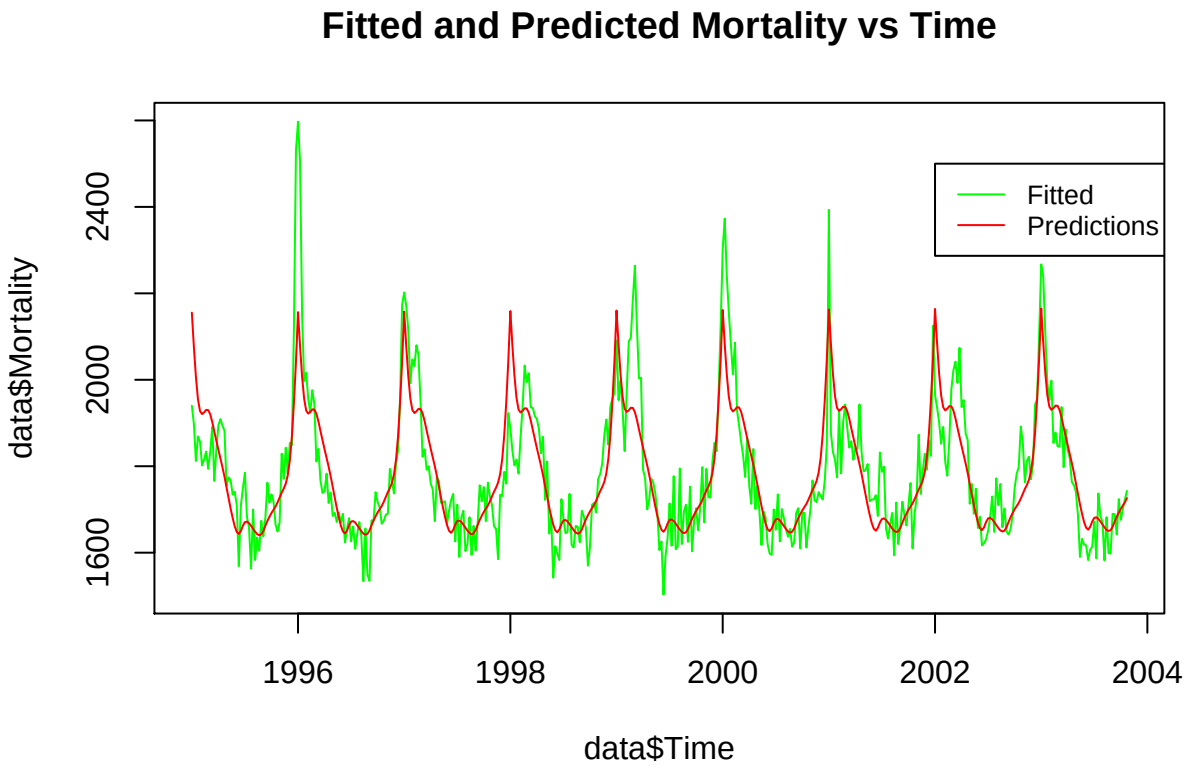


We can see from the plot that when we have a peak at mortality we also have a peak at influenza.Thus we can conclude that there is possitive correlation between mortality and Influenza

2

The underlying probabilistic model is $Mortality = N(Year + spline(Week), \sigma^2)$.Using the coefficients obtained the equation from the model is $Mortality = -680.598 + 1.233Year + 14.23spline(Week)$

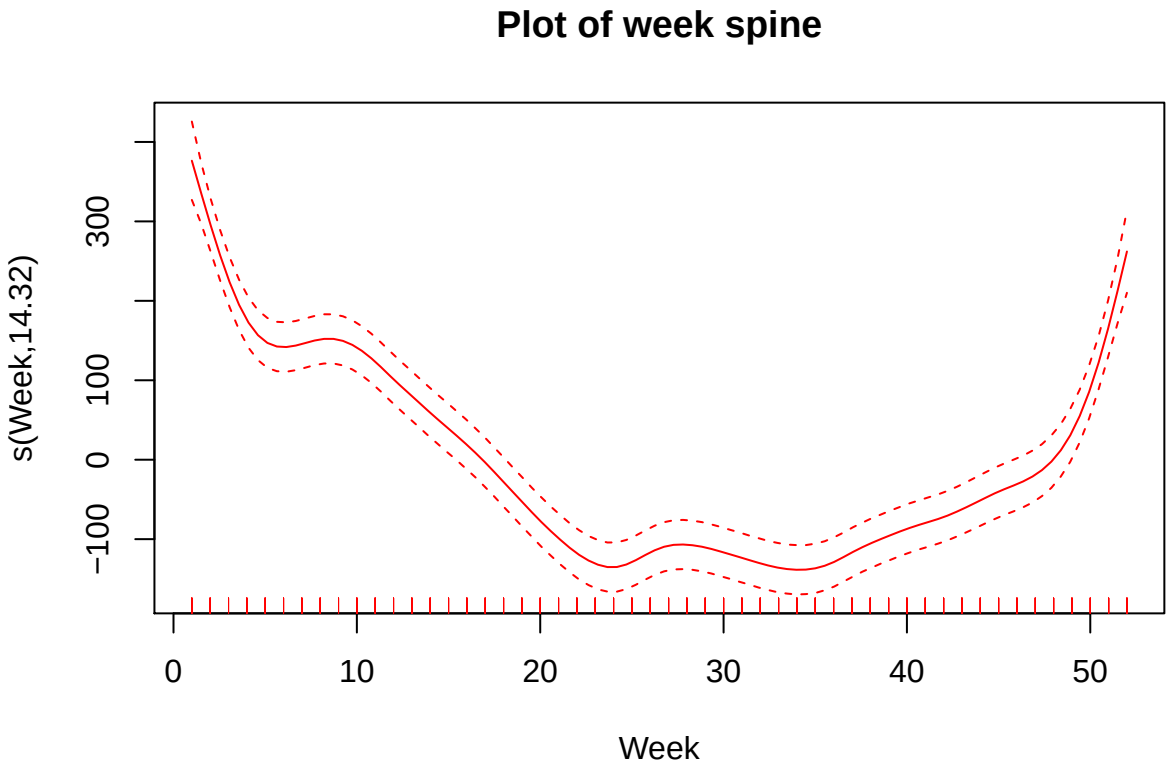
3



The above plot shows the fitted (green line) and predicted (red) values for Mortality.As we can see there is a trend in Mortality from one year to the other because there is allways one peak and one trough every year.In constrant we can see that the line of our predictions using gam follows quite well the real trend of Mortality although it is much smoother and doesn't map the highs and lows well.

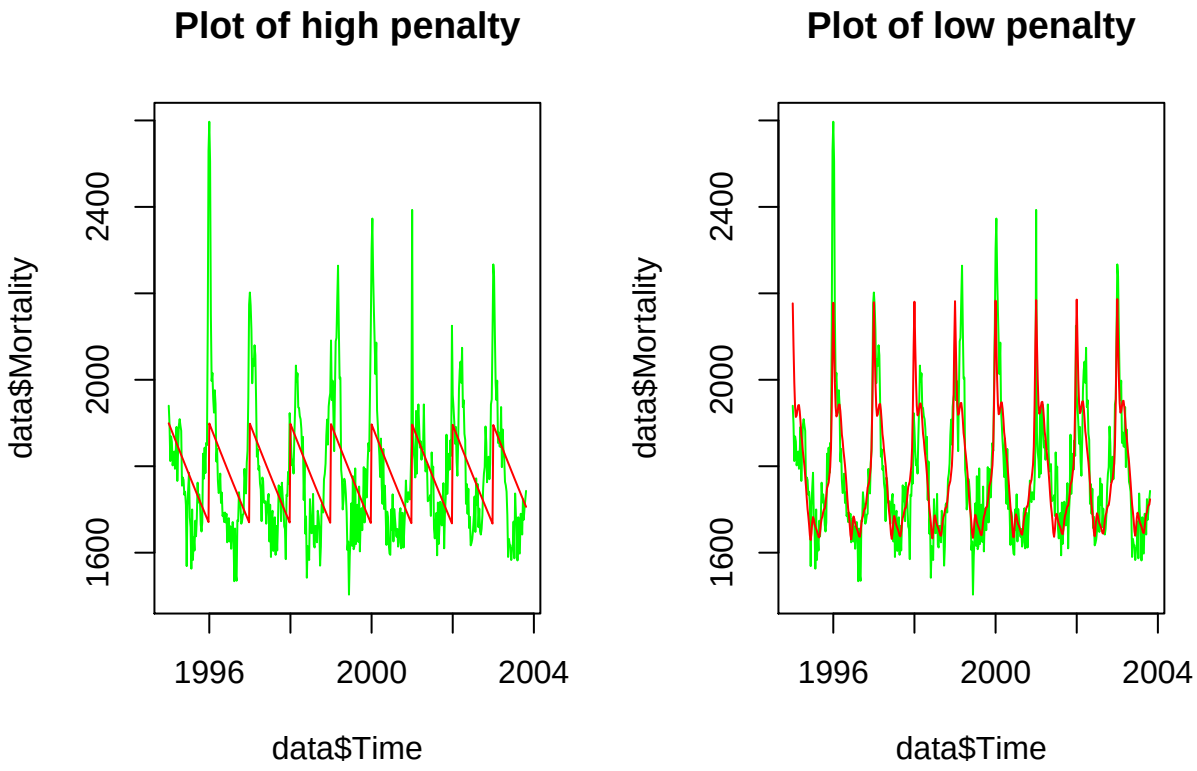
```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## Mortality ~ s(Week, k = length(unique(data$Week))) + Year
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -680.598   3367.760  -0.202    0.840
## Year           1.233      1.685   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
```

From the summary of our model we can see that the p-value for the intercept and the Year is not significant but for the Week is significant.



From the plot of the spline component we can see that there is a reasonal pattern in the week because it starts decreasing reaching until week 35 and then starts rising again.

4



We can see from the plots of fitted(green) and predicted(red) values for high and low penalty model that the model with the high peanalty is smoother compared with the one with the lower penalty beacause more penalty leads to more interpretable model.

```
## ============================================================================================
##  The deviance for low penalty model is:   3646841 and fo high penalty model is:  9804267
```

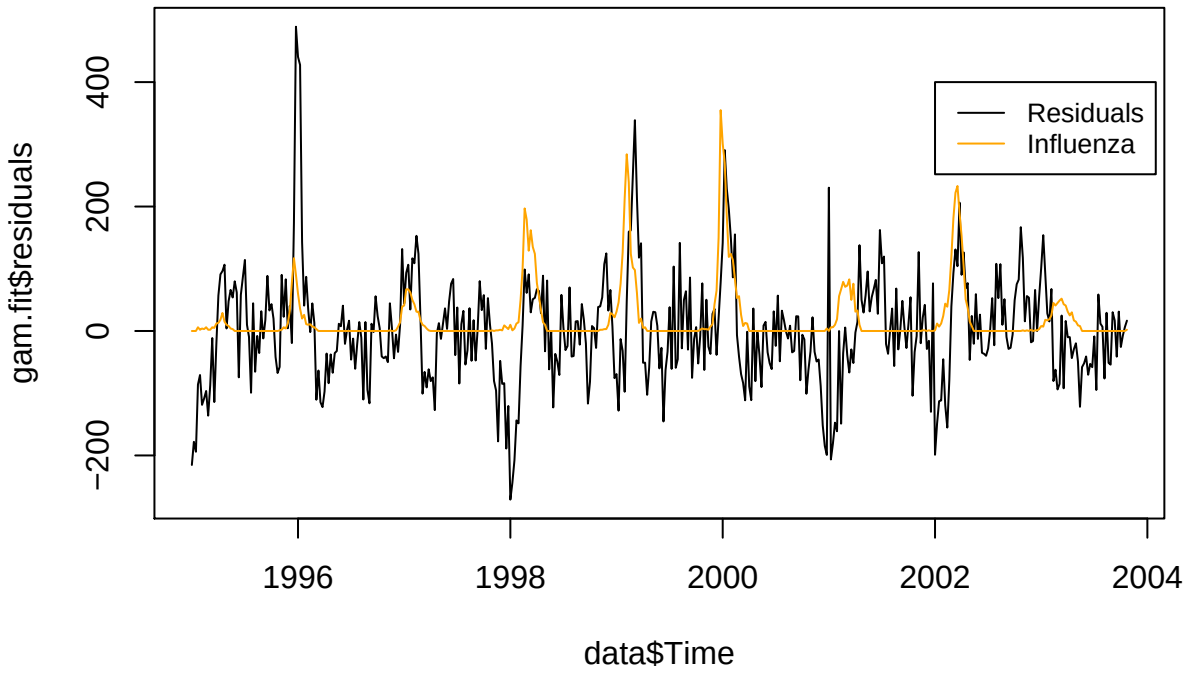
We can see comparing the deviances for the models that high penalty is related to higher deviance greater and low penalty with less deviance.

```
##              low_peanalty  high_peanlty
## (Intercept)     1.0000000  1.000000e+00
## Year            1.0000000  1.000000e+00
## s(Week).1       0.2038090  2.067485e-05
## s(Week).2       0.4236739  1.782799e-05
## s(Week).3       0.3904950  1.526864e-05
## s(Week).4       0.4321812  1.310983e-05
## s(Week).5       0.4225551  1.141598e-05
## s(Week).6       0.4706524  1.024038e-05
## s(Week).7       0.4919762  9.724288e-06
## s(Week).8       0.5028568  1.046104e-05
## s(Week).9       0.5051198  1.479284e-05
## s(Week).10      0.4971517  2.302393e-05
## s(Week).11      0.4898524  2.042234e-05
## s(Week).12      0.4851651  1.635105e-05
## s(Week).13      0.4776443  1.458902e-05
## s(Week).14      0.4545776  1.409421e-05
```

```
## s(Week).15     0.4372078  1.432154e-05
## s(Week).16     0.4407936  1.504731e-05
## s(Week).17     0.4247632  1.616748e-05
## s(Week).18     0.4213995  1.762129e-05
## s(Week).19     0.4189296  1.936244e-05
## s(Week).20     0.4097742  2.134809e-05
## s(Week).21     0.4149130  2.353492e-05
## s(Week).22     0.4039036  2.587839e-05
## s(Week).23     0.4120790  2.833332e-05
## s(Week).24     0.4024838  3.085522e-05
## s(Week).25     0.4117324  3.340171e-05
## s(Week).26     0.4028477  3.593401e-05
## s(Week).27     0.4129956  3.841820e-05
## s(Week).28     0.4058170  4.082636e-05
## s(Week).29     0.4161339  4.313738e-05
## s(Week).30     0.4097368  4.533764e-05
## s(Week).31     0.4205474  4.742146e-05
## s(Week).32     0.4163123  4.939156e-05
## s(Week).33     0.4261250  5.125966e-05
## s(Week).34     0.4250883  5.304743e-05
## s(Week).35     0.4325552  5.478847e-05
## s(Week).36     0.4390963  5.653179e-05
## s(Week).37     0.4395648  5.834801e-05
## s(Week).38     0.4580202  6.033930e-05
## s(Week).39     0.4543877  6.265157e-05
## s(Week).40     0.4710351  6.546842e-05
## s(Week).41     0.4959110  6.884000e-05
## s(Week).42     0.5109739  7.128451e-05
## s(Week).43     0.4835435  5.626509e-05
## s(Week).44     0.4922342 -3.407125e-06
## s(Week).45     0.4818735 -2.023107e-05
## s(Week).46     0.4966478 -5.168295e-06
## s(Week).47     0.4725357  5.696279e-06
## s(Week).48     0.3853328  1.254160e-05
## s(Week).49     0.2817319  1.740831e-05
## s(Week).50     0.9936556  5.843520e-03
## s(Week).51     0.9999987  1.000000e+00
```

Finally,from the reported table we can see that when the penalty is larger, the degrees of freedom are smaller than the degrees of freedom for low penalty.

5

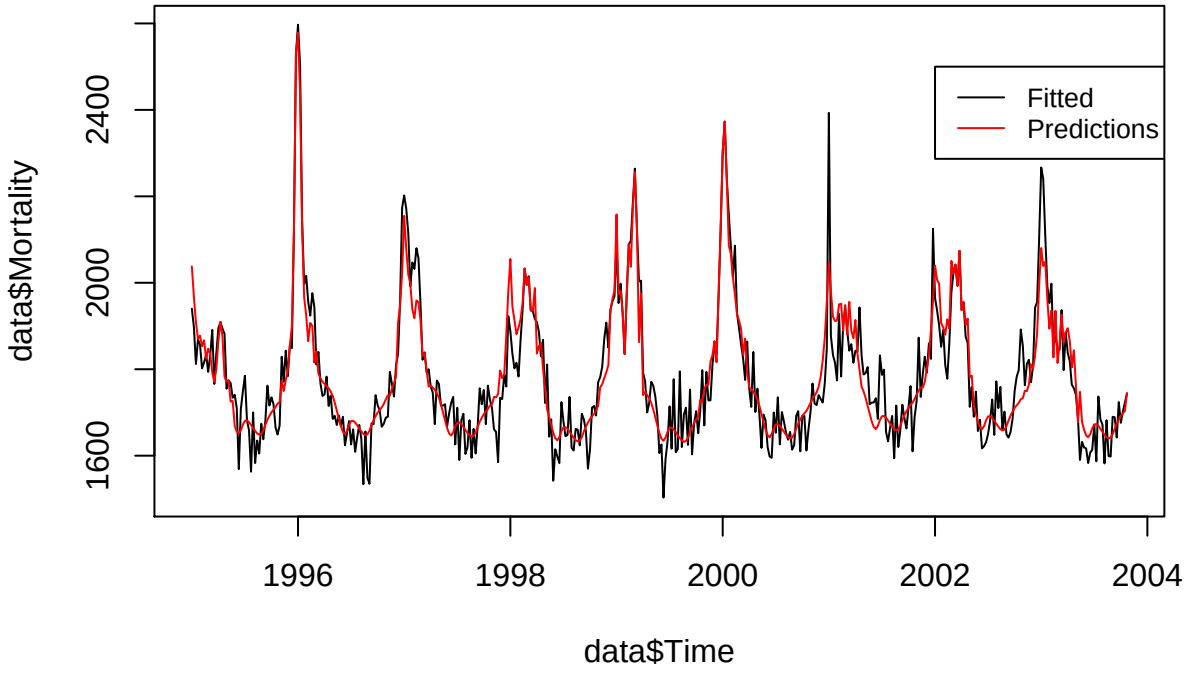


The above plot confirms that the outbreaks of Influenza follow the trend of residuals.When we have a peak at influenza we have a peak in the residuals and from that we can say that they correlated.

```
## The correlaation between residuals and Influenza is : 0.3397395
```

We can confirm that there is positive correlation for Influenza and residuals although not very strong.

6



From the plot of the fitted and predicted values of Mortality we can conclude that the model with the splines of Year,Influenza and Week fits very well in the Mortality trend and maps its trend quite well.Also we can conclude that Influenza plays a significant part in the explaination of Mortality which is something very reasonable.

```
## The correlation of fitted and predicted values for Mortality is :
##  0.9244462
```

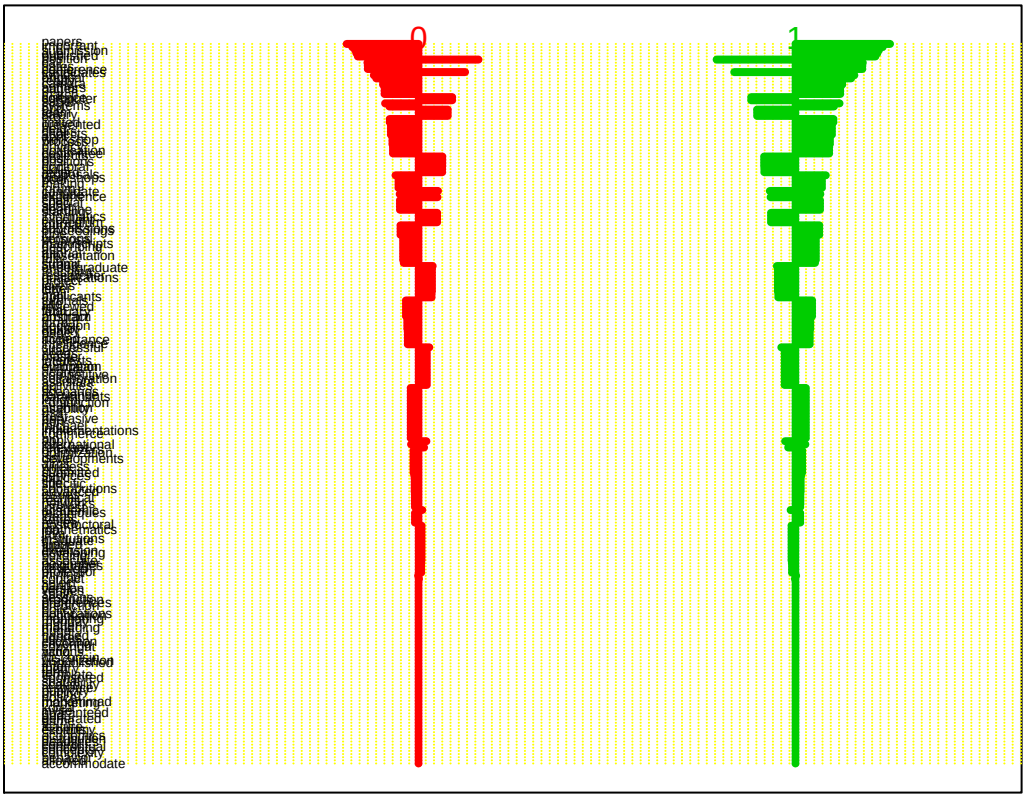
We can confirm are previous claim from the correlation of fitted and true values of Mortality above which indicated a strong correlation.

Assignment 2-High-dimensional methods

1

```
## 12345678910111213141516171819202122232425262728293O
```

```
## 12Fold 1 :12345678910111213141516171819202122232425262728293O
## Fold 2 :12345678910111213141516171819202122232425262728293O
## Fold 3 :12345678910111213141516171819202122232425262728293O
## Fold 4 :12345678910111213141516171819202122232425262728293O
## Fold 5 :12345678910111213141516171819202122232425262728293O
## Fold 6 :12345678910111213141516171819202122232425262728293O
## Fold 7 :12345678910111213141516171819202122232425262728293O
## Fold 8 :12345678910111213141516171819202122232425262728293O
## Fold 9 :12345678910111213141516171819202122232425262728293O
## Fold 10 :12345678910111213141516171819202122232425262728293O
```



The centroid plot is describing the distance of each feature from the 2 classes (0 and 1).The red column is coresponding to class 0 and green ceresponds to class 1

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

It its reasonable that the most cotributing features reported have a strong effect on classifing a mail as "announces of conferences" or not.Usually conference mails contain one or more of the words reported as most contributing.

```
## ============================================================================================
##  The misclssification error for test data is :  0.1 and the accuracy for test data is :  0.9
```

2a

```
## ============================================================================================
##  The misclssification error for test data is :  0.15 and the accuracy for test data is :  0.85
## The number of coefficients for elastic is:  12
```

2b

```
##  Setting default kernel parameters
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.
## ============================================================================================
##  The misclssification error for test data is :  0.05 and the accuracy for test data is :  0.95
## The number of coefficients for svm is:  43
```

Table 1: Summary table

misclass.error	accuracy	n_features	model
0.10	0.90	231	centroid
0.15	0.85	12	elastic
0.05	0.95	43	svm

The table provides misclassification error,accuracy and number of features selected for the 3 models we train.According to the table the svm looks more preferable from the rest of the models, it has a very low misclassification error(high accuracy) and with fewer features compared to other models.

3

```
## ========================================================
##  The cutoff value is:  0.0003762328
```

```
## ========================================================
##  The features coresponding to reject hypotheses are :
```

Table 2: Benjamin test

	p-value
papers	0.0000000
submission	0.0000000
position	0.0000000
published	0.0000002
important	0.0000003
call	0.0000004
conference	0.0000005
candidates	0.0000009
dates	0.0000014
paper	0.0000014
topics	0.0000051
limited	0.0000079
candidate	0.0000119
camera	0.0000210
ready	0.0000210
authors	0.0000215
phd	0.0000338
projects	0.0000350
org	0.0000374
chairs	0.0000586
due	0.0000649
original	0.0000649
notification	0.0000688
salary	0.0000797
record	0.0000909
skills	0.0000909
held	0.0001529
team	0.0001758
pages	0.0002007
workshop	0.0002007
committee	0.0002117

	p-value
proceedings	0.0002117
apply	0.0002166
strong	0.0002246
international	0.0002296
degree	0.0003762
excellent	0.0003762
post	0.0003762

From the table we can see the features that rejected from null hypotheses are the ones that are significant because the class of 1 which is the "announces of conferences" is more likely for these features.This seems reasonable because the words in the above talbe are more likely to be included in an email that is related to conference anouncement.

Appendix

```
set.seed(12345)
#read data xlsx
data<-readxl::read_xlsx("ïnfluenza.xlsx")
#plot time series
plot(data$Time,data$Mortality,type="l",col="red",ylim=c(0,2800),
     main="Mortality and Influenza vs Time",ylab="Influenza / Mortality",
     xlab= "Time")
lines(data$Time,data$Influenza,type="l",col="blue")
legend(2002,1000, legend=c("Mortality", "Infuenza"),
       col=c("red", "blue"), cex=0.8,lty = 1)
library(mgcv)
set.seed(12345)
gam.fit=gam(Mortality~s(Week,k=length(unique(data$Week)))+Year,
            data=data,method="GCV.Cp")


#gam.fit
#predictions using gam model
preds<-predict(gam.fit,data)
#plot predicitons and fitted Mortality
plot(data$Time,data$Mortality,type="l",col="green",
     main="Fitted and Predicted Mortality vs Time")
lines(data$Time,preds,col="red")
legend(2002,2500, legend=c("Fitted", "Predictions"),
       col=c("green", "red"), cex=0.8,lty = 1)

summary(gam.fit)

plot(gam.fit,main="Plot of week spine",col="red")

#gam model low penalty
gam.fit_low=gam(Mortality~s(Week,k=length(unique(data$Week)),sp=0.00001)+
                  Year, data=data,method="GCV.Cp")
#gam model high penalty
gam.fit_high=gam(Mortality~s(Week,k=length(unique(data$Week)),sp=100)+
                           Year, data=data,method="GCV.Cp")
```

```r
#predictions low penalty
preds_low<-predict(gam.fit_low,data)
#predictions high penalty
preds_high<-predict(gam.fit_high,data)

par(mfrow=c(1,2))
#plot of preds and fitted high penalty
plot(data$Time,data$Mortality,type="l",col="green",
     main="Plot of high penalty")
lines(data$Time,preds_high,col="red")

#plots of preds and fitted low penalty
plot(data$Time,data$Mortality,type="l",col="green",
     main="Plot of low penalty")
lines(data$Time,preds_low,col="red")


cat("================================================================================================\n",
    "The deviance for low penalty model is:  ",gam.fit_low$deviance,
    "and fo high penalty model is: ",gam.fit_high$deviance)


df_data<-data.frame(low_peanalty=gam.fit_low$edf,high_peanlty=gam.fit_high$edf)
df_data
plot(data$Time,gam.fit$residuals,type="l")
lines(data$Time,data$Influenza,col="orange")
legend(2002,400, legend=c("Residuals", "Influenza"),
       col=c("black", "orange"), cex=0.8,lty = 1)


cat("The correlaation between residuals and Influenza is :",cor(data$Influenza,gam.fit$residuals))


gam.fit1=gam(Mortality~s(Week,k=length(unique(data$Week)))+
               s(Year,k=length(unique(data$Year)))+
               s(Influenza,k=length(unique(data$Influenza))),data=data)

#summary(gam.fit)

plot(data$Time,data$Mortality,type="l")
lines(data$Time,predict(gam.fit1,data),col="red")
legend(2002,2500, legend=c("Fitted", "Predictions"),
       col=c("black", "red"), cex=0.8,lty = 1)


cat("The correlation of fitted and predicted values for Mortality is :\n",
    cor(fitted(gam.fit1),data$Mortality))

#read data
df<-read.csv2("data.csv",sep=";")
#make Conference a factor
df$Conference<-as.factor(df$Conference)
set.seed(12345)
```

```r
#split data to train and test
ind<-sample(nrow(df),floor(0.7*nrow(df)))
dftrain<-df[ind,]
dftest<-df[-ind,]


library(pamr)
#get the features
x<-t(dftrain[,-which(names(dftrain) == "Conference")])
#get the class
y<-as.factor(dftrain$Conference)
#make a list of data for centroid model
list_df=list(x=x,y=y,
             geneid=as.character(1:nrow(x)), genenames=rownames(x))
#fit model
par.model=pamr.train(list_df)
#cross validation
cvmodel=pamr.cv(par.model,list_df)
#find threshold
thres=cvmodel$threshold[which(cvmodel$error==min(cvmodel$error))]

#plot centroids
pamr.plotcen(par.model, list_df, threshold=thres)

#make a matrix with the cetroids
mat_genes<-invisible(pamr.listgenes(par.model,list_df,threshold=thres,genenames=T))
#number of parameters selected
num_centr<-nrow(mat_genes)
cat("The number of parametrs selected are: ",num_centr)
#10 most contributing features
cat("The 10 most contributing features are :\n",
    mat_genes[1:10,2])

#get the test features ->transpose it
dftest_features<-t(dftest[,-which(names(dftest) == "Conference")])
#make predictions
pred_pamr<-pamr.predict(par.model,dftest_features, threshold=thres,type="class")
#misclassificaiton error centroid
mis_error_cen<-mean(pred_pamr!=dftest$Conference)
#accuracy centroid
accuracy_cen<-mean(pred_pamr==dftest$Conference)
cat("=============================================================================================\n",
    "The misclssification error for test data is : ",mis_error_cen,
    "and the accuracy for test data is : ",accuracy_cen)

set.seed(12345)
library(glmnet)
#library(parallel)

#make feature matrix
X_data=dftrain[,-which(names(dftrain) == "Conference")]
y_data=as.factor(dftrain$Conference)
```

```r
#fit elastic model
elastic.fit<-cv.glmnet(as.matrix(X_data),y_data,family="binomial",alpha=0.5)


#test data
fitted_test<-as.matrix(dftest[,-which(names(dftest)=="Conference")])
#make predictions using lambda.min
elastic_preds<-predict(elastic.fit,fitted_test,type="class",s="lambda.1se")

#misclassification error elastic
mis_error_el<-mean(elastic_preds!=dftest$Conference)
#acuracy elastic
accuracy_el<-mean(elastic_preds==dftest$Conference)
cat("=============================================================================================\n",
    "The misclssification error for test data is : ",mis_error_el,
    "and the accuracy for test data is : ",accuracy_el)


#length of coefficients
num_elastic<-length(coef(elastic.fit,s=elastic.fit$lambda.1se)@x)
cat("The number of coefficients for elastic is: ",num_elastic)

library(kernlab)
#fit svm model with vanilladot kernel
svm<-ksvm(as.matrix(X_data),y_data,kernel="vanilladot") #getting a warring
#make predictions
svm_preds<-predict(svm,fitted_test,type="response")

mis_error_svm<-mean(svm_preds!=dftest$Conference)
accuracy_svm<-mean(svm_preds==dftest$Conference)
cat("\n")
cat("=============================================================================================\n",
    "The misclssification error for test data is : ",mis_error_svm,
    "and the accuracy for test data is : ",accuracy_svm)
num_svm<-length(coef(svm)[[1]])
cat("The number of coefficients for svm is: ",num_svm )

library(xtable)
#
sum_data<-data.frame(c(mis_error_cen,mis_error_el,mis_error_svm),
                     c(accuracy_cen,accuracy_el,accuracy_svm),
                     c(num_centr,num_elastic,num_svm),
                     c("centroid","elastic","svm"))
colnames(sum_data)<-c("misclass.error","accuracy","n_features","model")


library(knitr)
#make a table
kable(sum_data, caption = "Summary table")


t<-sapply(df[,-which(names(df)=="Conference")],
          function(x){ t.test(x[df$Conference==1],x[df$Conference==0])[["p.value"]]})
```

```r
benj<-function(p_values,alpha=0.05){
  p_values<-sort(p_values)
  indexes<-c(1:length(p_values))
  L<-p_values-((alpha*indexes)/length(p_values))
  best_p<-max(L[which(L<0)])
  cutoff<-p_values[L==best_p]
  rejected_values<-p_values[p_values<=cutoff]
  list(cutoff,rejected_values)


}

rej<-as.data.frame(benj(t)[[2]])
colnames(rej)<-c("p-value")
cat("============================================================\n",
    "The cutoff value is: ",benj(t)[[1]])
cat("\n")
cat("============================================================\n",
    "The features coresponding to reject hypotheses are : \n")

kable(rej, caption = "Benjamin test")
```