Homework3

Chen Yi-Ju(Ernie)

2020/6/4

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

I would consider oil price being a good situation to use exponential smoothing. would be somewhere closer to 1 than 0 because the oil price(under current situations) are known to have big fluncuations due to random events happening.

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.

My answer is No. According to exponential smoothing, summer has not gotten latter over the 20 years.This is the proccess for proving it.

Setup:

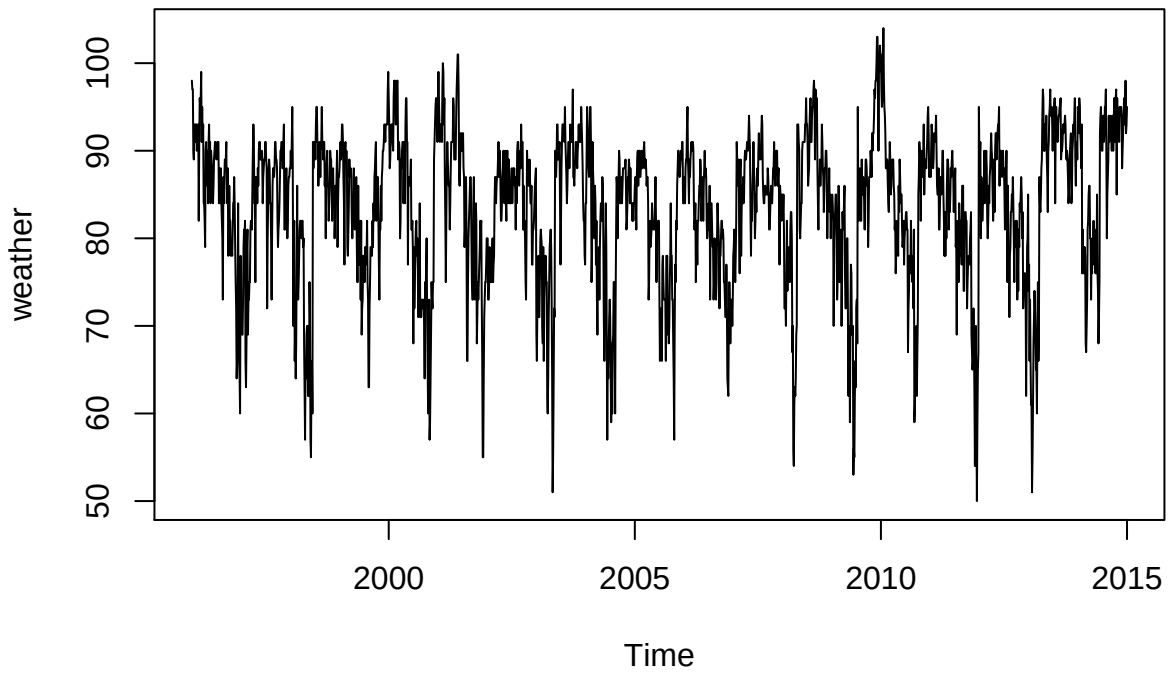
```
setwd("D:/ernie/self-study/GTxMicroMasters/Introduction to Analytics Modeling/week3/homework")
library(magrittr)
library(tidyverse)
library(lubridate)
library(corrplot)
library(leaps)
```

turning data into time. series format

```
weather <- data.frame(read.table("temps.txt" , header = T))%>%
  select(.,- DAY) %>%
  unlist()%>%
  as.vector()%>%
  ts(start = 1996 , end = 2015 , frequency = 100)
```

Graphically represented: It is hard to see actual trends

```
plot(weather)
```



Putting down the HoltWinters Function

```
HoltWinters(weather)
```

```
## Holt-Winters exponential smoothing with trend and additive seasonal component.
##
## Call:
## HoltWinters(x = weather)
##
## Smoothing parameters:
##  alpha: 0.7015953
##  beta : 0
##  gamma: 0.6813504
##
## Coefficients:
##              [,1]
## a    102.12556505
## b     -0.01546985
## s1   -10.48519724
## s2   -12.44318212
## s3   -10.99764581
## s4   -12.75209947
## s5   -12.62974056
## s6   -11.29182605
## s7   -10.93658628
```

```
## s8    -5.40303859
## s9    -6.79845771
## s10   -3.39388173
## s11   -6.86132351
## s12   -6.30334825
## s13   -9.27608984
## s14   -8.60454327
## s15   -9.35894641
## s16  -11.98293559
## s17  -10.08152712
## s18   -5.73359214
## s19   -4.21236012
## s20   -6.33379226
## s21   -5.18509908
## s22   -1.14584131
## s23   -2.59594123
## s24   -2.24708371
## s25   -1.20692084
## s26    1.98058414
## s27    1.13684508
## s28    4.26424229
## s29    4.68465208
## s30    7.72445661
## s31    4.97002347
## s32    7.89317690
## s33    3.94494859
## s34    3.91773586
## s35    1.96066900
## s36    3.12215825
## s37    3.36315823
## s38    3.39629121
## s39    7.73828726
## s40    7.98122071
## s41    8.30522935
## s42    9.35335153
## s43    9.59278999
## s44    7.23771484
## s45    7.81844143
## s46    5.46541873
## s47    5.04169526
## s48    5.02802732
## s49    5.36288062
## s50    7.31875821
## s51    7.81973186
## s52    4.87303155
## s53    6.23219406
## s54    3.63600461
## s55    6.98631128
## s56    6.72090297
## s57    5.95574278
## s58    5.40068097
## s59    2.13451835
## s60    2.66065990
## s61    1.01986035
```

```
## s62    2.08035759
## s63    2.76693224
## s64    1.80908389
## s65    3.14082126
## s66    1.13910230
## s67    1.42982098
## s68    0.15996809
## s69    0.73405805
## s70    4.42836996
## s71    4.11636798
## s72    4.56871751
## s73    4.89836713
## s74    5.94492884
## s75    4.02735738
## s76    4.43237712
## s77    4.85538584
## s78    5.15469049
## s79   -0.32306323
## s80    2.61750562
## s81   -0.96168460
## s82    0.75065521
## s83    1.48361721
## s84   -1.28613918
## s85   -0.38311462
## s86    0.17466496
## s87    0.76124762
## s88    3.07396243
## s89    2.29772241
## s90   -0.13094343
## s91   -0.94184454
## s92   -1.16934790
## s93   -1.93458425
## s94   -5.57050542
## s95  -11.59120257
## s96   -7.82643736
## s97   -3.97388600
## s98   -7.74355298
## s99   -5.55287173
## s100  -7.58855363
```

The main focus is here : Smoothing parameters: alpha: 0.7015953 beta : 0 gamma: 0.6813504

the beta of the Holt Winters Function is 0, indicating no overall trend, which matches our intuition.

Question 8.1

Describe a situation or problem from your job, everyday life, current events, etc., for which a linear regression model would be appropriate. List some (up to 5) predictors that you might use.

A good opportunity would be predicting a baseball team's winning chances. It would be through parameters including: team average ERA(Earned run average) team average batting average team average slugging average and team average fielding percentage.

Question 8.2

Using crime data , use regression (a useful R function islm or glm) to predict the observed crime rate in a cityin a city with the following data:

M = 14.0 So = 0 Ed = 10.0 Po1 = 12.0 Po2 = 15.5 LF = 0.640 M.F = 94.0 Pop = 150 NW = 1.1 U1 = 0.120 U2 = 3.6 Wealth = 3200 Ineq = 20.1 Prob = 0.04 Time = 39.0 Show your model (factors used and their coefficients), the software output, and the quality of fit.

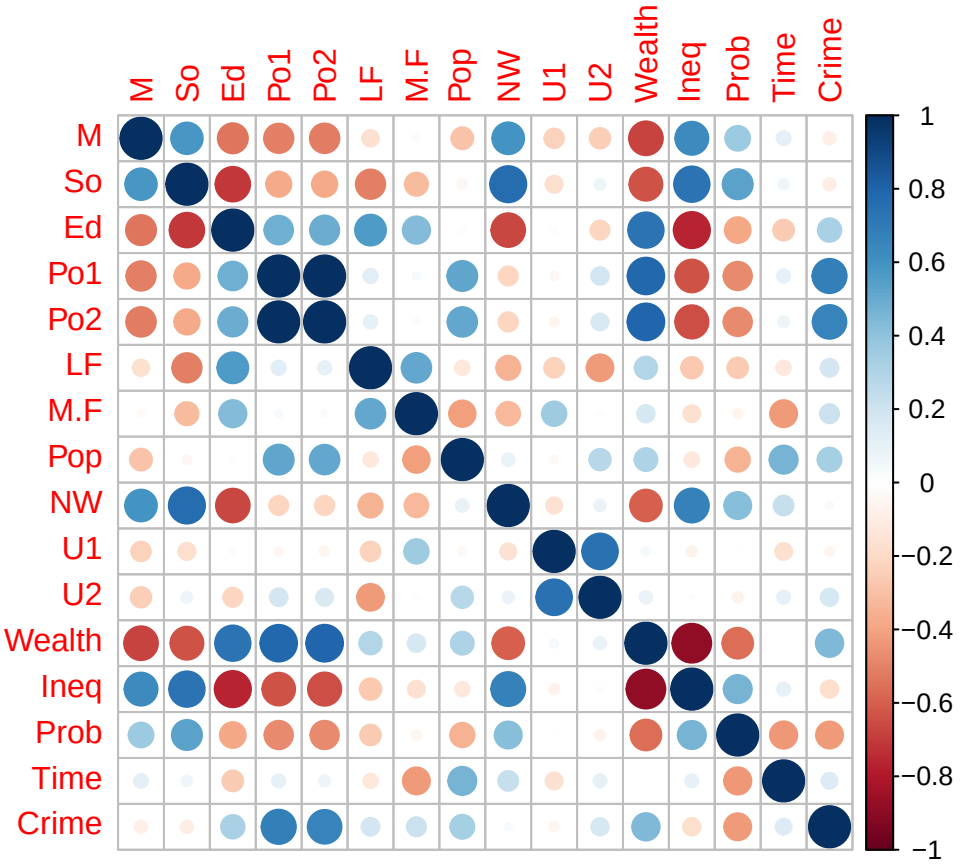
Answer : I created a model omitting parameters that are too low in correlation with the results or are highly correlated with other parameters, making them un-independent. The model I created has an 75% R-squared value and the prediction according to the model is 1177.978.

Read Data

```
crime <- read.table("uscrime.txt" , header = TRUE)%>%
  data.frame()
```

Showing the correlation between predictors

```
pl1 <- corrplot(cor(crime))
```



We eliminate predictor P02 due to its high correlation with p01

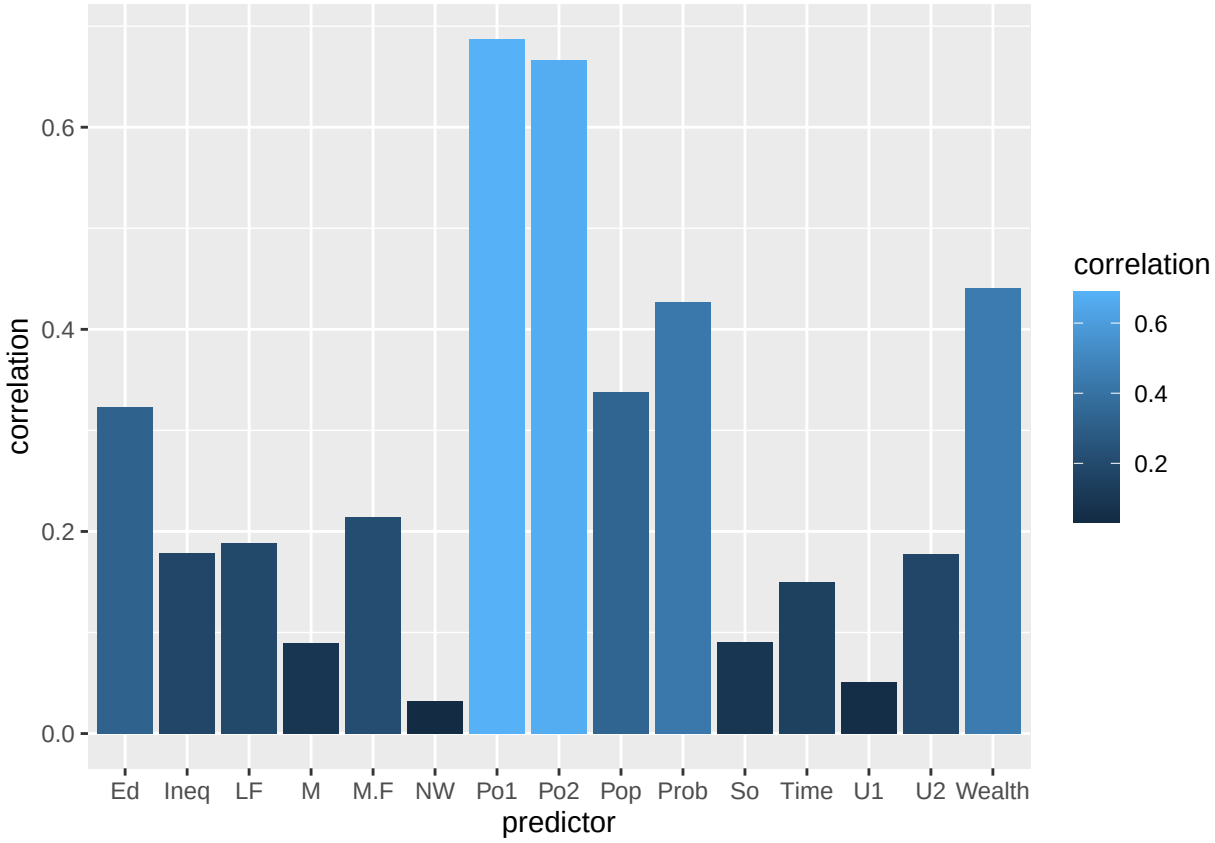
Numeric and Graphical representation of correlation with the crime variable

```
cor_relation <- abs(cor(crime$Crime , crime[,1:15]))%>%
  data.frame()
cor_relation <- cor_relation%>%
```

```
  gather(predictor, correlation)
cor_relation
```

```
##    predictor correlation
## 1          M  0.08947240
## 2         So  0.09063696
## 3         Ed  0.32283487
## 4        Po1  0.68760446
## 5        Po2  0.66671414
## 6         LF  0.18886635
## 7        M.F  0.21391426
## 8        Pop  0.33747406
## 9         NW  0.03259884
## 10        U1  0.05047792
## 11        U2  0.17732065
## 12    Wealth  0.44131995
## 13      Ineq  0.17902373
## 14      Prob  0.42742219
## 15      Time  0.14986606
```

```
pl2 <- ggplot(data = cor_relation, aes( x  = predictor , y = correlation , fill = correlation )) +
  geom_col()
pl2
```



We remove NW,U1 and So due to low correlation

Constructing the model:

```
model <- lm (data = crime , Crime ~ Ed + Ineq + LF + M + M.F +Po1 + Pop + Prob  + Time + Ed)
summary(model)
```

```
##
## Call:
## lm(formula = Crime ~ Ed + Ineq + LF + M + M.F + Po1 + Pop + Prob +
##     Time + Ed, data = crime)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -468.62 -100.73   -6.44  139.91  520.35
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5189.5782  1460.9341  -3.552 0.001063 **
## Ed            140.9730    57.8900   2.435 0.019823 *
## Ineq           68.7477    15.8765   4.330 0.000109 ***
## LF           -609.2340  1065.0117  -0.572 0.570751
## M              68.3357    35.3331   1.934 0.060784 .
## M.F            17.8666    15.2790   1.169 0.249738
## Po1           126.5215    17.3893   7.276 1.22e-08 ***
## Pop            -0.6526     1.2716  -0.513 0.610833
## Prob        -4006.6838  2033.8562  -1.970 0.056359 .
## Time            1.7858     6.6248   0.270 0.788995
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 213.8 on 37 degrees of freedom
## Multiple R-squared:  0.7543, Adjusted R-squared:  0.6945
## F-statistic: 12.62 on 9 and 37 DF,  p-value: 7.275e-09
```

The summary results show that R- squared is at about 75% which is a good enough result. One thing to notice is however the Variables LF,M.F., Pop and Time are not statistically significant. Nevertheless, We test out the model using the given numbers:

```
test <- data.frame(   M = 14.0,So = 0,Ed = 10.0,Po1 = 12.0,
                      Po2 = 15.5,LF = 0.640,M.F = 94.0 ,Pop = 150,
                      NW = 1.1,U1 = 0.120,U2 = 3.6,Wealth = 3200,Ineq = 20.1,Prob = 0.04,Time = 39.0
                      )
predict(model,test)
```

```
##        1
## 1177.978
```

Therefore my prediction is 1177.978.