Thomas Huelsnitz
ISYE 6501
Week 3 Homework

HW3

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

In my current role I work on forecasting how many field engineers of different skillsets we will need on a weekly basis over the next two years. This is done using a model which is trained on several years of historical timesheet data. The type of work that these engineers is doing is extremely seasonal and in addition to the seasonality, the historic data can be quite 'jagged'. There is not a lot of noise in the data, but slightly smoothing the training data improves the performance of the model. I think that the jaggedness in the unsmoothed data is a result of a lot of processes which we cannot observe so it can just be thought of as random noise in this case. I use an exponential smoothing algorithm to smooth the historical data before feeding it into the model. I do not know what the 'alpha' value is in this particular case, but I imagine that it is closer to 1 than 0 (assuming the equation is of the form in the lectures) since there is not a lot of noise.

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.)

We begin with loading required packages and the dataset. Also define the CUSUM function to use and a nice function for showing regression lines on scatter plots.

```
library(ggplot2)
library(reshape)

temps <- read.table("7.2tempsSummer2018.txt", header=TRUE)

temps$DAY <- as.Date(temps$DAY, '%e-%b')

cusum <- function(data, P, Q){
  ans <- data.frame(S=double(), alarm=integer())
  ans[nrow(ans)+1,] <- c(0,0)
  mu <- mean(data)
```

```
  std <- sd(data)
  C <- P * std
  thresh <- Q * std
  for (i in 2:length(data)){
    S <- max(0, ans[[i-1,1]] + (mu - data[i] - C))
    alarm <- S > thresh
    ans[nrow(ans)+1,] <- c(S, alarm)
  }
  ans
}

lm_eqn <- function(linmodel){
  m <- linmodel;
  eq <- substitute(italic(y) == a + b %.% italic(x)*","~~italic(r)^2~"="~r2,
                   list(a = format(coef(m)[1], digits = 2),
                        b = format(coef(m)[2], digits = 2),
                        r2 = format(summary(m)$r.squared, digits = 3)))
  as.character(as.expression(eq));
}
```

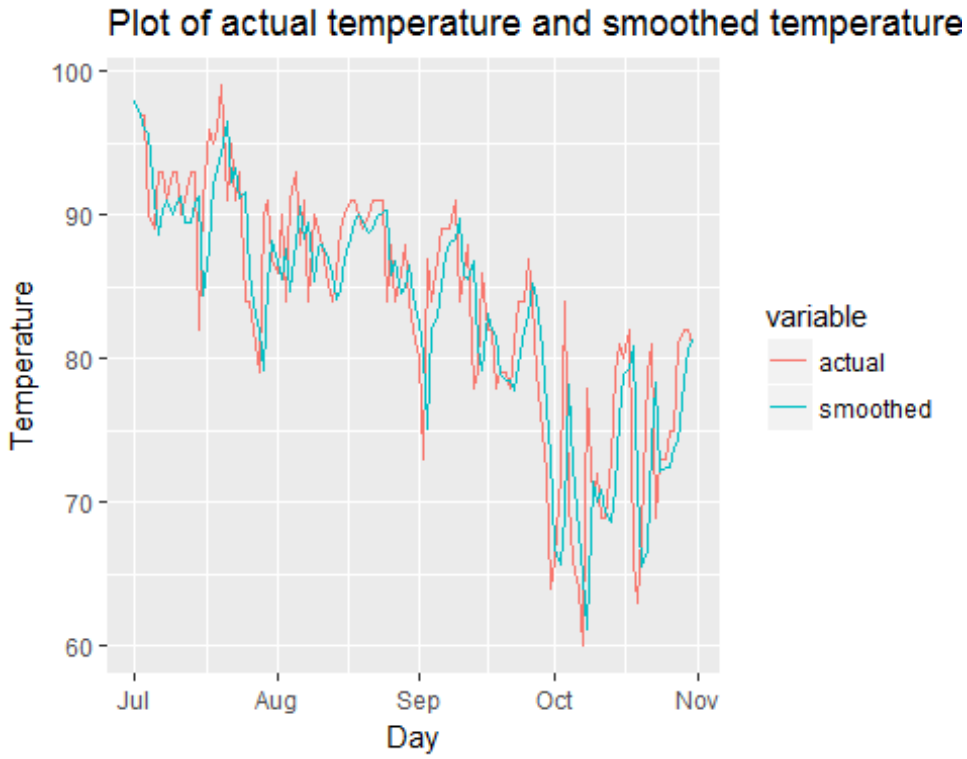
Analysis begins with a visual inspection of smoothing on temperatures from 1996. This uses alpha and beta parameters.

```
datchunk <- temps[,2]
smooth <- HoltWinters(datchunk, gamma=FALSE)
smootheddatchunk <- c(datchunk[1],datchunk[2],smooth$fitted[,'xhat'])

comparedat <- data.frame(DAY=temps['DAY'], actual=datchunk,
smoothed=smootheddatchunk)

meltedcompare <- melt(comparedat, id = 'DAY')
ggplot(meltedcompare, aes(x = DAY, y = value, colour = variable,
group=variable)) +
  geom_line() +
  ggtitle("Plot of actual temperature and smoothed temperature")+
  ylab("Temperature") +
  xlab("Day")
```

Thomas Huelsnitz
ISYE 6501
Week 3 Homework



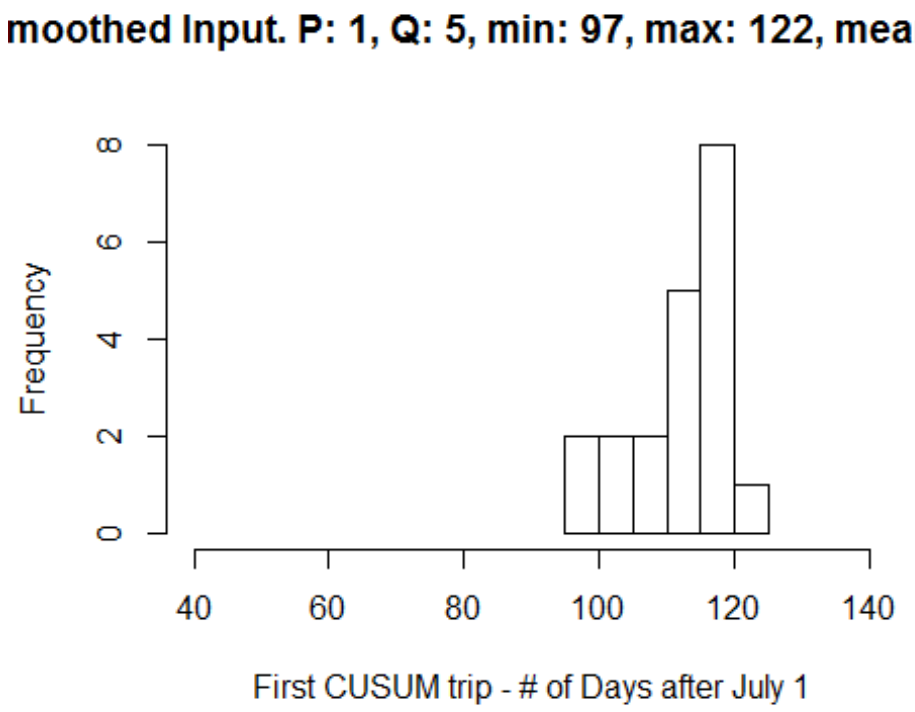
Next we apply HoltWinters smoothing to the raw temperature data for each year. Each year of this smoothed data is then fed into the CUSUM model with the same parameters as in homework 2. The output of this process is a value for each year indicating how many days passed until the CUSUM model registered a change (trip date). The plots below show a histogram of the trip dates and a timeseries plot of them.

```
P <- 1
Q <- 5
firsttriplist <- c()
for (cnum in 1:(ncol(temps) - 1)){
  tempdat <- temps[,cnum + 1]
  # datchunk <- temps[,2]
  smooth <- HoltWinters(tempdat, gamma=FALSE)
  smootheddatchunk <- c(tempdat[1],tempdat[2],smooth$fitted[,'xhat'])
  ans <- cusum(smootheddatchunk, P, Q)
  firsttrip <- which(ans$alarm == 1)[1]
  firsttriplist <- c(firsttriplist,firsttrip)
}

## Warning in HoltWinters(tempdat, gamma = FALSE): optimization difficulties:
## ERROR: ABNORMAL_TERMINATION_IN_LNSRCH

hist(firsttriplist, xlim=c(40,140), xlab="First CUSUM trip - # of Days after
July 1", main=sprintf("Smoothed Input. P: %s, Q: %s, min: %s, max: %s, mean:
%s\n", P, Q, min(firsttriplist), max(firsttriplist), mean(firsttriplist)))
```

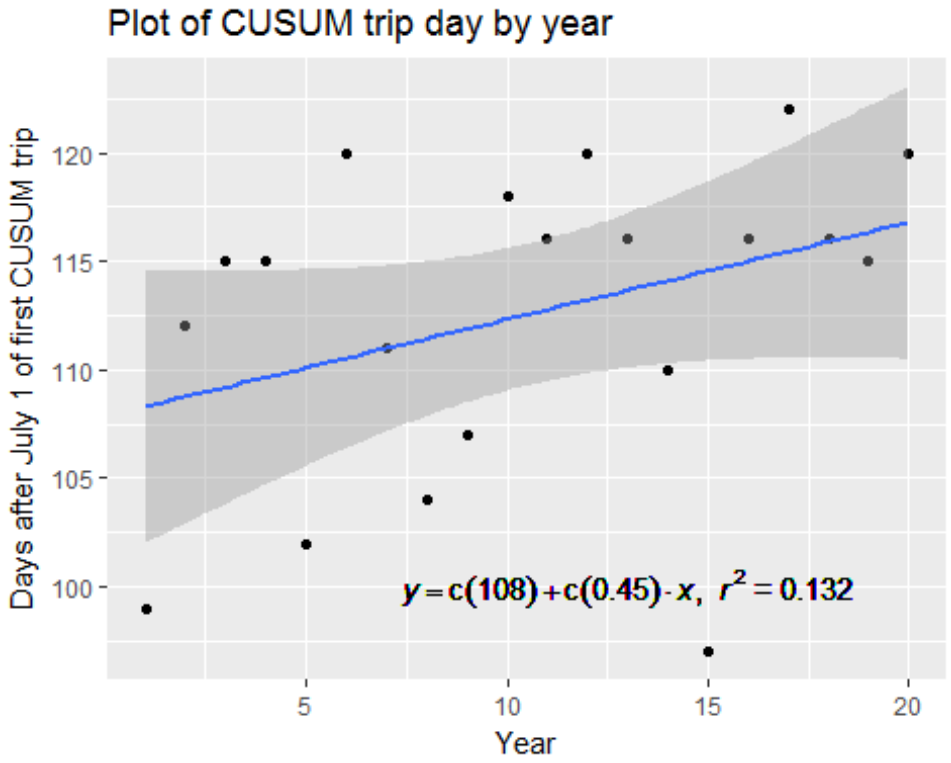
Thomas Huelsnitz
ISYE 6501
Week 3 Homework




```
firsttripdf <- data.frame(x=seq_along(firsttriplist), y=firsttriplist)

linmodel <- lm(y~x, firsttripdf)
ggplot(firsttripdf, aes(x=x, y=y)) +
geom_point() +
geom_smooth(method='lm') +
geom_text(x = 13, y = 100, label = lm_eqn(linmodel), parse = TRUE) +
ggtitle("Plot of CUSUM trip day by year")+
ylab("Days after July 1 of first CUSUM trip") +
xlab("Year")
```

Thomas Huelsnitz
ISYE 6501
Week 3 Homework

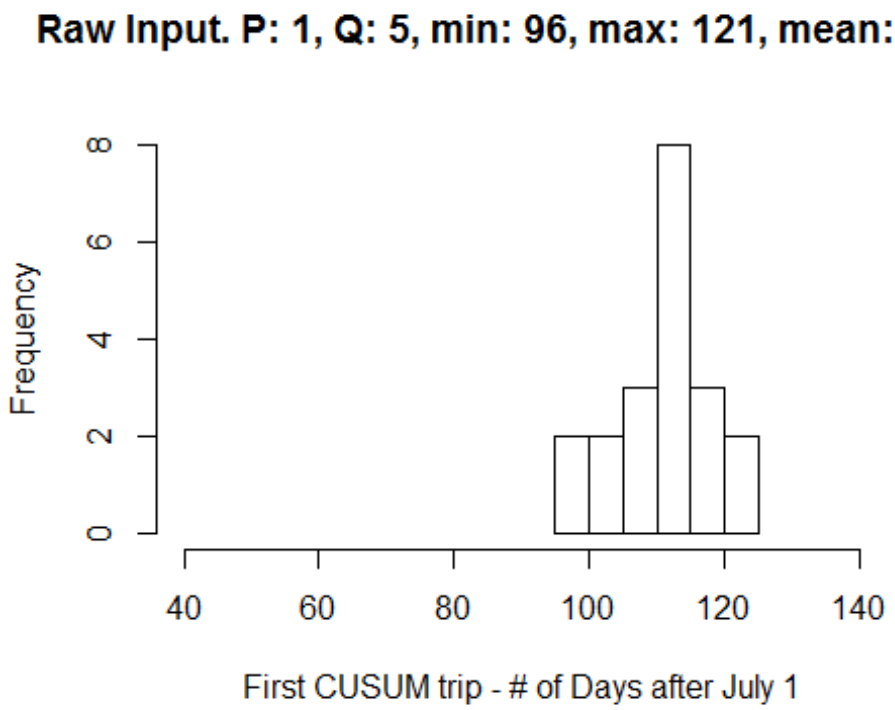


For comparison, below is the same process as above but without smoothing the temperature data first. This is the same process as in the last homework.

```
P <- 1
Q <- 5
firsttriplist <- c()
for (cnum in 1:(ncol(temps) - 1)){
  tempdat <- temps[,cnum + 1]
  ans <- cusum(tempdat, P, Q)
  firsttrip <- which(ans$alarm == 1)[1]
  firsttriplist <- c(firsttriplist,firsttrip)
}

hist(firsttriplist, xlim=c(40,140), xlab="First CUSUM trip - # of Days after
July 1", main=sprintf("Raw Input. P: %s, Q: %s, min: %s, max: %s, mean:
%s\n", P, Q, min(firsttriplist), max(firsttriplist), mean(firsttriplist)))
```

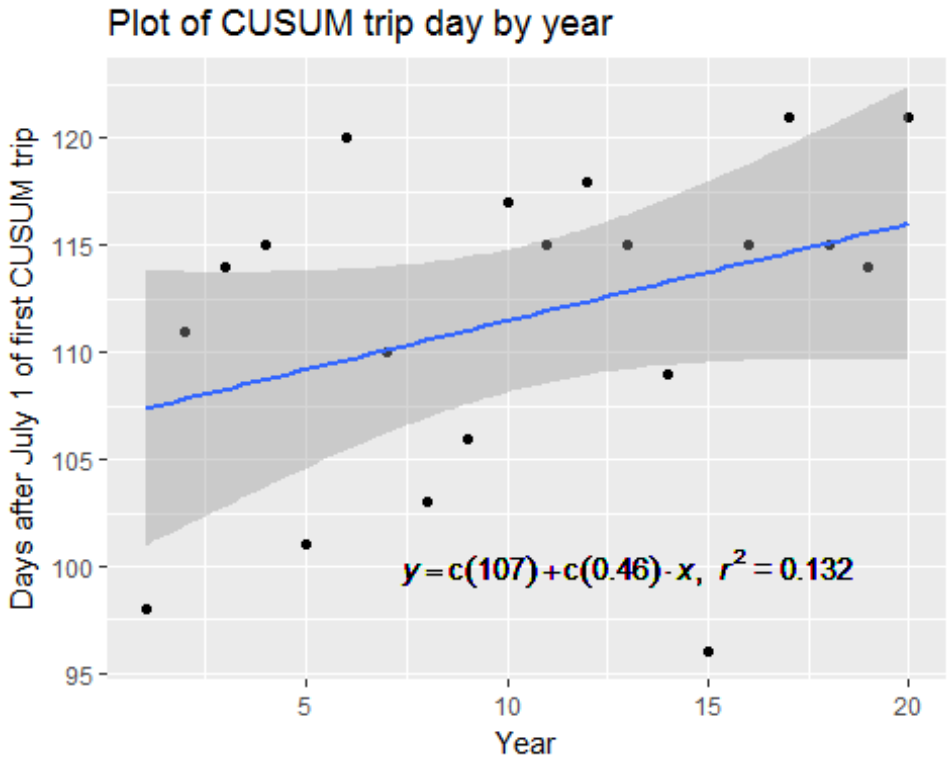
Thomas Huelsnitz
ISYE 6501
Week 3 Homework




```
firsttripdf <- data.frame(x=seq_along(firsttriplist), y=firsttriplist)
linmodel <- lm(y~x, firsttripdf)

ggplot(firsttripdf, aes(x=x, y=y)) +
geom_point() +
geom_smooth(method='lm') +
geom_text(x = 13, y = 100, label = lm_eqn(linmodel), parse = TRUE) +
ggtitle("Plot of CUSUM trip day by year")+
ylab("Days after July 1 of first CUSUM trip") +
xlab("Year")
```

Thomas Huelsnitz
ISYE 6501
Week 3 Homework

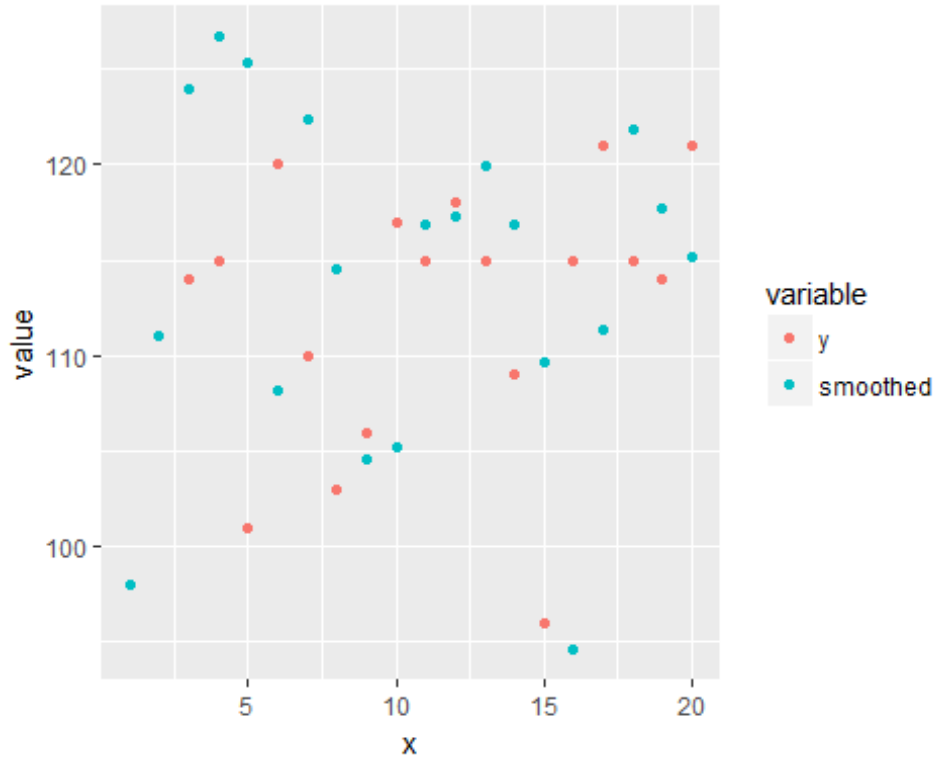


Judging from the results above, applying HoltWinters to the raw data doesn't really seem to change anything from the last lesson's analysis. Let's try raw data -> CUSUM -> HoltWinters and analyze the HoltWinters parameters to see if we can conclude anything about when summer is ending.

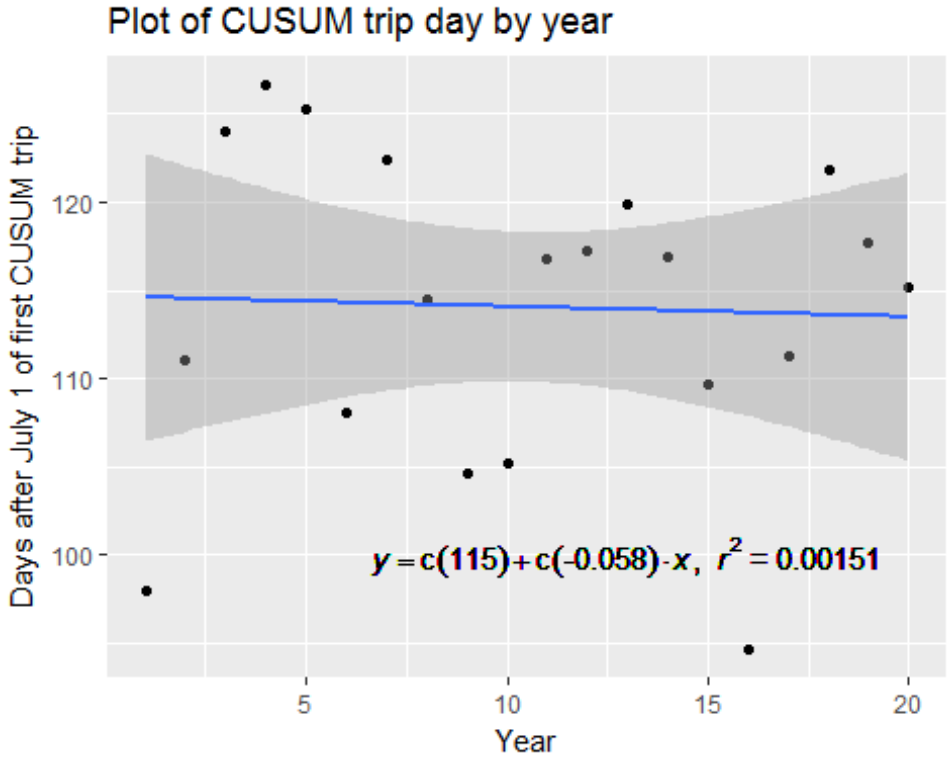
```
smooth <- HoltWinters(firsttriplist, gamma=FALSE)
smoothedfirsttriplist <-
c(firsttriplist[1],firsttriplist[2],smooth$fitted[,'xhat'])
firsttripdf[,'smoothed'] <- smoothedfirsttriplist

meltedfirsttripdf <- melt(firsttripdf, id = 'x')
ggplot(meltedfirsttripdf, aes(x = x, y = value, colour = variable,
group=variable)) +
  geom_point()
```

Thomas Huelsnitz
ISYE 6501
Week 3 Homework




```
linmodel <- lm(smoothed~x, firsttripdf)
ggplot(firsttripdf, aes(x = x, y = smoothed)) +
geom_point() +
geom_smooth(method='lm') +
geom_text(x = 13, y = 100, label = lm_eqn(linmodel), parse = TRUE) +
ggtitle("Plot of CUSUM trip day by year")+
ylab("Days after July 1 of first CUSUM trip") +
xlab("Year")
```



This second approach leads to a regression model with a very low r^2. There is no discernible linear trend in this data.

Neither of the exponential smoothing approaches shown above had an effect on the conclusion from the prior homework - namely that summer seems to be starting later in the year as the years progress in the dataset.

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.

I am in the market for buying a house. Predicting the price of a home could be solved using a regression model. The model could take the following into account:

1. Average price of the k nearest homes
2. Number of bedrooms
3. Number of bathrooms
4. Number of square feet
5. Age of the home

Thomas Huelsnitz
ISYE 6501
Week 3 Homework

8.2

Using the provided crime data, use regression (a useful R function is lm or glm) to predict the observed crime rate in 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.

Start with loading the necessary data and fitting a linear regression model to it. Here we report out the coefficients of the linear model, the r^2 value and the adjusted r^2 value.

```
crime <- read.table("8.2uscrimeSummer2018.txt", header=TRUE)

linmodel <- lm(Crime~., crime)

###Coefficients")###
summary(linmodel)$coefficients
```

```
##                  Estimate   Std. Error     t value     Pr(>|t|)
## (Intercept) -5.984288e+03 1628.3183733 -3.67513362 0.0008929887
## M            8.783017e+01   41.7138664  2.10553902 0.0434433942
## So          -3.803450e+00  148.7551399 -0.02556853 0.9797653725
## Ed           1.883243e+02   62.0883761  3.03316541 0.0048614327
## Po1          1.928043e+02  106.1096757  1.81702882 0.0788919769
## Po2         -1.094219e+02  117.4775356 -0.93142851 0.3588295738
## LF          -6.638261e+02 1469.7288208 -0.45166573 0.6546540941
## M.F          1.740686e+01   20.3538427  0.85521225 0.3989953316
## Pop         -7.330081e-01    1.2895554 -0.56841928 0.5738452309
## NW           4.204461e+00    6.4808922  0.64874725 0.5212791189
## U1          -5.827103e+03 4210.2890365 -1.38401489 0.1762380311
## U2           1.677997e+02   82.3359552  2.03798780 0.0501612829
## Wealth       9.616624e-02    0.1036661  0.92765416 0.3607537824
## Ineq         7.067210e+01   22.7165213  3.11104410 0.0039831365
## Prob        -4.855266e+03 2272.3746212 -2.13664850 0.0406269260
## Time        -3.479018e+00    7.1652752 -0.48553862 0.6307084351
```

```
###r-squared###
cat(summary(linmodel)$r.squared)
```

```
## 0.8030868
```

```
###Multiple r-squared###
cat(summary(linmodel)$adj.r.squared)
```

```
## 0.7078062
```

The r^2 value will always increase as we add new fields to the model. The adjusted r^2 accounts for this by penalizing the score as new fields are added to the model. A large

discrepancy in these two scores is indicative of overfitting. Comparing the two values for this model indicates that it is overfitting the provided data set.

Finally, we predict a crime rate given the test datapoint in the problem statement.

```
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
testpoint <- data.frame(M,So,Ed, Po1, Po2, LF, M.F, Pop, NW, U1, U2, Wealth,
Ineq, Prob, Time)

pred <- predict(linmodel, testpoint)
cat("Prediction: ", pred)

## Prediction:  155.4349
```