# Introduction to Machine Learning

Lab 1 Block 2

*Rasmus Holm*

*2016-11-19*

## Contents

# Assignment 1

## 2

Figure 1 shows that the fit is pretty reasonable, the knot at 4 should probably have been moved to around 5 for an even better fit.
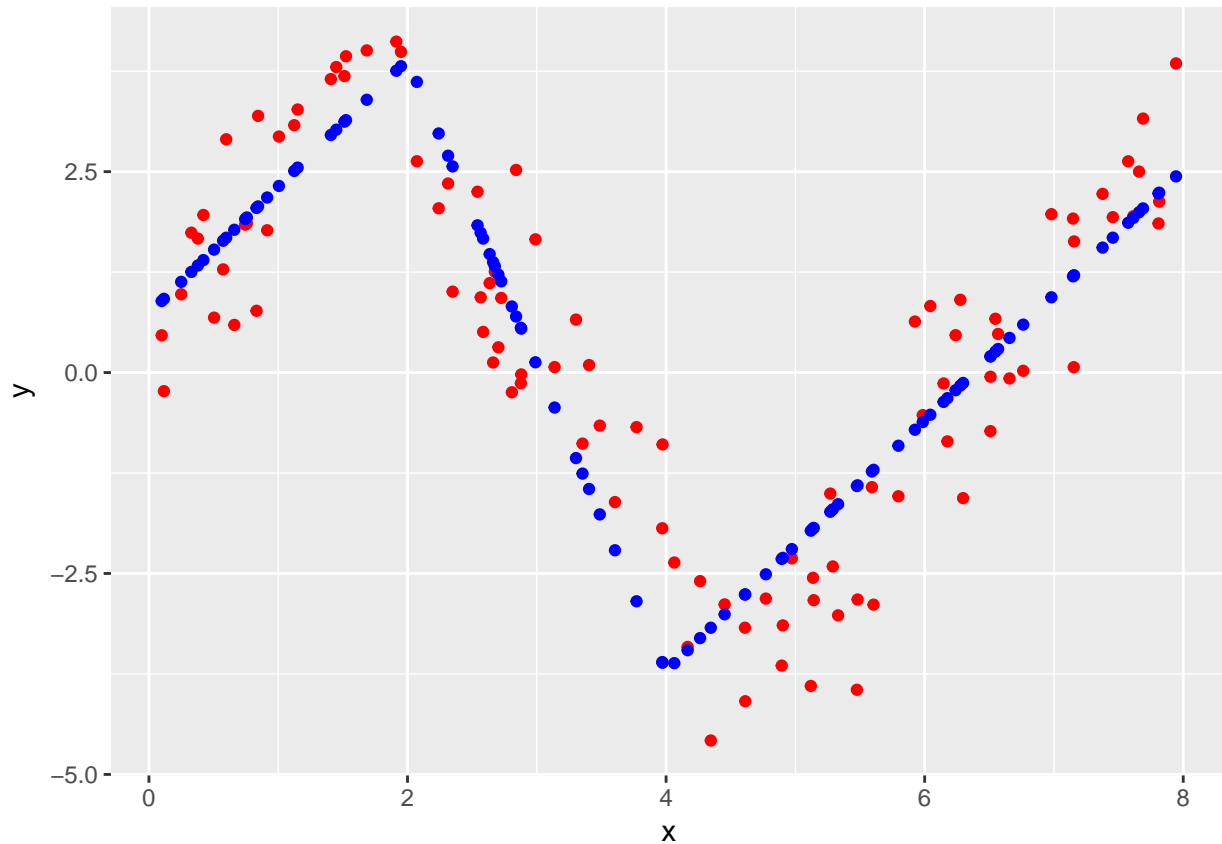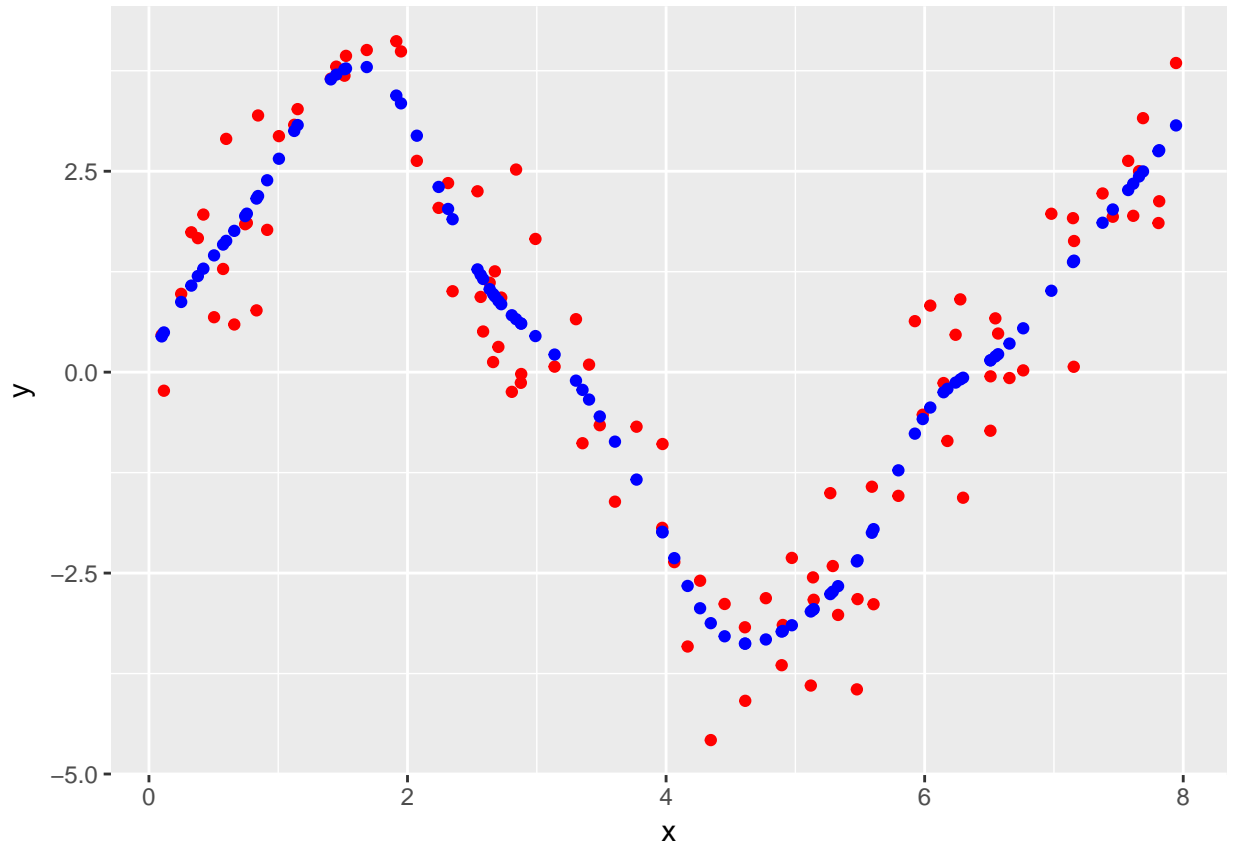


*Figure 1: Piecewise Linear Model*

It is a continuous piecewise linear function as it should be due to

$$h_3(X) = (X - \xi_1)_+,$$
$$h_4(X) = (X - \xi_2)_+$$

and it means that there are no derivatives at the knots.

## 3

The fitted line in figure 2 is much wigglier and seems to be highly affected by the noise in the data. It is however a much better fit where $x \in [4, 6]$.

*Figure 2: Cubic Smooth Spline*

Overall, I would say this fit is better than the previous one but maybe with a more thought out placement of the knots the piecewise linear may be more appropriate.

# Assignment 2

## 1

From figure 3 we can see that spikes in influenza also resulted in increased mortality cases over the complete timeline.
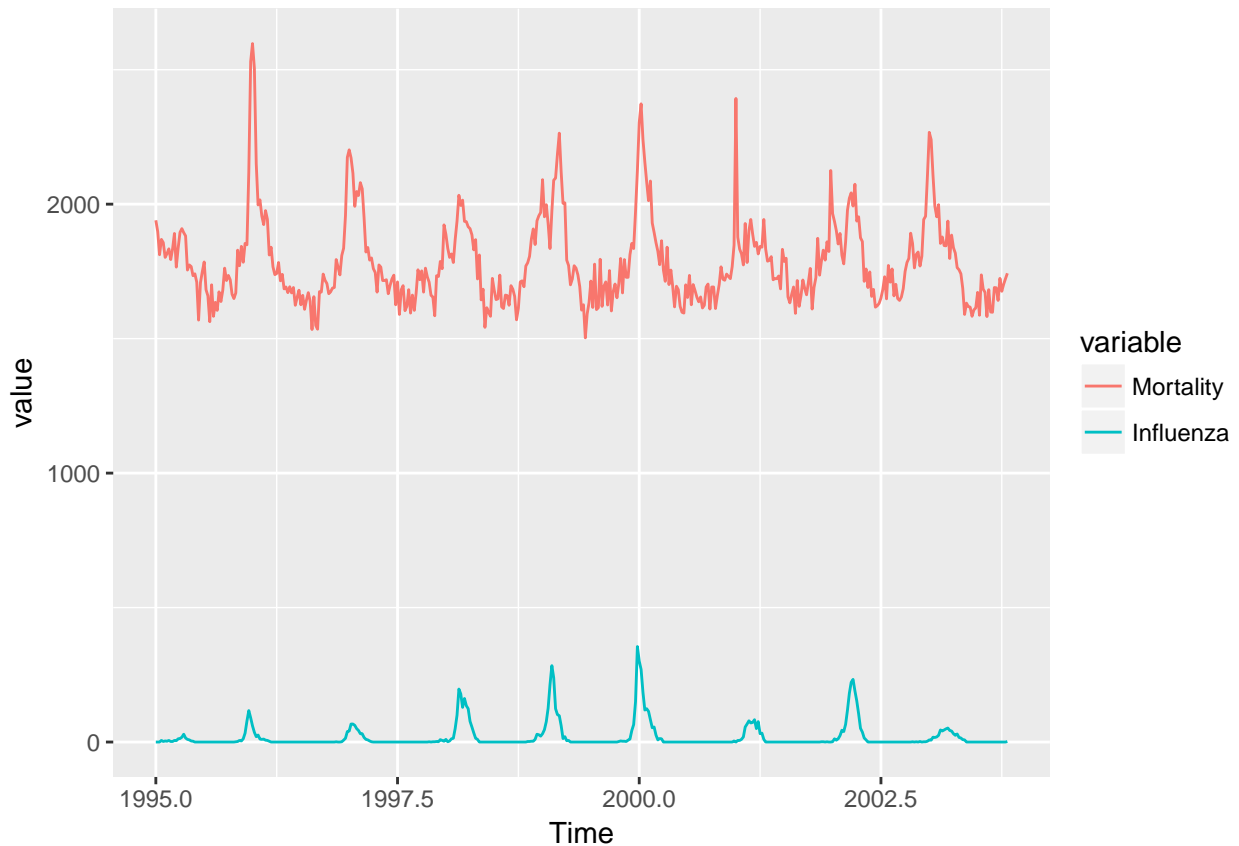


*Figure 3: Influenza and mortality cases as time series between 1995 and 2004*

## 2

**TODO**

In this exercise we assume $y \sim \mathcal{N}(\mu, \sigma^2)$ so the probabilistic model becomes

$$g(\mathbf{E}\left[Y|X\right]) = \alpha + \textit{Year} + f(\textit{Week}),$$

where $\mathrm{g}(\mu) = \mu$ is the link function and $f$ is a spline with an estimated degrees of freedom of 8.487.
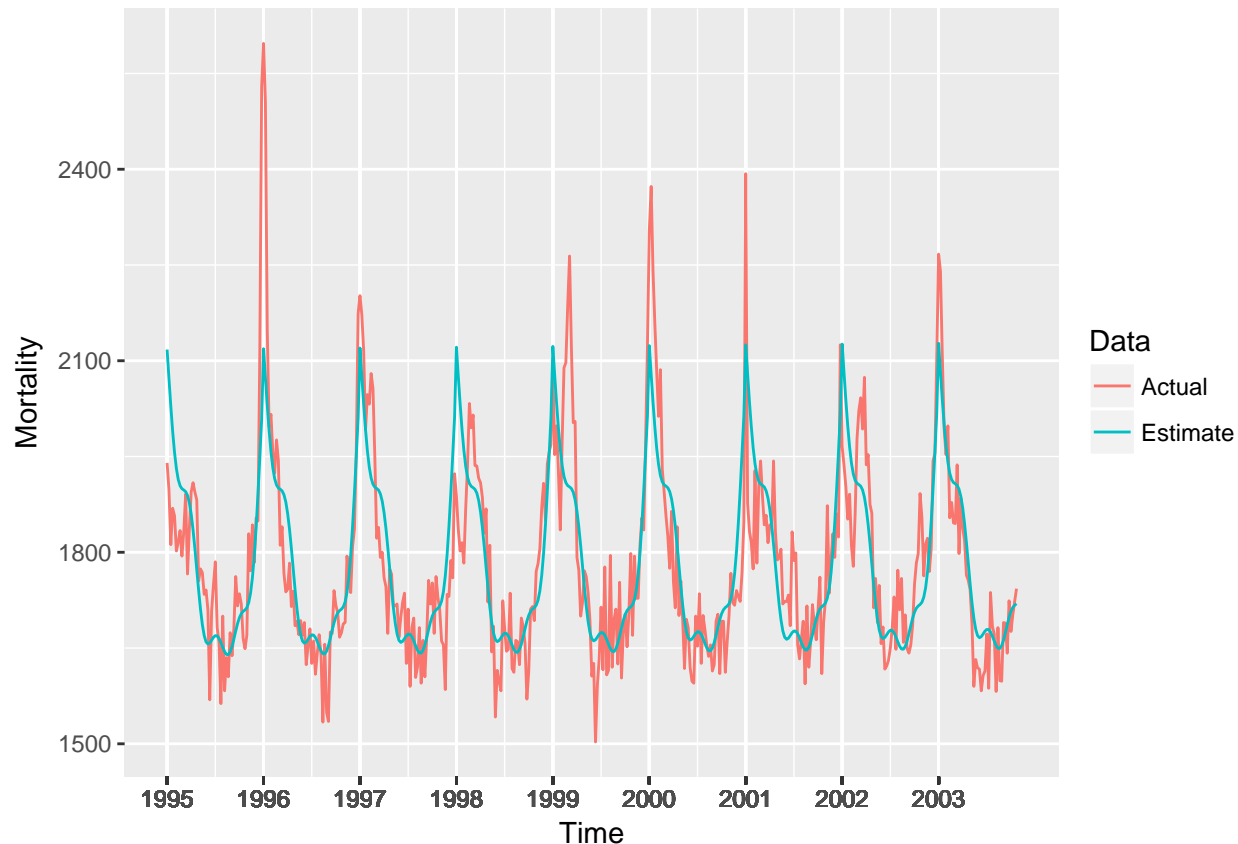
*Figure 4: Fitted model to the actual data.*

Figure 4 shows that the generalized additive model (GAM) is a decent fit to the data. The true data is very chaotic which the model does not capture, i.e. do not overfit to the noise, but it does underfit the peaks so overall I would say the model underfits the data.

We can also see that the mortality outbreak occurs in the beginning of each year and then decreases until winter is coming again, so the mortality is probably positively correlated with influenza cases which in turn are increased by the colder weather in the late/early months of each year.

```
#>
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> Mortality ~ Year + s(Week)
#>
#> Parametric coefficients:
#>             Estimate Std. Error t value Pr(>|t|)
#> (Intercept) -652.060   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  8.951 100.3  <2e-16 ***
```

```
#> ---
#> 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
```

From the summary above about the model we can see that the spline component is statistical significant while the intercept and year variable are not. This is not suprising given that the data is non-linear from figure 4.
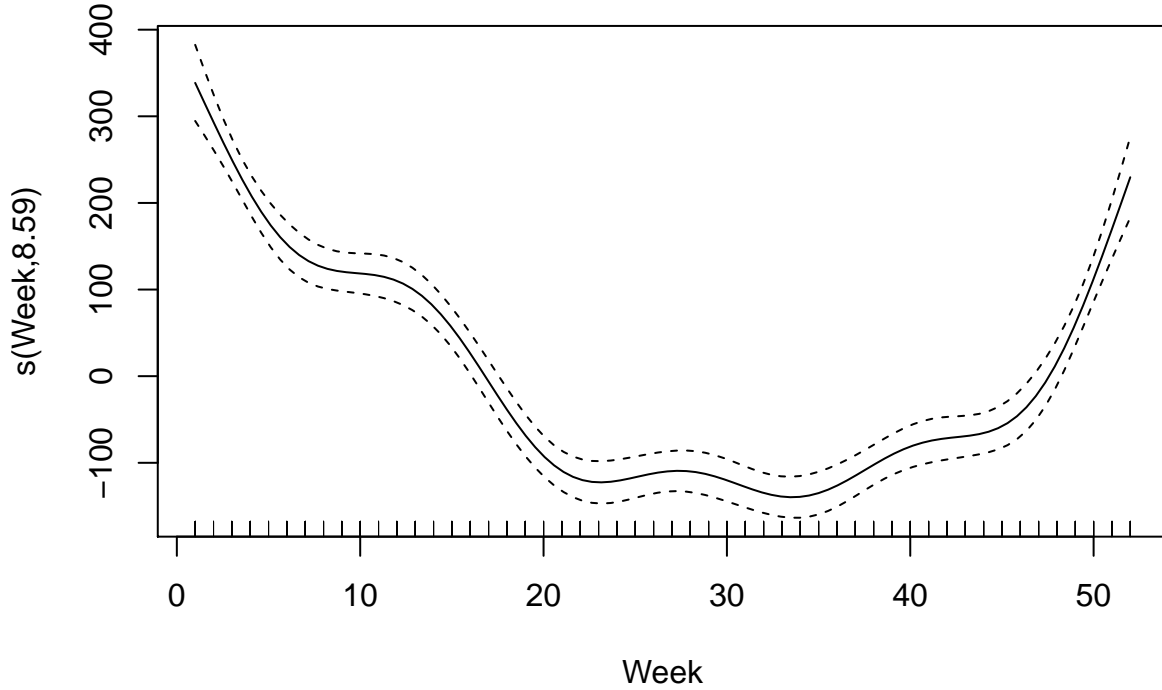


*Figure 5: The week spline function.*

The fitted *week* spline function can be seen in figure 5 with the pointwise standard errors included. It shows that the value is high in the early/late weeks and it is at its lowest point in the middle of the year, i.e. the summer months. This correspond to the same analysis as previously that mortality cases increase in the early/late months each year.

**4**

We know that the smoothing penalty factor, $\lambda$, and degrees of freedom for a spline has the following relationship.

$$\mathrm{df}_\lambda = \sum_{k=1}^{N} \frac{1}{1 + \lambda d_k}$$

implicates that as $\lambda$ increases the degrees of freedom decreases and vice verse. Figure 6 shows how the fit changes with varying penalty factors (sp parameter) with the maximum degrees of freedom set to a fixed value of 51 (k parameter).
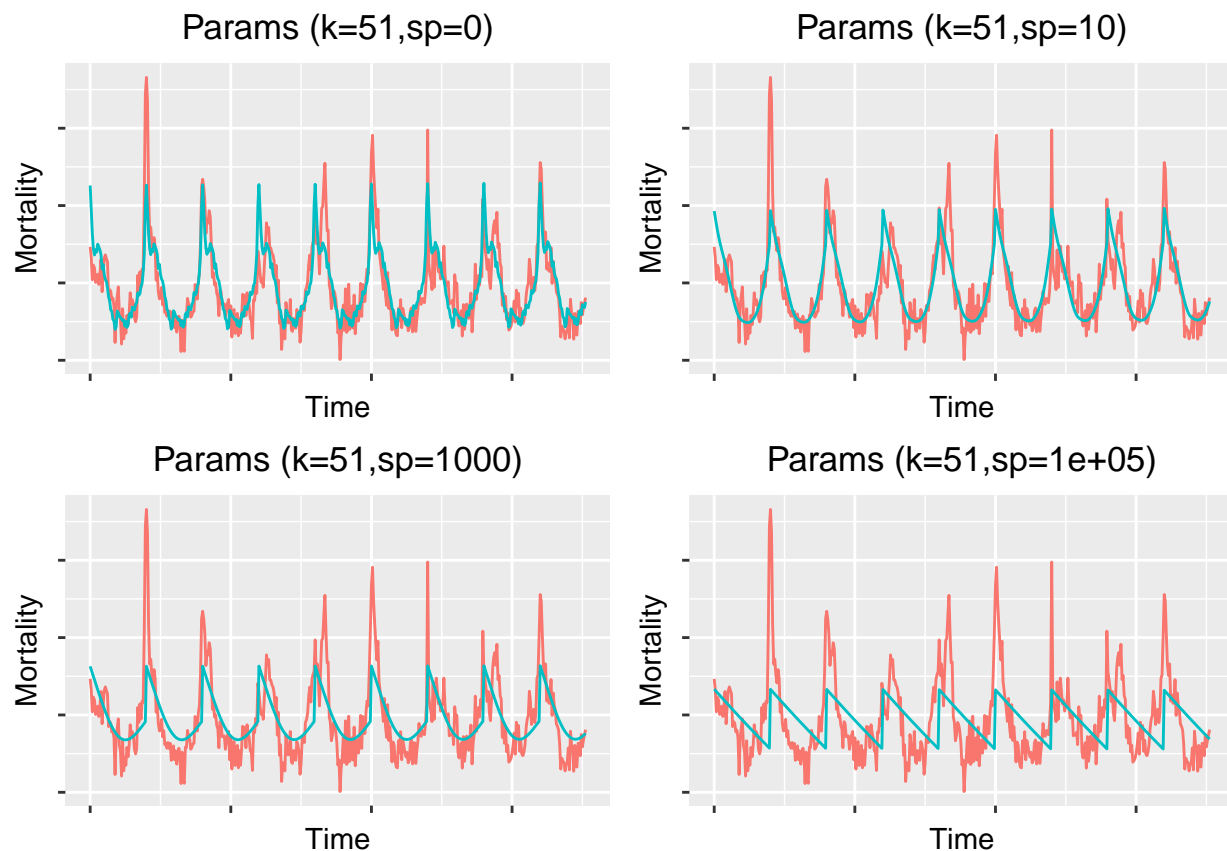
6

*Figure 6: Shows how the estimation changes with varying the penalty factor of the GAM.*

As the penalty factor increases the fit becomes less flexible, i.e. the bias increases, which results in a fit that does not match the data particular well and this follows directly from the theory above. A penalty factor of 0 or 10 are decent in this case but the models are still underfitting the real underlying data which means that the models are not complex enough. It is not even possible to increase the degrees of freedom when the penalty factor is set to 0 and this indicate that the *week* feature cannot explain the mortality completely so a better model would need to supplement it with further information.
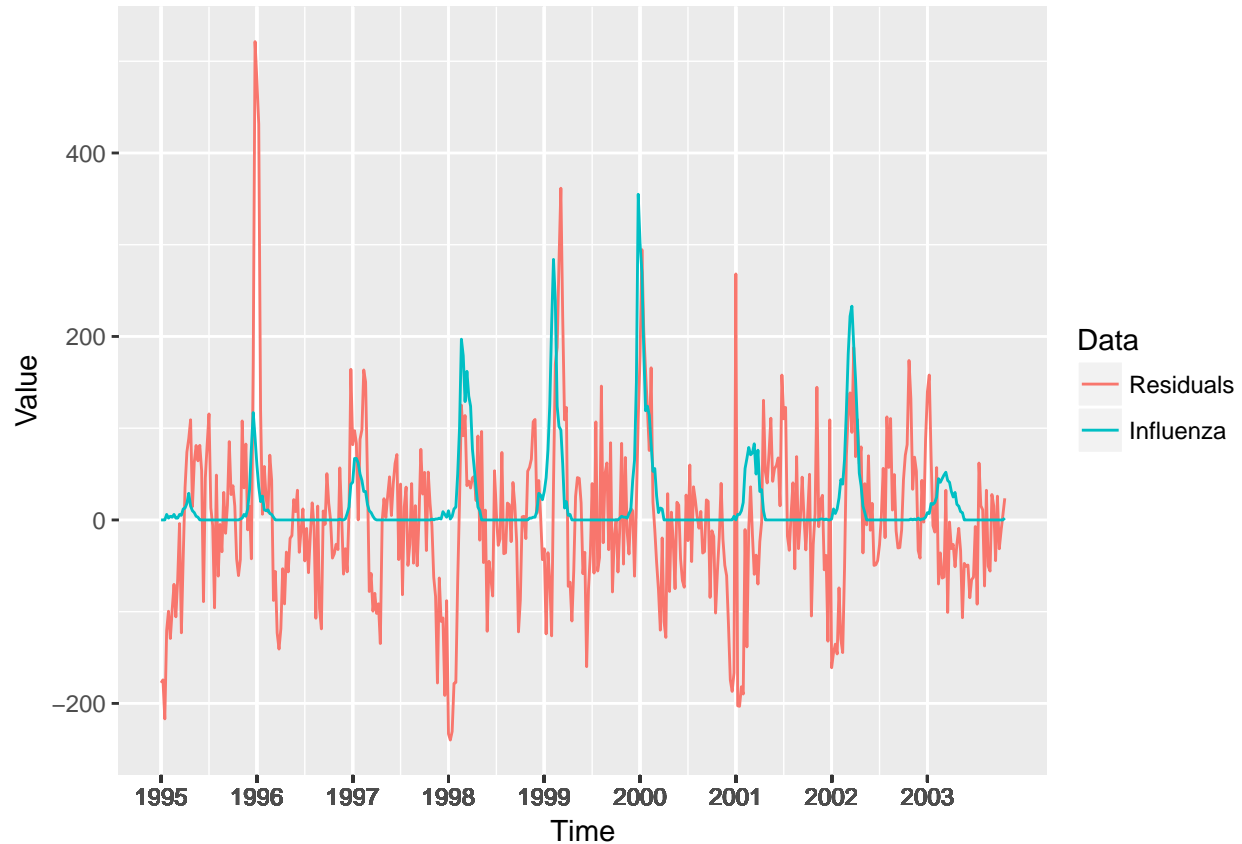
**5**



*Figure 7: Residuals from the GAM in exercise 2.*

The residuals are usually negative when *influenza* is at 0 and large positive spikes when the *influenza* is above 0 as shown in figure 7. This indicates that the model underestimates during the early/late months and usually overestimates the other months.

**6**

In this new model I have modelled *mortality* as an additive function of the spline functions of *year*, *week*, and *influenza*.

```
#>
#> Family: gaussian
#> Link function: identity
#>
#> Formula:
#> Mortality ~ s(Year, k = length(unique(data$Year)) - 1) + s(Week,
#>     k = length(unique(data$Week)) - 1) + s(Influenza, k = length(unique(data$Influenza)))
#>
#> Parametric coefficients:
#>             Estimate Std. Error t value Pr(>|t|)
#> (Intercept)  1783.77       3.28   543.8   <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)       3.907   4.75  1.178   0.292
#> s(Week)      13.831  17.20 20.342  <2e-16 ***
#> s(Influenza) 59.205  65.53  5.338  <2e-16 ***
#> ---
#> Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#>
#> Rank: 132/142
#> R-sq.(adj) =   0.81   Deviance explained = 84.2%
#> GCV = 5947.8  Scale est. = 4937.8    n = 459
```

The summary of the model above shows that the *influenza* is statistical significant in explaining the mortality.
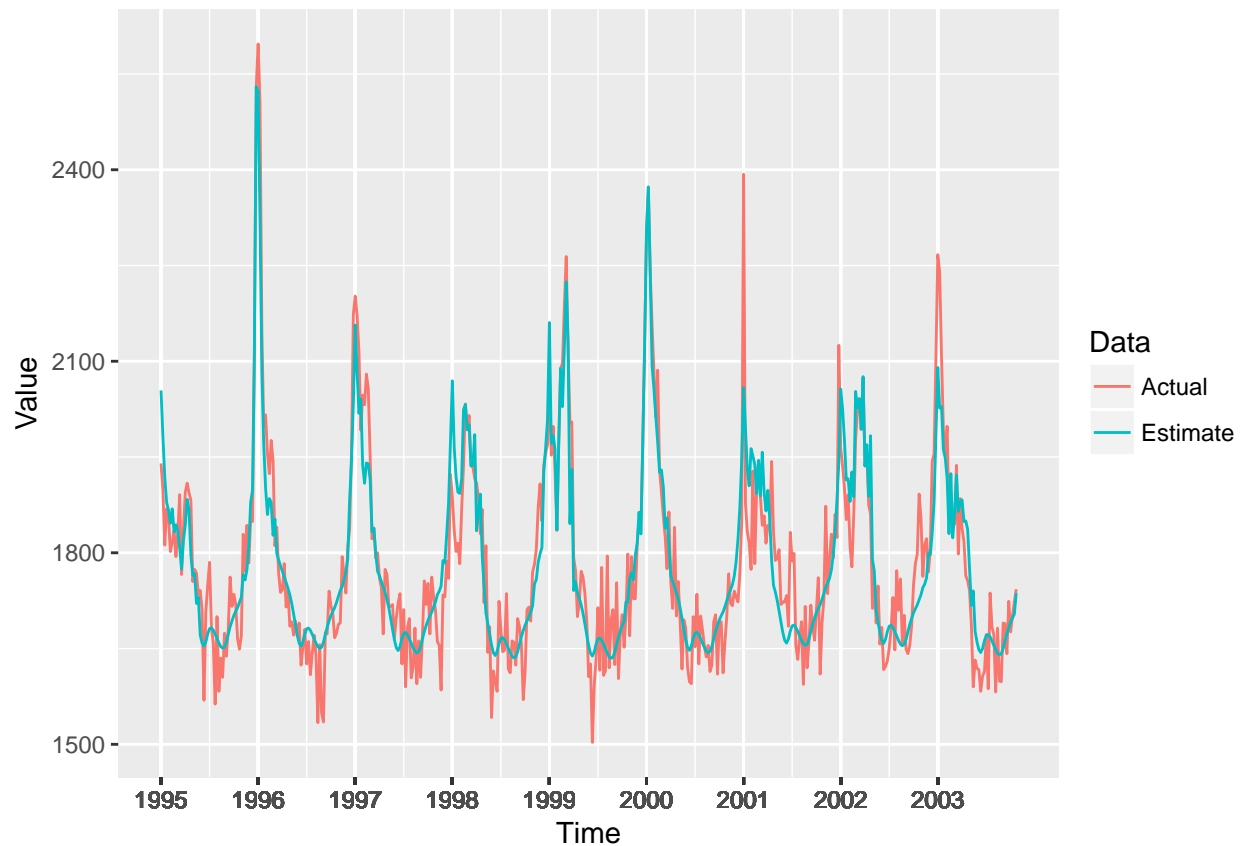


*Figure 8: Fitted model to the actual data.*

The new model is fitting the data much better than before as shown in figure 8 and from the summary we can conclude that the *week* and *influenza* variables explain most of the mortality cases, roughly 84%.

9

# Appendix

## Code for Assignment 1

```r
library(ggplot2)

myspline <- function(X, y, knots) {
    n <- length(X)
    m <- length(knots)
    df <- m + 2

    H <- matrix(0, nrow=n, ncol=df)
    H[, 1] <- 1
    H[, 2] <- X

    for (i in 3:df) {
        H[, i] <- pmax(X - knots[i - 2], 0)
    }

    data <- data.frame(y=y, H)
    ## Removes the intercept term (have it already)
    lmfit <- lm(y ~ 0 + ., data=data)
    coefficients <- as.numeric(coef(lmfit))
    yhat <- H %*% coefficients

    yhat
}

data <- read.csv2("../data/cube.csv", header=TRUE, sep=";")
knots <- c(2, 4)
yhat <- myspline(data$x, data$y, knots)

plot_data <- data.frame(x=data$x, y=data$y, yhat=yhat)

ggplot(plot_data) +
    geom_point(aes(x, y), color="red") +
    geom_point(aes(x, yhat), color="blue")
smooth_fit <- smooth.spline(x=data$x, y=data$y)
yhat <- fitted(smooth_fit)

## plot(smooth_fit, col="blue")
## points(data$x, data$y, col="red")

plot_data <- data.frame(x=data$x, y=data$y, yhat=yhat)

ggplot(plot_data) +
    geom_point(aes(x, y), color="red") +
    geom_point(aes(x, yhat), color="blue")
```

## Code for Assignment 2

```r
library(ggplot2)
library(readxl)
library(reshape2)
library(mgcv)
library(grid)
library(gridExtra)

data <- read_excel("../data/Influenza.xlsx")
plot_data <- melt(data[, c("Time", "Mortality", "Influenza")], id="Time")
ggplot(plot_data) +
    geom_line(aes(x=Time, y=value, color=variable))
gamfit <- gam(Mortality ~ Year + s(Week), family=gaussian, data=data, method="GCV.Cp")

yhat <- predict(gamfit, data)

plot_data <- data.frame(Time=data$Time, Actual=data$Mortality, Estimate=as.numeric(yhat))
plot_data <- melt(plot_data, id="Time", value.name="Mortality", variable.name="Data")

ggplot(plot_data) +
    geom_line(aes(x=Time, y=Mortality, color=Data)) +
    scale_x_continuous(breaks=data$Year)
summary(gamfit)
plot(gamfit)

k <- length(unique(data$Week)) - 1
penalty_values <- c(0, 10, 1000, 100000)

plots <- list()

for (i in 1:length(penalty_values)) {
    fit <- gam(Mortality ~ Year + s(Week, k=k, sp=penalty_values[i]),
               family=gaussian, data=data, method="GCV.Cp")

    title <- paste("Params (k=", k, ",sp=", penalty_values[i], ")", sep="")

    plot_data <- data.frame(Time=data$Time, Actual=data$Mortality, Estimate=fitted(fit))
    plot_data <- melt(plot_data, id="Time", value.name="Mortality", variable.name="Data")

    plots[[i]] <- ggplot(plot_data) +
        geom_line(aes(x=Time, y=Mortality, color=Data), show.legend=FALSE) +
        ggtitle(title) +
        theme(axis.text=element_blank(),
              plot.title=element_text(hjust=0.5))
}

do.call(grid.arrange, c(plots, list(ncol=2)))
gamfit <- gam(Mortality ~ Year + s(Week), family=gaussian, data=data, method="GCV.Cp")
residuals <- resid(gamfit)
plot_data <- data.frame(Time=data$Time, Residuals=residuals, Influenza=data$Influenza)
plot_data <- melt(plot_data, id="Time", value.name="Value", variable.name="Data")
```

```r
ggplot(plot_data) +
    geom_line(aes(x=Time, y=Value, color=Data)) +
    scale_x_continuous(breaks=data$Year)

gamfit <- gam(Mortality ~ s(Year, k=length(unique(data$Year)) - 1) +
                  s(Week, k=length(unique(data$Week)) - 1) +
                  s(Influenza, k=length(unique(data$Influenza))),
              data=data)
summary(gamfit)
plot_data <- data.frame(Time=data$Time, Actual=data$Mortality, Estimate=fitted(gamfit))
plot_data <- melt(plot_data, id="Time", value.name="Value", variable.name="Data")

ggplot(plot_data) +
    geom_line(aes(x=Time, y=Value, color=Data)) +
    scale_x_continuous(breaks=data$Year)
```