Introduction to Machine Learning - Lab 7

Gustav Sternelöv

Friday, November 27, 2015

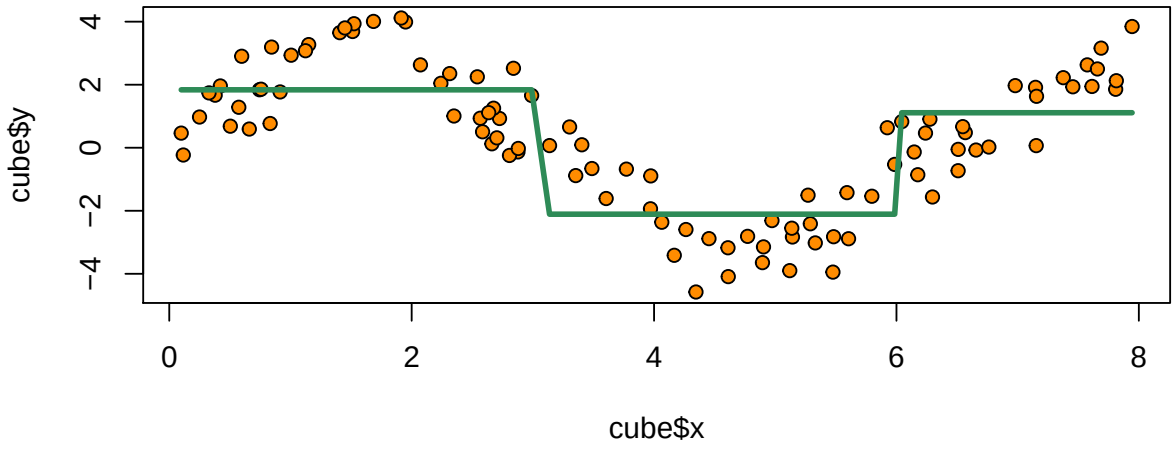
Assignment 1

1.1

In *1.1* a function that conducts a piecewise constant basis expansion is implemented. The function takes the argument *data*, where the first column is supposed to be the input vector and the second column the response vector, and the argument k which is the values for the knots. Returned by the function is a graph showing the results of the conducted expansion.

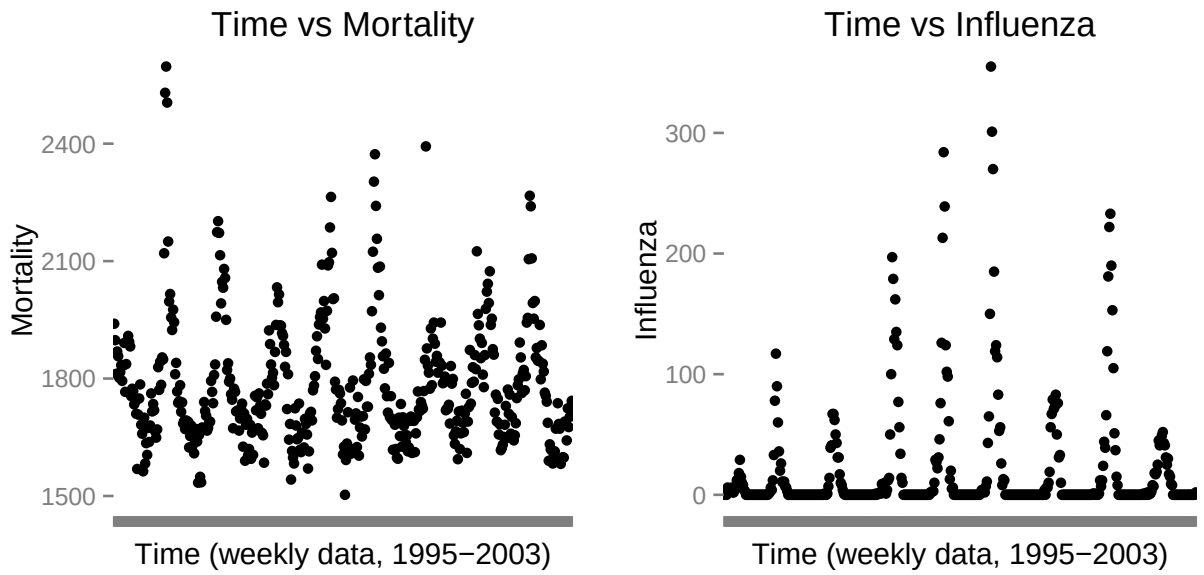
1.2

The created function is now tested on a dataset named *cube* which has two variables, x and y. Two knots are used, at x=3 and at x=6.
The result that is obtained by running the function on this data set is shown below.



Assignment 2

2.1



The graph over time versus mortality seem to follow a seasonal pattern. A closer look gives that the mortality is at its lowest during the summer and at its highest levels during the winter. This pattern is almost identical from year to year.

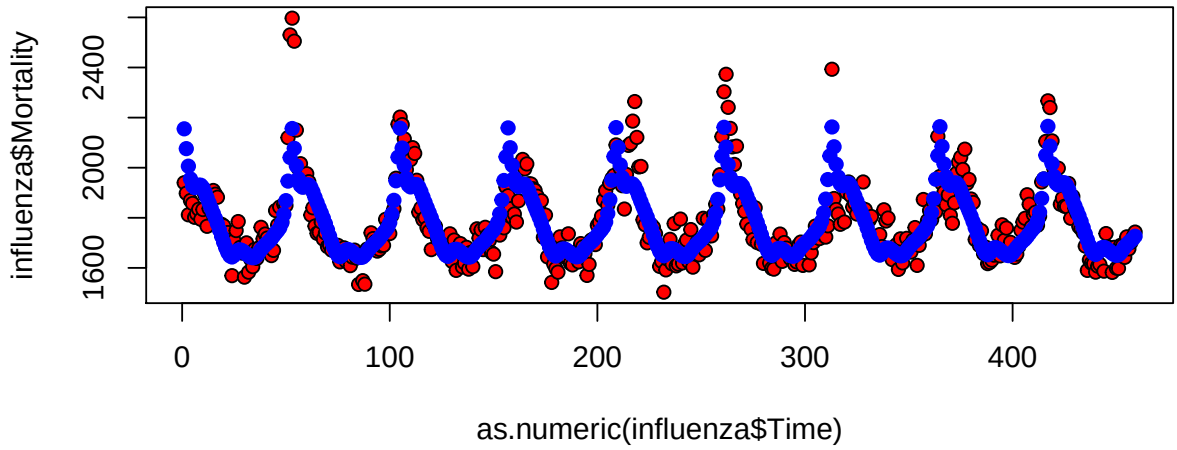
Regarding the graph over time versus influenza it can be seen the values are zero or very close to zero for the majority of the weeks during a year. It is during the coldest weeks of the winter the number of influenza cases rises. How many that is hit by the influenza is varying from year to year.

2.2

The probabilistic expression for the model fitted in *2.2* is:
E(Mortality) = -680.657 + 1.233 * Year + s(Week)

2.3

The predicted mortality by the fitted model in *2.2*, blue dots, is compared against the observed mortality, red dots.

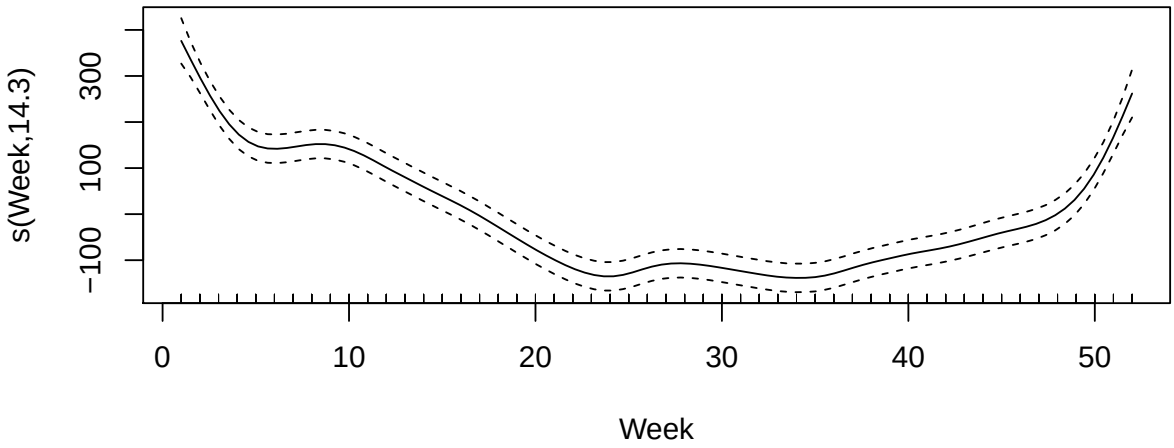


It seems to be a decent fit since the fitted values follows the general pattern for the mortality quite well. Although it also is a problem that the model is very general. A more adequate model would be one that is more flexible and has the ability to capture local changes in data.

Next the output of the GAM model is investigated. According to the output the parametric coefficient for the variable Year and the intercept is insignificant and the non-parametric coefficient for the variable week is significant. Hence, there is no significant trend saying that the mortality rate changes from one year to another.

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## Mortality ~ Year + s(Week, k = 51)
##
## Parametric coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -680.657   3367.966  -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) 14.3  17.85 53.91  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.677   Deviance explained = 68.8%
## GCV = 8709.3  Scale est. = 8399.9    n = 459
```

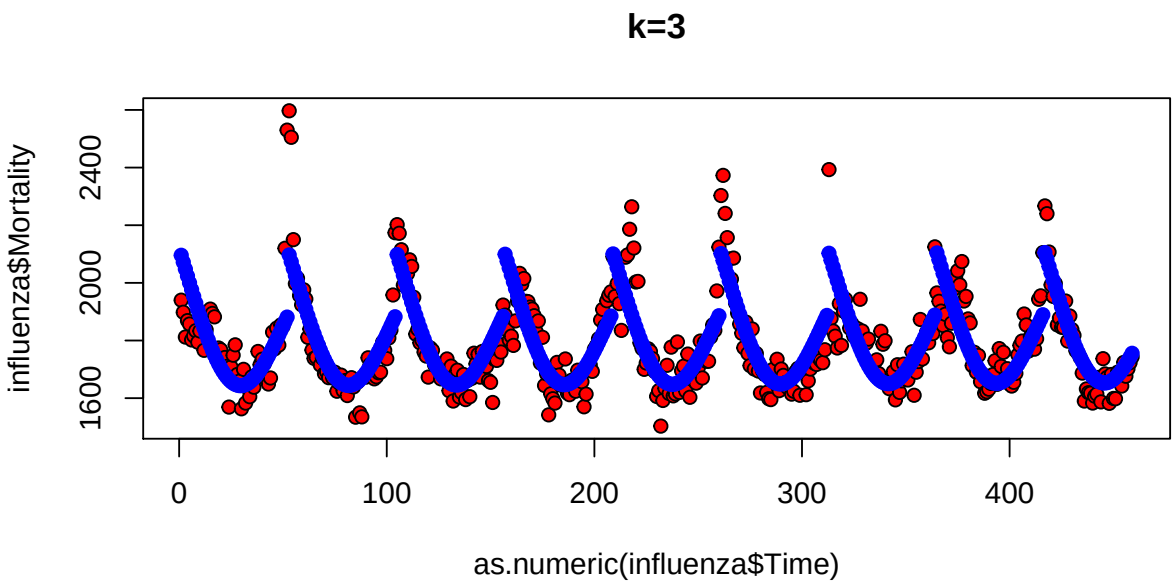
The spline component for the variable Week is visualized with the following plot.

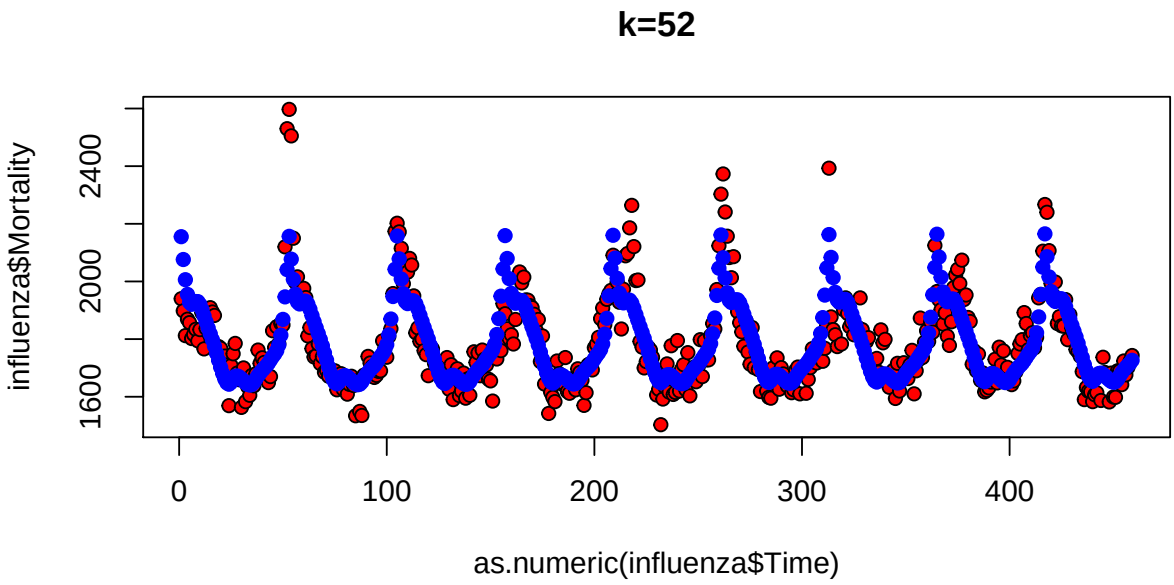


The plot over the spline component shows how this component affects the fitted values given by the model. For weeks at the start and at the end of the year it raises the mortality rate and during mid-year it lowers the mortality rate.

2.4

Two different adjustments are done for the model created in *2.2*. In the first case the number of knots for the spline function is set to 3 and in the second case it is set to 52. A comparison of the fitted values with the observed values is given by the plots below.



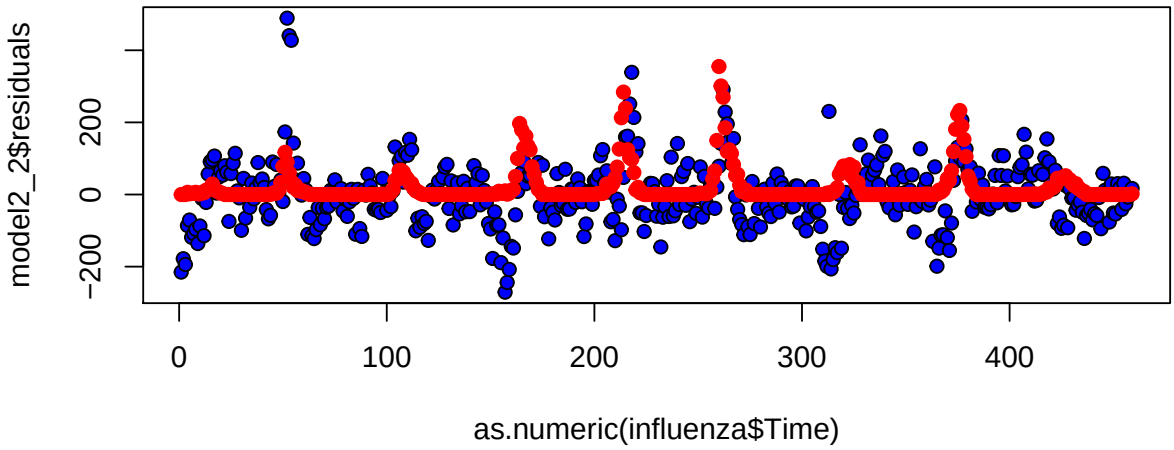


When the number of knots is low the fitted values almost looks exactly the same from year to year. For a higher number of knots the model becomes more flexible and a bit more of the yearly variation is captured.

The relation between the number of knots and the degrees of freedom is that the number of degress of freedom is higher for a higher number of knots. For the model with k equal to 52 the degress of freedom is 17.87 and for the model qith k equal to 3 it is 2.36. This is because a model with a higher number of knots uses more coefficients than a model with just 3 knots.

2.5

The residuals for the model created in *2.2* and the observed influenza values are plotted against time.



The residuals seem to be correlated to the outbreaks of influenza. This correlation indicates that the variable Influenza might be useful when trying to find a model that well describes the mortality rates.

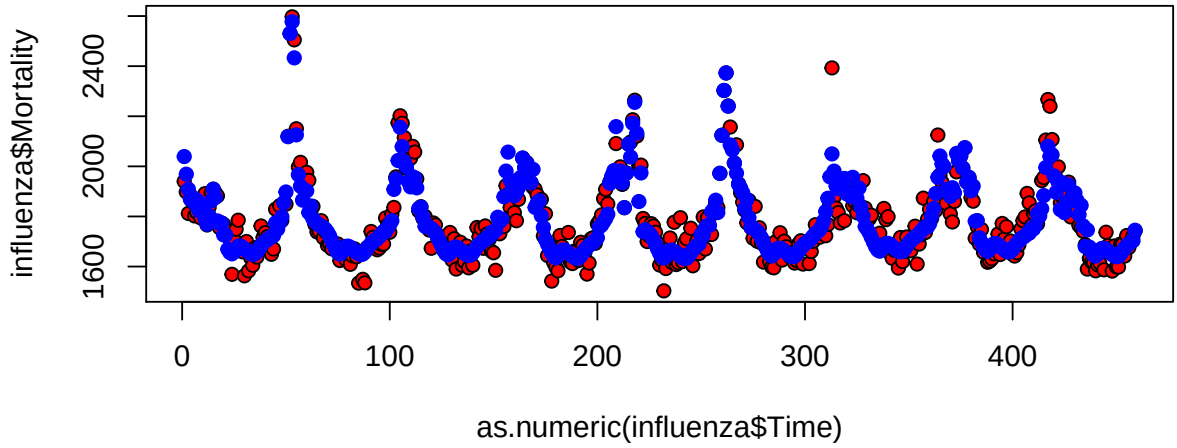
2.6

Next, a GAM model is fitted where mortality is described by spline functions of Year, Week and the number of confirmed cases of influenza. For this model the intercept is the only parametric coefficient, all the others are non-parametric coefficients. The coefficients for both Week and outbreaks of influenza are significant, and the coefficient for Year is insignificant. Regarding the outbreaks of influenza it seems like this variable influences the mortality rate.

```
##
## Family: gaussian
## Link function: identity
##
## Formula:
## Mortality ~ s(Week, k = 52) + s(Year, k = 9) + s(Influenza, k = 85)
##
## 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(Week)      14.641 18.248 18.516  <2e-16 ***
## s(Year)       4.663  5.677  1.487   0.184
## s(Influenza) 69.735 72.840  5.593  <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
```

The fitted values for the created model is compared against the observed valuse for mortality.



Compared to the model created in *2.2* this model is capable to capture more than the week to week variation. As can be seen in the plot above this model is more flexible and it manages to find at least some of the yearly variation.

The SSE for the model:

```
## [1] 1734020
```

Appendix

R-code

```
# Assignment 1
# 1.2
PiecewiseCBS <- function(data, k){
  y <- matrix(0, ncol=1, nrow=length(data[,1]))
  for (j in 1:length(data[,1])){
    if(data[j,1] < k[1]){
      y[j] <-  mean(subset(data, data[,1] < k[1])[,2])
    }else{y[j] <- y[j]  }
  }

  for (i in 2:length(k)){
    for (j in 1:length(data[,1])){

      if(k[i] > k[1] & data[j,1] < k[i] & data[j,1] > k[i-1] ){
        y[j] <- mean(subset(data, data[,1] >= k[i-1] & data$x < k[i])[,2])
```

```
      }else{y[j] <- y[j]  }

      if(data[j,1] >= k[length(k)]){
        y[j] <- mean(subset(data, data[,1] > k[i])[,2])
      }else{y[j] <- y[j]  }}}

  YhY <- cbind(data[,2], y, data[,1])
  YhY <- YhY[order(YhY[,3]),]

  plot(x=cube$x, y=cube$y, pch=21, bg="darkorange")
  points(x=YhY[,3], y=YhY[,2], col="seagreen", type="l", lwd=3)
}
cube <- read.csv("C:/Users/Gustav/Documents/Machine-Learning/Lab 7/cube.csv", sep=";", header=TRUE)

testData <- data.frame(x=cube$x, y =cube$y)
PiecewiseCBS(data = testData, k = c(3,6))
influenza <- read.csv("C:/Users/Gustav/Documents/Machine-Learning/Lab 7/influenza.csv", sep=";", header=
# 2.1
library(ggplot2)
library(gridExtra)
# time versus mortality
Mort <- ggplot(influenza, aes(x=Time, y=Mortality),) + geom_point() +
  theme(axis.text.x = element_blank()) + labs(x = "Time (weekly data, 1995-2003)")+ ggtitle("Time vs Mo
# time versus influenza
Infl <- ggplot(influenza, aes(x=Time, y=Influenza),) + geom_point() +
  theme(axis.text.x = element_blank()) + labs(x = "Time (weekly data, 1995-2003)")+ ggtitle("Time vs In
grid.arrange(Mort, Infl, ncol=2)
library(mgcv)
model2_2 <- gam(formula=Mortality ~ Year + s(Week, k=51), data=influenza)
# Evaluate the fit
plot(x=as.numeric(influenza$Time), y=influenza$Mortality, pch=21, bg="red")
points(x=influenza$Time, y=model2_2$fitted.values, col="blue", pch=21, bg="blue")

# Investigate output
summary(model2_2)

# Visualize the spline component
plot(model2_2)
# How penalty factor(k) affects the devaince.
# Compares for two different values of K (2, 30)
model2_4v1 <- gam(formula=Mortality ~ Year + s(Week, k=3), data=influenza)
model2_4v2 <- gam(formula=Mortality ~ Year + s(Week, k=52), data=influenza)
# Compares fitted and original values for each case
par(mfrow=c(2,1))
# v1
plot(x=as.numeric(influenza$Time), y=influenza$Mortality, pch=21, bg="red", main="k=3")
points(x=influenza$Time, y=model2_4v1$fitted.values, col="blue", pch=21, bg="blue")
# v2
plot(x=as.numeric(influenza$Time), y=influenza$Mortality, pch=21, bg="red", main = "k=52")
points(x=influenza$Time, y=model2_4v2$fitted.values, col="blue", pch=21, bg="blue")
par(mfrow=c(1,1))

# 2.5
```

```r
# Residuals and influenza values plotted against time
plot(x=as.numeric(influenza$Time), y=model2_2$residuals, pch=21, bg="blue")
points(x=influenza$Time, y=influenza$Influenza, col="red", pch=21, bg="red")
# 2.6
# Mortality described as spline functions of year, week and influenza.
model2_6 <- gam(formula=Mortality ~ s(Week, k=52) + s(Year, k=9) + s(Influenza, k=85), data=influenza)
# Use output to test whether or not mortality is influenced by influenza.
summary(model2_6)
# plot fitted against original values
plot(x=as.numeric(influenza$Time), y=influenza$Mortality, pch=21, bg="red")
points(x=influenza$Time, y=model2_6$fitted.values, col="blue", pch=21, bg="blue")
# compute SSE
SSE2_6 <- sum((influenza$Mortality-model2_6$fitted.values)^2)
SSE2_6
##
```