ISYE 6501

Homework 3

Artur Cabral, Marta Bras, Pedro Pinto, Katie Price

2019-09-16

Question 7.1

Describe a situation or problem from your job, everyday life, current events, etc., for which exponential smoothing would be appropriate. What data would you need? Would you expect the value of α (the first smoothing parameter) to be closer to 0 or 1, and why?

Cellphone manufacturers such as Apple or Samsumg, are constantly trying to improve the battery life of their devices. They could study how users' use of the device can impact the deterioration of battery life. For users who agree to share their usage data, the company would select a sample, and record screen time for app usage, and charging time of the devices every day. To eliminate randomness and variability of battery life, all devices in the sample should be purchased at the same day.
For each type of device, the company would build an eponential smoothing model with screen time for app usage and daily charging time. The data could include cyclic effects (such as times of the day the user utilises the phone more often, or days in the week where the usage is also increased) and the trend would show which type of usage has a higher impact battery life. Combining the CUSUM with exponential smoothing models, it would be possible to constantly monitor usage behavior that result in battery deterioration, and forecast how long it would take for a device to have its baterry life decreased.

Question 7.2

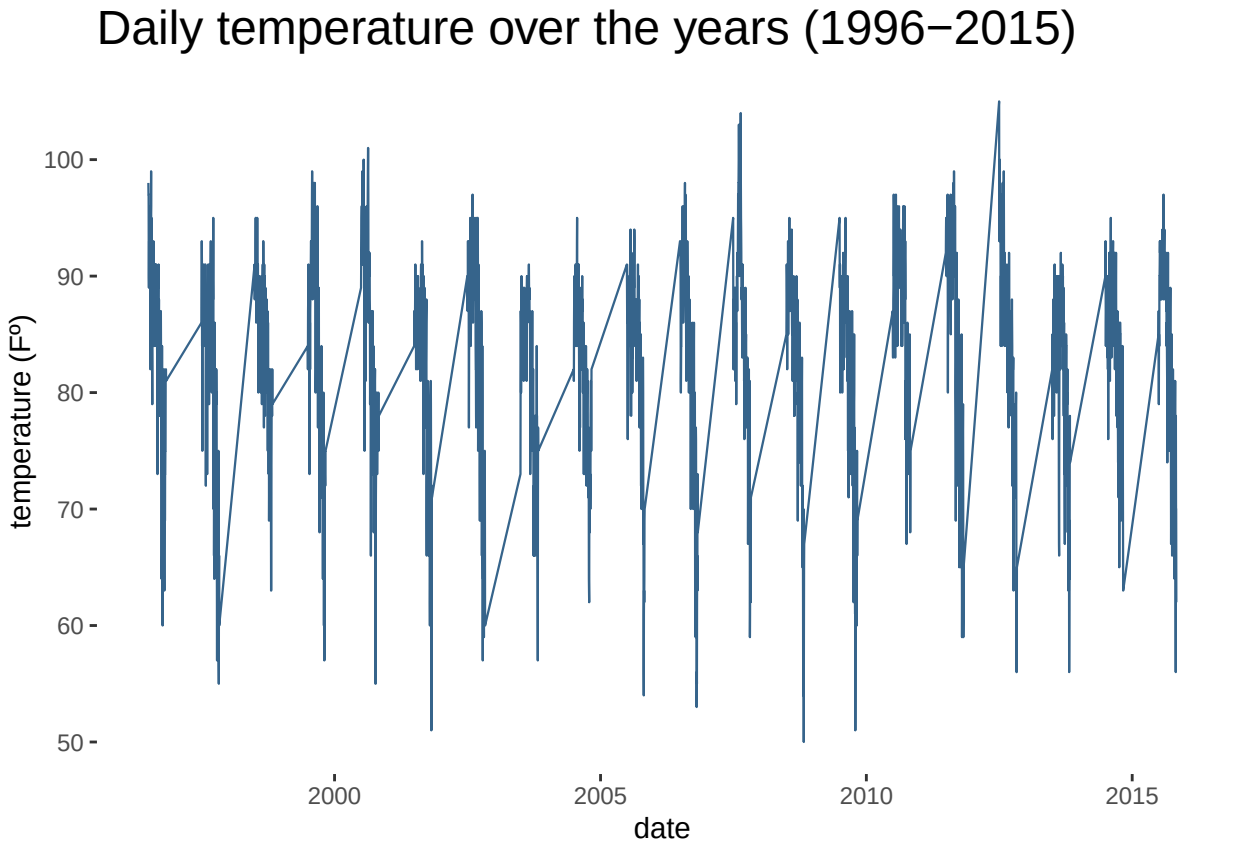
Using the 20 years of daily high temperature data for Atlanta (July through October) from Question 6.2 (file temps.txt), build and use an exponential smoothing model to help make a judgment of whether the unofficial end of summer has gotten later over the 20 years. (Part of the point of this assignment is for you to think about how you might use exponential smoothing to answer this question. Feel free to combine it with other models if you'd like to. There's certainly more than one reasonable approach.)

Note: in R, you can use either HoltWinters (simpler to use) or the smooth package's es function (harder to use, but more general). If you use es, the Holt-Winters model uses model="AAM" in the function call (the first and second constants are used "A"dditively, and the third (seasonality) is used "M"ultiplicatively; the documentation doesn't make that clear).

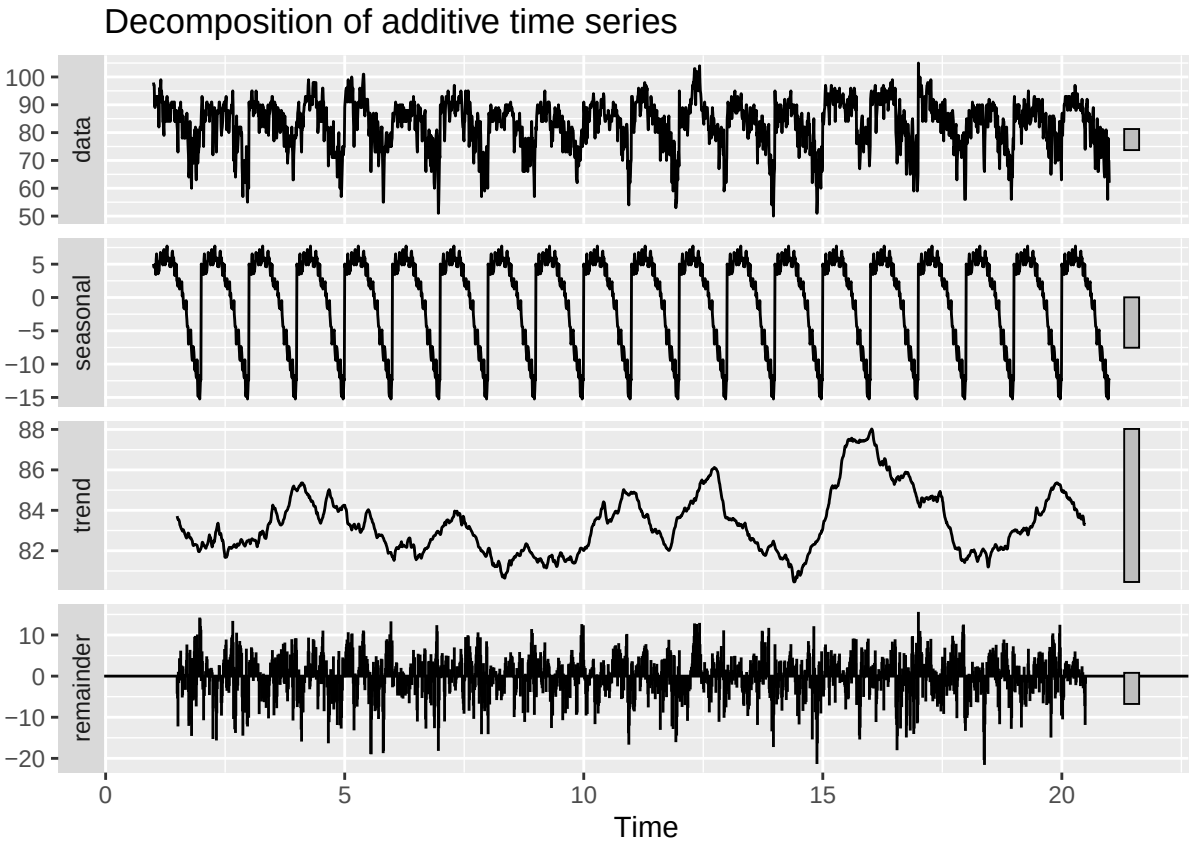
1. Importing and initial analysis of the data

```
#reading the data
t_data <- read.table("temps.txt", header = TRUE)

#creating a date column
date <- t_data %>% gather(X, Temp, X1996:X2015)
date$X <- gsub("^.{0,1}", "", date$X)
date <- date %>% mutate(x=str_c(DAY,"-",X))
```

```r
date[,4] <- as.Date(date[,4], format = "%d-%b-%Y")
colnames(date) = c("Day", "Year", "Temperature", "Date")

#plotting tempeature over time
ggplot(date, aes(x=Date, y=Temperature)) +
  geom_line(color="steelblue4", lwd=0.4) +
    ggtitle("Daily temperature over the years (1996-2015)") +
    ylab("temperature (Fº)") +
    xlab("date") +
    theme_bw() + theme(panel.border = element_blank(), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank()) +
    theme(plot.title = element_text(size=18))
```




```r
#saving data as vector
years <- as.vector(unlist(t_data[,2: length(t_data)]))

#converting data to time series
ts <- ts(years,frequency = 123)

#decomposition between trend and seasonality
autoplot(decompose(ts))
```



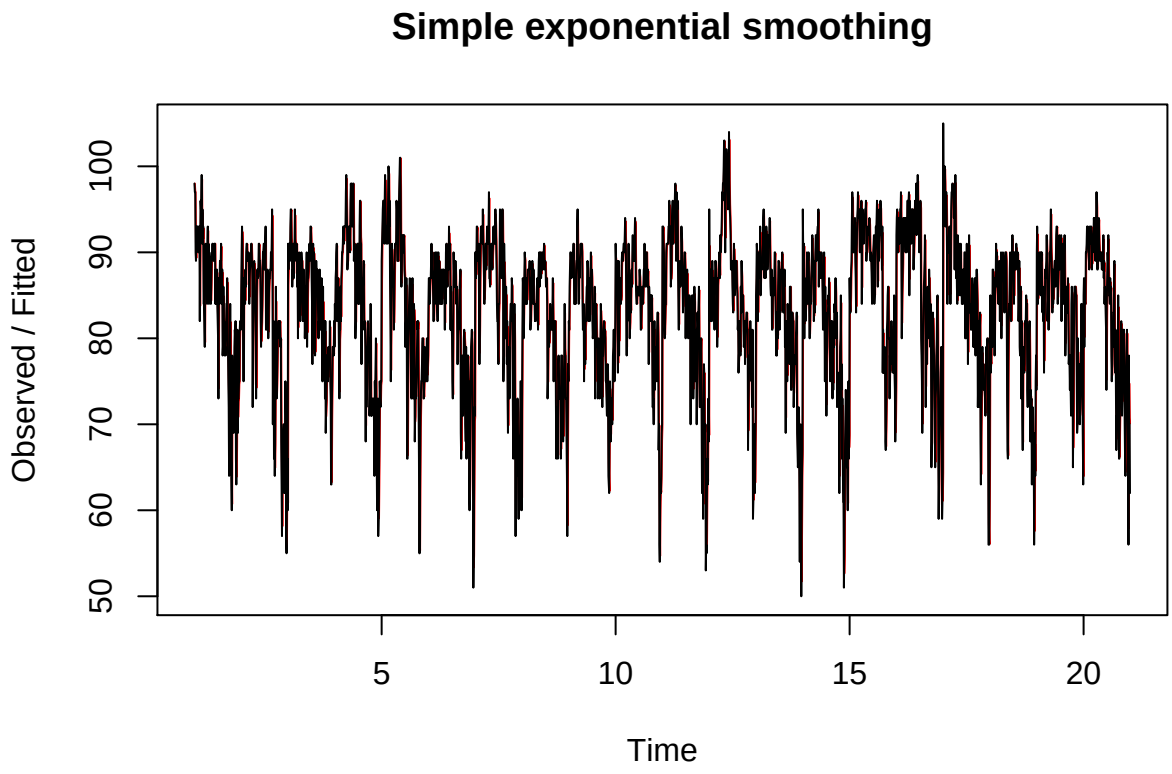
Looking at the decomposition of the time series data between seasonality and trend, we can see that there is high seasonality in the data. This means that from July to October, some days have much higher temperatures than others.

As we had concluded in the previous assignment, the trend in the daily temperatures is inconclusive, with a visible positive trend around 2010, followed by a clear negative trend around 2013.

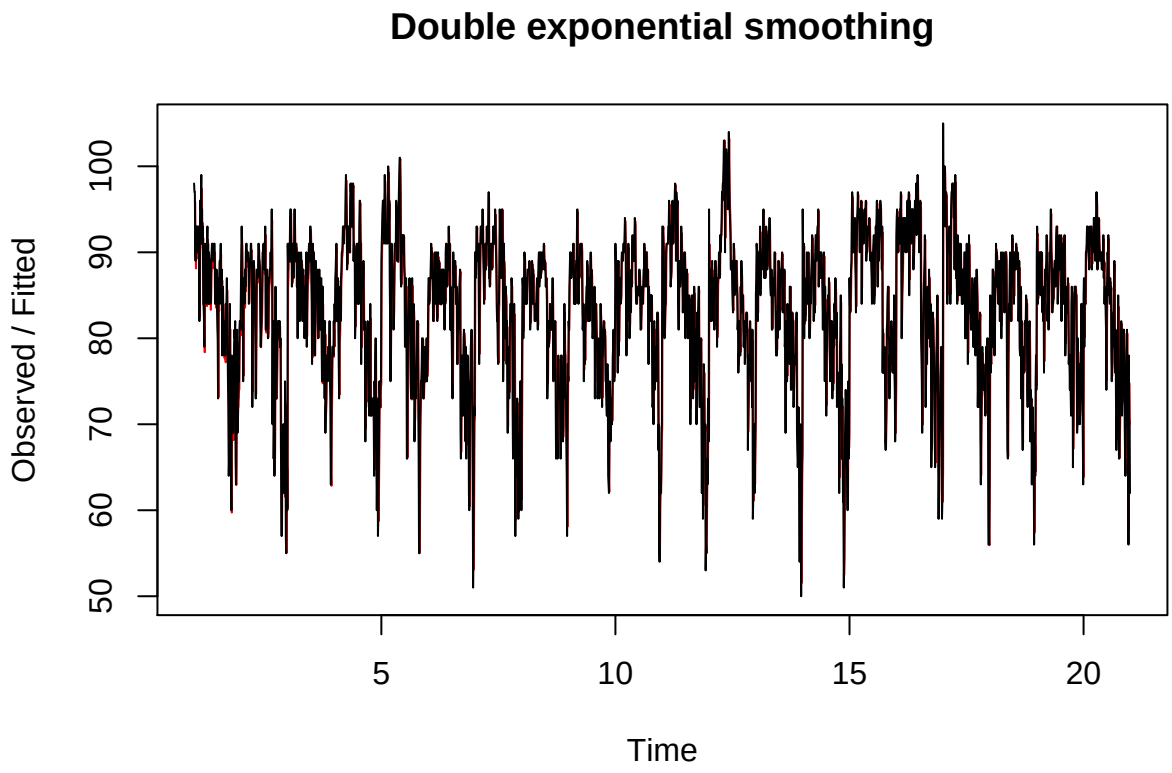
2. HoltWinters Models

To answer the question in the assignment, we started by fitting different HoltWinters models to the data to understand how they are performing.

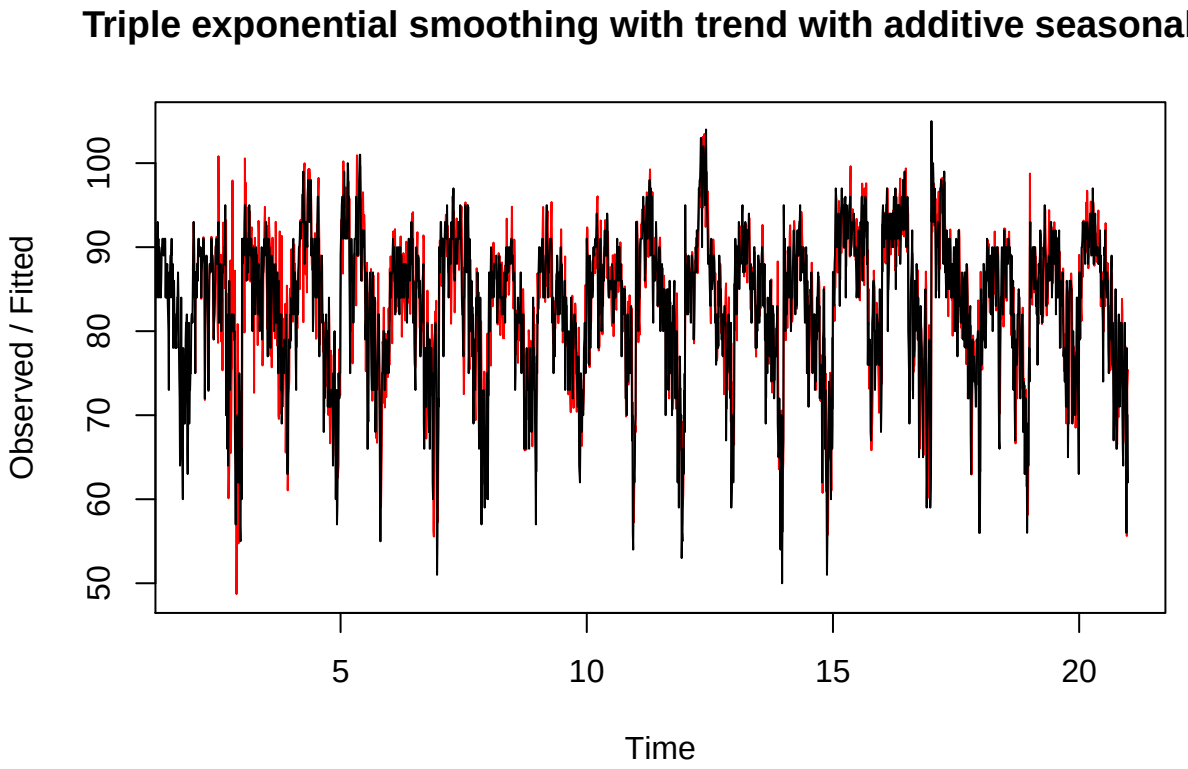
```
#Simple Exponential Smoothing:
smooth_1 <- HoltWinters(ts,beta = FALSE, gamma = FALSE)

#plot of smooth_1
plot(smooth_1, main = "Simple exponential smoothing")
```



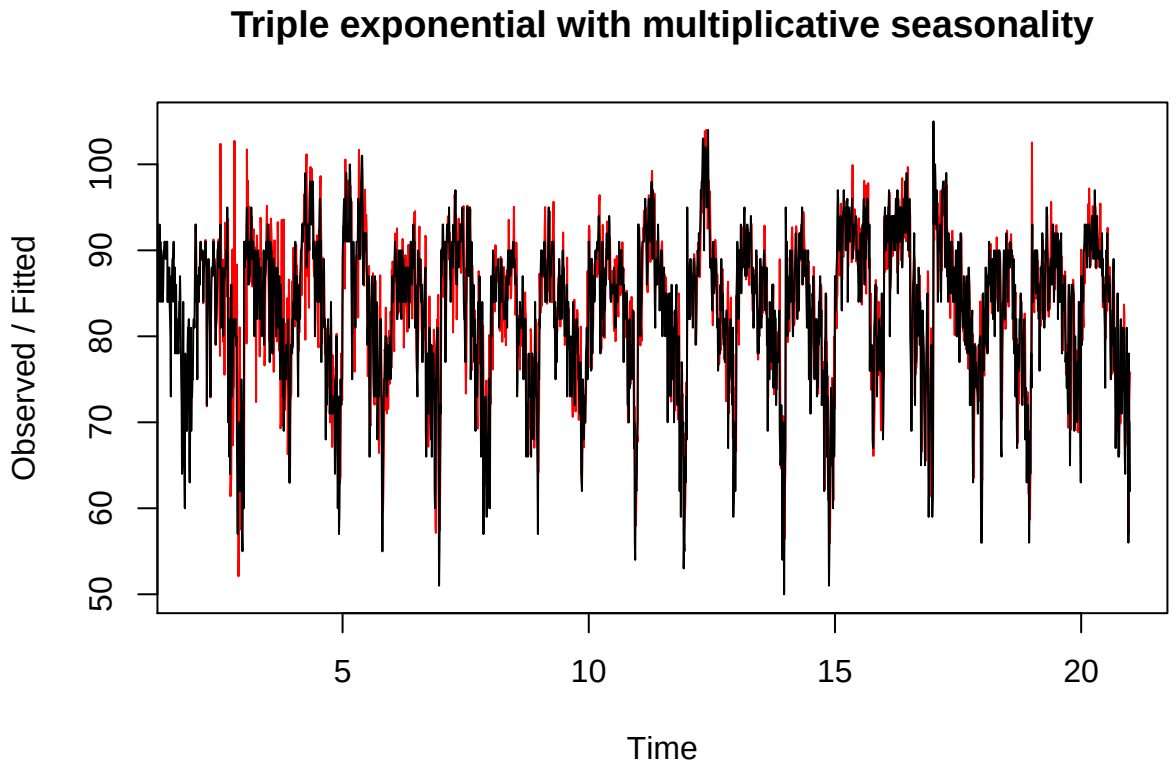

```
# Double Exponential Smoothing with trend:s
smooth_2 <- HoltWinters(ts,gamma = FALSE)

#plot of smooth_2
plot(smooth_2,  main = "Double exponential smoothing")
```



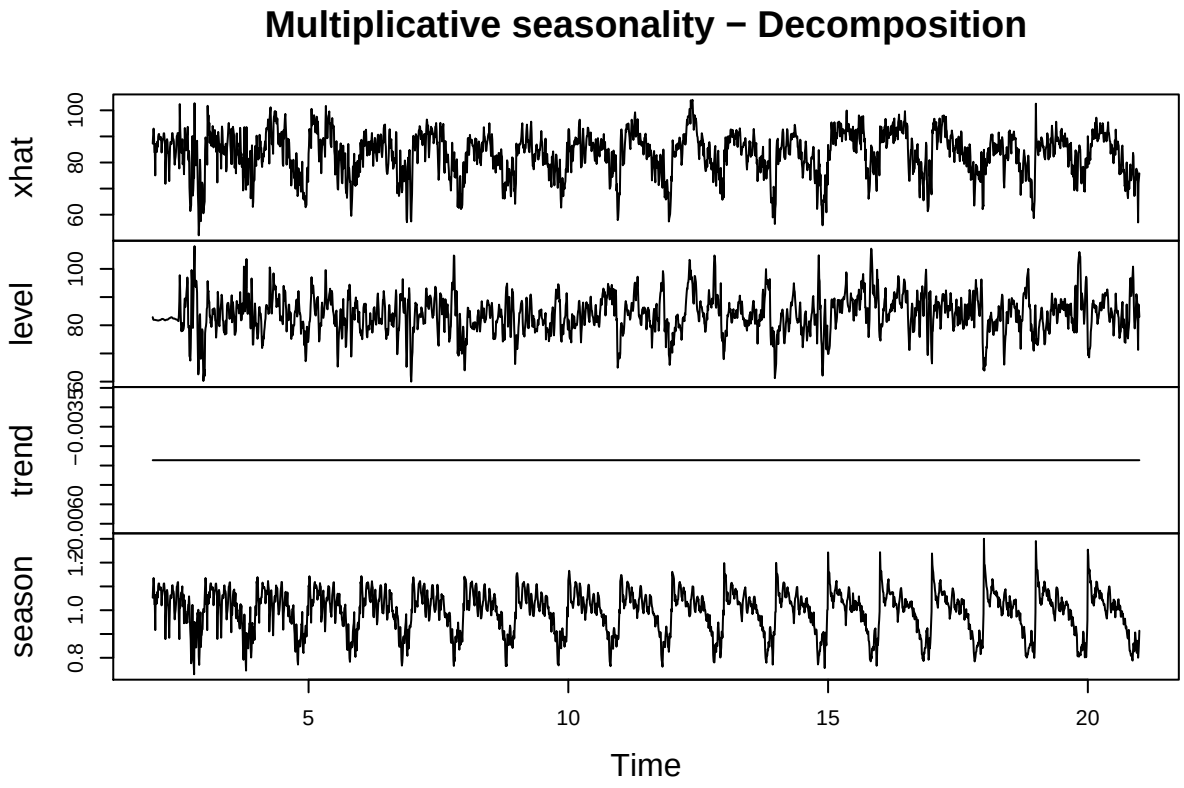
```r
#Triple Exponential Smoothing with trend with additive seasonality:
smooth_3 <- HoltWinters(ts)

#plot of smooth_3
plot(smooth_3,  main = "Triple exponential smoothing with trend with additive seasonality")
```



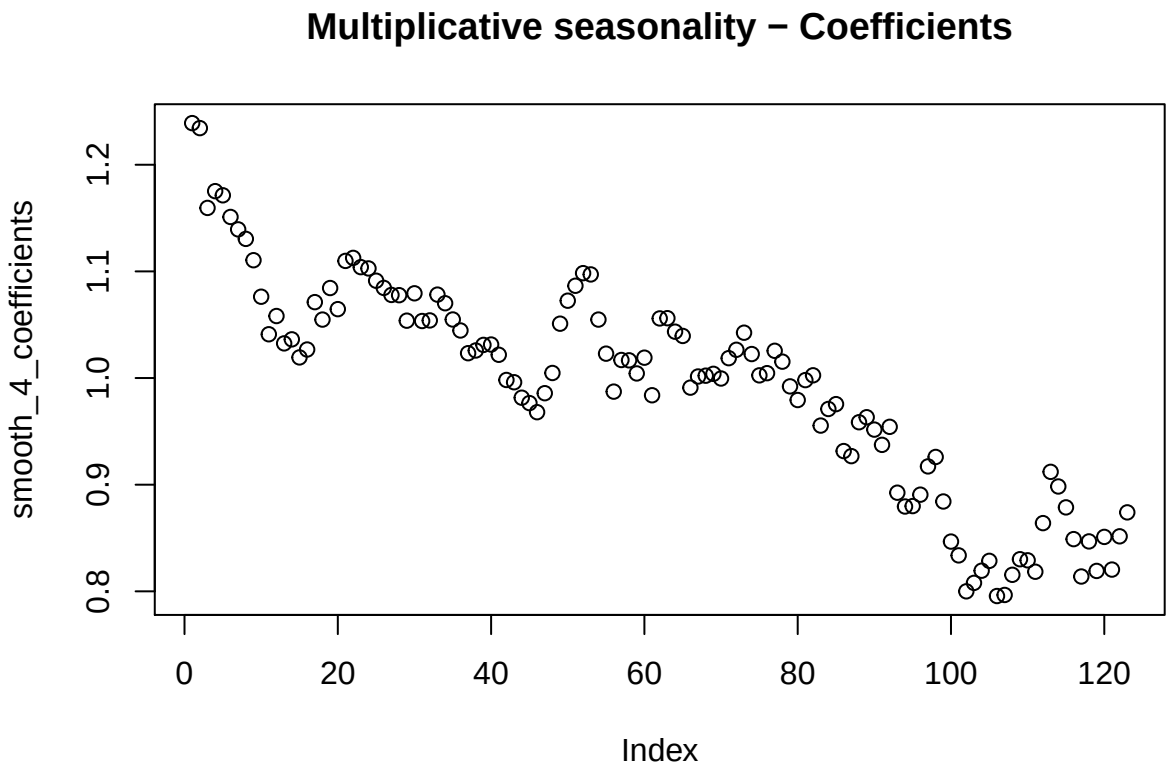

```r
#Triple Smoothing with trend with multiplicative seasonality:
smooth_4 <- HoltWinters(ts, seasonal="multiplicative")

#plot of smooth_4
plot(smooth_4,  main = "Triple exponential with multiplicative seasonality")
```




```
#decomposition  of smooth_4 - level, trend and seasonality
plot(fitted(smooth_4),  main = "Multiplicative seasonality - Decomposition")
```



```r
#coefficients of smooth_4
smooth_4_coefficients <- coefficients(smooth_4)[3:125]
plot(smooth_4_coefficients, main = "Multiplicative seasonality - Coefficients")
```



When we did the initial decomposition of the time series data, we concluded that there is seasonality in the data. This is a good indicator that a HW model with a seasonality parameter can be a good fit for our data.

From the *Triple exponential with multiplicative seasonality* plot, we can observe a good fit to the data, since the actual values represented by the black line are close to the fitted values represented by the red line. The multiplicative model doesn't start until 1997, after it processed one year of data.

There are some interesting conclusions when comparing data **before and after** smoothing that we can get from the decomposition graphs, before and after smoothing:

- **Seasonality** - There is seasonality in the data before and after multiplicative smoothing. From the smoothing model that we can observe that there is a difference in the seasonality pattern over the years, namely around period 10 and then again around period 15. After period 15, some days appear to have much more extreme temperatures than the others and it seems likke the temperature is decreasing after an initial peak but to levels higher than the minimum temperatures in the previous years.
- **Trend** - On the other hand, the trend in the data after smoothing is close to 0, which contrasts to the unclear upward and downward trend in the initial data. This means, that on average, there is no visible trend in the data.

3. Applying the CUSUM function to the smoothed data

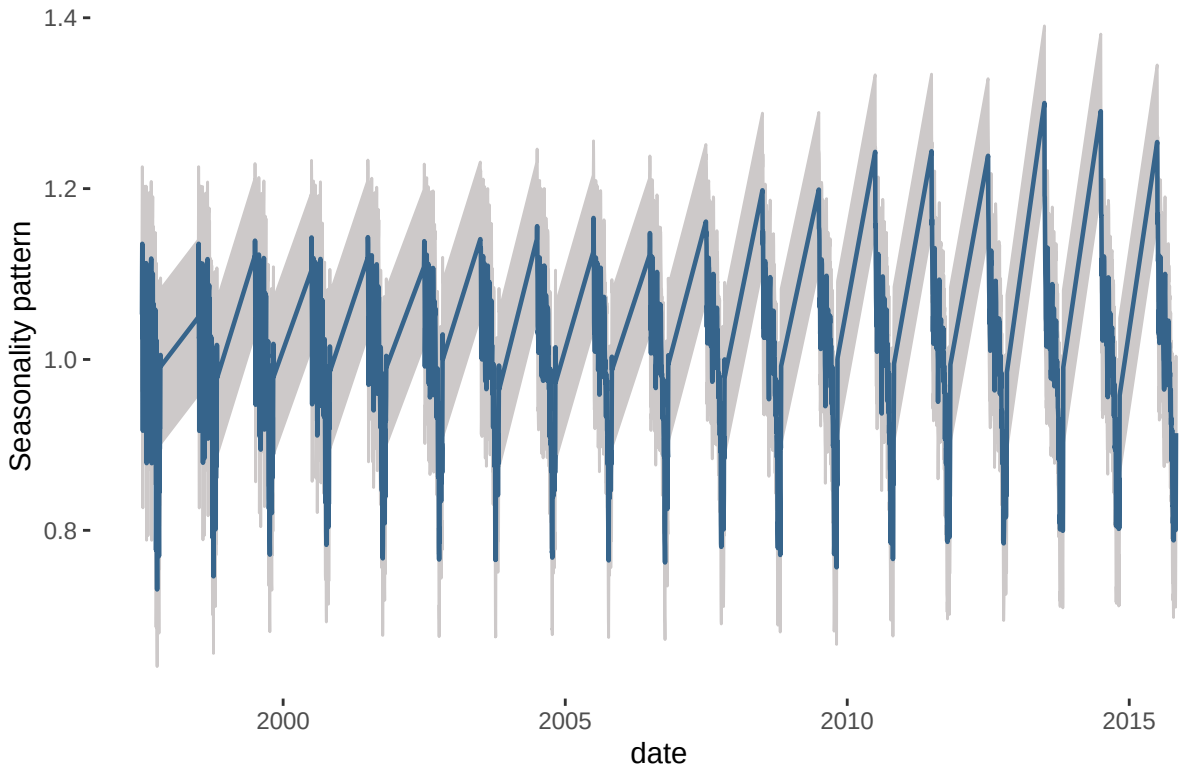
We can apply the CUSUM function to the smoothed data and see if we can detect a change in the last day of the Summer. We can do it by using either the seasonality or the overall smoothed values.

We will start by analysing the changes in the seasonality pattern, as we concluded before that there might have been changes in the pattern of daily temperatures throughout the years.

For the CUSUM function we will use the average daily seasonality over the years and compare the seasonality in a certain day with the overall average for that day. By doing that, we will try to detect which days the temperatures are lower than the average (plus the threshold).

```
#getting the seasonality estimates from the multiplicative model
fitted <- as.data.frame(smooth_4$fitted[,4])

#filtering the date column (day-month and date) from the original dataseet to start in 1997 because the
date_2 <- date$Date
day <- date$Day

date_2 <- date_2[124:length(date_2)]
day <- day[124:length(day)]

#adding date column to the fitted values
fitted <- cbind(date_2, fitted, day)

#plotting the seasonality pattern over the years
#adding a confidence interval
fitted$mino3<-fitted$x-sd(fitted$x, na.rm=T)
fitted$maxo3<-fitted$x+sd(fitted$x, na.rm=T)

ggplot(fitted, aes(x=date_2, y=x)) +
  geom_ribbon(aes(ymin=mino3, ymax=maxo3), fill="snow3", color="snow3")+
  geom_line(color="steelblue4", lwd=0.8)+
    ggtitle("Smoothed seasonality pattern over the years") +
    ylab("Seasonality pattern") +
    xlab("date") +
    theme_bw() + theme(panel.border = element_blank(), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank()) +
    theme(plot.title = element_text(size=18))
```

```
## Don't know how to automatically pick scale for object of type ts. Defaulting to continuous.
```



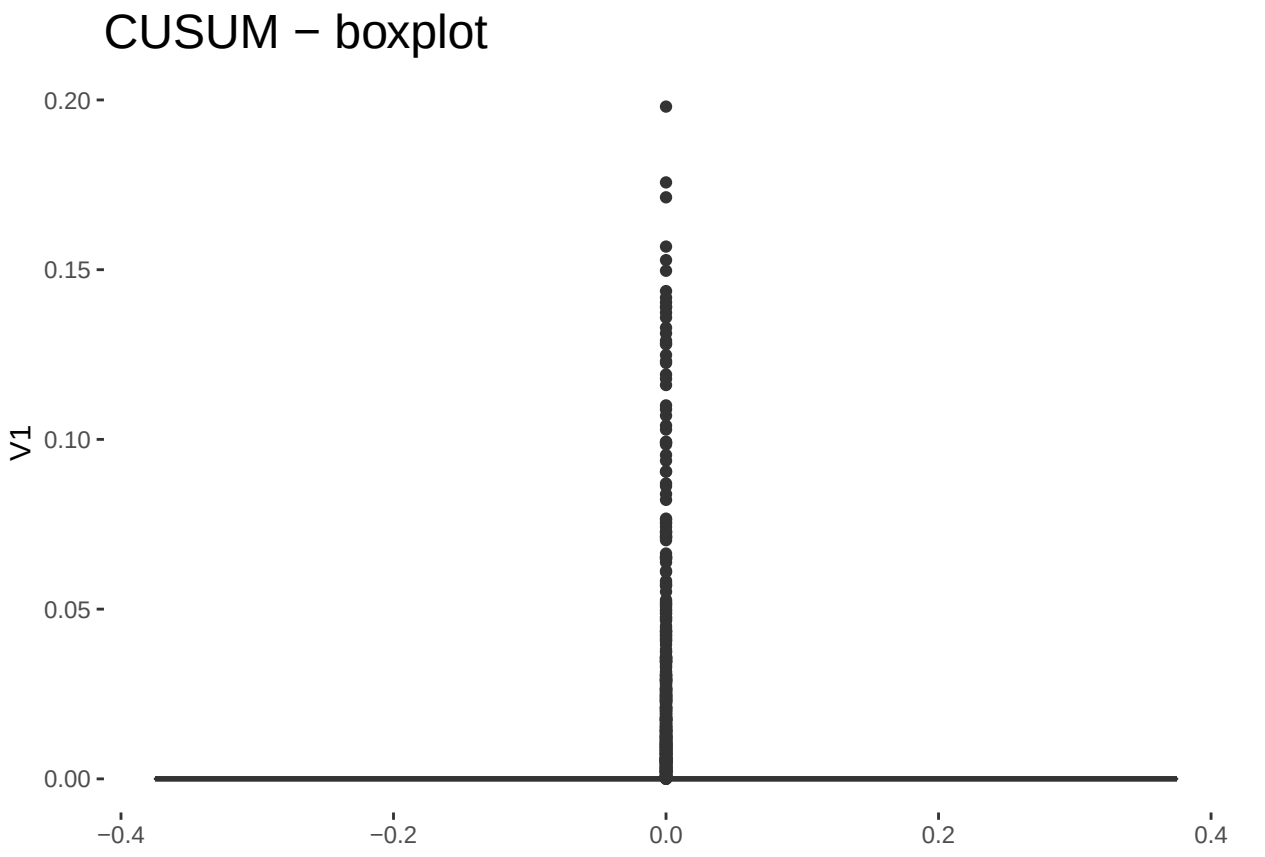

```
#removing max and min columns
fitted <- fitted[-c(4,5)]
```

As we can see from the plot above, the seasonality pattern changed first around 2008. The difference from the highest temperatures to the lowest temperatures seem to be higher from this year on. This is significant because we are comparing the average temperatures per day with the temperature in that day over the years. Also the lowest temperature seems to be increasing over the years, which might indicate the summer is lasting longer.

To understand if the temperatures have been higher than the overall average, we computed the average daily temperature over the years and compared to the specific temperature in that day. As C value we used the standard deviation in the daily temperatures over the years.

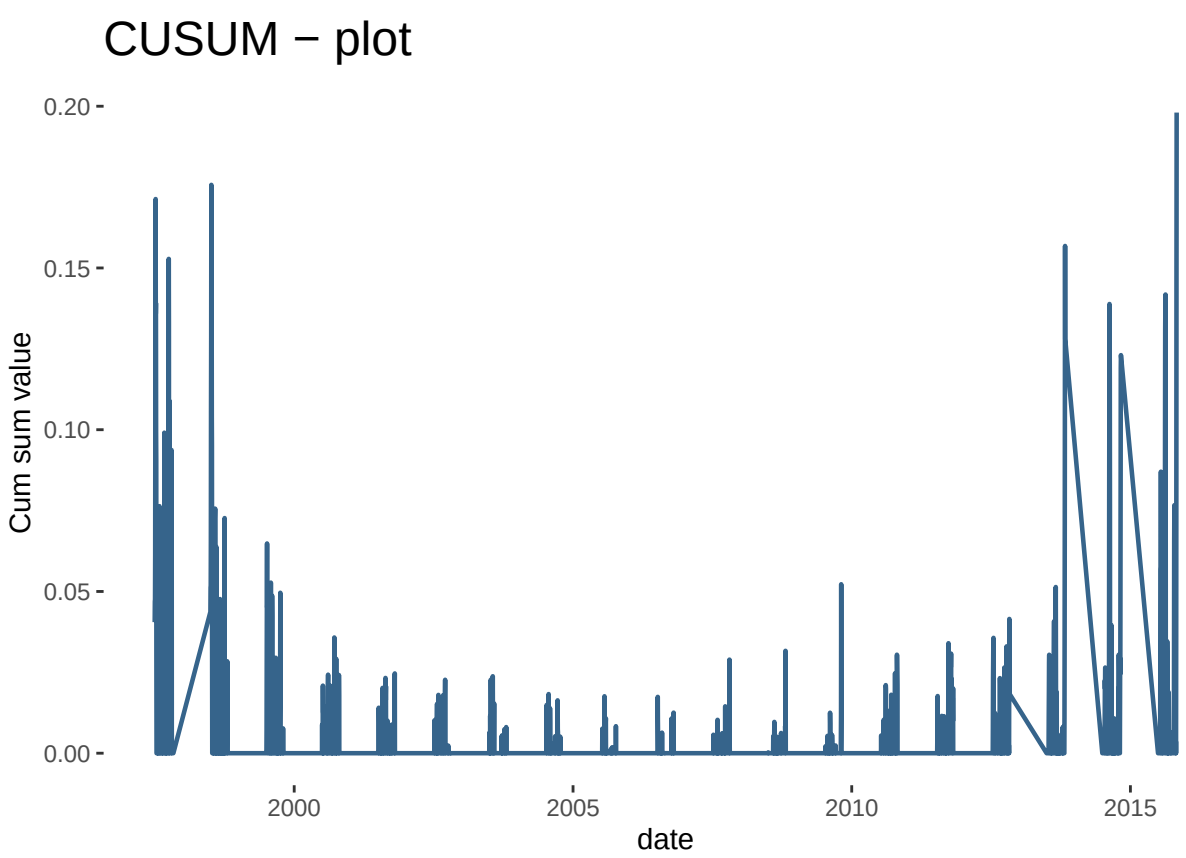
```
#creating average for each day
parameters <- fitted %>% group_by(day) %>% summarize(mean = mean(x), stdd = sd(x))


#adding everything to the same database
fitted <- left_join(fitted,  parameters, by ="day")

#initiating an empty matrix to store the CUSUM
CUSUM_matrix= matrix(nrow=length(fitted$day), ncol = 1)

st <- 0


#comptuing the CUSUM function
```

```
for (i in (1: length(fitted$day))) {
  CUSUM_matrix[i,1] = max(0, (st + (fitted[[i,4]] - fitted[[i,2]] - fitted[[i,5]])))
  st <- CUSUM_matrix[i,1]
}


#adding the results to a dataframe with the date
CUSUM_matrix_2 <- as.data.frame(CUSUM_matrix)
CUSUM_matrix_2 <- cbind(fitted$date_2, CUSUM_matrix_2, fitted$day)


#Boxplot of results
ggplot(CUSUM_matrix_2, aes(y=V1)) +
    geom_boxplot() +
    ggtitle("CUSUM - boxplot") +
    theme_bw() + theme(panel.border = element_blank(), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank()) +
    theme(plot.title = element_text(size=18))
```




```
#CUSUM plot
ggplot(CUSUM_matrix_2, aes(x=fitted$date_2, y=V1)) +
  geom_line(color="steelblue4", lwd=0.8) +
    ggtitle("CUSUM - plot") +
    ylab("Cum sum value") +
    xlab("date") +
    theme_bw() + theme(panel.border = element_blank(), panel.grid.major = element_blank(),
```

```
    panel.grid.minor = element_blank()) +
    theme(plot.title = element_text(size=18))
```



As we had previosuly concldued, the difference in the seasonality pattern results in CUSUM values decreasing over time. This indicates that the seasonality was more visible from 1997 to around 2001 - the difference in temperatures in summer is lower after 2001. The seasonality started being more visible again from 2013 to 2015. To understand if there are differences in the last day of summer, we compute the average and standard deviation of the values in the cusum function over the years per day. We then find which days had a cusum value that was higher than the average cusum for that day plus a standard deviation and we consider this to be the last day.

```
#adding year and month to the cusum matrix
CUSUM_matrix_2 <- CUSUM_matrix_2 %>%
  mutate(year = year(fitted$date_2), month = month((fitted$date_2)), day_number = day(fitted$date_2))

#computing the mean and standard deviation
mean <- CUSUM_matrix_2 %>% group_by(fitted$day) %>% summarize(mean = mean(V1), standard_dev = sd(V1))

#adding the mean and standard deviation to the cusum matrix
CUSUM_matrix_2 <- left_join(CUSUM_matrix_2, mean, by = "fitted$day")

#finding the values that are a standard dev away from the mean
results <- CUSUM_matrix_2 %>% filter(month > "8") %>% group_by(year) %>% mutate(diff = V1 - (mean - sta
```

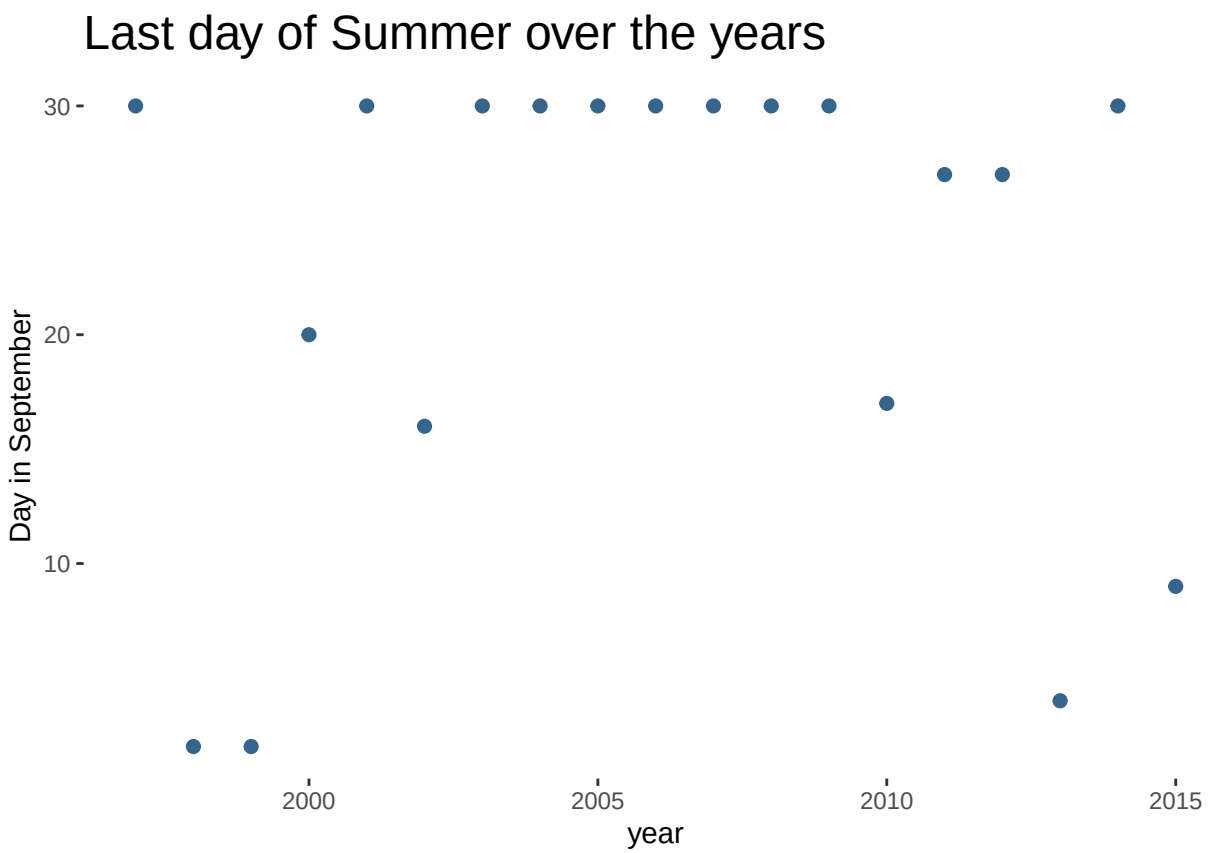
```
## Selecting by diff
```

Table 1: Last day of Summer over the years

year	day_month	day_number
2015	9-Sep	9
2014	30-Sep	30
2013	4-Sep	4
2012	27-Sep	27
2011	27-Sep	27
2010	17-Sep	17
2009	30-Sep	30
2008	30-Sep	30
2007	30-Sep	30
2006	30-Sep	30
2005	30-Sep	30
2004	30-Sep	30
2003	30-Sep	30
2002	16-Sep	16
2001	30-Sep	30
2000	20-Sep	20
1999	2-Sep	2
1998	2-Sep	2
1997	30-Sep	30

```
colnames(results) <- c("date", "cusum", "day_month", "year", "month", "day_number", "mean", "standard_d

#table with the last day of summer per year

last_day <- results %>% select(year, day_month,day_number)

kable(last_day, caption = "Last day of Summer over the years") %>%
  kable_styling(bootstrap_options = c("striped", "hover", "condensed", "bordered"))
```

```
#plotting results
ggplot(last_day, aes(x = year, y = day_number)) +
  geom_point(color="steelblue4", lwd=2) +
  ggtitle("Last day of Summer over the years") +
    ylab("Day in September") +
    xlab("year") +
    theme_bw() + theme(panel.border = element_blank(), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank()) +
    theme(plot.title = element_text(size=18))
```



As we can see from the graph and table above, the last day of Summer changed from being the 2 of September in 1998 and 1999, to being 20 September in 2000 and 30 Septmber until 2009.

As previously hypothesized, with exception of 2014, in 2013 and 2015, Summer seems to be ending sooner again. The main conclusions for this question are:

- The exponential smoothing using multiplicative seasonality helped us indentify differences in the seasonality patters over the years.
- The CUSUM function helped us identify if those patterns were related to changes in the last day of Summer.
- Even though it is difficult to identify the last day of Summer, we can observe that during a period of years (from 2000 to 2009), Summer appeared to have finished later than the remaining years.
- Different values of C and K might have resulted in different days but the years with longer summers would have remained the same.

The analysis could have been done using the overall fitted values of the HoltWinters model but since we are concerned about seasonality in the data, using the estimates for seasonality seemed a reasonable approach.