Lab2block2

Thijs Quast

10-12-2018

Contents

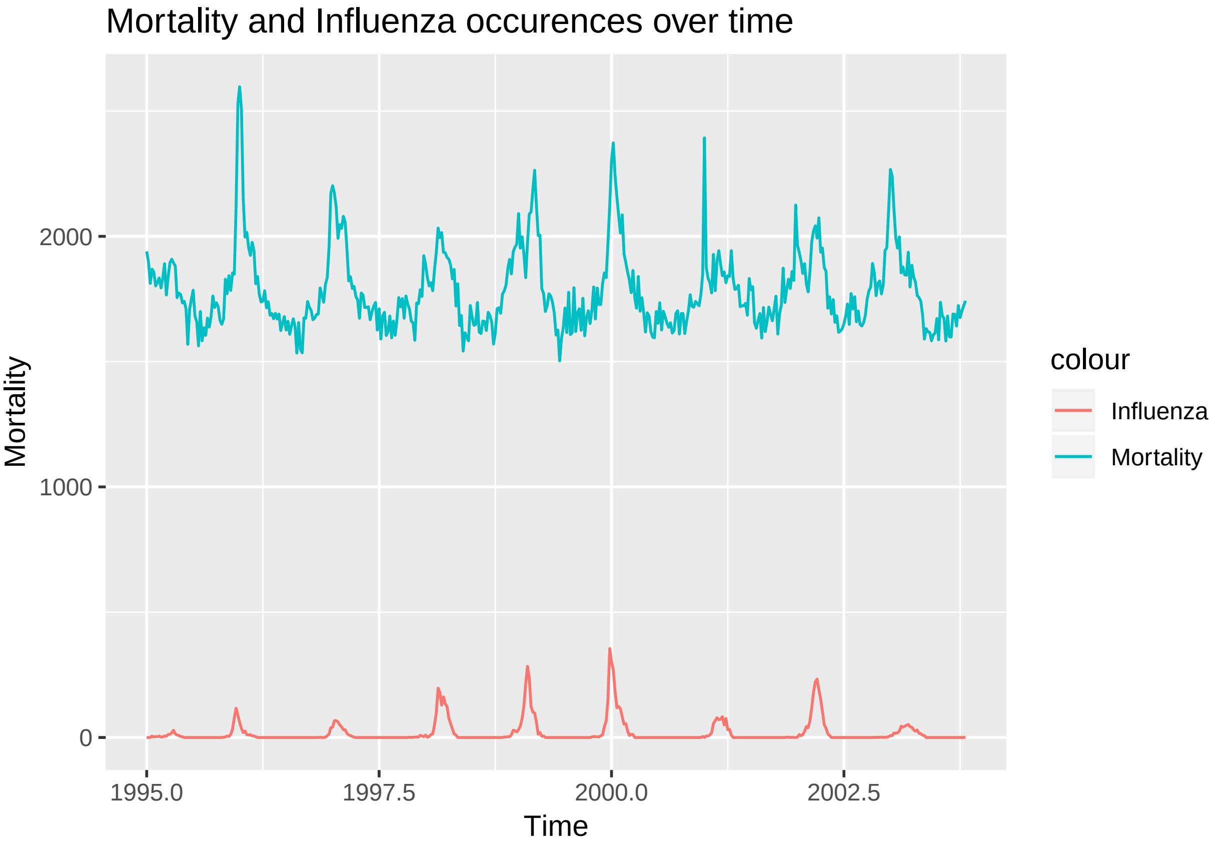
Assignment 1

1.1

```r
library(readxl)
options(scipen = 999)
influenza <- read_xlsx("influenza.xlsx")

library(ggplot2)
plot <- ggplot(data = influenza, aes(x = Time, y = Mortality, color = "Mortality")) +
  geom_line() +
  geom_line(aes(y = Influenza, color = "Influenza")) + ggtitle("Mortality and Influenza occurences over

plot
```



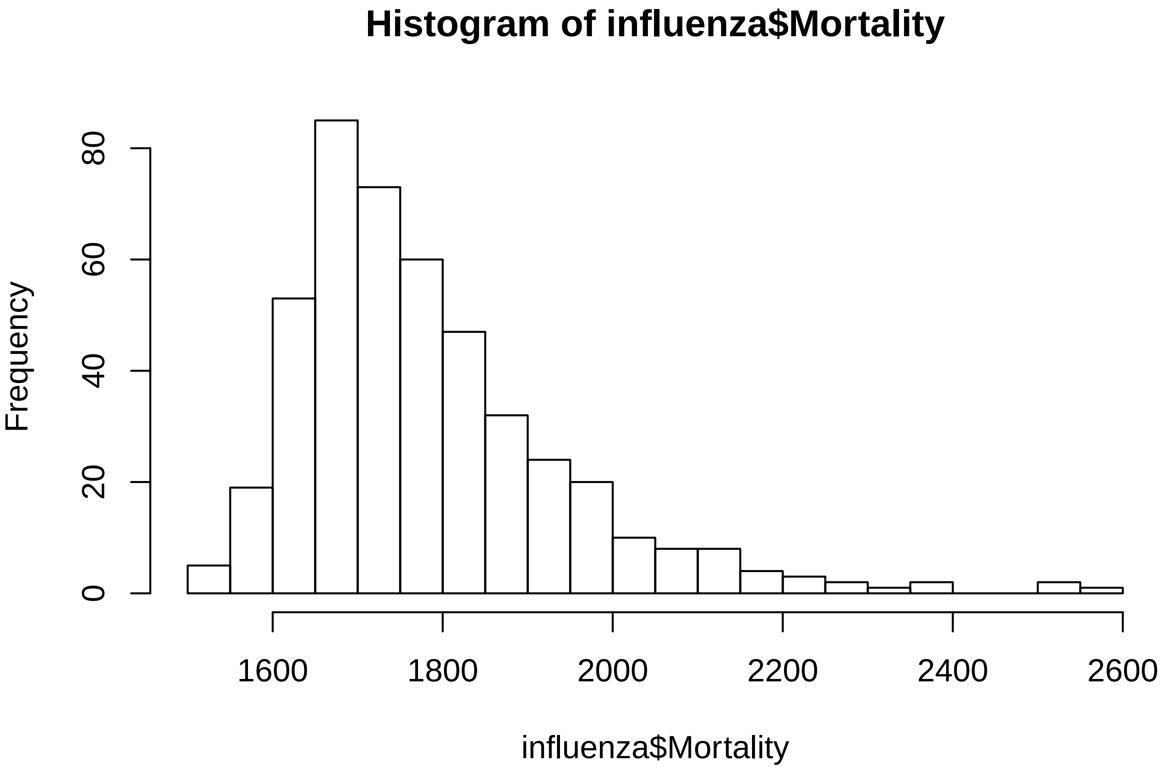
When looking at the plot of Mortality and Influenza cases over time, one can see a similarity in the patterns. When Influenze reaches a spike, so does the Mortality rate. From such a plot one is then tempted to argue that Influenza causes the mortality to rate to go up. Given that Influenza is a disease, I would say it is reasonable to argue that spikes in Influenza cases lead to spikes in the Mortality rate.

1.2

```r
library(mgcv)
```

```
## Loading required package: nlme
## This is mgcv 1.8-26. For overview type 'help("mgcv-package")'.
```

```
hist(influenza$Mortality, breaks = 20)
```



```
gam <- gam(Mortality ~ s(Week) + Year, data = influenza)
summary(gam)
```

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## Mortality ~ s(Week) + Year
##
## Parametric coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -652.058   3448.379  -0.189     0.85
## Year           1.219      1.725   0.706     0.48
##
## Approximate significance of smooth terms:
##           edf Ref.df     F              p-value
## s(Week) 8.587  8.951 100.6 <0.0000000000000002 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.661   Deviance explained = 66.8%
## GCV = 9014.6  Scale est. = 8806.7    n = 459
```

The default values of the function assumes a normal distribution and smoothing parameters are obtained using generalized cross validation. The underlying probabilistic model is:

$$Mortality = N(\mu, \sigma^2)$$

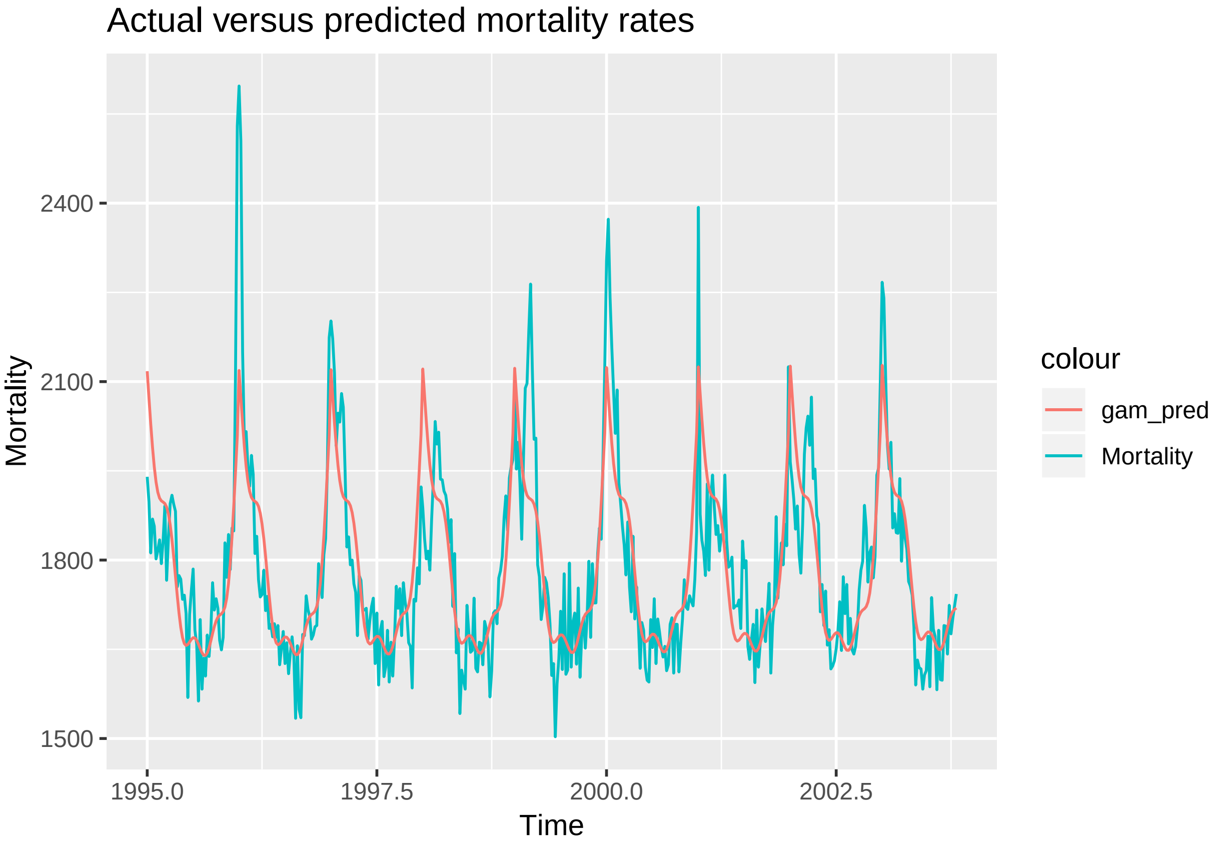
$$g(\mu) = Intercept + Beta_{year} * Year + s(Week)$$

In this situation g is a link function with a normal distribution.

1.3

```r
gam_pred <- predict.gam(gam, newdata = influenza)
influenza <- cbind(influenza, gam_pred)

plot_gam <- ggplot(data = influenza, aes(x = Time, y = Mortality, color = "Mortality")) +
  geom_line() +
  geom_line(aes(y = gam_pred, color = "gam_pred")) + ggtitle("Actual versus predicted mortality rates")
plot_gam
```

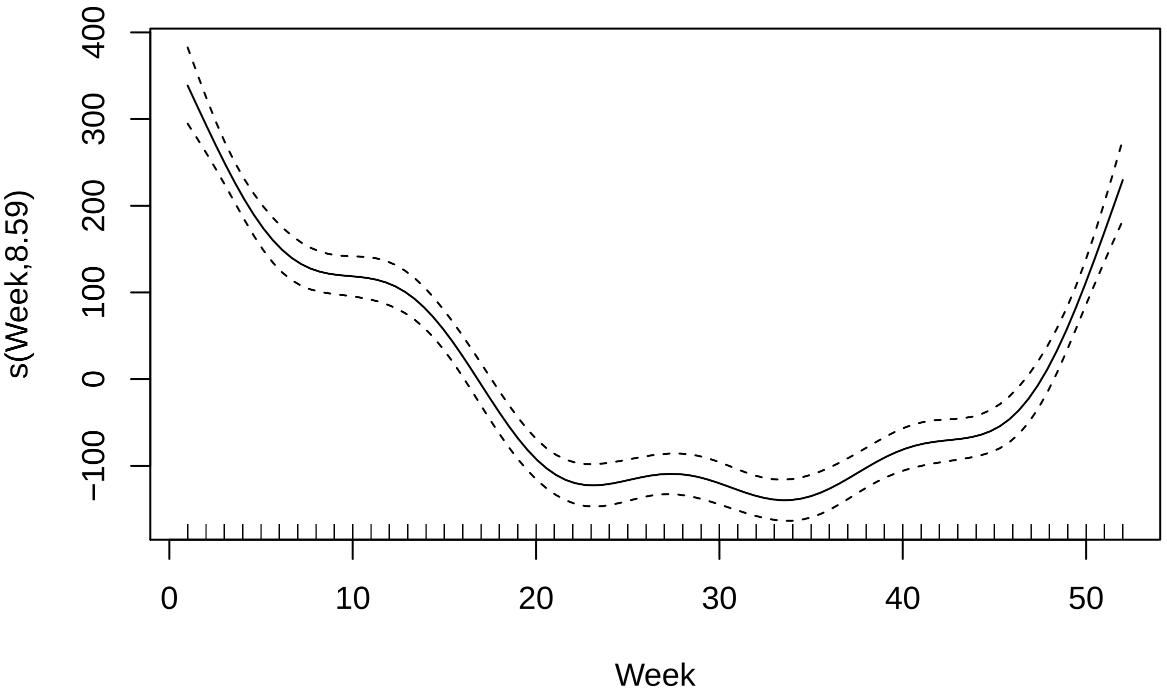


The predicted values for Mortality are shown in the red line, whereas actual values are shown in the blue line. The patterns of both line correspond, meaning the model estimates the dependent variable in a good way. Therefore I would say the fit is good. Still it has to be mentioned that the fitted values do not fully capture the extremes of the actual mortality rate.

Results from step 1.2 imply that the parametric coefficients are insignificantly different from zero, therefore we cannot assume the coefficients have an influence on the target variable. However, the smoothing terms result in a significant p value for the Week variable. Meaning, week has a significant influence on the target variable. Given the adjusted R-squared value, 66.1% of the variance is explained by this model.

The plot above show that Mortality rates peak each year. Therefore I would say there is not trend in mortality rate from one year to another. I would rather say, mortality rates show the same trend within each year, namely a peak at a certain time of the year.

```r
plot(gam)
```



The plot of the spline component shows how the response variable (Mortality) varies with the weeks of the year. Clearly, at the beginning and end of the year mortality rates are very much higher than in the middel of the year. When one thinks of this, this makes sense. Most likely will people suffer from influenzia in winter periods, thus the beginning and end of the calendar year, whereas in summer, the middle of the calendar year, people suffer less from influenzia, and thus less people die.

The curves in the shape is due to the fact that smoothing factors were implented in the model, and is due to non-linearity in the data. Dotted lines around the line represent standard errors of the fit.

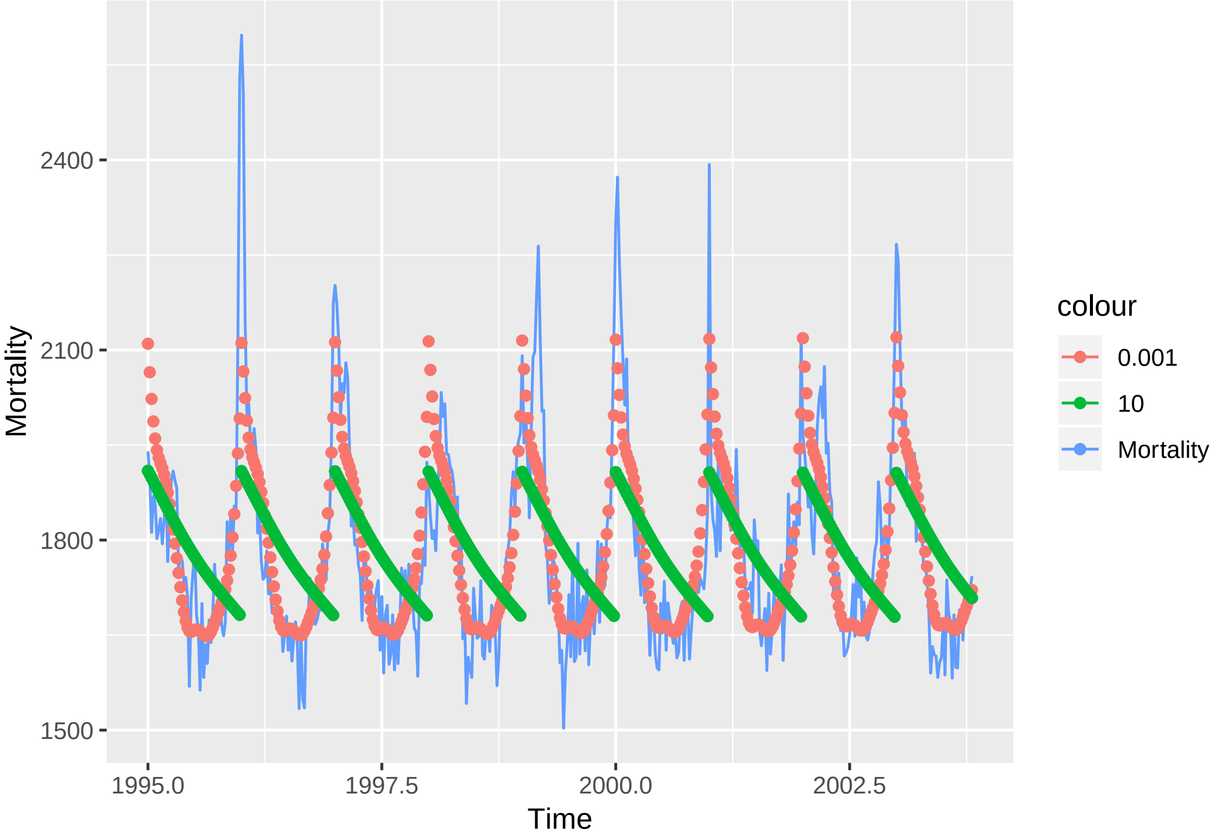
1.4

```r
gamma_df <- data.frame(influenza$Mortality, influenza$Time)
gammas <- c(0.001, 10)
j <- 3
k <- length(unique(influenza$Week))
gamma_list <- list()
x <- 1

for (i in gammas){
gamma <- gam(Mortality ~ s(Week, k = k, sp = i) + Year, data = influenza)
gamma_pred <- predict.gam(gamma, newdata = influenza)
gamma_df[, j] <- gamma_pred
j <- j + 1
gamma_list[[x]] <- gamma
x <- x+1
}
```

```
colnames(gamma_df) <- c("Mortality", "Time", "sp_0.001", "sp_10")
```

```
gammas <- ggplot(data = gamma_df, aes(x = Time, y = Mortality, color = "Mortality")) +
  geom_line() +
  geom_point(aes(y = sp_0.001, color = "0.001")) +
  geom_point(aes(y = sp_10, color = "10"))
gammas
```



```
lapply(gamma_list, summary)
```

```
## [[1]]
##
## Family: gaussian
## Link function: identity
##
## Formula:
## Mortality ~ s(Week, k = k, sp = i) + Year
##
## Parametric coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -745.837   3442.424  -0.217    0.829
## Year           1.265      1.722   0.735    0.463
##
## Approximate significance of smooth terms:
##           edf Ref.df     F             p-value
## s(Week) 8.333  10.41 85.18 <0.0000000000000002 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.662   Deviance explained = 66.9%
## GCV = 8979.7  Scale est. = 8777.6    n = 459
##
## [[2]]
##
## Family: gaussian
## Link function: identity
##
## Formula:
## Mortality ~ s(Week, k = k, sp = i) + Year
##
## Parametric coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2614.4022  5234.1496   0.499    0.618
## Year          -0.4155     2.6185  -0.159    0.874
##
## Approximate significance of smooth terms:
##           edf Ref.df     F             p-value
## s(Week) 1.069  1.136 106.1 <0.0000000000000002 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.217   Deviance explained = 22.1%
## GCV =  20478  Scale est. = 20341     n = 459
```

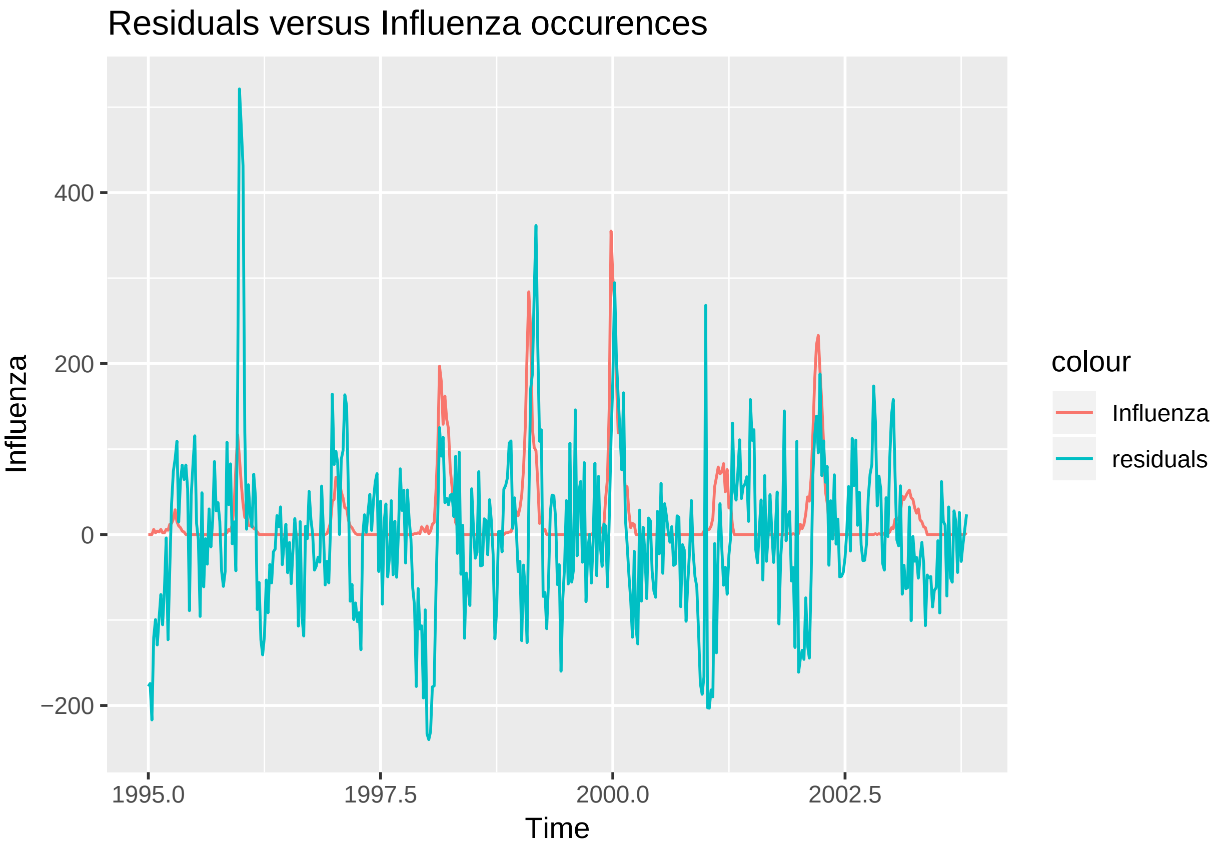
A gamma model with a small penalty factor results in more degrees of freedom and higher percentage of deviance explained than the gamma model with a high penalty factor. Therefore the penalty factor negatively relates to deviance and degrees of freedom. The fact that this relationship holds can be seen from the plot above, in which a penalty factor of 10 shows a severly worse fit to the data.

1.5

```r
residuals <- influenza$Mortality - gam_pred
df2 <- data.frame(cbind(influenza$Time, influenza$Influenza, residuals))
colnames(df2) <- c("Time", "Influenza", "residuals")

residuals_plot <- ggplot(data = df2, aes(x = Time, y = Influenza, color = "Influenza")) +
  geom_line() +
  geom_line(aes(y = residuals, color = "residuals")) + ggtitle("Residuals versus Influenza occurences")

residuals_plot
```



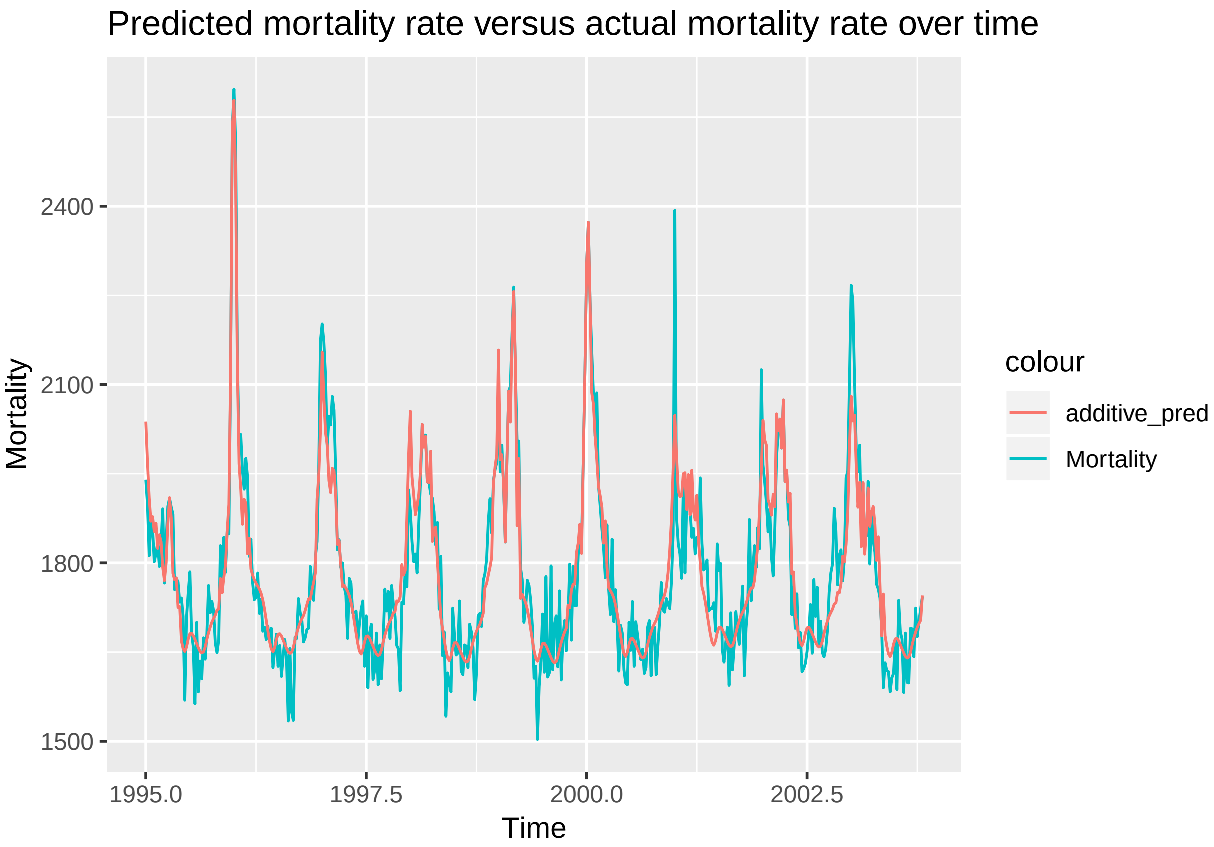
Some of the beaks in Influenza outbreaks correspond to peaks in the residuals of the fitted model. Still, however, a lot of variance in the residuals is not correlated to Influenza outbreaks. Therefore, I would say that the Influenza outbreaks are not correlated to the residuals.

1.6

```
additive_gam <- gam(Mortality ~ s(Year, k=length(unique(influenza$Year))) + s(Week, k=length(unique(infl

additive_pred <- predict.gam(additive_gam, newdata = influenza)
```

```
influenza <- cbind(influenza, additive_pred)
plot_additive <- ggplot(data = influenza, aes(x = Time, y = Mortality, color = "Mortality")) +
  geom_line() +
  geom_line(aes(y = additive_pred, color = "additive_pred")) + ggtitle("Predicted mortality rate versus
plot_additive
```



The additive GAM model clearly has the best fit. Much of the variance of the data is captured by the model, given the R-squared statistic of 0.819. Given that the GAM models in step 2 and step 4 do not include the influenza variable from the dataset, and the the model above does, one can say that most likely mortality is influenced by the outbreaks of influenza.

Assignment 2

2.1

```
library(readr)
data <- read.csv2(file = "data.csv", sep = ";", header = TRUE, fileEncoding = "ISO-8859-1")
data$Conference = as.factor(data$Conference)
n <- dim(data)[1]
set.seed(12345)
id <- sample(1:n, floor(n*0.7))
train <- data[id,]
test <- data[-id,]

rownames(train) <- 1:nrow(train)
x_train <- t(train[,-4703])
y_train <- train[[4703]]

rownames(test) <- 1:nrow(test)
x_test <- t(test[, -4703])
y_test <- test[[4703]]
```

```
library(pamr)
```

```
## Loading required package: cluster

## Loading required package: survival
```

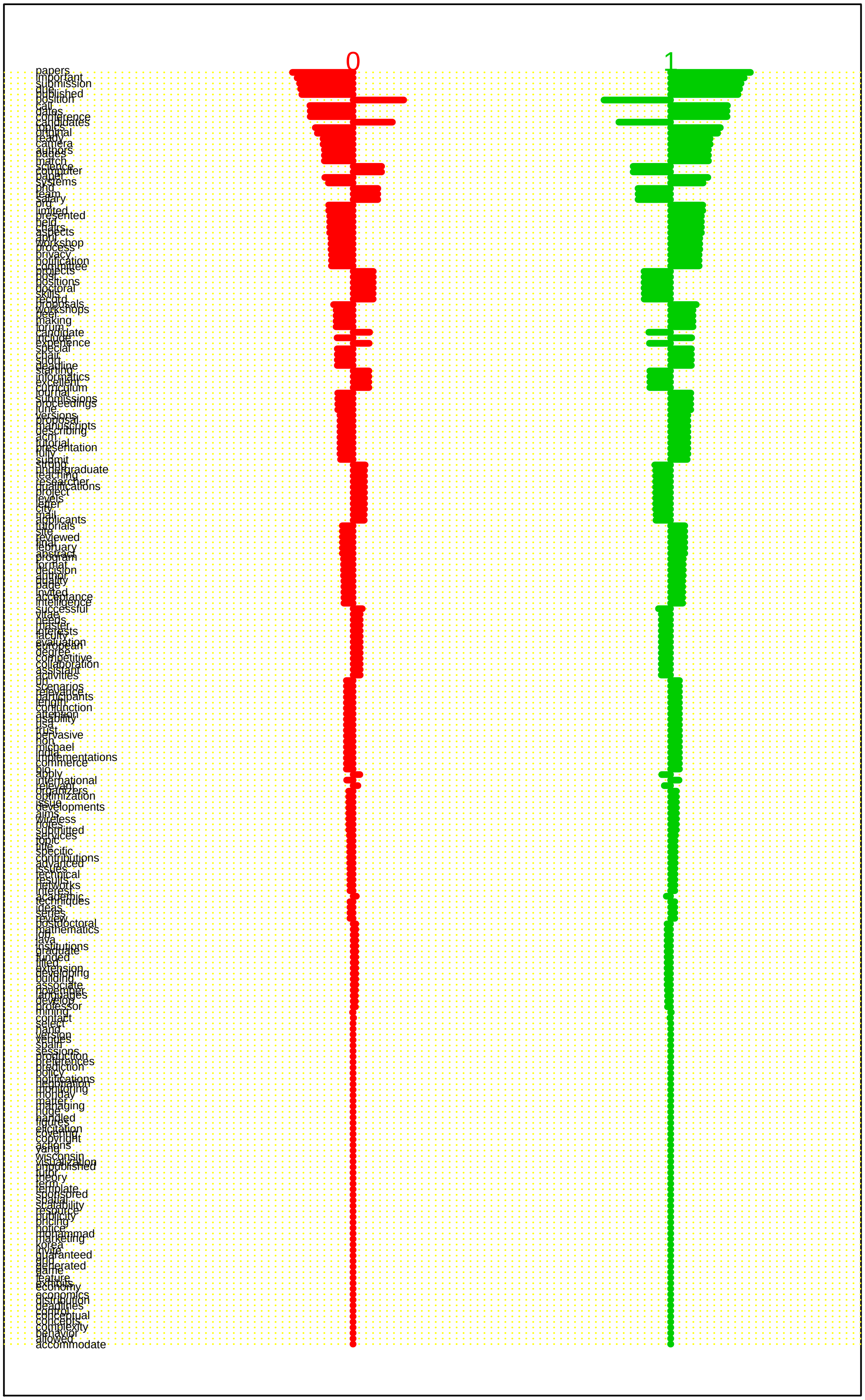
```
mydata_train <- list(x=x_train,y=as.factor(y_train),geneid=as.character(1:nrow(x_train)), genenames=rown
mydata_test <- list(x=x_test,y=as.factor(y_test),geneid=as.character(1:nrow(x_train)), genenames=rowname
model <- pamr.train(mydata_train,threshold=seq(0,4, 0.1))
```

```
## 1234567891011121314151617181920212223242526272829303132333435363738394041
```

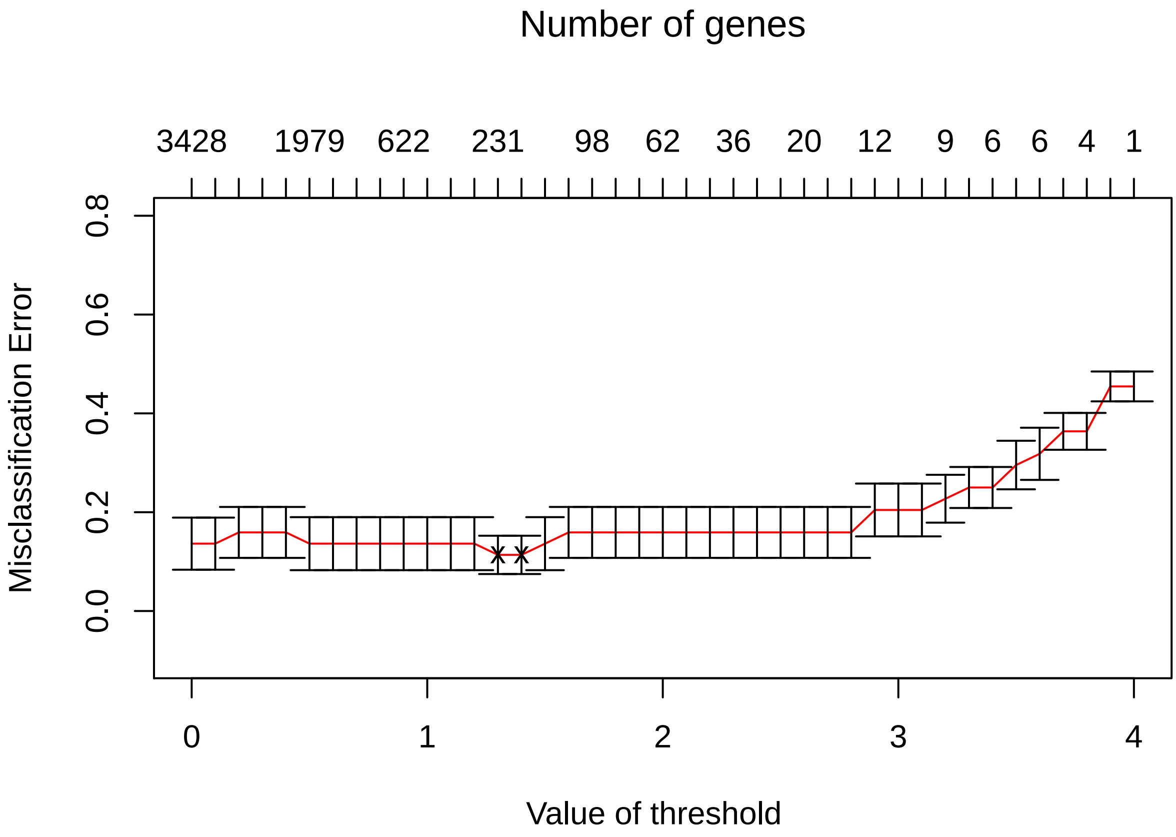
```
cvmodel <- pamr.cv(model, mydata_train)
```

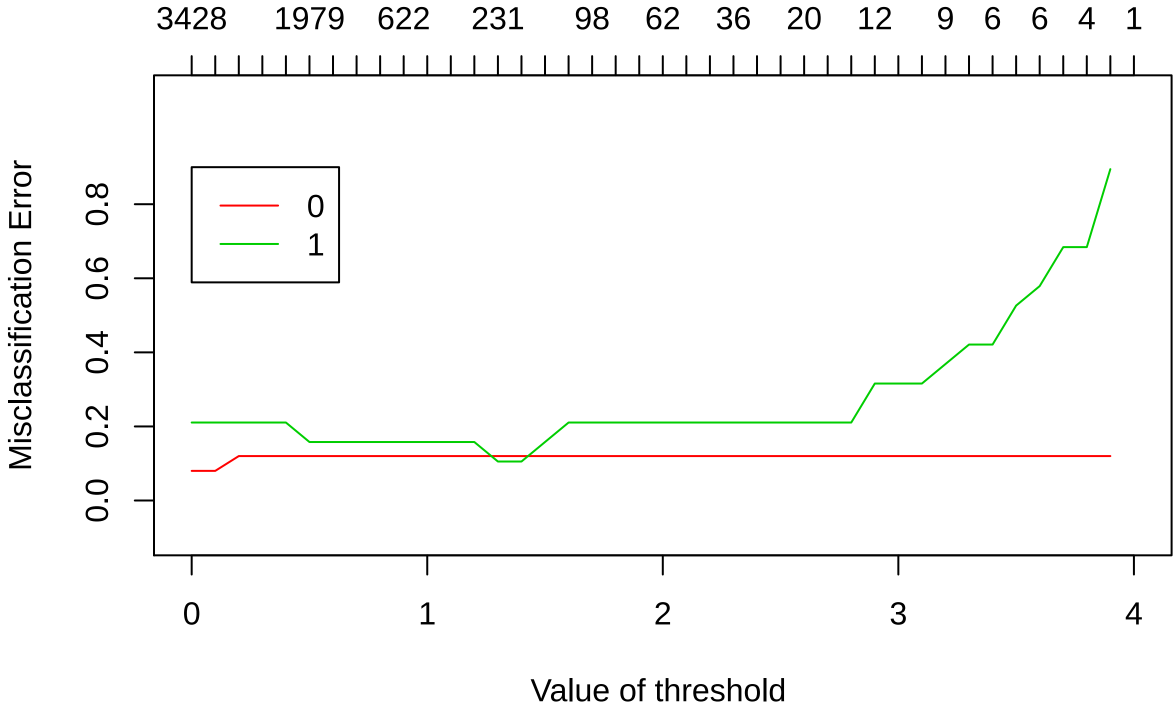
```
## 12Fold 1 :1234567891011121314151617181920212223242526272829303132333435363738394041
## Fold 2 :1234567891011121314151617181920212223242526272829303132333435363738394041
## Fold 3 :1234567891011121314151617181920212223242526272829303132333435363738394041
## Fold 4 :1234567891011121314151617181920212223242526272829303132333435363738394041
## Fold 5 :1234567891011121314151617181920212223242526272829303132333435363738394041
## Fold 6 :1234567891011121314151617181920212223242526272829303132333435363738394041
## Fold 7 :1234567891011121314151617181920212223242526272829303132333435363738394041
## Fold 8 :1234567891011121314151617181920212223242526272829303132333435363738394041
## Fold 9 :1234567891011121314151617181920212223242526272829303132333435363738394041
## Fold 10 :1234567891011121314151617181920212223242526272829303132333435363738394041
```

```
pamr.plotcen(model, mydata_train, threshold = 1.3)
```


0
1
papers
important
submission
due
published
position
call
dates
conference
candidates
topics
original
ready
camera
authors
pages
march
science
computer
paper
systems
phd
team
salary
org
limited
presented
held
chairs
aspects
april
workshop
process
privacy
notification
committee
projects
post
positions
doctoral
skills
record
proposals
workshops
peer
making
forum
candidate
include
experience
special
chair
short
deadline
starting
informatics
excellent
curriculum
journal
submissions
proceedings
june
versions
proposal
manuscripts
describing
acm
tutorial
presentation
fully
submit
strong
undergraduate
teaching
researcher
qualifications
project
levels
letter
city
mail
applicants
tutorials
site
reviewed
final
february
abstract
program
format
decision
author
quality
page
invited
acceptance
intelligence
successful
vitae
needs
master
interests
faculty
evaluation
european
degree
competitive
collaboration
assistant
activities
url
scenarios
relevance
participants
length
conjunction
attention
usability
usa
trust
pervasive
non
michael
india
implementations
commerce
bio
apply
international
relevant
organizers
optimization
issue
developments
aims
wireless
notes
submitted
services
topic
title
specific
contributions
advanced
issues
technical
results
networks
interest
academic
techniques
ideas
series
review
postdoctoral
mathematics
job
java
institutions
graduate
funded
filled
extension
developing
building
associate
november
languages
develop
professor
mining
contact
select
hand
version
venues
spain
sessions
production
preferences
prediction
policy
notifications
negotiation
monitoring
monday
matter
managing
huge
handled
figures
elicitation
covering
copyright
actions
yang
wisconsin
visualization
unpublished
tutor
theory
term
template
sponsored
spatial
scalability
resource
publicity
pricing
notice
mohammad
marketing
korea
invite
guaranteed
grid
generated
game
feature
exhibits
economy
economics
distribution
deadlines
control
conceptual
concepts
complexity
behavior
allowed
accommodate

```
pamr.plotcv(cvmodel)
```


Number of genes
3428
1979
622
231
98
62
36
20
12
9
6
6
4
1
Misclassification Error
0.0
0.2
0.4
0.6
0.8
0
1
2
3
4
Value of threshold


3428
1979
622
231
98
62
36
20
12
9
6
6
4
1
Misclassification Error
0.0
0.2
0.4
0.6
0.8
0
1
0
1
2
3
4
Value of threshold

```
predicted <- pamr.predict(model, newx = x_test, threshold = 1)
conf_matrix <- table(y_test, predicted)
conf_matrix
```

```
##       predicted
## y_test  0  1
##      0 10  0
##      1  2  8
```

```
test_error <- (conf_matrix[2,1] + conf_matrix[1,2])/nrow(test)
test_error
```

```
## [1] 0.1
```

2.2

```
library(glmnet)
```

```
## Loading required package: Matrix
```

```
## Loading required package: foreach
```

```
## Loaded glmnet 2.0-16
```

```
x_train <- as.matrix(train[,-4703])
y_train <- train[,4703]

x_test <- as.matrix(test[,-4703])
y_test <- test[,4703]

cv_elastic <- cv.glmnet(x=x_train, y=y_train, alpha = 0.5, family = "binomial")
predict_elastic <- predict.cv.glmnet(cv_elastic, newx = x_test, s = "lambda.min", type = "class")

elastic_conf <- table(y_test, predict_elastic)
elastic_error <- (elastic_conf[2,1] + elastic_conf[1,2]) / nrow(test)

coefficients <- coef(cv_elastic, s = "lambda.min")
variables <- data.frame(coefficients@Dimnames[[1]][coefficients@i + 1], coefficients@x)
colnames(variables) <- c("Variable", "Coefficient")

elastic_conf
```

```
##       predict_elastic
## y_test  0  1
##      0 10  0
##      1  2  8
```

```
elastic_error
```

```
## [1] 0.1
```

```
variables
```

```
##       Variable  Coefficient
## 1  (Intercept) -1.018931295
## 2    abstracts -0.301126433
## 3      aspects  0.073677580
## 4          bio  0.022876514
## 5         call  0.331990016
## 6   candidates -0.187831077
## 7     computer -0.283206491
## 8   conceptual  0.038084357
## 9   conference  0.196532966
## 10       dates  0.241663004
## 11         due  0.521172495
## 12  evaluation -0.179640082
## 13    exhibits  0.378269987
## 14   important  0.392427522
## 15   languages -0.025846994
## 16      making  0.189239367
## 17 manuscripts  0.032558442
## 18    original  0.055820470
## 19      papers  0.385380979
## 20        peer  0.096721108
## 21    position -0.375082994
## 22     process  0.001623837
## 23    projects -0.190407998
## 24   proposals  0.055355377
## 25   published  0.281820589
## 26     queries -0.300245879
## 27      record -0.116251400
## 28    relevant -0.113556406
## 29   scenarios  0.005346950
## 30     spatial  0.192500683
## 31  submission  0.280351935
## 32        team -0.129127761
## 33    versions  0.154574908
```

```
library(kernlab)
```

```
##
## Attaching package: 'kernlab'
```

```
## The following object is masked from 'package:ggplot2':
##
##     alpha
```

```
svm_model <-ksvm(x_train, y_train, kernel="vanilladot", scale = FALSE, type = "C-svc")
```

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

```
predicted_svm <- predict(svm_model, x_test, type="response")

svm_conf <- table(y_test, predicted_svm)
svm_error <- (svm_conf[2,1] + svm_conf[1,2]) / nrow(test)

coefficients_svm <- coef(svm_model)
coefficients_svm <- length(coefficients_svm[[1]])

svm_conf
```

```
##       predicted_svm
## y_test  0  1
##      0 10  0
##      1  1  9
```

```
svm_error
```

```
## [1] 0.05
```

```
coefficients_svm
```

```
## [1] 43
```

```
final_errors <- cbind("Error of Nearest Shrunken Centroid Model" = test_error, "Error of ElasticNet" =

knitr::kable(final_errors, caption = "Error terms of three different models")
```

Table 1: Error terms of three different models

Error of Nearest Shrunken Centroid Model	Error of ElasticNet	Error of Support Vector Machine
0.1	0.1	0.05

Comparing all models, the support vector machine model results in the lowest error. Therefore I would prefer this model.

2.3

```
p_value <- c()
for (i in 1:4702){
  x <- data[,i]
  res <- t.test(x ~ Conference, data = data, alternative = "two.sided")
  p <- res$p.value
  p_value[i] <- p
}
p_value <- as.data.frame(p_value)
p_value$reject_flag <- as.factor(ifelse(p_value$p_value <0.05, "Retain", "Drop"))
p_value$column_index <- row.names(p_value)

keep <- ifelse(p_value$reject_flag == "Retain", as.numeric(p_value$column_index), NA)
keep <- na.omit(keep)

rejected <- colnames(data[,keep])
rejected
```

```
##   [1] "abstract"        "academic"        "acceptance"
##   [4] "accepted"        "access"          "acm"
##   [7] "action"          "activities"      "advanced"
##  [10] "affirmative"     "agents"          "aims"
##  [13] "allowed"         "applicants"      "apply"
##  [16] "appointment"     "april"           "arrangements"
##  [19] "artificial"      "aspects"         "assistant"
##  [22] "associate"       "australia"       "author"
##  [25] "authors"         "background"      "beginning"
##  [28] "bio"             "call"            "calls"
##  [31] "camera"          "canada"          "candidate"
##  [34] "candidates"      "carry"           "chair"
##  [37] "chairs"          "chen"            "closing"
##  [40] "cloud"           "com"             "commerce"
##  [43] "commission"      "committee"       "communications"
##  [46] "company"         "competitive"     "computer"
##  [49] "concepts"        "conduct"         "conference"
##  [52] "conjunction"     "contact"         "contract"
##  [55] "contributions"   "copyright"       "covering"
##  [58] "cross"           "curriculum"      "date"
##  [61] "dates"           "david"           "deadline"
##  [64] "degree"          "degrees"         "describing"
##  [67] "desirable"       "detailed"        "develop"
##  [70] "developments"    "directly"        "doc"
##  [73] "doctoral"        "due"             "economics"
##  [76] "employer"        "english"         "equal"
##  [79] "equivalent"      "european"        "excellent"
##  [82] "expected"        "experience"      "extension"
##  [85] "feature"         "february"        "figures"
##  [88] "filled"          "final"           "format"
##  [91] "forum"           "foundation"      "fp7"
##  [94] "france"          "funded"          "fusion"
##  [97] "general"         "germany"         "graduate"
## [100] "green"           "grid"            "hand"
## [103] "handled"         "held"            "hiring"
## [106] "ideas"           "implementations" "important"
## [109] "include"         "included"        "india"
## [112] "infrastructures" "initially"       "institution"
## [115] "institutions"    "interest"        "interests"
## [118] "international"   "internet"        "invite"
## [121] "invited"         "issue"           "issues"
## [124] "japan"           "java"            "jin"
## [127] "job"             "jobs"            "journal"
## [130] "june"            "kevin"           "keynote"
## [133] "korea"           "largest"         "length"
## [136] "letter"          "levels"          "limited"
## [139] "liu"             "looking"         "madison"
## [142] "mail"            "making"          "manuscripts"
## [145] "march"           "master"          "mathematics"
## [148] "media"           "member"          "michael"
## [151] "mit"             "mobile"          "models"
## [154] "months"          "motivated"       "nanyang"
## [157] "networks"        "non"             "notes"
## [160] "notification"    "obtained"        "ongoing"
## [163] "ontologies"      "opportunity"     "optimization"
## [166] "org"             "organizers"      "organizing"
## [169] "original"        "page"            "pages"
## [172] "paper"           "papers"          "parallel"
## [175] "participants"    "participated"    "peer"
## [178] "phd"             "position"        "positions"
## [181] "post"            "postdoctoral"    "poster"
## [184] "posting"         "practitioners"   "presentation"
## [187] "presentations"   "presented"       "pricing"
## [190] "privacy"         "proceedings"     "process"
## [193] "professor"       "proficiency"     "program"
## [196] "programming"     "project"         "projects"
## [199] "proposal"        "proposals"       "protocols"
## [202] "proven"          "publicity"       "published"
## [205] "qualifications"  "ready"           "record"
## [208] "ref"             "relevance"       "relevant"
## [211] "researcher"      "resource"        "results"
## [214] "reviewed"        "reviewing"       "salary"
## [217] "scalability"     "scenarios"       "science"
## [220] "scope"           "security"        "series"
## [223] "services"        "sessions"        "share"
## [226] "short"           "site"            "skills"
## [229] "smart"           "spain"           "special"
## [232] "springer"        "start"           "starting"
## [235] "statement"       "strong"          "students"
## [238] "submission"      "submissions"     "submit"
## [241] "successful"      "supervision"     "systems"
## [244] "taiwan"          "takes"           "tasks"
## [247] "teaching"        "team"            "technical"
## [250] "template"        "tenure"          "term"
## [253] "thesis"          "title"           "top"
## [256] "topic"           "topics"          "tracks"
## [259] "trust"           "tutorial"        "tutorials"
## [262] "ubiquitous"      "undergraduate"   "universite"
## [265] "universities"    "university"      "unpublished"
## [268] "usa"             "usability"       "version"
## [271] "versions"        "vienna"          "visualization"
## [274] "vitae"           "wang"            "wireless"
## [277] "wisconsin"       "women"           "workshop"
## [280] "workshops"       "yang"
```

For all abovementioned features, according to Bejamini-Hochberg method, their p-values are lower than the threshold of 0.05. This means that all these features have a significant influence on the target variable.

Appendix

```
library(readxl)
options(scipen = 999)
influenza <- read_xlsx("influenza.xlsx")

library(ggplot2)
plot <- ggplot(data = influenza, aes(x = Time, y = Mortality, color = "Mortality")) +
  geom_line() +
  geom_line(aes(y = Influenza, color = "Influenza")) + ggtitle("Mortality and Influenza occurences over

plot
library(mgcv)
hist(influenza$Mortality, breaks = 20)
gam <- gam(Mortality ~ s(Week) + Year, data = influenza)
summary(gam)
gam_pred <- predict.gam(gam, newdata = influenza)
influenza <- cbind(influenza, gam_pred)

plot_gam <- ggplot(data = influenza, aes(x = Time, y = Mortality, color = "Mortality")) +
  geom_line() +
  geom_line(aes(y = gam_pred, color = "gam_pred")) + ggtitle("Actual versus predicted mortality rates")
plot_gam
plot(gam)
gamma_df <- data.frame(influenza$Mortality, influenza$Time)
gammas <- c(0.001, 10)
j <- 3
k <- length(unique(influenza$Week))
gamma_list <- list()
x <- 1

for (i in gammas){
gamma <- gam(Mortality ~ s(Week, k = k, sp = i) + Year, data = influenza)
gamma_pred <- predict.gam(gamma, newdata = influenza)
gamma_df[, j] <- gamma_pred
j <- j + 1
gamma_list[[x]] <- gamma
x <- x+1
}

colnames(gamma_df) <- c("Mortality", "Time", "sp_0.001", "sp_10")
gammas <- ggplot(data = gamma_df, aes(x = Time, y = Mortality, color = "Mortality")) +
  geom_line() +
  geom_point(aes(y = sp_0.001, color = "0.001")) +
  geom_point(aes(y = sp_10, color = "10"))
gammas
lapply(gamma_list, summary)
residuals <- influenza$Mortality - gam_pred
df2 <- data.frame(cbind(influenza$Time, influenza$Influenza, residuals))
colnames(df2) <- c("Time", "Influenza", "residuals")

residuals_plot <- ggplot(data = df2, aes(x = Time, y = Influenza, color = "Influenza")) +
  geom_line() +
  geom_line(aes(y = residuals, color = "residuals")) + ggtitle("Residuals versus Influenza occurences")

residuals_plot
additive_gam <- gam(Mortality ~ s(Year, k=length(unique(influenza$Year))) + s(Week, k=length(unique(infl

additive_pred <- predict.gam(additive_gam, newdata = influenza)
influenza <- cbind(influenza, additive_pred)
plot_additive <- ggplot(data = influenza, aes(x = Time, y = Mortality, color = "Mortality")) +
  geom_line() +
  geom_line(aes(y = additive_pred, color = "additive_pred")) + ggtitle("Predicted mortality rate versus
plot_additive
library(readr)
data <- read.csv2(file = "data.csv", sep = ";", header = TRUE, fileEncoding = "ISO-8859-1")
data$Conference = as.factor(data$Conference)
n <- dim(data)[1]
set.seed(12345)
id <- sample(1:n, floor(n*0.7))
train <- data[id,]
test <- data[-id,]

rownames(train) <- 1:nrow(train)
```

```r
x_train <- t(train[,-4703])
y_train <- train[[4703]]

rownames(test) <- 1:nrow(test)
x_test <- t(test[, -4703])
y_test <- test[[4703]]

library(pamr)
mydata_train <- list(x=x_train,y=as.factor(y_train),geneid=as.character(1:nrow(x_train)), genenames=rown
mydata_test <- list(x=x_test,y=as.factor(y_test),geneid=as.character(1:nrow(x_train)), genenames=rowname
model <- pamr.train(mydata_train,threshold=seq(0,4, 0.1))

cvmodel <- pamr.cv(model, mydata_train)
pamr.plotcen(model, mydata_train, threshold = 1.3)
pamr.plotcv(cvmodel)
predicted <- pamr.predict(model, newx = x_test, threshold = 1)
conf_matrix <- table(y_test, predicted)
conf_matrix
test_error <- (conf_matrix[2,1] + conf_matrix[1,2])/nrow(test)
test_error
library(glmnet)

x_train <- as.matrix(train[,-4703])
y_train <- train[,4703]

x_test <- as.matrix(test[,-4703])
y_test <- test[,4703]

cv_elastic <- cv.glmnet(x=x_train, y=y_train, alpha = 0.5, family = "binomial")
predict_elastic <- predict.cv.glmnet(cv_elastic, newx = x_test, s = "lambda.min", type = "class")

elastic_conf <- table(y_test, predict_elastic)
elastic_error <- (elastic_conf[2,1] + elastic_conf[1,2]) / nrow(test)

coefficients <- coef(cv_elastic, s = "lambda.min")
variables <- data.frame(coefficients@Dimnames[[1]][coefficients@i + 1], coefficients@x)
colnames(variables) <- c("Variable", "Coefficient")

elastic_conf
elastic_error
variables
library(kernlab)

svm_model <-ksvm(x_train, y_train, kernel="vanilladot", scale = FALSE, type = "C-svc")
predicted_svm <- predict(svm_model, x_test, type="response")

svm_conf <- table(y_test, predicted_svm)
svm_error <- (svm_conf[2,1] + svm_conf[1,2]) / nrow(test)

coefficients_svm <- coef(svm_model)
coefficients_svm <- length(coefficients_svm[[1]])

svm_conf
```

```r
svm_error
coefficients_svm
final_errors <- cbind("Error of Nearest Shrunken Centroid Model" = test_error, "Error of ElasticNet" = 

knitr::kable(final_errors, caption = "Error terms of three different models")
p_value <- c()
for (i in 1:4702){
  x <- data[,i]
  res <- t.test(x ~ Conference, data = data, alternative = "two.sided")
  p <- res$p.value
  p_value[i] <- p
}
p_value <- as.data.frame(p_value)
p_value$reject_flag <- as.factor(ifelse(p_value$p_value <0.05, "Retain", "Drop"))
p_value$column_index <- row.names(p_value)

keep <- ifelse(p_value$reject_flag == "Retain", as.numeric(p_value$column_index), NA)
keep <- na.omit(keep)

rejected <- colnames(data[,keep])
rejected
```