Homework Week 4

Chris Messer

2022-09-17

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 a (the first smoothing parameter) to be closer to 0 or 1, and why?

> Exponential Smoothing has many uses, including forecasting and anomally detection. As a forecasting tool, exponential smoothing would be useful in predicting web for a cloud hosted website, to ensure that cloud resources are spun up to handle increased traffic at appropriate times. We may uncover there is a strong "seasonal" trend in traffic, for example, an entertainment site may see a spike in traffic Monday through Friday around lunch time, and they could use that forecast to ensure the resources are in place to support that traffic during peak times.
>
> I would need web traffic data such as visitors per minute, and I would expect an alpha parameter closer to 0, as there is likely a lot of randomness in the observed visitor traffic, but as a whole should be rather consistent on a weekly cycle.

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.)

> Several methods of evaluation are shown below, but in the end, when extracting the seasonality of the time series, I have concluded that the summer season is in fact ending later and later. However, to note, this is not a statement that summer is "lasting longer", since we do not have the data to determine whether summer is also potentially *starting longer*. See analysis below for how I arrived as this conclusion.

Analysis

Why smoothing?

First, I'd like to walk you through why we might want to use a smoothing technique in making the determination that summer is ending later. To do so, I will use a CUSUM model on the unsmoothed data, similar to the prior week's lesson.

No Smoothing

Now, we have some decisions to make. What should our mu value be? Our T (threshold)? Our C (shift)? How do we determine when summer has officially ended, and is not just a cold front that blew in?

Choosing Mu

To find an appropriate value of mu, we first look to the question asked. What is the *unofficial* end date of summer? Since we are unofficially looking for a date, this implies there is an "official" end date of summer! A quick google search tells us in the northern hemisphere, summer officially starts on June 21st and ends on September 22. Since our data set starts on July 1st, we can assume the official summer are the dates the first 84 rows of data. As such, we can conclude the average temperature of summer is the mean temperature during these days.

Choosing T

The qcc package makes this a little easier for us. It defaults to a cummulative 5 standard deviations from the mean before considering something as "out of threshold". I am electing to use a threshold of a cumulative 10 standard deviations. Why? because I used a C value of 1 standard deviations (see below), a threshold of 10 standard deviations equates to a temperature decrease of at least a cumulative 10 days of temps below 1 standard deviation away from the mean. (Note, I say the equivalent of 10 days… the model would also show a breach if we had 5 days of temps 2 standard deviations below the mean).

Choosing C

Now we must consider- at what point is a summer day *abnormally hot* or *abnormally cold*? Standard deviation is a great fence post for this metric. As long as our data is pseudo normally distributed, 1 standard deviation1 from the mean would encompass 68% of all data points. Since we saw in the previous lesson the data was pseudo-normal, I will proceed with this assumption. First, read the data in:

```
library(qcc)
```

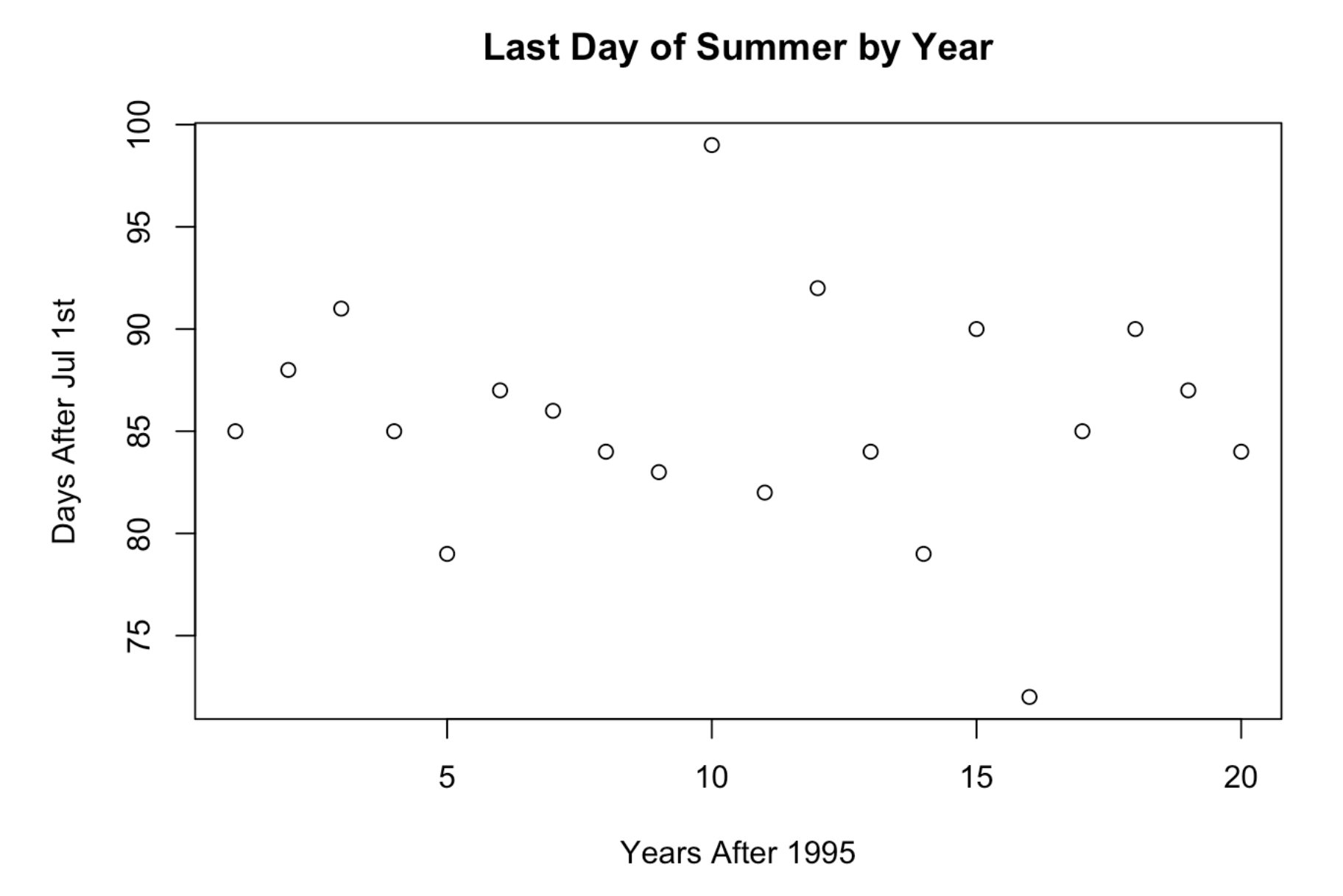
```
## Package 'qcc' version 2.7
```

```
## Type 'citation("qcc")' for citing this R package in publications.
```

```
temps <- read.csv('temps.txt', sep ='\t')
```

Now, build a CUSUM model to determine when the last day of every summer is and plot the result:

```
eos <- 84 #set end of summer length
data <- temps[,2:21]
first_breach <- c()
for (year in seq(ncol(data))){

  summer_mean <- mean(data[1:eos,year])
  summer_std <- sd(data[1:eos,year])
  cusum_year <- cusum(data[1:eos,year], newdata = data[(eos+1):123,year], std.dev = summer_std, decision.interval
= 10, plot =F)

  first_breach[year] <- cusum_year$violations$lower[[1]]
}

plot(first_breach,
     ylab = 'Days After Jul 1st',
     xlab = 'Years After 1995',
     main = 'Last Day of Summer by Year')
```



Conclusion

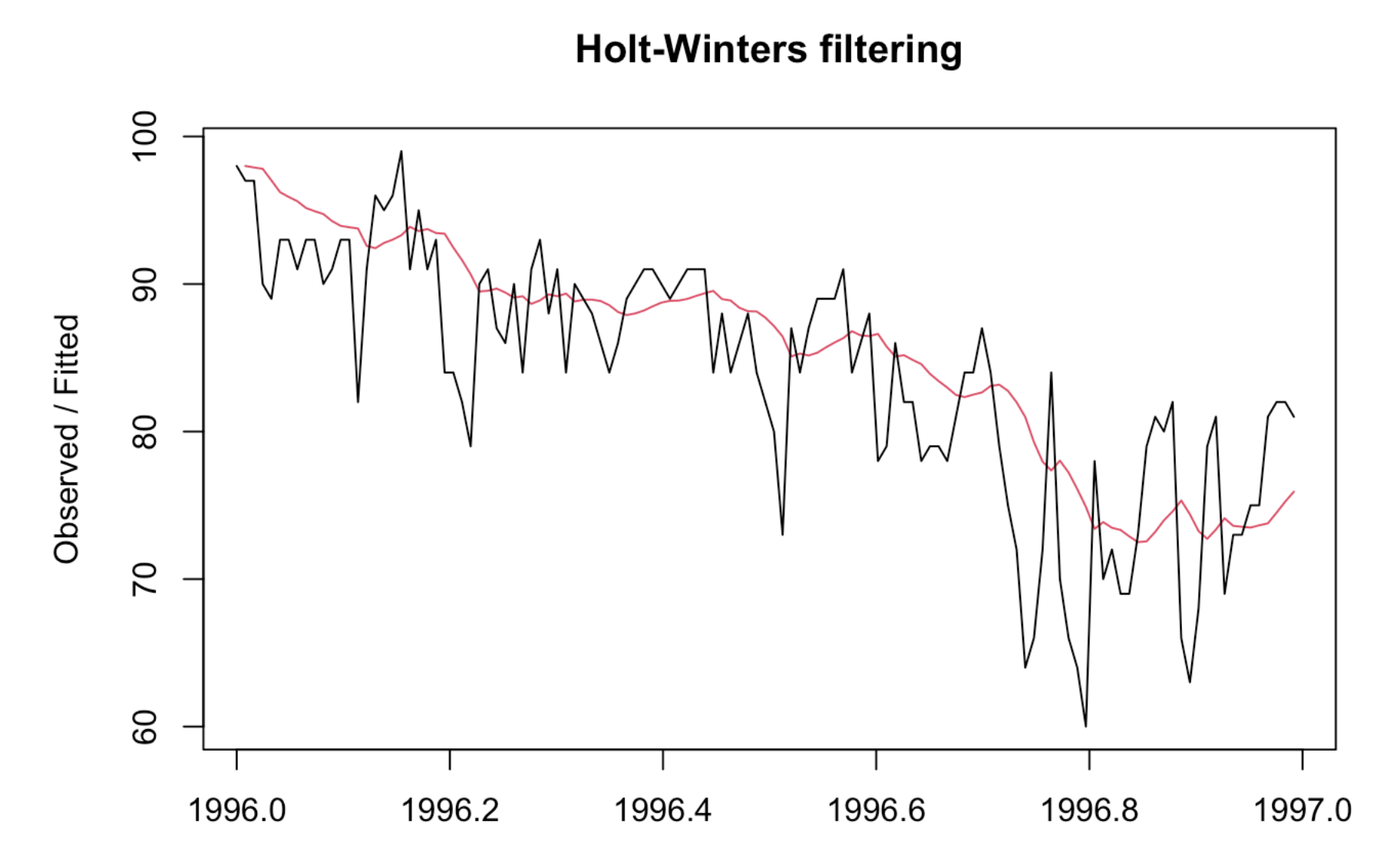
As we can see in the above plot, there is no dicernable trend in the summer end date. Because temperature data is so noisy, we should consider smoothing our data out to get a clearer picture of the trend.

Smoothing Each Year

Next, I'd like to use single exponential smoothing on each year, considering only the data from a given year for the smoothing. Note, we cannot use double/triple (trend/seasonal) smoothing in this approach because we are only using one season for each smoothing operation.

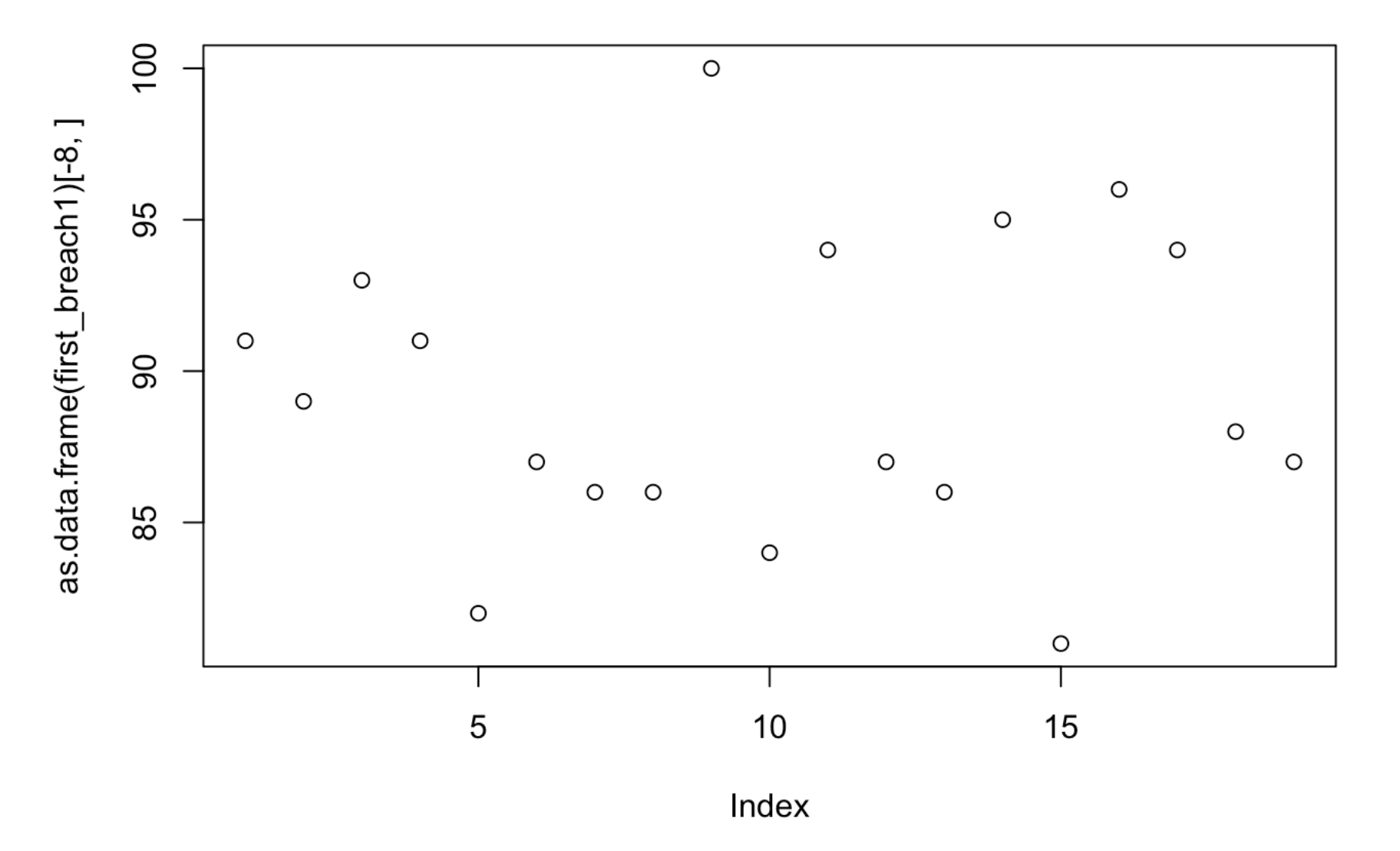
For the single exponential smoothing, we must provide a value for alpha, as it defaults to one, which would just be the observed value. I have chose a value much closer to 0, as we know that intrinsically temperature data can be very unpredictable day to day, thus we want to weight our St value close to the previous day's temperature.

```
eos <- 84 #set end of summer length
data <- temps[,2:21]
first_breach1 <- c()
for (year in seq(ncol(data))){

  #use HoltWinters to smooth the data
  data_for_ts <- as.vector(unlist(data[,year]))
  ts <- ts(data_for_ts, frequency = 123, start =(1995 + year))
  hw <- HoltWinters(ts, alpha = .1, gamma =F, beta = F)
  if (year ==1){plot(hw)} #plot one year of smoothed data for explanation

  #store the smoothed xhat values into a dataframe
  values <- as.data.frame(fitted(hw))[,1]
  values <- as.data.frame(values)

  #run a CUSUM model on the new smoothed temperatures
  summer_std <- sd(values[1:eos,1])
  summer_center <- mean(values[1:eos,1])
  cusum_year <- cusum(values[1:eos,1], newdata = values[(eos+1):121,1], center = summer_center, std.dev = summer_
std, decision.interval = 10, , se.shift = 2, plot =F)

  #error handling, if no lowerbound breach, then consider the last day of the dataset as the last day of summer
  if (length(cusum_year$violations$lower) == 0){
    first_breach1[year] <- 123
  }
  else{
  first_breach1[year] <- cusum_year$violations$lower[[1]]
  }


  #fitted(hw)
}
```



The above red line in this chart shows what the single exponential smoothing did to our data. We then used those smoothed temperatures and ran a CUSUM (using the above discussed values of mu/C/T), and the below plotted the calculated end of summer dates.

```
#plot the data, exclude the 8th year- after examination, an unusually cold start of the summer caused a breach in
the first week of summer, so exluding that datapoint rather than rewriting the above code to handle it, since it
is only one datapoint in an overall trend.
plot(as.data.frame(first_breach1)[-8,])
```



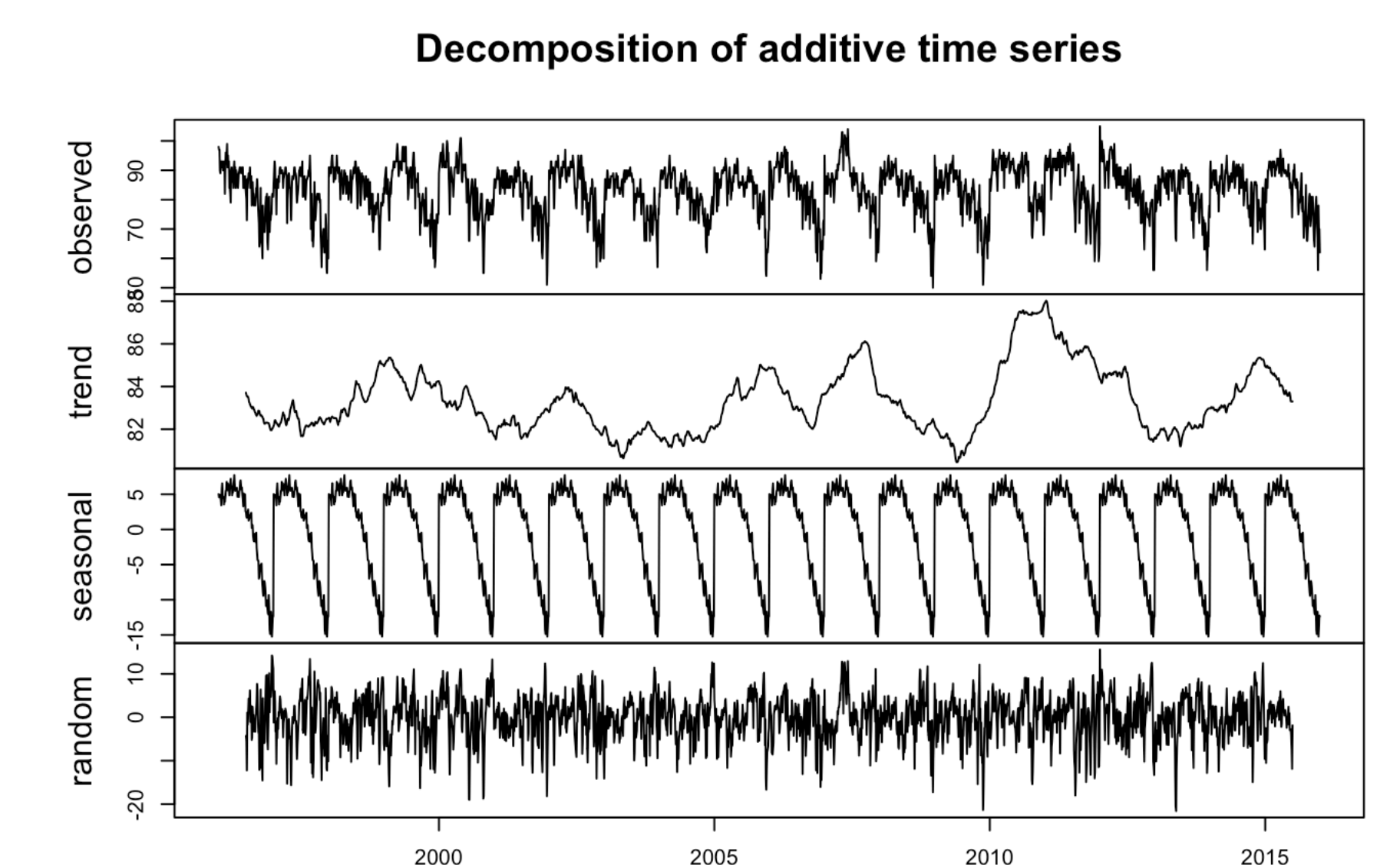
Conclusion

Interestingly enough, using smoothed data gives us once again end of summer dates with no pattern. However, there is one last trick we should consider here to tell if there is in fact a change in the summer end date year over year.

Holt-Winters Triple Exponential Smoothing

Holt Winters give's us a little more insight into the data. Lets first dump the temperature data into a time series and do some exploratory data analysis to make a prediction.

```
#put the data into a time series
data <- temps[,2:21]
data_for_ts <- as.vector(unlist(data[,1:20]))
full_ts <- ts(data_for_ts, frequency = 123, start = 1996)

plot(decompose(full_ts))
```



The above decomposition of the data shows some interesting points, but primarily it looks like we have a strong seasonality to our data, as shown by the apparent cycles in the seasonality column.

Next, we can use Holt-Winters triple exponential smoothing to extract out that seasonality component from our data. Essentially, that will tell us the slope of the seasonality in the above chart. I'll break this down further below.

First, lets see which model, additive or multiplicative, returns the lower SSE.

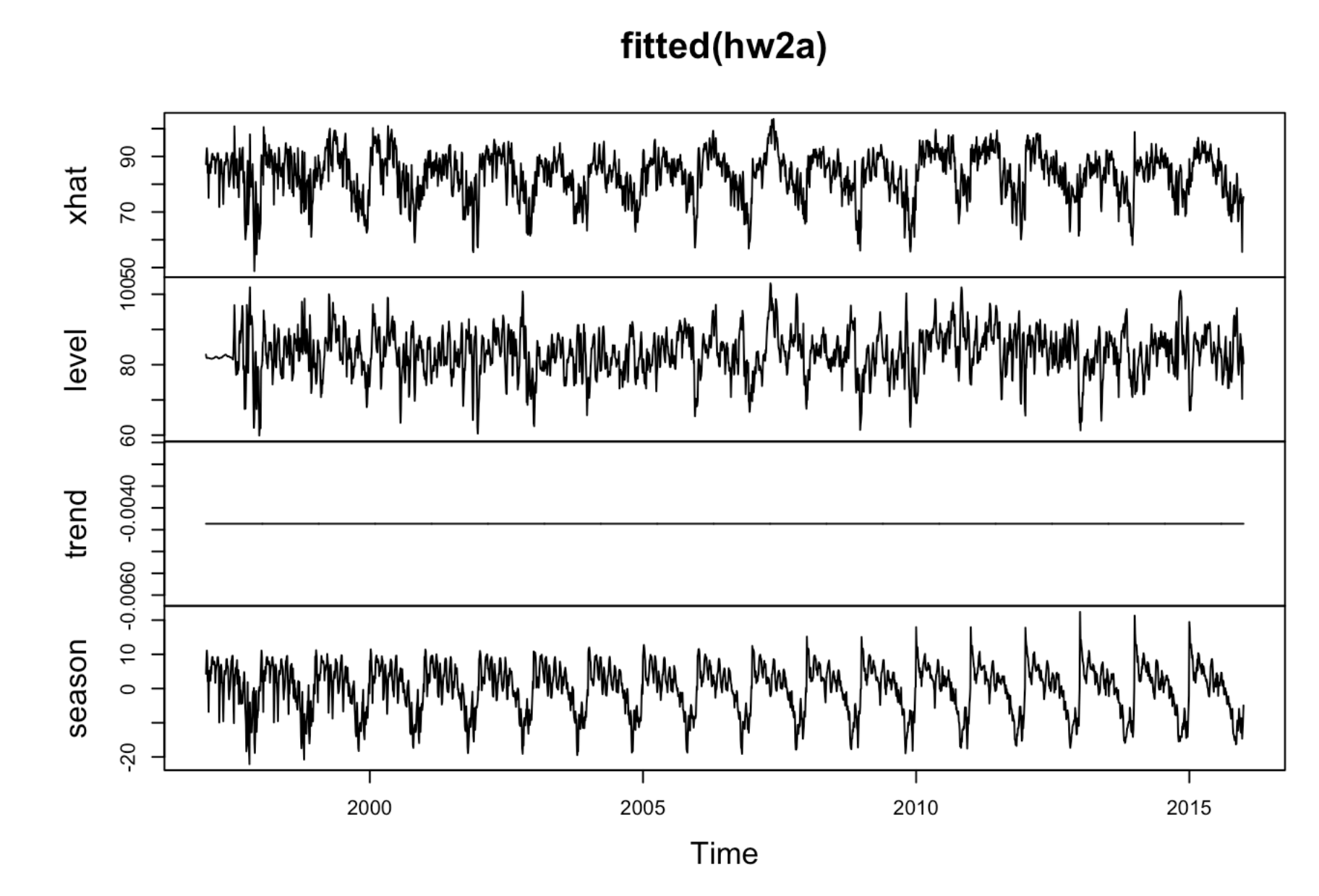
NOTE: I have not input a value for alpha/gamma/beta. The HoltWinters() function takes in a data set and optimizes those parameters for us to get the closest fit (lowest SSE). Because we are not using HW in this case for forecasting, we do not need to be concerned if the model is overfit, which may be a concern if we were forecasting into the future. For these purposes, optimized values of these variables is ideal.

```
hw2a <- HoltWinters(full_ts, seasonal = "additive")
hw2m <- HoltWinters(full_ts, seasonal = "multiplicative")

print(c(hw2a$SSE, hw2m$SSE))
```

```
## [1] 66244.25 68904.57
```

Looks like the additive seasonality gives a lower SSE, so we will use that for our below analysis as the smoothed data is better fit to the actual data.

```
plot(fitted(hw2a))
```



Here, we can see some new insights. First, we can see there is infact no trend to the smoothed data over the 14 years (the first year is used as base data). This essentially means there is virtually no upward trend in the overall temperature in the summer, i.e. it is not getting hotter or colder *on average* from the months July through October. However, that doesn't mean we don't have more extreme hot/cold days in that time period.

Next lets look at the seasonality. This is essentially the slop of the seasonality shown in the decomposition chart above. We can now see there is an increase in the seasonal factor towards the latter half of our data. But what does that *mean?* Well, essentially, the seasonal component is getting stronger as the years go on.

Hot dog Example

If we were to think of the seasonality of selling hot dogs at a baseball game over 20 years, and saw the number sold going up every year, that could be driven by a number of things. If we did a similar analysis as above, we may see a few things. An increase in *trend* over the years would indicate that there may just be more people attending the games over the years.

But an increase in seasonality would mean there are certain points in the season where we are selling more hot dogs. Maybe there is a special that runs during pre-season, and an multiplicative seasonality of 1.2 for those months would imply we are selling 20% more hot dogs because of that special. Now if we were to look and see when that seasonality factor dropped below 1.2, that would tell us when the seasonality component no longer impacts hot dog sales, i.e., the special ended.

We can apply that same logic to temperature data. By looking at when the seasonality (time of year) no longer impacts the temperature being near the mean summer temperatures, we can make a call on when it is no longer summer just by looking at when the seasonality drops below the average summer seasonality.

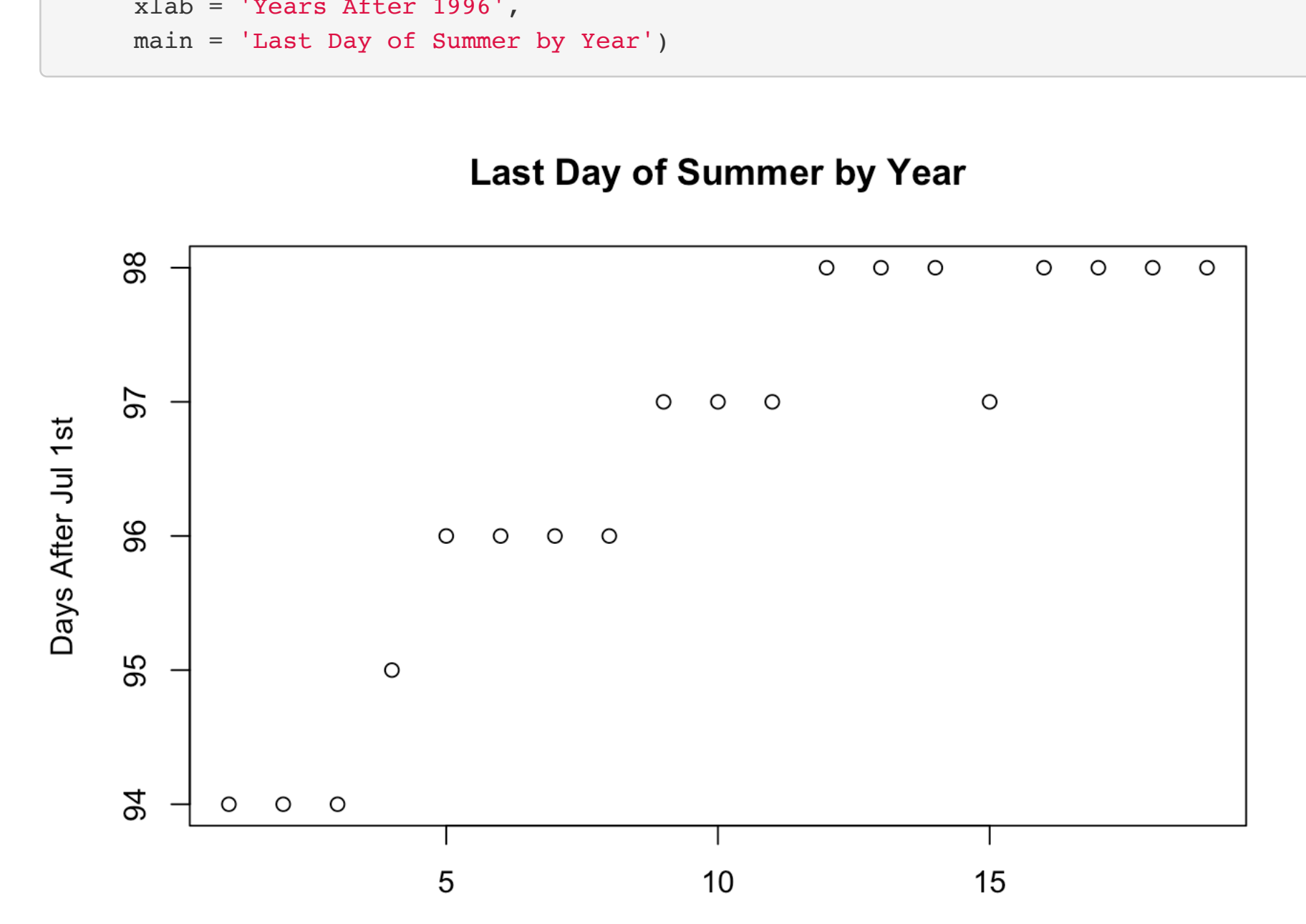
Temperature Data

So lets break our data into a time series and do a Holt-Winters Analysis, and see when the CUSUM of the seasonality breaches the lower bound for each year.

```
#pull the seasonality factors out into a new object
hwa.seasonality <- matrix(hw2a$fitted[,4], nrow = 123)
rownames(hwa.seasonality) <- temps[,1]
colnames(hwa.seasonality) <- colnames(temps[3:21])
```

Now, run a CUSUM model using the same mu/C/T values as our first CUSUM model with no smoothing, because those assumptions still hold.

```
eos <- 84 #set end of summer length
data <- hwa.seasonality
first_breach2 <- c()

for (year in seq(ncol(data))){


  #run a CUSUM model on the new smoothed temperatures
  summer_std <- sd(data[1:eos,1])
  summer_center <- mean(data[1:eos,1])

  cusum_year <- cusum(data[1:eos,year], newdata = data[(eos+1):121,year], center = summer_center, std.dev = summe
r_std, decision.interval = 10, , se.shift = 2, plot =F)

  #store the last day of summer into a vector to plot
  first_breach2[year] <- cusum_year$violations$lower[[1]]


}

plot(first_breach2,
     ylab = 'Days After Jul 1st',
     xlab = 'Years After 1996',
     main = 'Last Day of Summer by Year')
```



At last, we have a clear pattern, summer is progressively ending later and later each year! By only examining the seasonality of the temperatures, we strip out a lot of noise from the data. For example, on the extreme end, if summer got hotter every year by 5 degrees on average, just using the data in raw format would be difficult to say when summer was ending using the first year as base data, because the *trend* of the temperature would cloud our judgement of when summer was ending. But introducing the seasonality adjustment, and examining just how much of the temperature change is due to *only seasonality* gives us a much clear picture, which is what we have done here.