Homework 2

Xiao Wang

Question 4.1

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

A university may want to find out the pattern of the alumni who make donations.
Predictors:

1. Highest degree of education
2. Major
3. Length of the stay in this university
4. If received any kind of finicial aids from the university
5. Annual income

Question 4.2

Use the R function kmeans to cluster the points as well as possible. Report the best combination of predictors, your suggested value of k, and how well your best clustering predicts flower type.

```
head(iris)

##   Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1         3.5          1.4         0.2  setosa
## 2          4.9         3.0          1.4         0.2  setosa
## 3          4.7         3.2          1.3         0.2  setosa
## 4          4.6         3.1          1.5         0.2  setosa
## 5          5.0         3.6          1.4         0.2  setosa
## 6          5.4         3.9          1.7         0.4  setosa

str(iris)

## 'data.frame':    150 obs. of  5 variables:
##  $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
##  $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
##  $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
##  $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
##  $ Species     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 
1 1 1 1 ...
```

Only keep the predictor variables

```
iris4.2<-iris[,1:4]
head(iris4.2)
```

```
##   Sepal.Length Sepal.Width Petal.Length Petal.Width
## 1          5.1         3.5          1.4         0.2
## 2          4.9         3.0          1.4         0.2
## 3          4.7         3.2          1.3         0.2
## 4          4.6         3.1          1.5         0.2
## 5          5.0         3.6          1.4         0.2
## 6          5.4         3.9          1.7         0.4
```

Scale to the normal distribution

```
meansl=mean(iris4.2$Sepal.Length)
sdsl=sd(iris4.2$Sepal.Length)

meansw=mean(iris4.2$Sepal.Width)
sdsw=sd(iris4.2$Sepal.Width)

meanpl=mean(iris4.2$Petal.Length)
sdpl=sd(iris4.2$Petal.Length)

meanpw=mean(iris4.2$Petal.Width)
sdpw=sd(iris4.2$Petal.Width)

attach(iris4.2)
iris4.2$s.lgth<-(Sepal.Length-meansl)/sdsl
iris4.2$s.wdth<-(Sepal.Width-meansw)/sdsw
iris4.2$p.lgth<-(Petal.Length-meanpl)/sdpl
iris4.2$p.wdth<-(Petal.Width-meanpw)/sdpw
```

Only keep the scaled data

```
irisf<-iris4.2[,5:8]
head(irisf)
```

```
##       s.lgth      s.wdth    p.lgth    p.wdth
## 1 -0.8976739  1.01560199 -1.335752 -1.311052
## 2 -1.1392005 -0.13153881 -1.335752 -1.311052
## 3 -1.3807271  0.32731751 -1.392399 -1.311052
## 4 -1.5014904  0.09788935 -1.279104 -1.311052
## 5 -1.0184372  1.24503015 -1.335752 -1.311052
## 6 -0.5353840  1.93331463 -1.165809 -1.048667
```

```
set.seed(1234)
```

Method 1. Using all predictors (Septal Length, Septal Width,Petal Length, and Petal Width)

```
k2 <- kmeans(irisf, centers=  2, nstart = 25)
k3 <- kmeans(irisf, centers = 3, nstart = 25)
k4 <- kmeans(irisf, centers = 4, nstart = 25)
k5 <- kmeans(irisf, centers = 5, nstart = 25)
```

Function to compute total within-cluster sum of square

```
wss <- function(k) {
  kmeans(irisf, k, nstart = 25 )$tot.withinss
}
```

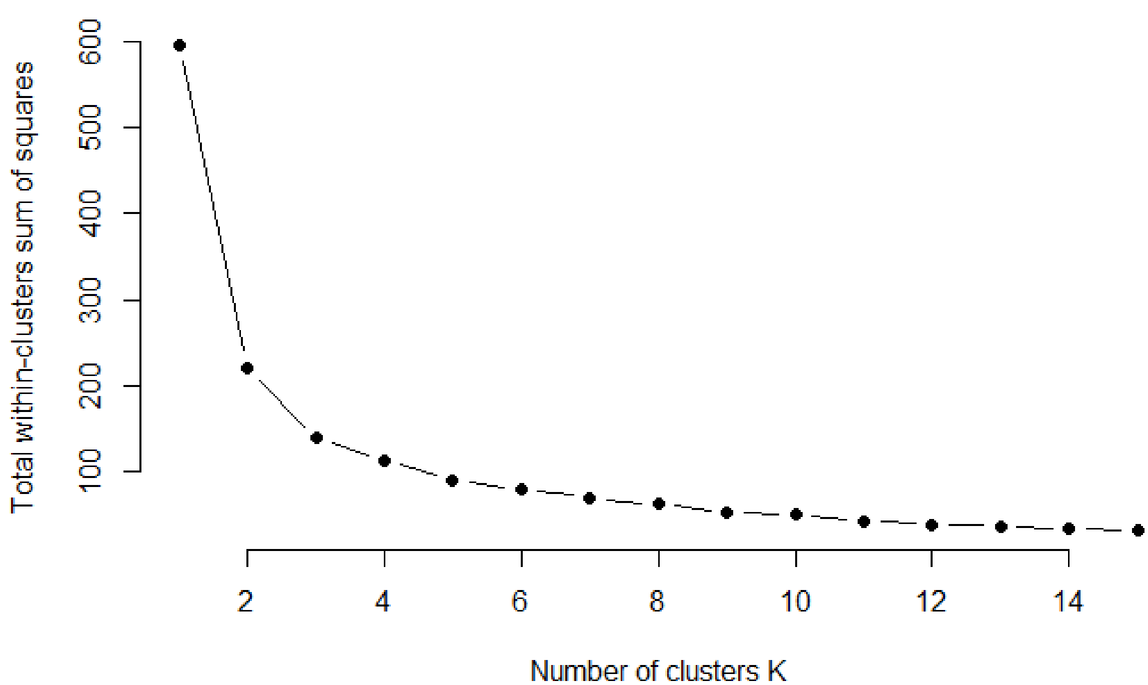
Compute and plot wss for k = 1 to k = 15

```
kvalues <- 1:15
```

Extract wss for 1-15 clusters

wss_values <- map_dbl(kvalues, wss)

plot(kvalues, wss_values, type="b", pch = 19, frame = FALSE, xlab="Number of clusters K", ylab="Total within-clusters sum of squares")



3 is the desired number of clusters

```
table(k3$cluster, iris$Species)

##
##     setosa versicolor virginica
##   1     50          0         0
##   2      0         39        14
##   3      0         11        36
```

All the setosa was perfectly clustered in one group, while the versicolor and virginica were clustered in 2 groups.

Method 2. Using Two predictors (Septal Length, Septal Width)

```
slw2 <- kmeans(irisf[,1:2], centers=  2, nstart = 25)
slw3 <- kmeans(irisf[,1:2], centers = 3, nstart = 25)
slw4 <- kmeans(irisf[,1:2], centers = 4, nstart = 25)
slw5 <- kmeans(irisf[,1:2], centers = 5, nstart = 25)
```

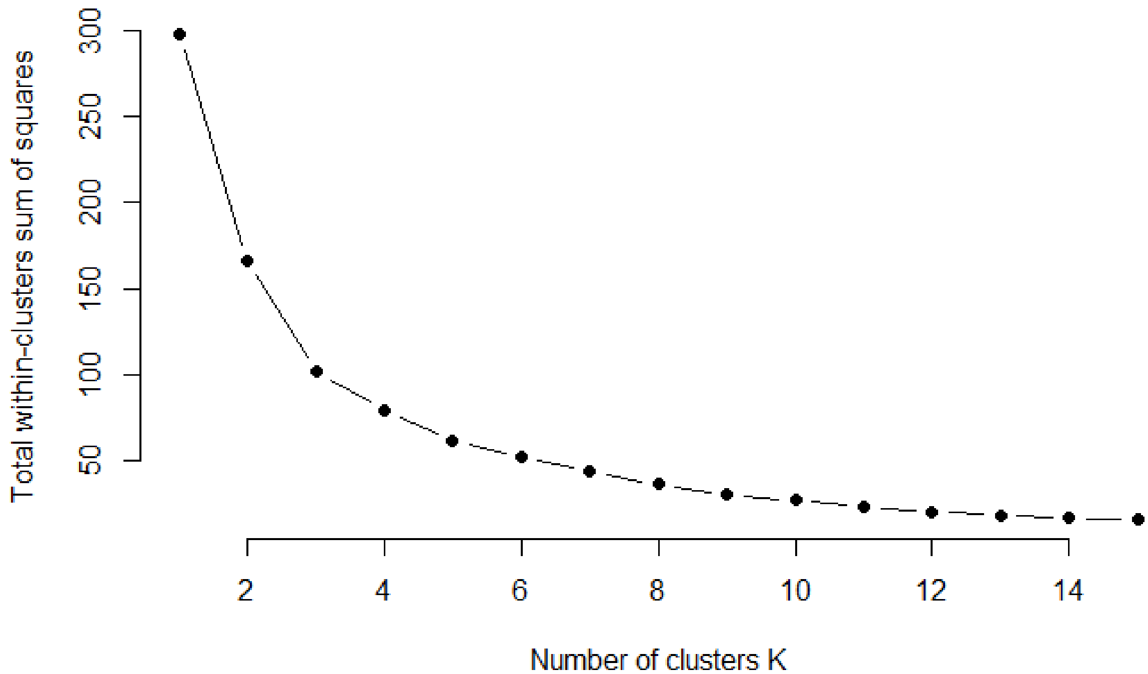
Function to compute total within-cluster sum of square

```
wssslw <- function(k) {
  kmeans(irisf[1:2], k, nstart = 25 )$tot.withinss
}
```

Extract wss for 1-15 clusters

wss_valuesslw <- map_dbl(kvalues, wssslw) #kvalues are the same to the previous, which was 1:15

plot(kvalues, wss_valuesslw, type="b", pch = 19, frame = FALSE, xlab="Number of clusters K", ylab="Total within-clusters sum of squares")



Same to the previous one, 3 is the desired number of clusters, where we got significant improvement.

```
table(slw3$cluster, iris$Species)

##
##     setosa versicolor virginica
##   1      0         14        31
##   2      1         36        19
##   3     49          0         0
```

Compare to the previous method, I don't see great improvement in this one. Instead, setosa species was clustered into two groups.

Method 3. Using Two predictors (Pettal Length, Pettal Width)

```
plw2 <- kmeans(irisf[,3:4], centers=  2, nstart = 25)
plw3 <- kmeans(irisf[,3:4], centers = 3, nstart = 25)
plw4 <- kmeans(irisf[,3:4], centers = 4, nstart = 25)
plw5 <- kmeans(irisf[,3:4], centers = 5, nstart = 25)
```

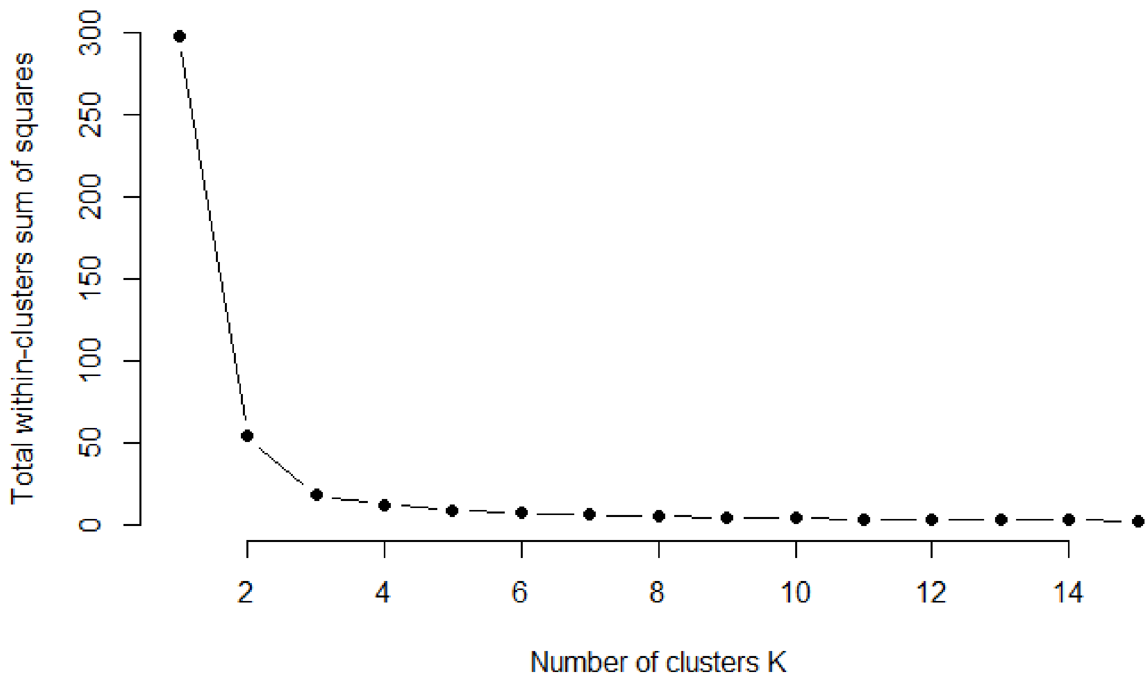
Function to compute total within-cluster sum of square

```
wssplw <- function(k) {
  kmeans(irisf[3:4], k, nstart = 25 )$tot.withinss
}
```

Extract wss for 1-15 clusters

wss_valuesplw <- map_dbl(kvalues, wssplw)

plot(kvalues, wss_valuesplw, type="b", pch = 19, frame = FALSE, xlab="Number of clusters K", ylab="Total within-clusters sum of squares")



Same to the previous ones, 3 is the desired number of clusters,where we got significant improvement.

```
table(plw3$cluster, iris$Species)

##
##      setosa versicolor virginica
##    1     50          0         0
##    2      0          2        46
##    3      0         48         4
```

Plots for visualization

p3 <- fviz_cluster(plw3, geom = "point", data = irisf) + ggtitle("k = 3") grid.arrange(p3)

Compare to the previous methods, this one is much better: Setosa was perfectly clustered as the method 1 (also shown in the picture above), and versicolor and virginica were also nicely clustered with only 2-4 misclassification. I would recommend to use Pettal Length and Pettal Width as predictors, k value equals to 3 for this model. It can correctly predicts 144 out of 150 cluters.

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.

```
crime<- read.table("http://www.statsci.org/data/general/uscrime.txt",header=T
RUE)
head(crime)

##      M So   Ed  Po1  Po2    LF   M.F Pop   NW    U1  U2 Wealth Ineq
## 1 15.1  1  9.1  5.8  5.6 0.510  95.0  33 30.1 0.108 4.1   3940 26.1
## 2 14.3  0 11.3 10.3  9.5 0.583 101.2  13 10.2 0.096 3.6   5570 19.4
## 3 14.2  1  8.9  4.5  4.4 0.533  96.9  18 21.9 0.094 3.3   3180 25.0
## 4 13.6  0 12.1 14.9 14.1 0.577  99.4 157  8.0 0.102 3.9   6730 16.7
## 5 14.1  0 12.1 10.9 10.1 0.591  98.5  18  3.0 0.091 2.0   5780 17.4
## 6 12.1  0 11.0 11.8 11.5 0.547  96.4  25  4.4 0.084 2.9   6890 12.6
##       Prob    Time Crime
```

```
## 1 0.084602 26.2011   791
## 2 0.029599 25.2999  1635
## 3 0.083401 24.3006   578
## 4 0.015801 29.9012  1969
## 5 0.041399 21.2998  1234
## 6 0.034201 20.9995   682

crime<-crime$Crime
```

Test if the lowest and highest value are two outliers on opposite tails of sample.

grubbs.test(crime,type=11)

```
Grubbs test for two opposite outliers

data:  crime
G = 4.26880, U = 0.78103, p-value = 1
alternative hypothesis: 342 and 1993 are outliers
```

p-value=1, so at least one of the values (342, 1993) is not an outlier.

Test if the highest value is aN outlier.

grubbs.test(crime,type=10)

```
Grubbs test for one outlier

data:  crime
G = 2.81290, U = 0.82426, p-value = 0.07887
alternative hypothesis: highest value 1993 is an outlier
```

If we set the threshold p-value=0.05, then we didn't detect the outlier. If we set the threshold p-value=0.1, then we detected highest value 1993 as an outlier.

Test if the lowest value is a outlier.

grubbs.test(crime,type=10,opposite=TRUE)

```
Grubbs test for one outlier

data:  crime
G = 1.45590, U = 0.95292, p-value = 1
alternative hypothesis: lowest value 342 is an outlier
```

The p-value rounds to 1, so the lowest-crime city does not p-value=1, so the lowest value is not a outlier. This is consistent with our first test,where we found at least one of the extreme values is not an outlier. In conclusion,the highest value is an outlier at p-value=0.1; it is not an outlier at p-value=0.05.

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?

Machine may gets heated after running for a long time. Technicians would want to prevent it gets overheated before it's too late. CUSUM can be used to monitor the temperature of the machine,and detect a change when the temperature gets above a certain threhold, indicating overheat. The cost of can't detect the overheat in time is much greater than a false alarm. So the criticial value and the threhold would be some small values, based on the historical data.

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.

   I am using Excel to do this analysis:

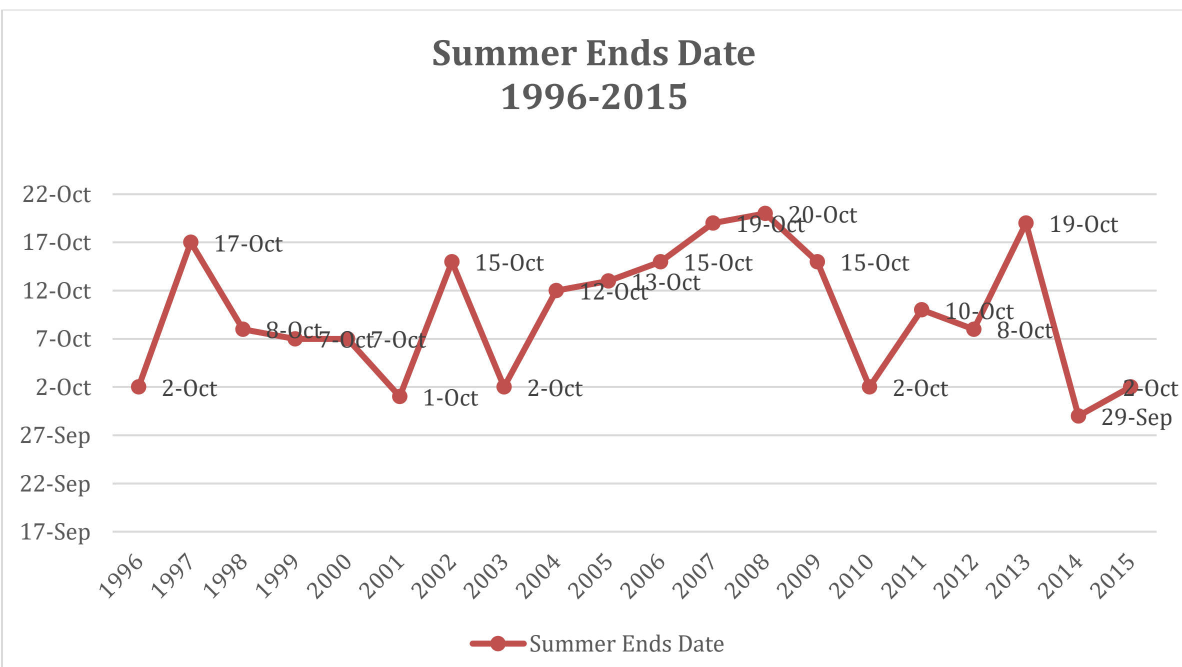
Year	Mean Temperature (July-October)	SD	C	T	Summer Ends Date	Daily high Temperature on summer ends date
1996	83.71545	8.548339	4.27417	-42.7417	2-Oct	72
1997	81.6748	9.319023	4.659512	-46.5951	17-Oct	66
1998	84.26016	6.409314	3.204657	-32.0466	8-Oct	69
1999	83.35772	9.723328	4.861664	-48.6166	7-Oct	73
2000	84.03252	9.518692	4.759346	-47.5935	7-Oct	66
2001	81.55285	8.224517	4.112258	-41.1226	1-Oct	75
2002	83.58537	9.426095	4.713047	-47.1305	15-Oct	57
2003	81.47967	7.017951	3.508975	-35.0898	2-Oct	68
2004	81.76423	6.66294	3.33147	-33.3147	12-Oct	73
2005	83.35772	7.733396	3.866698	-38.667	13-Oct	64

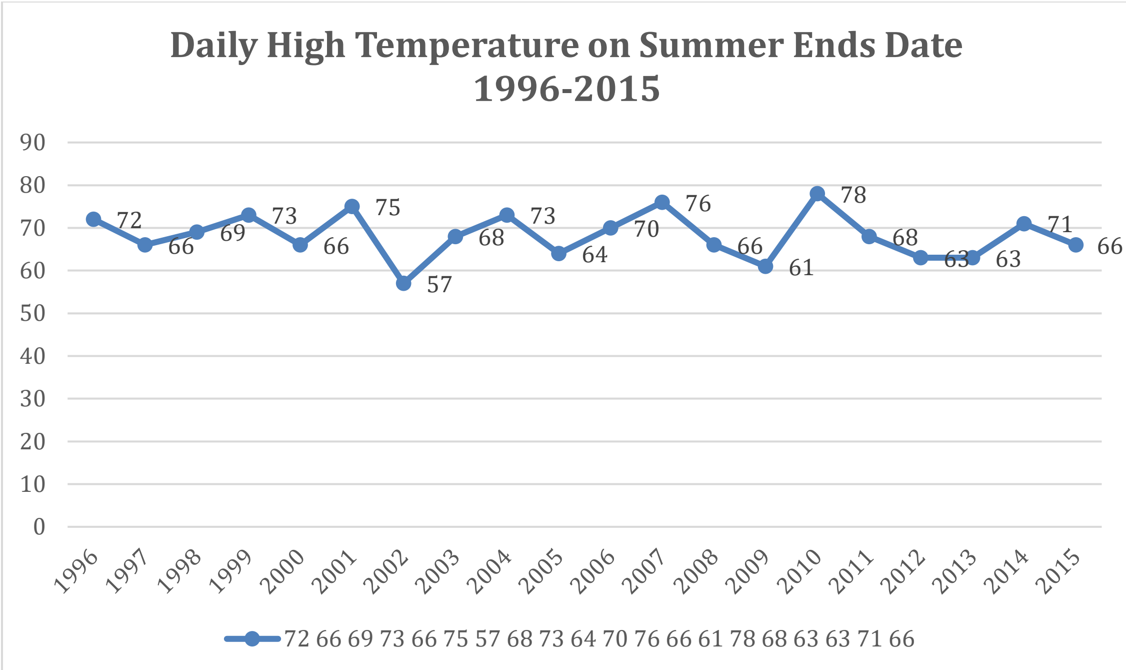
2006	83.04878	9.793653	4.896826	-48.9683	15-Oct	70
2007	85.39837	9.033399	4.516699	-45.167	19-Oct	76
2008	82.5122	8.733172	4.366586	-43.6659	20-Oct	66
2009	80.99187	9.013192	4.506596	-45.066	15-Oct	61
2010	87.21138	7.445157	3.722578	-37.2258	2-Oct	78
2011	85.27642	9.931157	4.965579	-49.6558	10-Oct	68
2012	84.65041	9.252367	4.626183	-46.2618	8-Oct	63
2013	81.66667	7.726542	3.863271	-38.6327	19-Oct	63
2014	83.94309	6.591476	3.295738	-32.9574	29-Sep	71
2015	83.30081	8.709271	4.354635	-43.5464	2-Oct	66

Each year, I calculated the mean temperature, as well as the standard deviation from July to October. My C is half of the standard deviation, and the T is 5 times of the standard deviation. I detect the unofficial summer ends dates as the first date when the min{T (t-1)+Daily Temperature (t)-Mean+C, 0}<T

If I average the result, Oct 9th would be the unofficial date when summer ends.

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).





Based on the table from question 1 and the two charts above, I don't see Atlanta's summer climate has gotten warmer in that time.