Homework 3

6/3/2020

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

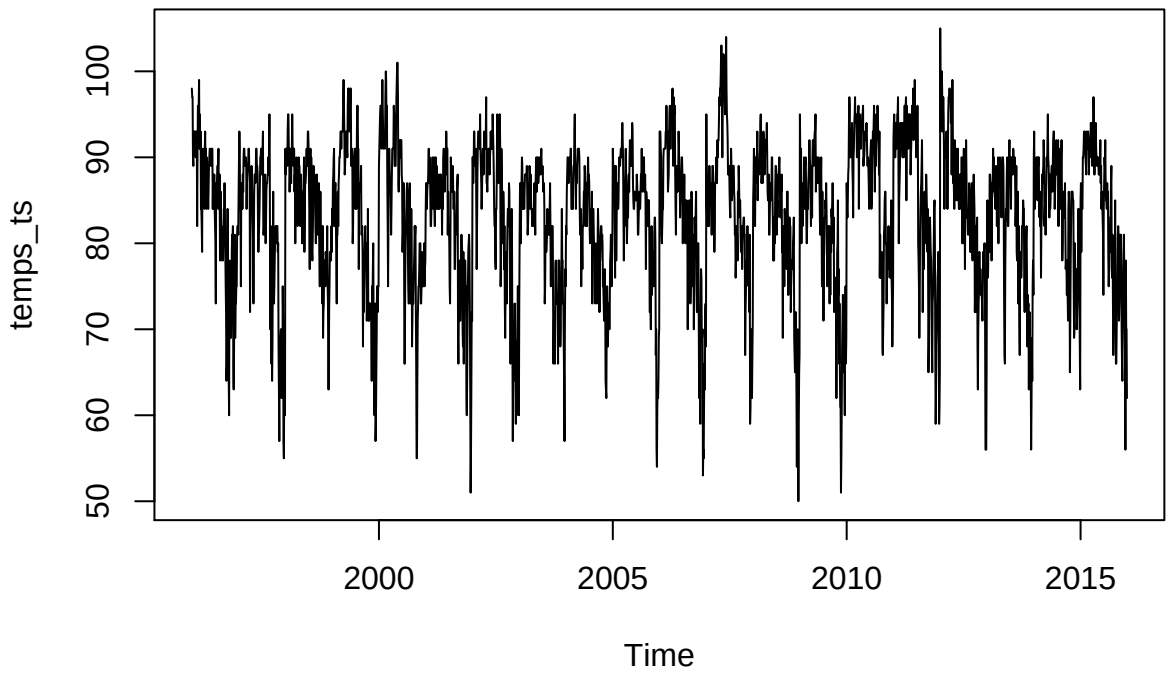
I would use exponential smoothing to forecast daily sales for a retail company in the next quarter. Looking at daily data, there are a lot of fluctuations in sales day to day. There are a couple seasonality factors that need to be taken into account when forecasting like the day of week and time of year. For example, the weekends tend to have more sales than weekdays and weeks leading up to major holidays also tend to have an increase in sales. I would expect alpha to be between 0.5 and 1 as there are other factors like market trends, promotions and product launches that can't be attributed to seasonality and are more important to creating forecasts of the next quarter's sales.

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.

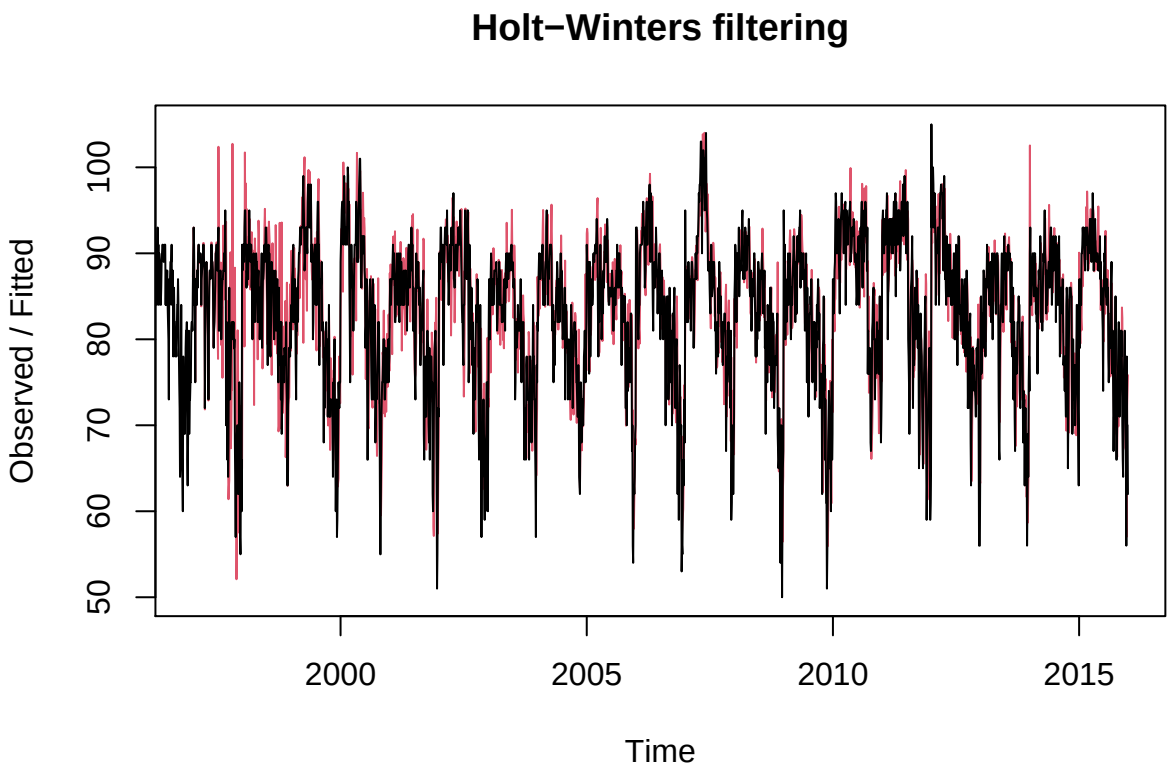
To use the Holt Winters smoothing function, I have to turn the temps data into a time series.

```
temps_data <- read.delim('temps.txt')
temps <- as.vector(unlist(temps_data[,2:21]))
temps_ts <- ts(temps,start=1996, frequency=123)
plot(temps_ts)
```



Reading the documentation, there are two ways to define the seasonal argument: multiplicative and additive. In order to pick the best option, I will test both and choose the one with the lower sum of square error.

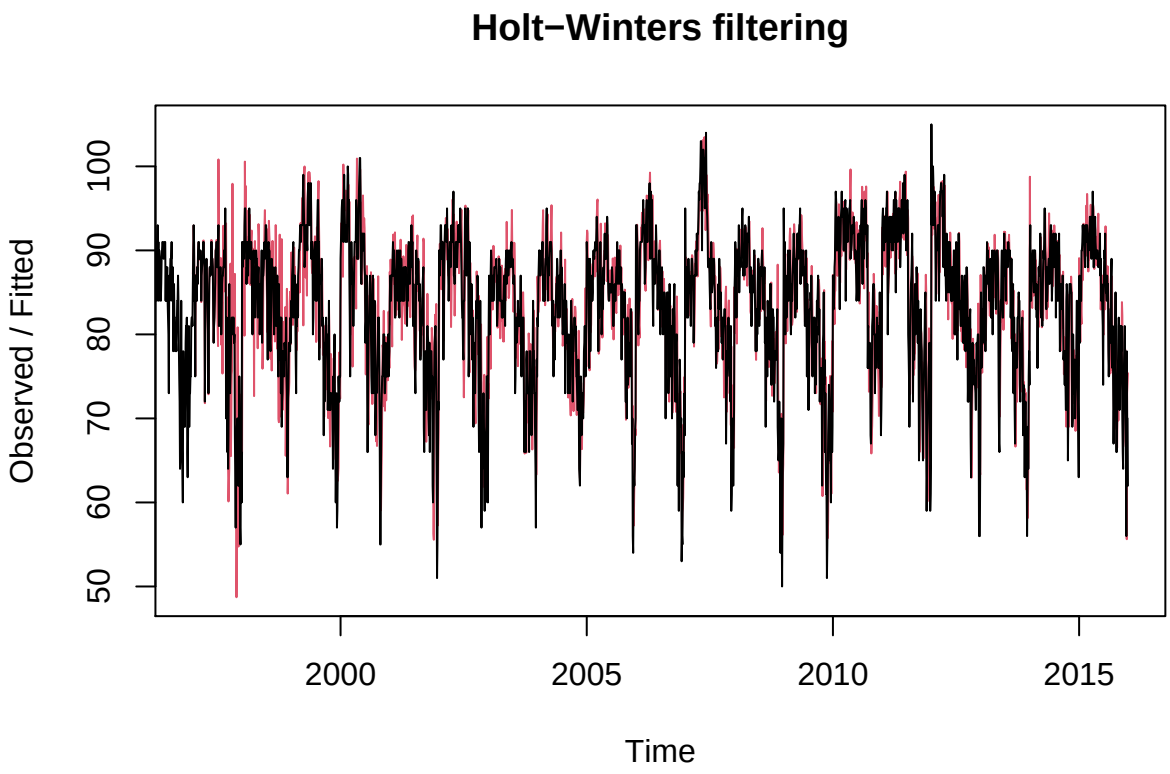
```
HW_M <- HoltWinters(temps_ts, alpha=NULL, gamma=NULL, seasonal='multiplicative')
plot(HW_M)
```




```
HW_M['SSE']
```

```
## $SSE
## [1] 68904.57
```

```
HW_A <- HoltWinters(temps_ts, alpha=NULL, gamma=NULL, seasonal='additive')
plot(HW_A)
```




```
HW_A['SSE']
```

```
## $SSE
## [1] 66244.25
```

Since the SEE for the additive model is lower than that of the multiplicative, I'll use the additive model going forward. Let's look at the model parameters and output:

```
HW_A[["alpha"]]
```

```
##     alpha
## 0.6610618
```

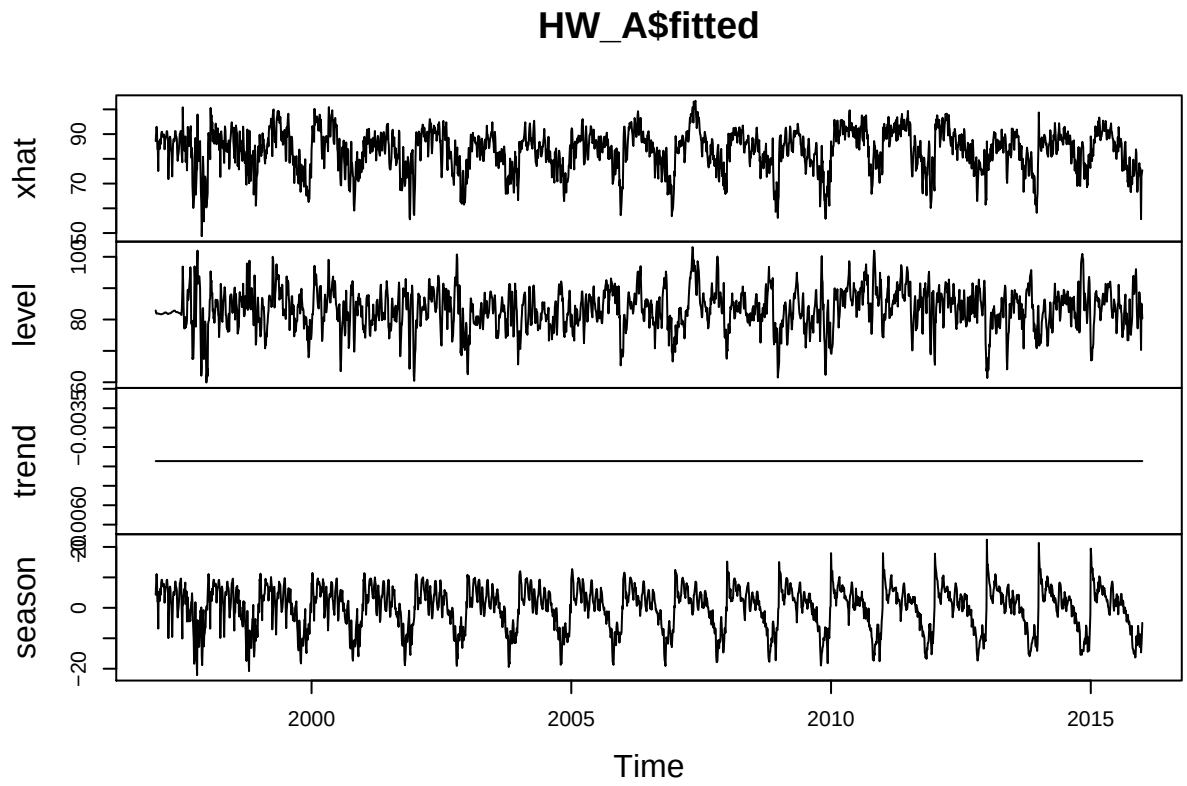
```
HW_A[["gamma"]]
```

```
##     gamma
## 0.6248076
```

```
HW_A[["beta"]]
```

```
## beta
##    0
```

```
plot(HW_A$fitted)
```

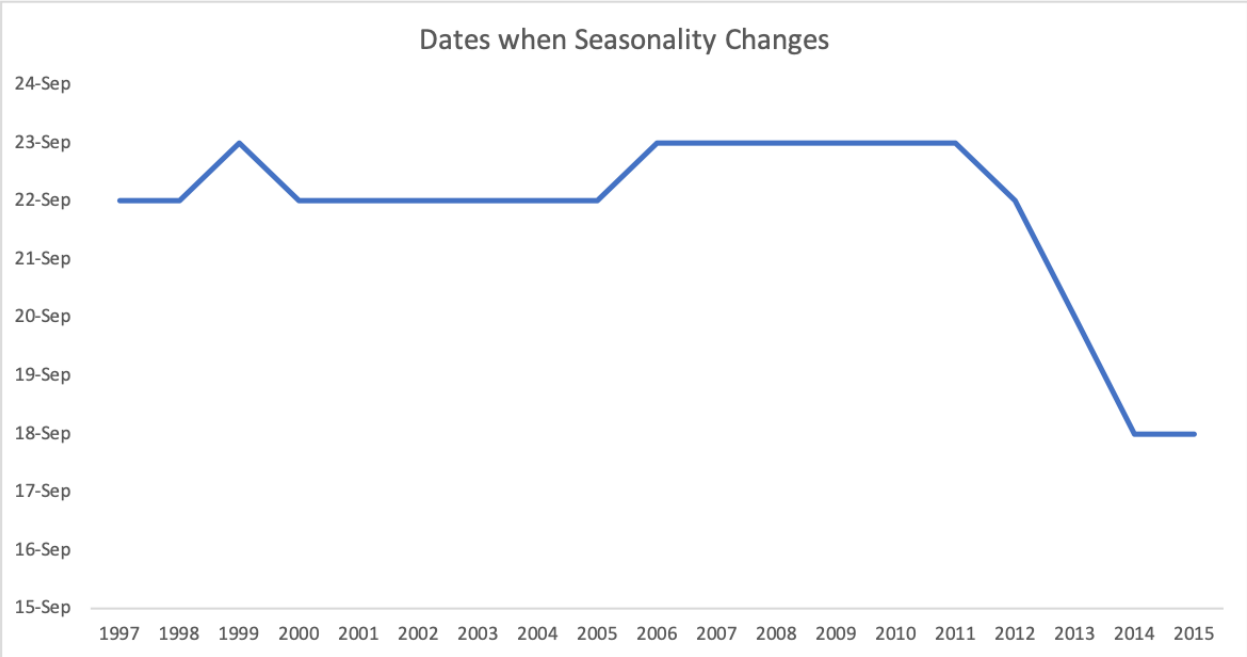


The alpha value chosen by the Holt Winters model is closer to 1 which means that the most current observations are more important than previous observations when predicting. The gamma value closer to 1 means that the more recent trends in seasonality in the data are more important in forecasting compared to the older seasonality trends. **The beta value of 0 suggests that there is no change in the temperature values over time.** However, the chart of season values is increasing slightly and is showing higher peaks from about 2008/2009 onwards.

To investigate this, I will export the model's seasonal values to use as input to a CUSUM model in the attached Excel sheet:

```
data_export <- matrix(HW_A$fitted[,1], nrow=123)
write.csv(data_export, file = 'homework_3_data.csv', sep = ",")
```

If summer is ending later, I would expect seasonality to decrease at a later date every year. Using C = 5 and T = 30. Here are my CUSUM results:



Based on this, I can conclude that summer is not ending later every year. It has been ending earlier in the year, starting from 2011.

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.

I work for a loyalty company that awards points to its members for shopping at participating stores. I would use linear regression to predict the amount of points members will earn in the next quarter. Some indictors I would include:

- Number of active members
- Total number of points earned in the last 12 months
- Number of trips in the last 12 months
- Avg basket size

Question 8.2

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

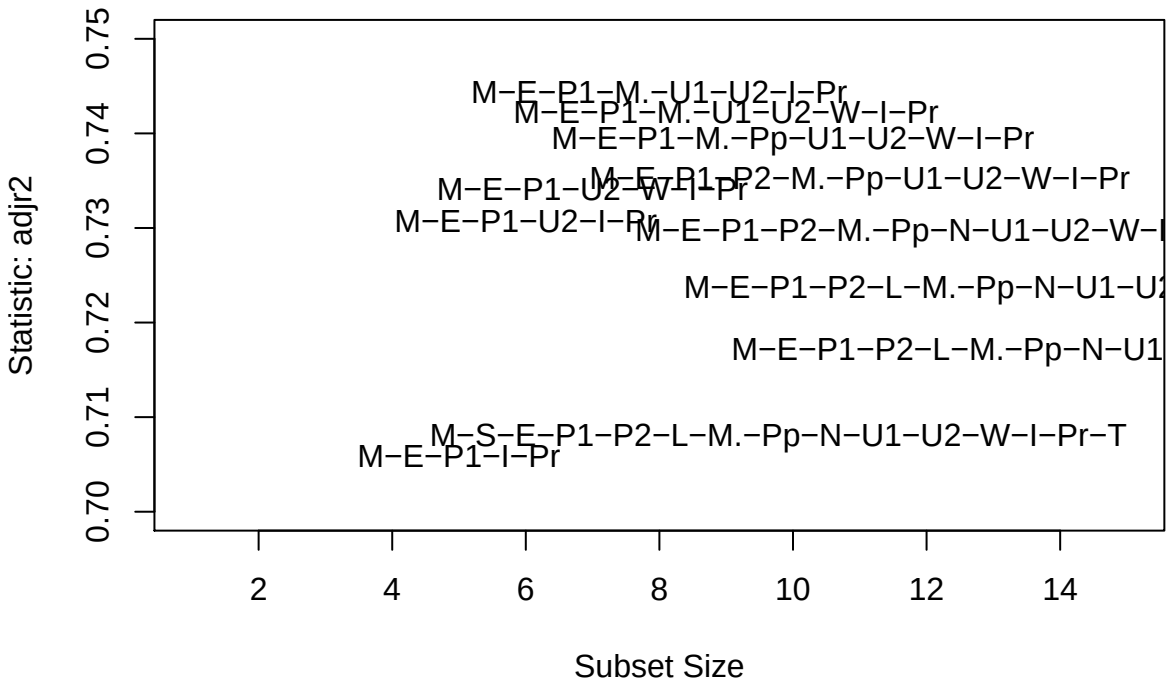
Show your model (factors used and their coefficients), the software output, and the quality of fit.

```
uscrime <- read.delim('uscrime.txt')
summary(uscrime$Crime)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   342.0   658.5   831.0   905.1  1057.5  1993.0
```

To choose predictors, I'm using the regsubsets function in the leaps package. This function runs through the subsets of predictors and calculates metrics like adjusted r squared, BIC, etc

```
choose_predictors <- regsubsets(Crime~.,
         data = uscrime,
         nbest = 1,       # 1 best model for each number of predictors
         nvmax = 15,    # NULL for no limit on number of variables
         force.in = NULL, force.out = NULL,
         method = "exhaustive")
subsets(choose_predictors, statistic = "adjr2", legend=c(0.0), ylim = c(0.7,0.75))
```



The graph above shows the adj. r squared values, which predictors correspond to the value and how the adj. r squared value compares between different combinations of predictors. The model with the highest adjusted r squared value has 8 predictors: M, Ed, Po1, M.F, U1, U2, Ineq and Prob.

```
set.seed(8010)
lr_crime <- lm(Crime ~ M + Ed + Po1 + M.F + U1 + U2 + Ineq + Prob, data = uscrime)
summary(lr_crime)
```

```
##
## Call:
## lm(formula = Crime ~ M + Ed + Po1 + M.F + U1 + U2 + Ineq + Prob,
##     data = uscrime)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -444.70 -111.07    3.03  122.15  483.30
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -6426.10    1194.61  -5.379 4.04e-06 ***
## M              93.32      33.50   2.786  0.00828 **
## Ed            180.12      52.75   3.414  0.00153 **
## Po1           102.65      15.52   6.613 8.26e-08 ***
## M.F            22.34      13.60   1.642  0.10874
## U1          -6086.63    3339.27  -1.823  0.07622 .
## U2            187.35      72.48   2.585  0.01371 *
## Ineq           61.33      13.96   4.394 8.63e-05 ***
## Prob        -3796.03    1490.65  -2.547  0.01505 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 195.5 on 38 degrees of freedom
## Multiple R-squared:  0.7888, Adjusted R-squared:  0.7444
## F-statistic: 17.74 on 8 and 38 DF,  p-value: 1.159e-10
```

We have a very good adjusted r squared value of 0.74. To test for collinearity, I'll use VIF (Variance Inflation Factor). VIF > 10 indicates that variables are collinear.

```
vif(lr_crime)
```

```
##        M       Ed      Po1      M.F       U1       U2     Ineq     Prob
## 2.131963 4.189684 2.560496 1.932367 4.360038 4.508106 3.731074 1.381879
```

Since VIF values are less than 10, we can conclude that there is no collinearity between these predictors and can proceed in using this model to predict the crime rate for the data given in the problem.

Out of curiosity, I want to see what the adjusted r squared value will be if I used all the predictors:

```
set.seed(8010)
lr_crime2 <- lm(Crime ~ ., data = uscrime)
summary(lr_crime2)
```

```
##
## Call:
## lm(formula = Crime ~ ., data = uscrime)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -395.74  -98.09   -6.69  112.99  512.67
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5.984e+03  1.628e+03  -3.675 0.000893 ***
## M            8.783e+01  4.171e+01   2.106 0.043443 *
## So          -3.803e+00  1.488e+02  -0.026 0.979765
## Ed           1.883e+02  6.209e+01   3.033 0.004861 **
## Po1          1.928e+02  1.061e+02   1.817 0.078892 .
## Po2         -1.094e+02  1.175e+02  -0.931 0.358830
```

```
## LF          -6.638e+02  1.470e+03  -0.452 0.654654
## M.F          1.741e+01  2.035e+01   0.855 0.398995
## Pop         -7.330e-01  1.290e+00  -0.568 0.573845
## NW           4.204e+00  6.481e+00   0.649 0.521279
## U1          -5.827e+03  4.210e+03  -1.384 0.176238
## U2           1.678e+02  8.234e+01   2.038 0.050161 .
## Wealth       9.617e-02  1.037e-01   0.928 0.360754
## Ineq         7.067e+01  2.272e+01   3.111 0.003983 **
## Prob        -4.855e+03  2.272e+03  -2.137 0.040627 *
## Time        -3.479e+00  7.165e+00  -0.486 0.630708
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 209.1 on 31 degrees of freedom
## Multiple R-squared:  0.8031, Adjusted R-squared:  0.7078
## F-statistic: 8.429 on 15 and 31 DF,  p-value: 3.539e-07
```

Interesting! The adjusted r squared value using all predictors is lower than the model that uses 8 predictors.

Now we use the the first model generated with 8 predictors to predict the crime rate for the data given.

```
data <- data.frame(M = 14.0, So = 0, Ed = 10.0, Po1 = 12.0, Po2 = 15.5, LF = 0.640, M.F = 94.0, Pop = 15

crime <- predict(lr_crime,data)
crime
```

```
##        1
## 1038.413
```

Based on the data given, this city's crime rate is 1038 out of 100,000 or 1.04%.