Homework 3

Richard Albright
ISYE6501
Spring 2018

1/26/2019

Question 5.1

Using crime data from the file uscrime.txt (http://www.statsci.org/data/general/uscrime.txt, description at http://www.statsci.org/data/general/uscrime.html), test to see whether there are any outliers in the last column (number of crimes per 100,000 people). Use the grubbs.test function in the outliers package in R.

Read in the CSV

```
data <-
  read.table(
    "/Users/ralbright/Dropbox/ISYE6501/week3/homework/uscrime.txt",
    header=TRUE,
    sep="\t"
  )
```

Head:

```
table <- xtable(head(data))
print(table, type='latex', comment=FALSE, scalebox='0.75')
```

	M	So	Ed	Po1	Po2	LF	M.F	Pop	NW	U1	U2	Wealth	Ineq	Prob	Time	Crime
1	15.10	1	9.10	5.80	5.60	0.51	95.00	33	30.10	0.11	4.10	3940	26.10	0.08	26.20	791
2	14.30	0	11.30	10.30	9.50	0.58	101.20	13	10.20	0.10	3.60	5570	19.40	0.03	25.30	1635
3	14.20	1	8.90	4.50	4.40	0.53	96.90	18	21.90	0.09	3.30	3180	25.00	0.08	24.30	578
4	13.60	0	12.10	14.90	14.10	0.58	99.40	157	8.00	0.10	3.90	6730	16.70	0.02	29.90	1969
5	14.10	0	12.10	10.90	10.10	0.59	98.50	18	3.00	0.09	2.00	5780	17.40	0.04	21.30	1234
6	12.10	0	11.00	11.80	11.50	0.55	96.40	25	4.40	0.08	2.90	6890	12.60	0.03	21.00	682

Tail:

```
table <- xtable(tail(data))
print(table, type='latex', comment=FALSE, scalebox='0.75')
```

	M	So	Ed	Po1	Po2	LF	M.F	Pop	NW	U1	U2	Wealth	Ineq	Prob	Time	Crime
42	14.10	0	10.90	5.60	5.40	0.52	96.80	4	0.20	0.11	3.70	4890	17.00	0.09	12.20	542
43	16.20	1	9.90	7.50	7.00	0.52	99.60	40	20.80	0.07	2.70	4960	22.40	0.05	32.00	823
44	13.60	0	12.10	9.50	9.60	0.57	101.20	29	3.60	0.11	3.70	6220	16.20	0.03	30.00	1030
45	13.90	1	8.80	4.60	4.10	0.48	96.80	19	4.90	0.14	5.30	4570	24.90	0.06	32.60	455
46	12.60	0	10.40	10.60	9.70	0.60	98.90	40	2.40	0.08	2.50	5930	17.10	0.05	16.70	508
47	13.00	0	12.10	9.00	9.10	0.62	104.90	3	2.20	0.11	4.00	5880	16.00	0.05	16.10	849

Summary:

```
table <- xtable(summary(data))
print(table, type='latex', comment=FALSE, scalebox='0.4')
```

	M	So	Ed	Po1	Po2	LF	M.F	Pop	NW	U1	U2	Wealth	Ineq	Prob	Time	Crime
X	Min. :11.90	Min. :0.0000	Min. : 8.70	Min. : 4.50	Min. : 4.100	Min. :0.4800	Min. : 93.40	Min. : 3.00	Min. : 0.20	Min. :0.07000	Min. :2.000	Min. :2880	Min. :12.60	Min. :0.00690	Min. :12.20	Min. : 342.0
X.1	1st Qu.:13.00	1st Qu.:0.0000	1st Qu.: 9.75	1st Qu.: 6.25	1st Qu.: 5.850	1st Qu.:0.5305	1st Qu.: 96.45	1st Qu.: 10.00	1st Qu.: 2.40	1st Qu.:0.08050	1st Qu.:2.750	1st Qu.:4595	1st Qu.:16.55	1st Qu.:0.03270	1st Qu.:21.60	1st Qu.: 658.5
X.2	Median :13.60	Median :0.0000	Median :10.80	Median : 7.80	Median : 7.300	Median :0.5600	Median : 97.70	Median : 25.00	Median : 7.60	Median :0.09200	Median :3.400	Median :5370	Median :17.60	Median :0.04210	Median :25.80	Median : 831.0
X.3	Mean :13.86	Mean :0.3404	Mean :10.56	Mean : 8.50	Mean : 8.023	Mean :0.5612	Mean : 98.30	Mean : 36.62	Mean :10.11	Mean :0.09547	Mean :3.398	Mean :5254	Mean :19.40	Mean :0.04709	Mean :26.60	Mean : 905.1
X.4	3rd Qu.:14.60	3rd Qu.:1.0000	3rd Qu.:11.45	3rd Qu.:10.45	3rd Qu.: 9.700	3rd Qu.:0.5930	3rd Qu.: 99.20	3rd Qu.: 41.50	3rd Qu.:13.25	3rd Qu.:0.10400	3rd Qu.:3.850	3rd Qu.:5915	3rd Qu.:22.75	3rd Qu.:0.05445	3rd Qu.:30.45	3rd Qu.:1057.5
X.5	Max. :17.70	Max. :1.0000	Max. :12.20	Max. :16.60	Max. :15.700	Max. :0.6410	Max. :107.10	Max. :168.00	Max. :42.30	Max. :0.14200	Max. :5.800	Max. :6890	Max. :27.60	Max. :0.11980	Max. :44.00	Max. :1993.0

Example analysis from http://www.statsci.org/data/general/uscrime.html

This is performing a linear regression using the lm() function using the last column Crime vs its predictor columns. The leaps functions is an all subsets regression function that attempts to find the best predictors for use in a linear regression model.

```
lm.crime <- lm(Crime~., data=data, names=names(data))
```

```
## Warning: In lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...) :
##  extra argument 'names' will be disregarded
```

```
summary(lm.crime,correlation=FALSE)
```

```
##
## Call:
## lm(formula = Crime ~ ., data = data, names = names(data))
##
## 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) -5984.28760  1628.31837  -3.675 0.000893 ***
## M              87.83017    41.71387   2.106 0.043443 *
## So             -3.80345   148.75514  -0.026 0.979765
## Ed            188.32431    62.08838   3.033 0.004861 **
## Po1           192.80434   106.10968   1.817 0.078892 .
## Po2          -109.42193   117.47754  -0.931 0.358830
## LF           -663.82615  1469.72882  -0.452 0.654654
## M.F            17.40686    20.35384   0.855 0.398995
## Pop            -0.73301     1.28956  -0.568 0.573845
## NW              4.20446     6.48089   0.649 0.521279
## U1          -5827.10272  4210.28904  -1.384 0.176238
## U2            167.79967    82.33596   2.038 0.050161 .
## Wealth          0.09617     0.10367   0.928 0.360754
## Ineq           70.67210    22.71652   3.111 0.003983 **
## Prob        -4855.26582  2272.37462  -2.137 0.040627 *
## Time           -3.47902     7.16528  -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: 0.0000003539
```

We find that the best predictors after performing a linear regression are M, Ed, Po1, U2, Ineq, and Prob.

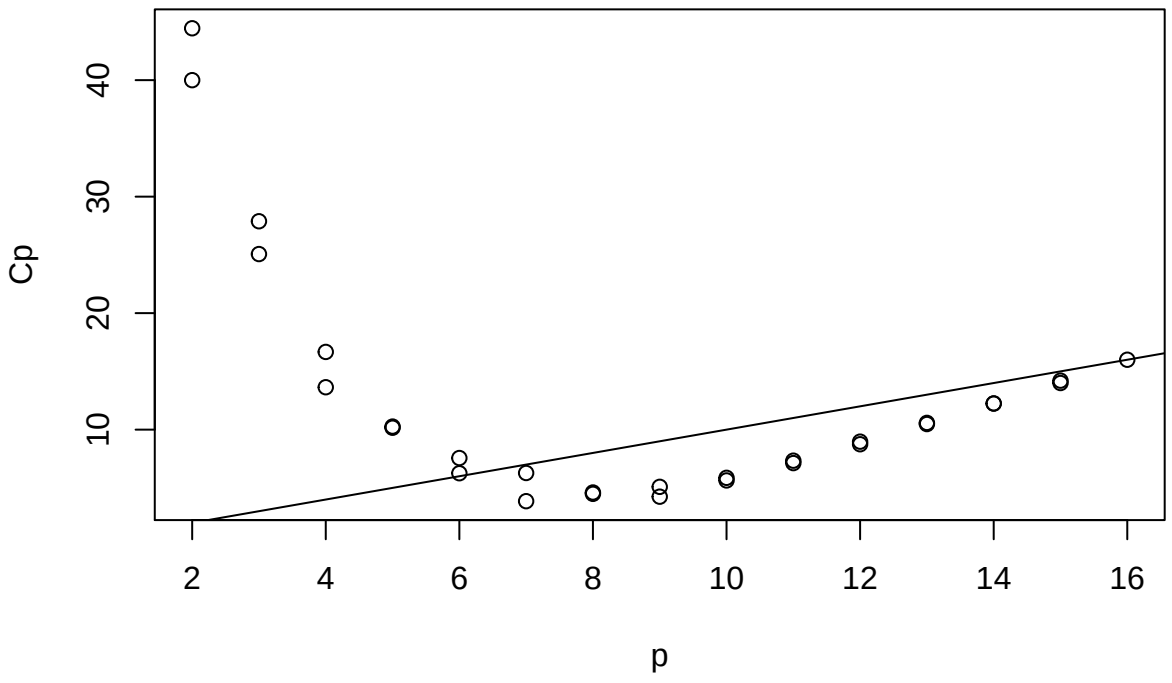
We can then run our predictors through the leaps functions to verify if in fact our predictors are the best ones to use (Information about leaps here: http://www2.hawaii.edu/~taylor/z632/Rbestsubsets.pdf). We want to find the combination of number of p predictors is closest in value to Mallows C_p Statistic (p=C_p) (https://en.wikipedia.org/wiki/Mallows's_Cp).

```
leaps.crime <- leaps(data[,1:15],data$Crime,nbest=2, names=names(data[,1:15]))
```

```
leaps.tab <- data.frame(p=leaps.crime$size,Cp=leaps.crime$Cp)
round(leaps.tab,2)
```

```
##     p    Cp
## 1   2 40.00
## 2   2 44.45
## 3   3 25.07
## 4   3 27.89
## 5   4 13.64
## 6   4 16.67
## 7   5 10.16
## 8   5 10.26
## 9   6  6.26
## 10  6  7.56
## 11  7  3.86
## 12  7  6.28
## 13  8  4.49
## 14  8  4.61
## 15  9  4.24
## 16  9  5.09
## 17 10  5.64
## 18 10  5.86
## 19 11  7.13
## 20 11  7.34
## 21 12  8.75
## 22 12  8.97
## 23 13 10.48
## 24 13 10.58
## 25 14 12.24
## 26 14 12.25
## 27 15 14.00
## 28 15 14.20
## 29 16 16.00
```

```
plot(leaps.tab)
abline(0,1)
```



We can see from the chart that using 6 predictors gives you the best linear regession model (The 1st point where the AB line crosses a scatter point from left to right). Now lets generate a linear regression model using only these factors as identifiified as significant in the lm model. This is before we go through and look for any outliers to remove.

```
lm.crime <- lm(Crime~M+Ed+Po1+U2+Ineq+Prob,data=data)
summary(lm.crime)
```

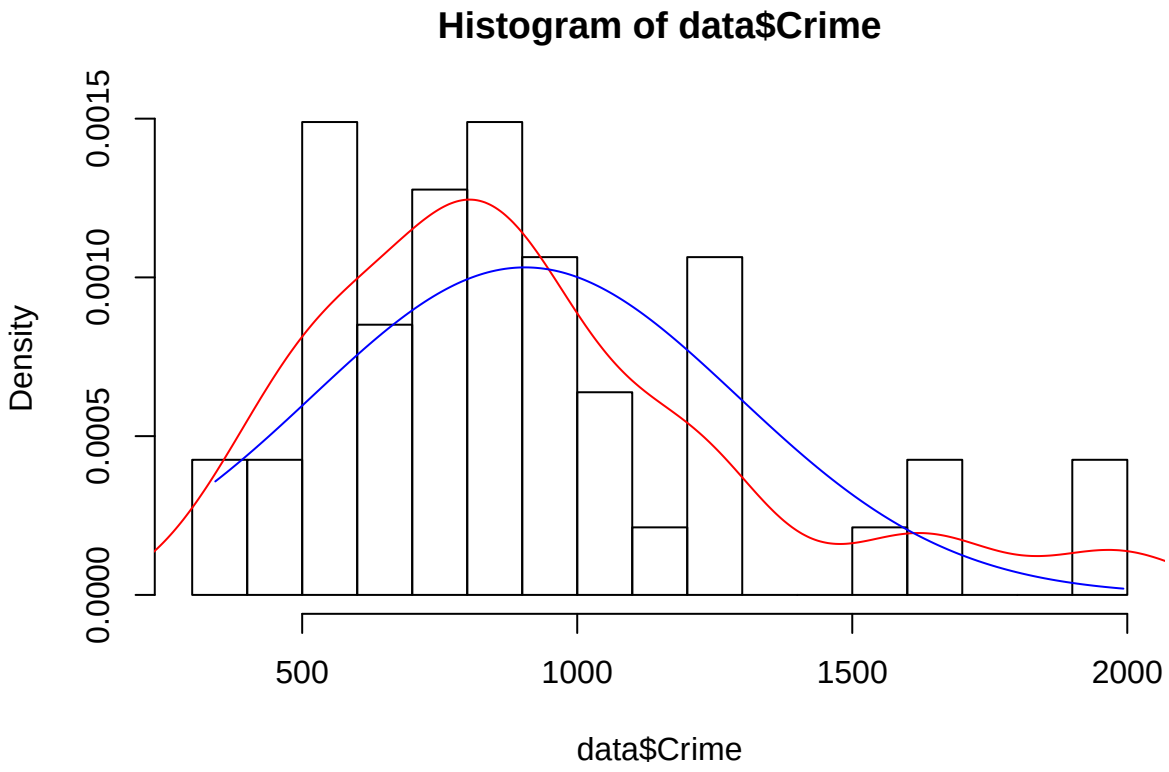
```
##
## Call:
## lm(formula = Crime ~ M + Ed + Po1 + U2 + Ineq + Prob, data = data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -470.68  -78.41  -19.68  133.12  556.23
##
## Coefficients:
##             Estimate Std. Error t value       Pr(>|t|)
## (Intercept) -5040.50     899.84  -5.602 0.000001715273 ***
## M             105.02      33.30   3.154        0.00305 **
## Ed            196.47      44.75   4.390 0.000080720160 ***
## Po1           115.02      13.75   8.363 0.000000000256 ***
## U2             89.37      40.91   2.185        0.03483 *
## Ineq           67.65      13.94   4.855 0.000018793765 ***
## Prob        -3801.84    1528.10  -2.488        0.01711 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 200.7 on 40 degrees of freedom
## Multiple R-squared:  0.7659, Adjusted R-squared:  0.7307
## F-statistic: 21.81 on 6 and 40 DF,  p-value: 0.00000000003418
```

Our model has the above characteristics prior to removing any outliers.

Outlier Removal Using grubbs.test

Lets 1st plot a histogram of our Crime Response variable vs its density and a overlay of the normal distribution.

```
hist(data$Crime, freq=F, breaks=12)
lines(density(data$Crime), col="red")
lines(seq(min(data$Crime), max(data$Crime)), dnorm(seq(min(data$Crime), max(data$Crime)),mean(data$Crim
```



The left tail seems to indicate there may be some outliers in our data set. Lets take a look at a box plot of our Crime response variable as well.

```
boxplot(data$Crime, main="Crime", boxwex=0.1)
```


```
possible_outliers <- boxplot.stats(data$Crime)$out
possible_outliers
```

```
## [1] 1969 1674 1993
```

This histogram and boxplot both point to possible outliers in the upper tail. Output from boxplot.stats

indicates that the 3 possible outliers are 1969, 1674, & 1993. We will now use the grubbs.test function to remove the outliers from the data set.

The grubbs.test function can perform 3 tests (taken directly from the R Documentation).

First test (10) is used to detect if the sample dataset contains one outlier, statistically different than the other values. Test is based by calculating score of this outlier G (outlier minus mean and divided by sd) and comparing it to appropriate critical values. Alternative method is calculating ratio of variances of two datasets - full dataset and dataset without outlier. The obtained value called U is bound with G by simple formula.

Second test (11) is used to check if lowest and highest value are two outliers on opposite tails of sample. It is based on calculation of ratio of range to standard deviation of the sample.

Third test (20) calculates ratio of variance of full sample and sample without two extreme observations. It is used to detect if dataset contains two outliers on the same tail.

The homework question asks us to look for outliers in the Crime response vairable. The third test fails on a code error.

```
# > gtest <- grubbs.test(as.vector(data$Crime), type=20)
# > gtest
# Error in qgrubbs(q, n, type, rev = TRUE) : n must be in range 3-30
```

So we will loop through the 1st two test types on the Crime column.

```
tests <- c(10, 11)
for(test in tests) {
  for(truth in c(TRUE,FALSE)) {
    gtest <- grubbs.test(as.vector(data$Crime), type=test, opposite=truth)
    print(paste('Grubbs Test Type:', test, collapse=' '))
    print(gtest)
  }
}
```

```
## [1] "Grubbs Test Type: 10"
##
##  Grubbs test for one outlier
##
## data:  as.vector(data$Crime)
## G = 1.45590, U = 0.95292, p-value = 1
## alternative hypothesis: lowest value 342 is an outlier
##
## [1] "Grubbs Test Type: 10"
##
##  Grubbs test for one outlier
##
## data:  as.vector(data$Crime)
## G = 2.81290, U = 0.82426, p-value = 0.07887
## alternative hypothesis: highest value 1993 is an outlier
##
## [1] "Grubbs Test Type: 11"
##
##  Grubbs test for two opposite outliers
##
## data:  as.vector(data$Crime)
## G = 4.26880, U = 0.78103, p-value = 1
## alternative hypothesis: 342 and 1993 are outliers
```

```
##
## [1] "Grubbs Test Type: 11"
##
##  Grubbs test for two opposite outliers
##
## data:  as.vector(data$Crime)
## G = 4.26880, U = 0.78103, p-value = 1
## alternative hypothesis: 342 and 1993 are outliers
```

If we assumed a 95% or 99% confidence interval, We would accept the null hypothesis that there are not any outliers in our Crime reponse variable. At a 90% confidence interval, we would reject the null hypothesis and accecpt the alternate hypothesis that the value 1993 in a one tailed test is an outlier. So let's go down the rabbit hole at a 90% confidence interval and see if there are any additional high values in the Crime response variable that can be removed.

We will remove the highest Crime data point from our set and run the Grubbs test again.

```
max_crime1 <- which.max(data$Crime)
max_crime1
```

```
## [1] 26
```

```
data1 <- data[-max_crime1,]
gtest <- grubbs.test(as.vector(data1$Crime), type=10)
print(gtest)
```

```
##
##  Grubbs test for one outlier
##
## data:  as.vector(data1$Crime)
## G = 3.06340, U = 0.78682, p-value = 0.02848
## alternative hypothesis: highest value 1969 is an outlier
```

With a confidence interval of 90%, and a p-value of 0.02848, 1969 can also be considered an outlier.

```
max_crime2= which.max(data1$Crime)
max_crime2
```

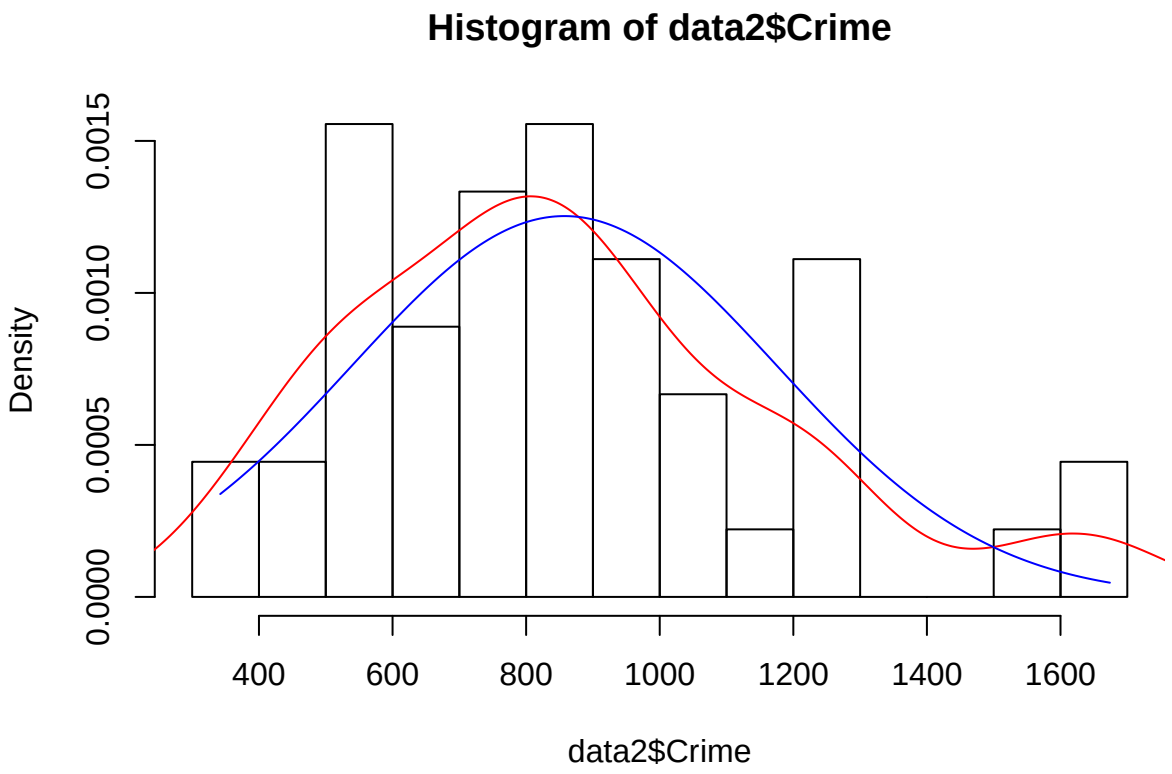
```
## [1] 4
```

```
data2 = data1[-max_crime2,]
gtest <- grubbs.test(as.vector(data2$Crime), type=10)
print(gtest)
```

```
##
##  Grubbs test for one outlier
##
## data:  as.vector(data2$Crime)
## G = 2.56460, U = 0.84712, p-value = 0.1781
## alternative hypothesis: highest value 1674 is an outlier
```

After removing 1993 and 1969, we have no more outliers to remove. Let's look at our histogram again.

```
hist(data2$Crime, freq=F, breaks=12)
lines(density(data2$Crime), col="red")
lines(seq(min(data2$Crime), max(data2$Crime)), dnorm(seq(min(data2$Crime), max(data2$Crime)),mean(data2
```



Our histogram indicates our data with outliers removed more closely resembles a normal distribution.

Let's now run the sample analysis from the http://www.statsci.org/data/general/uscrime.html web page again.

```
lm.crime <- lm(Crime~., data=data2, names=names(data2))
```

```
## Warning: In lm.fit(x, y, offset = offset, singular.ok = singular.ok, ...) :
##  extra argument 'names' will be disregarded
```

```
summary(lm.crime,correlation=FALSE)
```

```
##
## Call:
## lm(formula = Crime ~ ., data = data2, names = names(data2))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -381.84  -98.79   -6.45  106.81  536.88
##
## Coefficients:
##               Estimate  Std. Error t value Pr(>|t|)
## (Intercept) -5265.07699  1879.53527  -2.801  0.00897 **
## M              73.63238    44.42254   1.658  0.10819
## So              7.02787   150.91532   0.047  0.96318
## Ed            172.07458    64.53666   2.666  0.01241 *
## Po1           196.37090   118.27335   1.660  0.10763
## Po2          -115.17225   126.20814  -0.913  0.36900
## LF           -547.32904  1492.33795  -0.367  0.71646
## M.F            14.48640    22.05403   0.657  0.51645
## Pop            -1.46821     1.56000  -0.941  0.35440
```

```
## NW             4.45177     6.58392   0.676  0.50430
## U1         -5794.79350  4368.15821  -1.327  0.19499
## U2           163.35528    85.74484   1.905  0.06672 .
## Wealth         0.08985     0.10613   0.847  0.40412
## Ineq          66.71207    23.32256   2.860  0.00777 **
## Prob       -4864.50231  2315.55124  -2.101  0.04447 *
## Time          -2.63775     7.34175  -0.359  0.72199
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 211.7 on 29 degrees of freedom
## Multiple R-squared:  0.7089, Adjusted R-squared:  0.5583
## F-statistic: 4.708 on 15 and 29 DF,  p-value: 0.000177
```

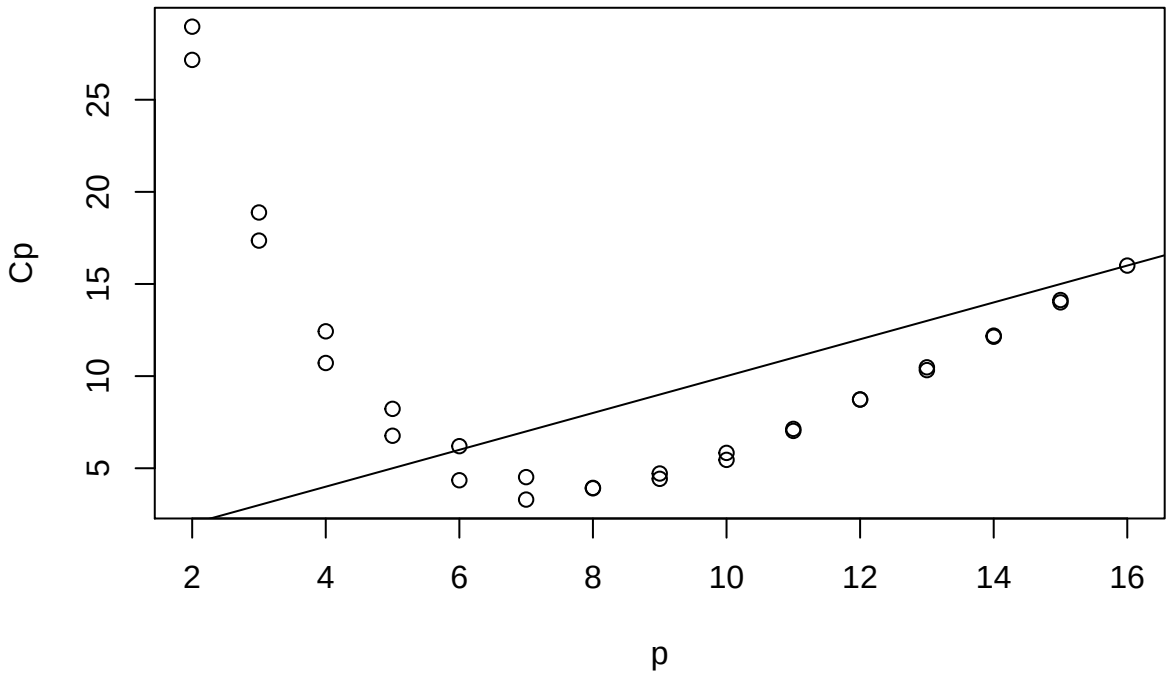
We find that the best predictors have changed after performing a linear regression using lm() with our outliers removed. Our best predictors are now Ed, U2, Ineq, and Prob. Resulting in a much simpler model. Let's double check our predictors using leaps to see if the results agee with using a model with less predictors.

```
leaps.crime2 <- leaps(data2[,1:15],data2$Crime,nbest=2, names=names(data2[,1:15]))
```

```
leaps.tab2 <- data.frame(p=leaps.crime2$size,Cp=leaps.crime2$Cp)
round(leaps.tab2,2)
```

```
##     p    Cp
## 1   2 27.16
## 2   2 28.97
## 3   3 17.36
## 4   3 18.88
## 5   4 10.71
## 6   4 12.43
## 7   5  6.76
## 8   5  8.22
## 9   6  4.35
## 10  6  6.20
## 11  7  3.30
## 12  7  4.52
## 13  8  3.91
## 14  8  3.93
## 15  9  4.43
## 16  9  4.71
## 17 10  5.45
## 18 10  5.83
## 19 11  7.03
## 20 11  7.14
## 21 12  8.73
## 22 12  8.73
## 23 13 10.33
## 24 13 10.48
## 25 14 12.14
## 26 14 12.20
## 27 15 14.00
## 28 15 14.13
## 29 16 16.00
```

```
plot(leaps.tab2)
abline(0,1)
```



The leaps function still indicates our original model of 6 predictors is still the best one to use. So running our same model above with 2 outliers removed yields:

```
lm.crime2 <- lm(Crime~M+Ed+Po1+U2+Ineq+Prob,data=data2)
summary(lm.crime2)
```

```
##
## Call:
## lm(formula = Crime ~ M + Ed + Po1 + U2 + Ineq + Prob, data = data2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -434.55  -94.97   -9.64  130.58  571.48
##
## Coefficients:
##             Estimate Std. Error t value    Pr(>|t|)
## (Intercept) -4456.60    1016.64  -4.384 0.000089066 ***
## M              94.65      34.44   2.748    0.009117 **
## Ed            175.39      48.05   3.650    0.000785 ***
## Po1           104.42      15.98   6.533 0.000000106 ***
## U2             77.30      42.06   1.838    0.073915 .
## Ineq           63.25      14.48   4.367 0.000093588 ***
## Prob        -4034.72    1554.74  -2.595    0.013365 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 201.1 on 38 degrees of freedom
## Multiple R-squared:  0.6557, Adjusted R-squared:  0.6013
## F-statistic: 12.06 on 6 and 38 DF,  p-value: 0.0000001517
```

Our model has the above characteristics after removing 2 outliers.

Question 6.1

Describe a situation or problem from your job, everyday life, current events, etc., for which a Change Detection model would be appropriate. Applying the CUSUM technique, how would you choose the critical value and the threshold?

I work for a FinTech company that analysis insider trading data that assigns a value of -4 to +4 on the significance of an insiders reported purchase, sale, and exercise and sale. Purchases are ranked from > 0 to +4 and sales and exercise and sales are ranked from -4 to <= 0, the ranks assigned are based on a normal distribution. There are numerous factors that go into this model: market capitalization at time of purchase, the value of an insiders transaction, his/her position, the price per share, the number of shares held, etc. This model was written in 2004, and we would want to know if it has remained stable. Using the cusum approach, we would want the ratios of count these scores that are abs value of >= |3| to be the number of transactions that have occured per day to be the same over time. model's stability. We would want to plot the cusum of sum(count((abs(score>= 3))))/# of transactions over time. We would want possibly use a c value of 1/2 standard deviation of the ratio count over our sample period, and 3.5 standard deviations for our critical t value.

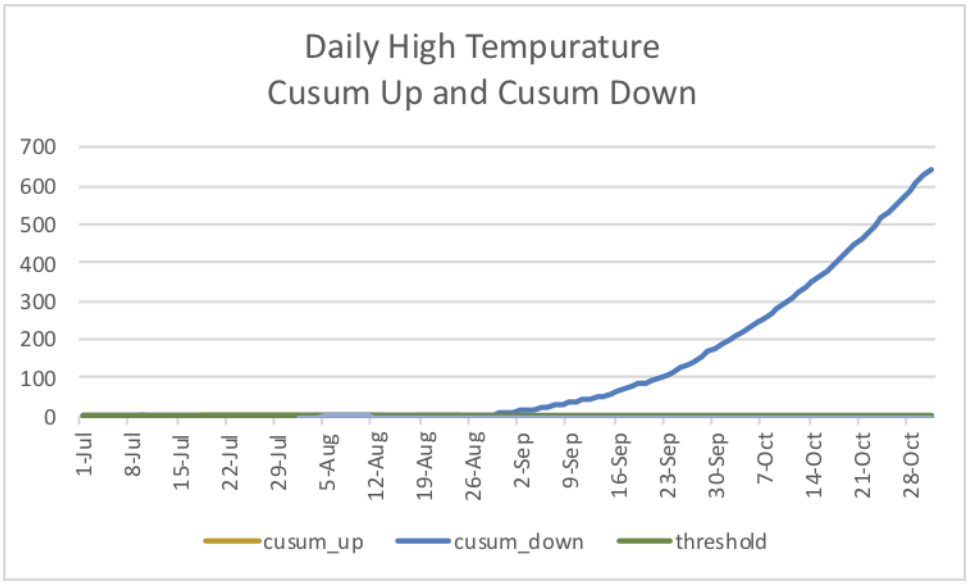
Question 6.2

1.

Using July through October daily-high-temperature data for Atlanta for 1996 through 2015, use a CUSUM approach to identify when unofficial summer ends (i.e., when the weather starts cooling off) each year. You can get the data that you need from the file temps.txt or online, for example at http://www.iweathernet.com/atlanta-weather-records or https://www.wunderground.com/history/airport/KFTY/2015/7/1/CustomHistory.html . You can use R if you'd like, but it's straightforward enough that an Excel spreadsheet can easily do the job too.

This analysis was done in my uploaded temps.xlsx file.

I averaged the high for each date across all years to get an average high per date. I then took the average high of all the July dates to get my initial mean of 88.75 degress and standard deviation of 0.900185166 to use for change detection analysis. I set my C value to be 0.450092583 (0.5 standard deviations). I picked a T value of 2.700555498 (3 d=standarrd deviations) based on the probablility of the event occuring yearly, assuming the data is normally distributed, since summer is once a year event (https://en.wikipedia.org/wiki/68%E2%80%9395%E2%80%9399.7_rule). I performed both a positive and negative cusum analysis in excel, but though the negative is only applicable to this question. I ruled out any change detection events that occured during the training period of July. Based upon the aforementioned parameters, the 1st date the cusum analysis crosses the threshold for a negative change detection is on August 27th, which is when summer unofficially ends based on tempurature.



/pagebreak

2.

Use a CUSUM approach to make a judgment of whether Atlanta's summer climate has gotten warmer in that time (and if so, when).

I transposed the data set in my excel file and performed the following analysis. I averaged all the highs for each year to get an average high per year. Since the data is severely constrained when analyzing by year, I used the 1st 5 years (1996-2000) of average high tempratures to calculate my initial mean of 83.40813008 and standard deviation of 1.026797404. My initial C value is 0.513398702 (0.5 standard deviations). My T value for this analysis was 3.593790914 (3.5 standard deviations), which has a probabiliy of occuring in 1 in 2149 times. I performed both a positive and negative change detection analysis, but only the positive is relevant for this question. My threshold is crossed for the years 2011 and 2012, but goes below the threshold again for 2013 - 2015. If my threshold was 3 instead of its current 4.620588318, it would indicate that Atlanta was warmer for the 2010-2014 period, but goes below the threshold again for 2015. So no, Atlanta is not getting warmer.

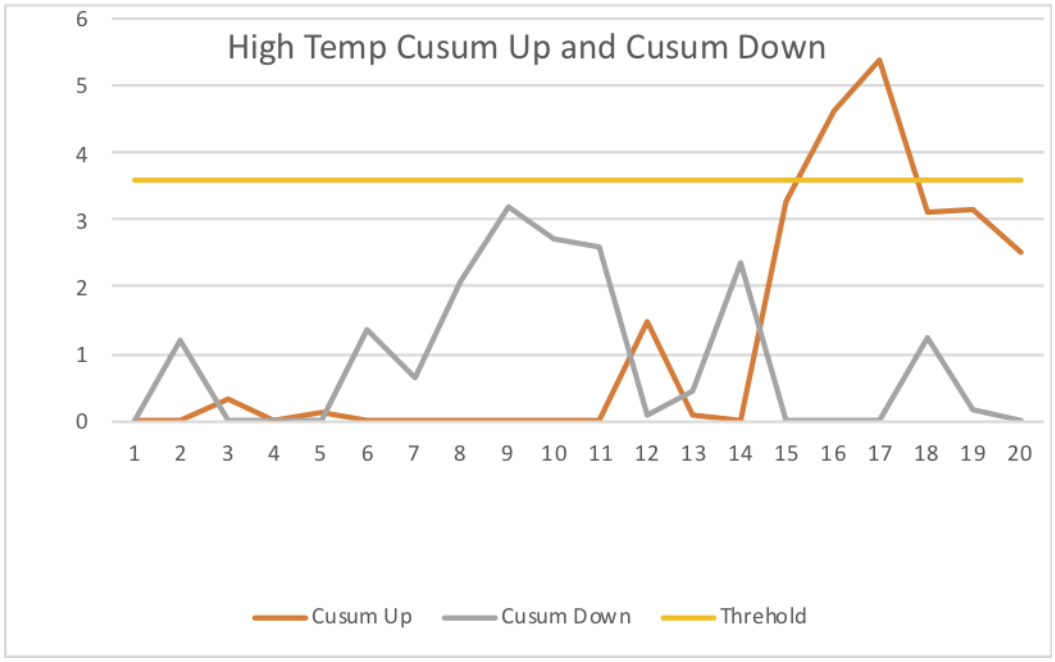


Figure 1: Cusum Analysis for Yearly High Tempurature Averages