Week 2 Homework

## Loding packages and datasets

#Loading Libraries  
library('ggplot2')  
library("outliers")  
library("qcc")  
  
  
#Loading data   
  
#temperature data file  
tempData <- read.table("https://d37djvu3ytnwxt.cloudfront.net/assets/courseware/v1/592f3be3e90d2bdfe6a69f62374a1250/asset-v1:GTx+ISYE6501x+2T2017+type@asset+block/temps.txt", header = TRUE)  
  
#Crime data file   
crimeData <- read.table("https://d37djvu3ytnwxt.cloudfront.net/assets/courseware/v1/17b85cea5d0e613bf08025ca2907b62f/asset-v1:GTx+ISYE6501x+2T2017+type@asset+block/uscrime.txt", header = TRUE)  
  
#iris data file   
irisData <- iris  
  
#Set seed for reproducibility  
set.seed(156)

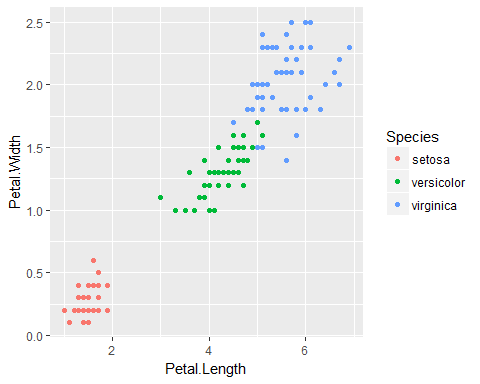
## Question 1:

A situtation where a clustering model would be appropriate. In my current job a clustering technique would be useful to find different customer cohorts that could be divided by certain behaviors. predictors could be OrderFrequency, OrderAmount, Tenure, MarketingType, Age

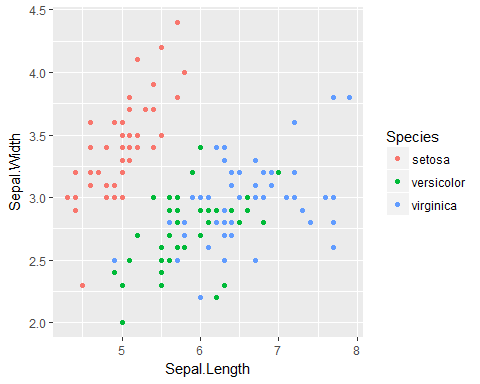
## Question 2: kmeans clustering to determine species from the iris data set.

I chose the cluster amount by examining the data and finding that there are 3 species in the dataset. Accuracy is 94%

#Visualize the dataset:  
ggplot(irisData, aes(Petal.Length, Petal.Width, color = Species)) + geom\_point()



ggplot(irisData, aes(Sepal.Length, Sepal.Width, color = Species)) + geom\_point()



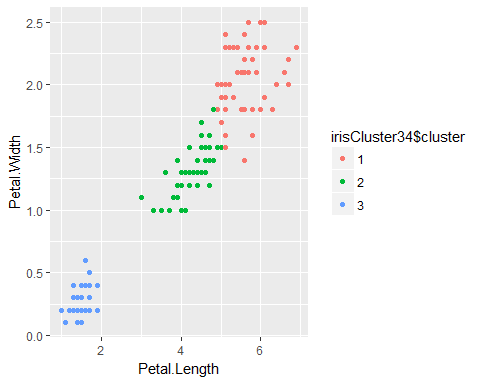
# these plots demonstrate that there are only three distinct species therefore the correct cluster amount should be 3  
  
#Most accurate column selection for clustering at 94%   
irisCluster34 <- kmeans(iris[, 3:4], 3)  
irisCluster34

## K-means clustering with 3 clusters of sizes 48, 52, 50  
##   
## Cluster means:  
## Petal.Length Petal.Width  
## 1 5.595833 2.037500  
## 2 4.269231 1.342308  
## 3 1.462000 0.246000  
##   
## Clustering vector:  
## [1] 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3  
## [36] 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2  
## [71] 2 2 2 2 2 2 2 1 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 1 1 1  
## [106] 1 2 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 2 1  
## [141] 1 1 1 1 1 1 1 1 1 1  
##   
## Within cluster sum of squares by cluster:  
## [1] 16.29167 13.05769 2.02200  
## (between\_SS / total\_SS = 94.3 %)  
##   
## Available components:  
##   
## [1] "cluster" "centers" "totss" "withinss"   
## [5] "tot.withinss" "betweenss" "size" "iter"   
## [9] "ifault"

table(irisCluster34$cluster, iris$Species)

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

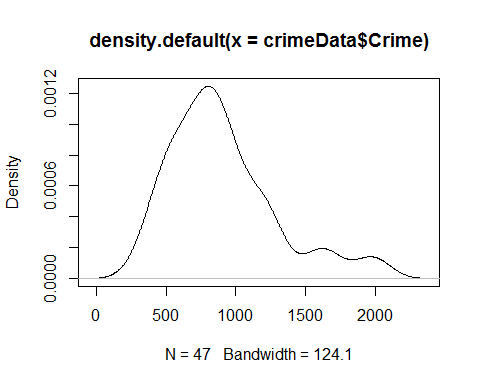
#----------------------------------------------  
  
#Plot of results  
irisCluster34$cluster <- as.factor(irisCluster34$cluster)  
ggplot(irisData, aes(Petal.Length, Petal.Width, color = irisCluster34$cluster)) + geom\_point()



## Question 3: Outliers

Conclusion: I found that the highest crime city is an outlier, however the lowest crime city is not as it should fit in the normal distribution.

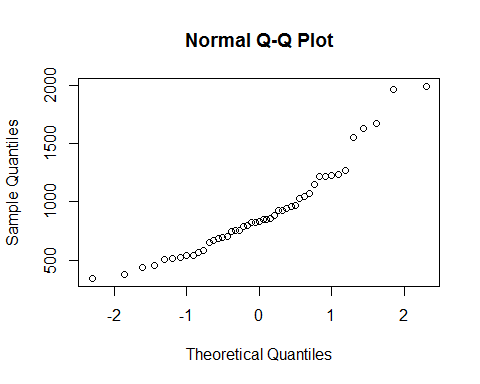
#The grubbs test is based on the assumption of normal distributed data. plots and shapiro test to determine if it meets this before running outlier package   
  
#this plot indicates to me that the data is fairly normal with a long tail where there may be some outliers   
plot(density(crimeData$Crime))



#as the p value is less than 0.05 we can assume there is normal distribution   
shapiro.test(crimeData$Crime)

##   
## Shapiro-Wilk normality test  
##   
## data: crimeData$Crime  
## W = 0.91273, p-value = 0.001882

#another technique to determine normality. Because the lines are not in a straight line in the upper quantiles this is indicitative of a heavy tail however the data seems fairly normally distributed   
qqnorm(crimeData$Crime)



#the density plot seems to show a heavy tail on the upper end so type=10 is used to find the correct outlier.   
grubbs.outlier <- grubbs.test(crimeData$Crime, type = 10)  
grubbs.outlier$alternative

## [1] "highest value 1993 is an outlier"

#I take this to mean that the highest crime city is an outlier, however the lowest crime city is not as it should fit in the normal distribution

## Question 4 CUSUM example

An area cusum would be useful in my job now is detecting if there is a change in order volumes by day. Most daily changes are well within 3 stdevs so CUSUM would work well. I would choose the critical value as 1 stdev of the last 3 months worth of data and the threshold would be determined by running the cusum model on periods of time where we know there was a shift and use that as a baseline to determine a good threshold value.

## Question 5: CUSUM of temperature data to detect temperature changes to inidicate end of Summer.

1: I used the CUSUM function from the package qcc. I ran this funtion against each year in the dataset and found the minimum date for each year that the shift occured. I then took a mean of each year to determine the date in which summer ends. Answer: 30th September.

2: After looking at the cusum charts for each year and examining the results output across all years I cannot conclude that the summer climate has gotten warmer. Some years it is warmer but others are not so warm. 2001 through 2003 were very warm as were 2011- 2013

#data exploration section --------------------------------------------------  
#Summary of dataset to get a general overview  
summary(tempData)

## DAY X1996 X1997 X1998   
## 1-Aug : 1 Min. :60.00 Min. :55.00 Min. :63.00   
## 1-Jul : 1 1st Qu.:79.00 1st Qu.:78.50 1st Qu.:79.50   
## 1-Oct : 1 Median :84.00 Median :84.00 Median :86.00   
## 1-Sep : 1 Mean :83.72 Mean :81.67 Mean :84.26   
## 10-Aug : 1 3rd Qu.:90.00 3rd Qu.:88.50 3rd Qu.:89.00   
## 10-Jul : 1 Max. :99.00 Max. :95.00 Max. :95.00   
## (Other):117   
## X1999 X2000 X2001 X2002   
## Min. :57.00 Min. : 55.00 Min. :51.00 Min. :57.00   
## 1st Qu.:75.00 1st Qu.: 77.00 1st Qu.:78.00 1st Qu.:78.00   
## Median :86.00 Median : 86.00 Median :84.00 Median :87.00   
## Mean :83.36 Mean : 84.03 Mean :81.55 Mean :83.59   
## 3rd Qu.:91.00 3rd Qu.: 91.00 3rd Qu.:87.00 3rd Qu.:91.00   
## Max. :99.00 Max. :101.00 Max. :93.00 Max. :97.00   
##   
## X2003 X2004 X2005 X2006   
## Min. :57.00 Min. :62.00 Min. :54.00 Min. :53.00   
## 1st Qu.:78.00 1st Qu.:78.00 1st Qu.:81.50 1st Qu.:79.00   
## Median :84.00 Median :82.00 Median :85.00 Median :85.00   
## Mean :81.48 Mean :81.76 Mean :83.36 Mean :83.05   
## 3rd Qu.:87.00 3rd Qu.:87.00 3rd Qu.:88.00 3rd Qu.:91.00   
## Max. :91.00 Max. :95.00 Max. :94.00 Max. :98.00   
##   
## X2007 X2008 X2009 X2010   
## Min. : 59.0 Min. :50.00 Min. :51.00 Min. :67.00   
## 1st Qu.: 81.0 1st Qu.:79.50 1st Qu.:75.00 1st Qu.:82.00   
## Median : 86.0 Median :85.00 Median :83.00 Median :90.00   
## Mean : 85.4 Mean :82.51 Mean :80.99 Mean :87.21   
## 3rd Qu.: 89.5 3rd Qu.:88.50 3rd Qu.:88.00 3rd Qu.:93.00   
## Max. :104.0 Max. :95.00 Max. :95.00 Max. :97.00   
##   
## X2011 X2012 X2013 X2014   
## Min. :59.00 Min. : 56.00 Min. :56.00 Min. :63.00   
## 1st Qu.:79.00 1st Qu.: 79.50 1st Qu.:77.00 1st Qu.:81.50   
## Median :89.00 Median : 85.00 Median :84.00 Median :86.00   
## Mean :85.28 Mean : 84.65 Mean :81.67 Mean :83.94   
## 3rd Qu.:94.00 3rd Qu.: 90.50 3rd Qu.:88.00 3rd Qu.:89.00   
## Max. :99.00 Max. :105.00 Max. :92.00 Max. :95.00   
##   
## X2015   
## Min. :56.0   
## 1st Qu.:77.0   
## Median :85.0   
## Mean :83.3   
## 3rd Qu.:90.0   
## Max. :97.0   
##

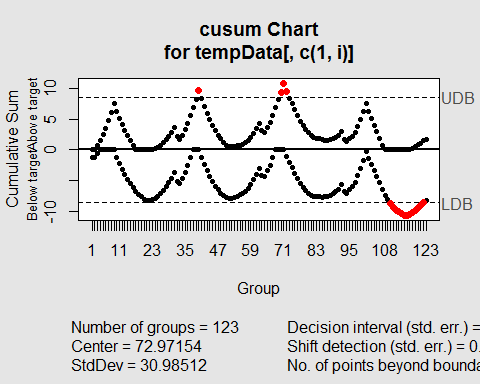
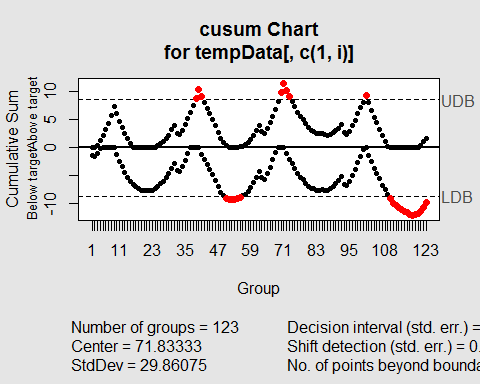
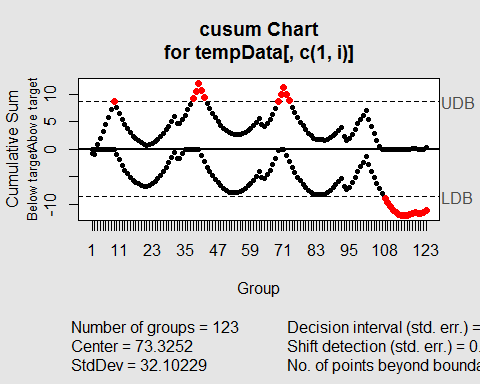
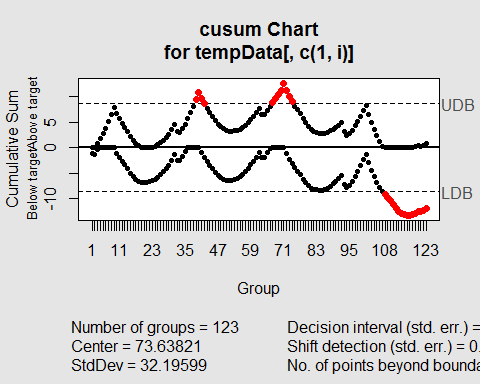
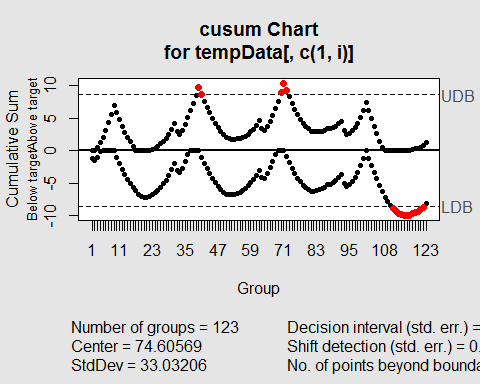
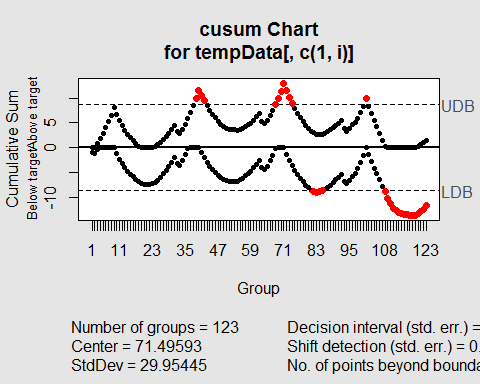
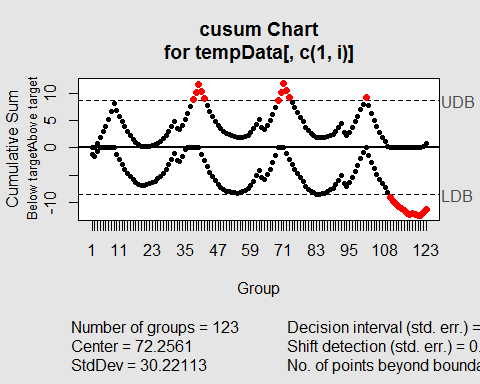
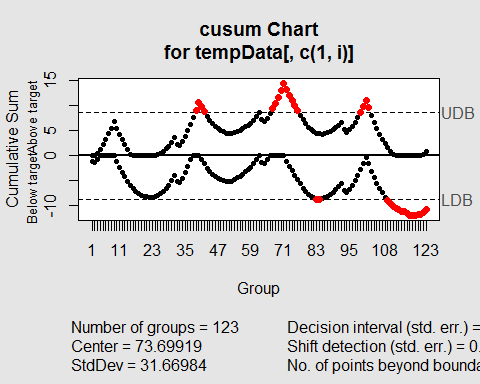
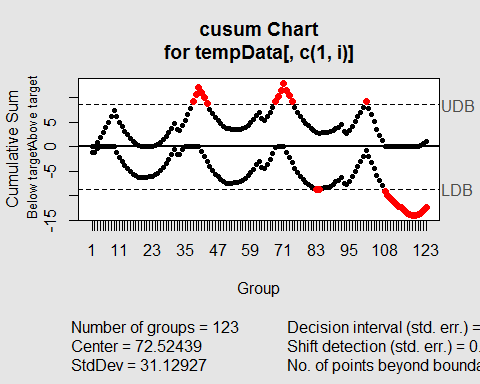
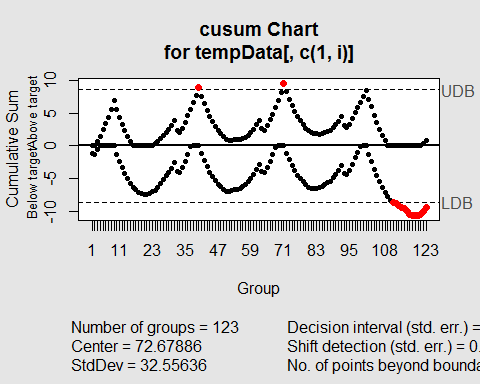
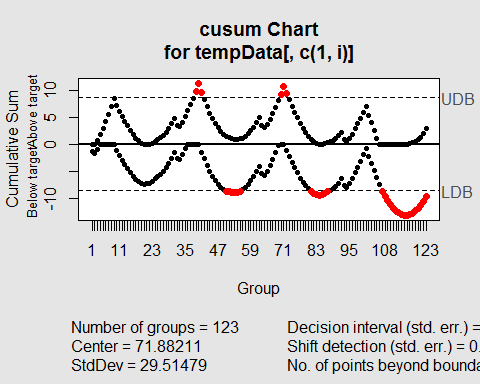
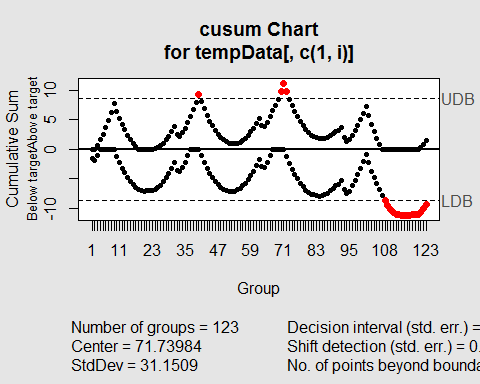
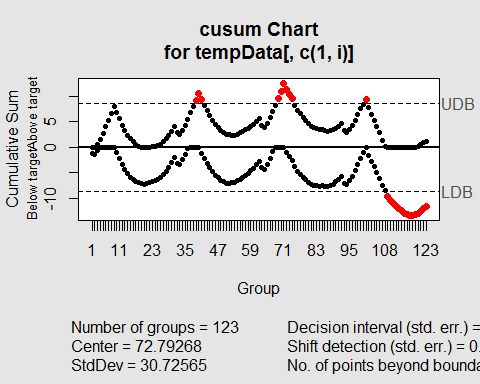
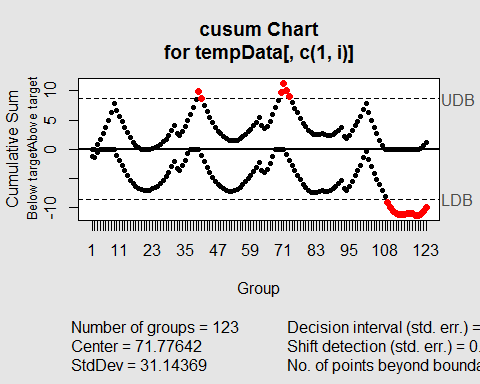
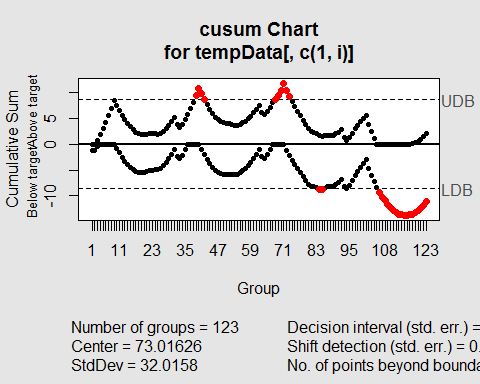
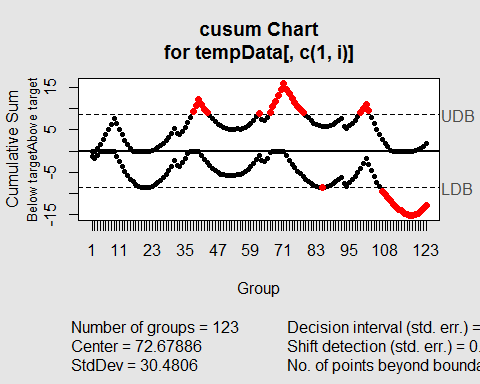
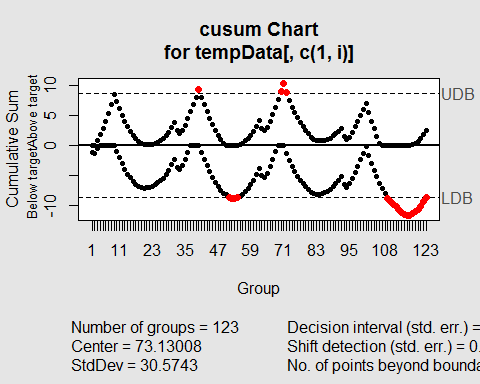
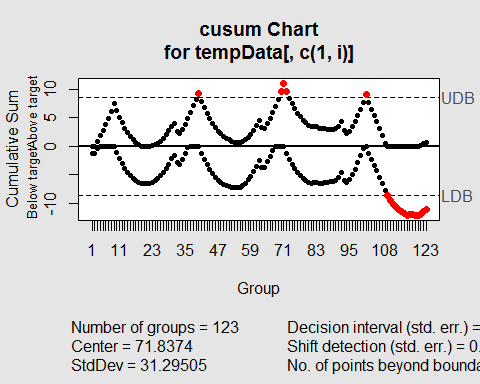
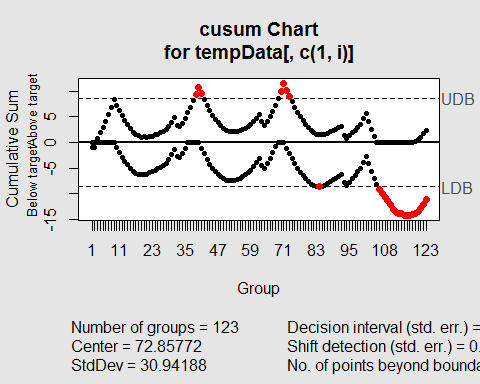
#stdev of temperaures  
cat("stdev of the temperature data: ",sd(as.matrix(tempData[,2:20])), "\n")

## stdev of the temperature data: 8.61742

temp\_stdev <- sd(as.matrix(tempData[,2:20]))  
  
#Starting with the C value as 1 stdev of data  
c <- (1 \* temp\_stdev)  
  
#Starting with T as 3 stdevs above and below   
tUpper <- (3 \* temp\_stdev)  
tLower <- (-3 \* temp\_stdev)  
  
cat("tUpper is: ", tUpper, "\ntLower is: ",tLower, "\n")

## tUpper is: 25.85226   
## tLower is: -25.85226

#targetValue temperature   
target <- mean(as.matrix(tempData[,2]))  
  
##-------------------------------------------------------------------------  
  
##building the cusum model using the qcc package - se.shift was optimized through manual testing to find the number that produced the best results  
results <- NULL  
for (i in 2:20) {  
 q <- cusum(tempData[,c(1,i)], decision.interval = temp\_stdev, se.shift = 0.1, add.stats = TRUE)  
 results[i] <- min(unlist(q$violations["lower"]))  
  
}



tempData[,1] <- as.character(tempData[,1])  
results <- unlist(results)  
daymean <- round(mean(results,na.rm = TRUE),0)  
  
#results  
cat("average day that summer ends: ",tempData[daymean,1])

## average day that summer ends: 30-Sep

results

## [1] NA 84 109 51 85 84 109 109 108 50 111 83 83 110 82 111 108  
## [18] 108 50 110