

Machine Learning - Block 2 Assignment 2

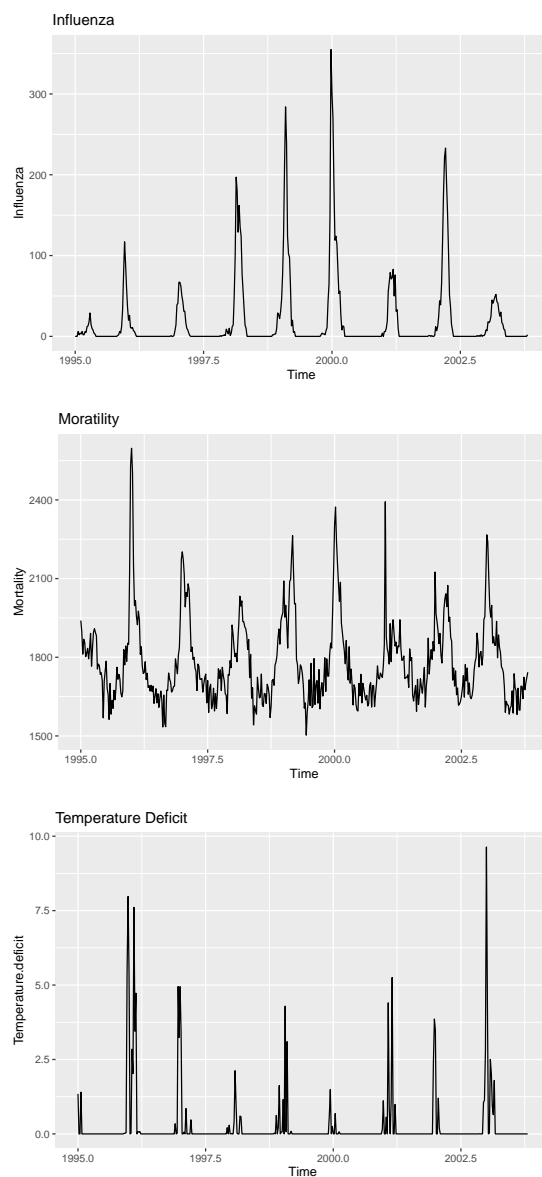
Agustín Valencia

12/9/2019

Assignment 1. Using GAM and GLM to examine mortality rates

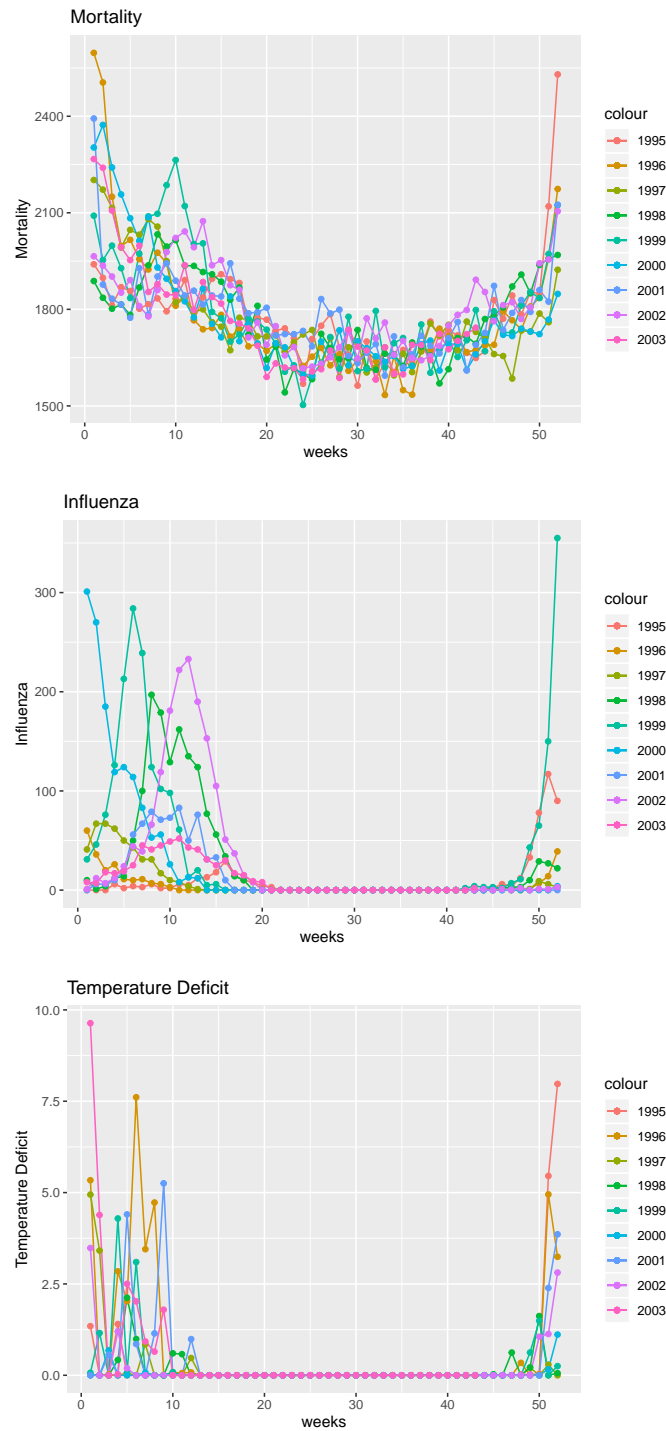
The Excel document `influenza.xlsx` contains weekly data on the mortality and the number of laboratory-confirmed cases of influenza in Sweden. In addition, there is information about population-weighted temperature anomalies (temperature deficits).

1. Use time series plots to visually inspect how the mortality and influenza number vary with time (use Time as X axis). By using this plot, comment how the amounts of influenza cases are related to mortality rates.



It seems to be a modality in the data. It can be seen that Influenza peaks seems to have a correlation with peaks in mortality and temperature deficit

As the model is meant to learn from weeks variations along the year, it is better to separate the data into several time series per year and see how the data looks now.



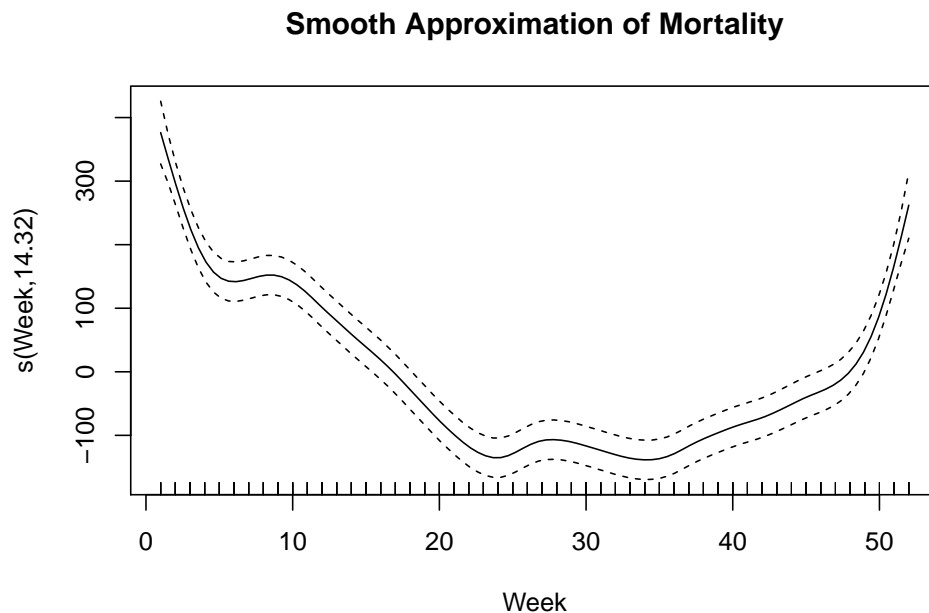
Now its quite clear that the data shows that during winter weeks Influenza cases and Mortality related to increases, also the temperature deficit seems to follow that trend.

2. Use `gam()` function from `mgcv` package to fit a GAM model in which Mortality is normally distributed and modelled as a linear function of Year and spline function of Week, and make sure that the model parameters are selected by the generalized cross-validation. Report the underlying probabilistic model.

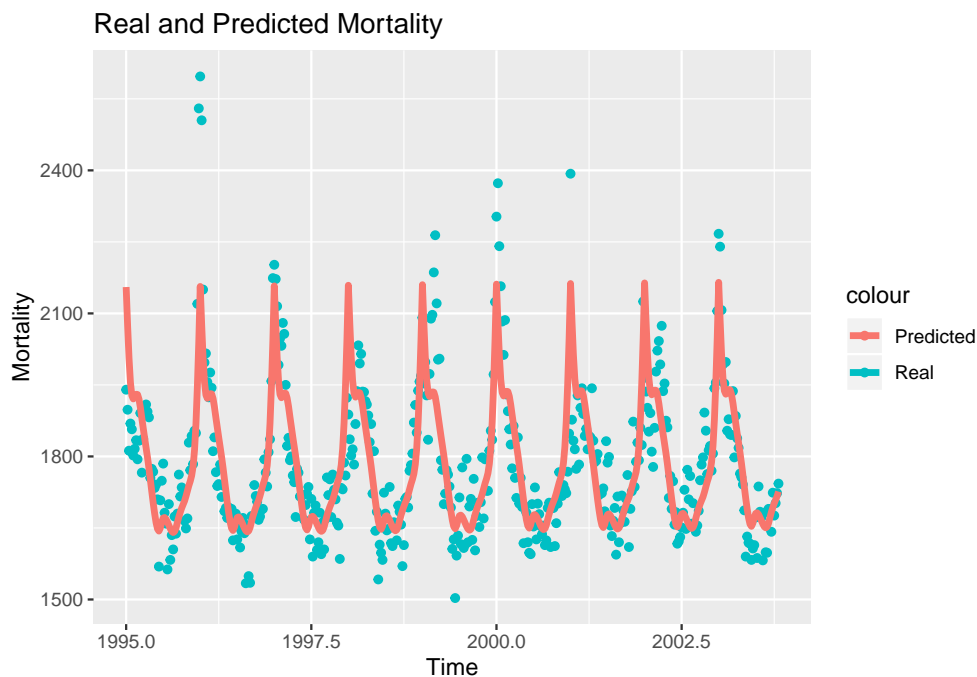
The summary of the obtained probabilistic model is:

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## Mortality ~ Year + s(Week, k = length(unique(data$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
```

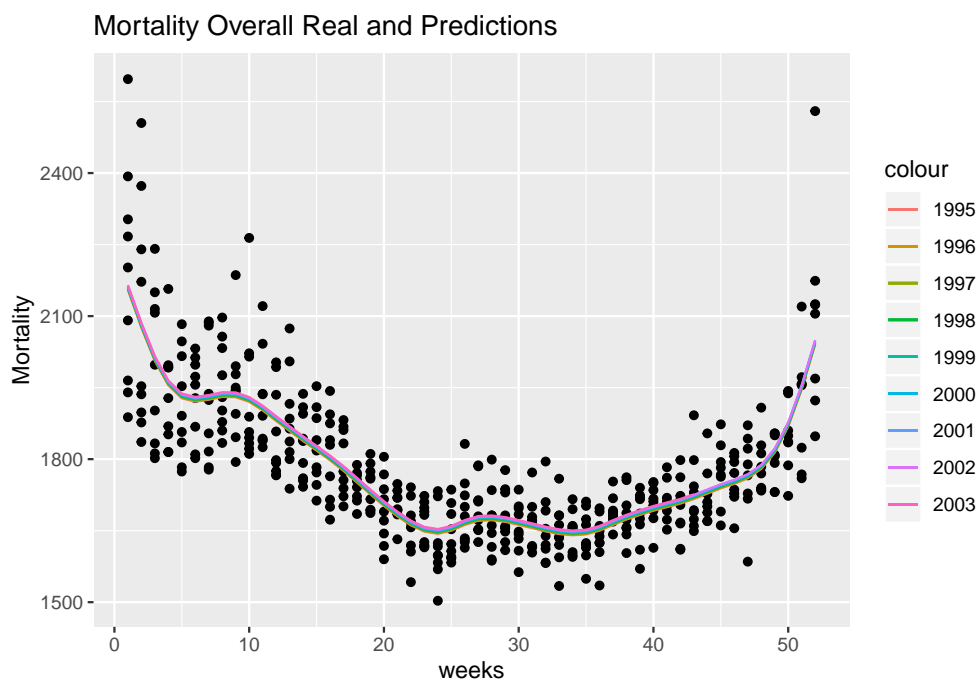
The smoothed curve approximated by the GAM model is:



- Plot predicted and observed mortality against time for the fitted model and comment on the quality of the fit. Investigate the output of the GAM model and report which terms appear to be significant in the model. Is there a trend in mortality change from one year to another? Plot the spline component and interpret the plot.



Although it seems to be a fair general approximation it seems to be missing some abnormal positive and negative peaks.



It can be seen that given the variability of mortality per week each year is the GAM fit acceptably well, though still a good predictor.

4. Examine how the penalty factor of the spline function in the GAM model from step 2 influences the estimated deviance of the model. Make plots of the predicted and observed mortality against time for cases of very high and very low penalty factors. What is the relation of the penalty factor to the degrees of freedom? Do your results confirm this relationship?
5. Use the model obtained in step 2 and plot the residuals and the influenza values against time (in one plot). Is the temporal pattern in the residuals correlated to the outbreaks of influenza?
6. Fit a 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 cases of influenza. Use the output of this GAM function to conclude whether or not the mortality is influenced by the outbreaks of influenza. Provide the plot of the original and fitted Mortality against Time and comment whether the model seems to be better than the previous GAM models.

Assignment 2. High-dimensional methods

The data file `data.csv` contains information about 64 e-mails which were manually collected from DBWorld mailing list. They were classified as: ‘announces of conferences’ (1) and ‘everything else’ (0) (variable `Conference`)

1. Divide data into training and test sets (70/30) without scaling. Perform nearest shrunken centroid classification of training data in which the threshold is chosen by cross-validation. Provide a centroid plot and interpret it. How many features were selected by the method? List the names of the 10 most contributing features and comment whether it is reasonable that they have strong effect on the discrimination between the conference mails and other mails? Report the test error.
2. Compute the test error and the number of the contributing features for the following methods fitted to the training data:
 - a. Elastic net with the binomial response and $\alpha = 0.5$ in which penalty is selected by the cross-validation
 - b. Support vector machine with “vanilladot” kernel.

Compare the results of these models with the results of the nearest shrunken centroids (make a comparative table). Which model would you prefer and why?

3. Implement Benjamini-Hochberg method for the original data, and use `t.test()` for computing p-values. Which features correspond to the rejected hypotheses? Interpret the result.

Appendix A : Code

```
knitr::opts_chunk$set(echo = FALSE)
library(openxlsx)
library(ggplot2)
library(mgcv)
library(akima)
library(plotly)
set.seed(12345)
#####
##                               Assignment 1
#####

# Import data
dataPath <- "data/influenza.xlsx"
data <- read.xlsx(dataPath)

# Time Series plotting
mortPlot <- ggplot(data) +
  geom_line(aes(x=Time, y=Mortality), color="black") + ggtitle("Moratality")
infPlot <- ggplot(data) +
  geom_line(aes(x=Time, y=Influenza), color="black") + ggtitle("Influenza")
tempPlot <- ggplot(data) +
  geom_line(aes(x=Time, y=Temperature.deficit), color="black") +
  ggtitle("Temperature Deficit")
infPlot
mortPlot
tempPlot

# Time series per year
years <- unique(data$Year)
weeks <- unique(data$Week)
mortData <- list()
infData <- list()
tempData <- list()
for(i in 1:length(years)) {
  year <- years[i]
  mortData[[i]] <- data$Mortality[which(data$Year == year)]
  infData[[i]] <- data$Influenza[which(data$Year == year)]
  tempData[[i]] <- data$Temperature.deficit[which(data$Year == year)]
}
names(mortData) <- years
names(infData) <- years
names(tempData) <- years

# create data.frames for ggplot
plotData <- function(d, title) {
  shortWeeks <- 1:length(d$'2003')
  p <- ggplot() +
    geom_line(aes(x=weeks, y=d$'1995', color="1995")) +
    geom_line(aes(x=weeks, y=d$'1996', color="1996")) +
    geom_line(aes(x=weeks, y=d$'1997', color="1997")) +
    geom_line(aes(x=weeks, y=d$'1998', color="1998")) +
```

```

      geom_line(aes(x=weeks, y=d$'1999', color="1999")) +
      geom_line(aes(x=weeks, y=d$'2000', color="2000")) +
      geom_line(aes(x=weeks, y=d$'2001', color="2001")) +
      geom_line(aes(x=weeks, y=d$'2002', color="2002")) +
      geom_line(aes(x=shortWeeks, y=d$'2003', color="2003")) +
      geom_point(aes(x=weeks, y=d$'1995', color="1995")) +
      geom_point(aes(x=weeks, y=d$'1996', color="1996")) +
      geom_point(aes(x=weeks, y=d$'1997', color="1997")) +
      geom_point(aes(x=weeks, y=d$'1998', color="1998")) +
      geom_point(aes(x=weeks, y=d$'1999', color="1999")) +
      geom_point(aes(x=weeks, y=d$'2000', color="2000")) +
      geom_point(aes(x=weeks, y=d$'2001', color="2001")) +
      geom_point(aes(x=weeks, y=d$'2002', color="2002")) +
      geom_point(aes(x=shortWeeks, y=d$'2003', color="2003")) +
      ggtitle(title) + ylab(title)
    return(p)
  }
mortPlot <- plotData(mortData, "Mortality")
infPlot <- plotData(infData, "Influenza")
tempPlot <- plotData(tempData, "Temperature Deficit")
mortPlot
infPlot
tempPlot
model <- gam (
  Mortality ~ Year +
    s(Week, k = length(unique(data$Week))),
  data=data,
  method = "GCV.Cp"
)
summary(model)
plot(model, main="Smooth Approximation of Mortality")

# Predictions
predictions <- predict(model, data)
ggplot(data) +
  geom_point(aes(x=Time, y=Mortality, color="Real")) +
  geom_line(aes(x=Time, y=predictions, color="Predicted"), size=1.5) +
  ggtitle("Real and Predicted Mortality")
plotAll <- function(d, p, title) {
  shortWeeks <- 1:length(d$'2003')
  a <- ggplot() +
    geom_point(aes(x=weeks, y=d$'1995')) +
    geom_point(aes(x=weeks, y=d$'1996')) +
    geom_point(aes(x=weeks, y=d$'1997')) +
    geom_point(aes(x=weeks, y=d$'1998')) +
    geom_point(aes(x=weeks, y=d$'1999')) +
    geom_point(aes(x=weeks, y=d$'2000')) +
    geom_point(aes(x=weeks, y=d$'2001')) +
    geom_point(aes(x=weeks, y=d$'2002')) +
    geom_point(aes(x=shortWeeks, y=d$'2003')) +
    geom_line(aes(x=weeks, y=p$'1995', color="1995")) +
    geom_line(aes(x=weeks, y=p$'1996', color="1996")) +
    geom_line(aes(x=weeks, y=p$'1997', color="1997")) +

```

```

    geom_line(aes(x=weeks, y=p$'1998', color="1998")) +
    geom_line(aes(x=weeks, y=p$'1999', color="1999")) +
    geom_line(aes(x=weeks, y=p$'2000', color="2000")) +
    geom_line(aes(x=weeks, y=p$'2001', color="2001")) +
    geom_line(aes(x=weeks, y=p$'2002', color="2002")) +
    geom_line(aes(x=shortWeeks, y=p$'2003', color="2003")) +
    ggtitle(title)
  return(a)
}

preds <- list()
for (i in 1:length(years)) {
  year <- years[i]
  d <- data[which(data$Year == year),]
  preds[[i]] <- predict(model, d)
}
names(preds) <- years
p <- plotAll(mortData, preds, "Mortality Overall Real and Predictions")
p <- p + ylab("Mortality")
p
dataPath <- "data/data.csv"
data <- read.csv2(dataPath)

```