ISyE 6501-HOMEWORK 4

Group Members:
Jinyu, Huang | jhuang472@gatech.edu | GTID: 903522245
Chengqi, Huang | chengqihuang@gatech.edu | GTID: 903534690
Yuefan, Hu | yuefanhu@gatech.edu | GTID:903543027
Jingyu, Li | alanli@gatech.edu | GTID: 903520148

Qusetion 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?.

Answer

For example, we can use exponential smoothing to deal with the randomness in the daily number of page views of a certain main page of the web portal. Our model will be: (PV=page views)

$$PVS_t = \alpha PVX_t + (1 - \alpha)PVS_{t-1}$$

We propose that the baseline volume of page views is more meaningful for us to get a impression of how popular the web portal is. And as for the daily number of page views on a certain day may varies much, resulting from like breaking big news, holidays or users random behavior. So we expect that α should be closer to 0 to lower the weight of oberseved number of page views on Day t.

Qusetion 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.)

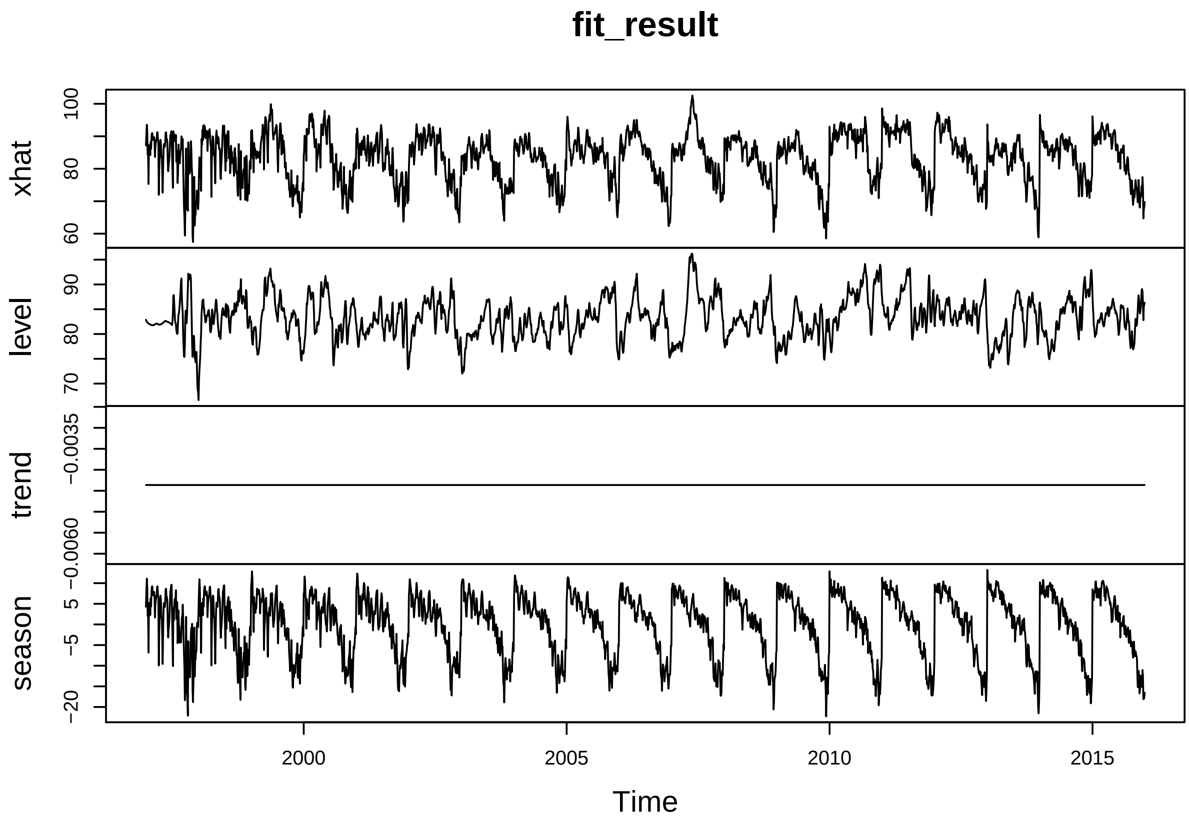
Answer

We import the data and change them into time series format. The frequency is the number of days from Jul 1st to Oct 31st. And the seasonality starts from Year 1996.

```
> data = read.table("temps.txt", header=TRUE)  # import data
> data_one_column = unlist(data[,2:ncol(data)])  # transform each year's data into one column
> data.ts = ts(data_one_column, frequency=nrow(data), start=c(1996,1))
>                                       # transform to time series data format
```

Based on HW3, we got know that the fluctuation in daily-high-temperature data for Atlanta is large. So we choose a small $\alpha(0.2)$ to smooth the daily various. Other parameters in the model we leave them as default. By using this exponential smoothing model, we split the seasonal component from the total fluctuation. To judge whether the unofficial end of summer has gotten later over the 20 years, we apply CUSUM model to the seasonal component.

```
> model = HoltWinters(data.ts, alpha=0.2)  # exponential smoothing model
> fit_result = model$fitted
> plot(fit_result)
```




```
> season_raw = fit_result[,"season"]  # seasonal component data
> # transfor time series data back to matrix
> data_season = matrix(season_raw, nrow=nrow(fit_result)/(ncol(data)-2),ncol=ncol(data)-2)
> colnames(data_season) = colnames(data[,c(-1,-2)])
```

Based on our exploration in HW3, we choose the average temperature of July as the baseline(μ) and the calculation starts from 1st, Aug. Then we assume that if data decrease 15 from the baseline, it means summer ends and weather starts to cool off. So for the threshold in each year's model, $T = -15$. As for C, based on the range of day-by-day change, we choose $C = 7$. We build the CUSUM function in R. This function use every year's data as input and return the row number when $S_t \leq -15$.

```
> cusum_model = function(data, column_num, T=-15, C=7){
+   base = mean(data[1:31, column_num])  # July average
+   s = 0  # cusum value
+   for (i in seq(32,nrow(data))){  # traverse from Aug-1
+     new_s = s-(base-data[i, column_num]-C)
+     if (new_s < 0){s = new_s}
+     else {s = 0}
+     if (s <= T){return (i)}  # return row number = number of days after Jun 30
+   }
+   return (0)
+ }
```

We use the function above to get number of days after Jun-30 in each year when summer ended. According to the plot, we can see that the date of summer ending did not get later in pase 20 years.

```
> days_after_Jun = matrix(nrow=ncol(data_season), ncol=2)
> colnames(days_after_Jun)=c("Year", "Days after Jun-30 when getting cooler")
> for (i in seq(1,nrow(days_after_Jun))){
+   days_after_Jun[i,1] = 1996+i
+   row_num = cusum_model(data_season,i)
+   if (row_num != 0){
+     days_after_Jun[i,2] = row_num
+   }
+   else{days_after_Jun[i,2] = NULL}
+ }
> plot(days_after_Jun,
+      type="b",
+      ylim=c(0,90),
+      main="Number of days after Jun-30 when summer ends")
> abline(h=c(60,80), lty=2)
```