snewt
ISYE6501x - HW#2
Due May 31, 2018

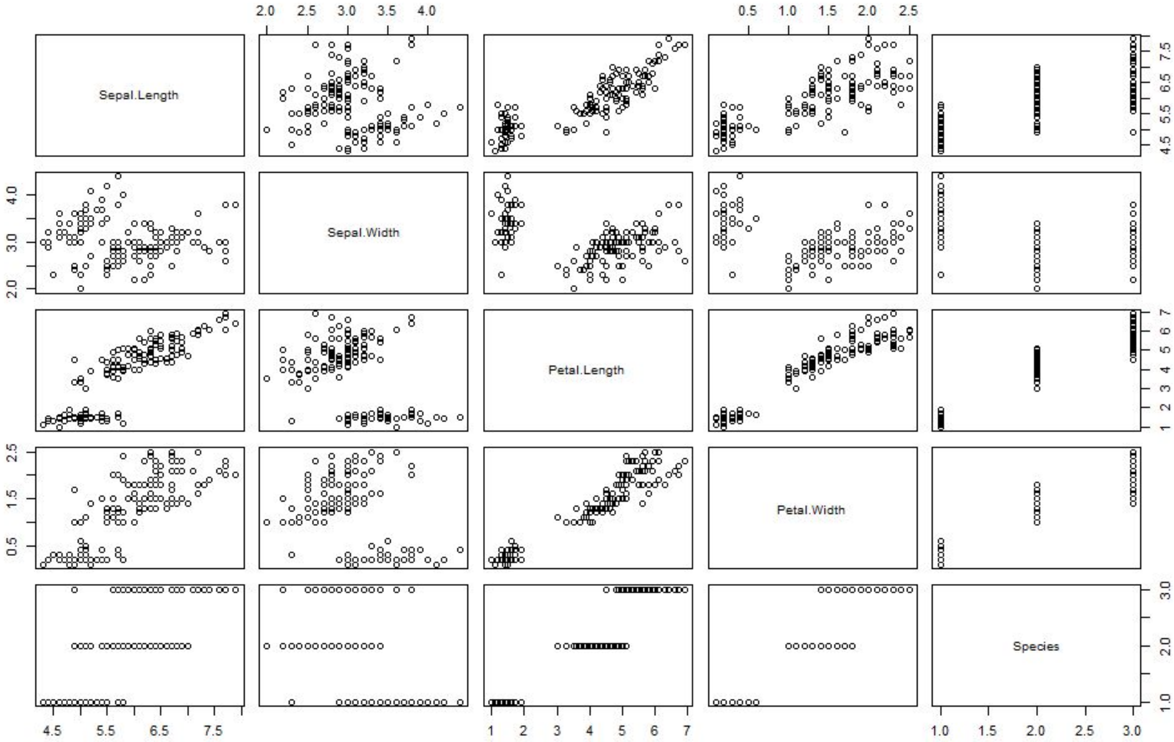
<u>Question 4.1:</u>

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.

Answer:
I used a clustering model in my thesis work to see if you could tell students' education level (in chemistry) by their observed skill level at various laboratory tasks. Spoiler alert: you can't. There was no significant difference whatsoever between students who didn't finish general chemistry 1, students who finished 1 course, and students who finished 2 courses (general chemistry 1&2). We also tried to see if we could predict sex by clustering on overperformance/underperformance from their own expectations (ie are women students more likely to underassess their own performance) - that was also a no-go. TL;DR: I got a degree for trying a bunch of models that didn't work. It's the saddest consolation prize.

<u>Question 4.2:</u>

First plot(iris) to see shape of data:



So it looks like petal length is pretty strongly correlated with each of the other numeric variables, and that there are pretty well separated groups for species by petal length, width as well, though the virginica and versicolor species appear to overlap some on both, while the setosa species is fairly well separated for both predictors alone.

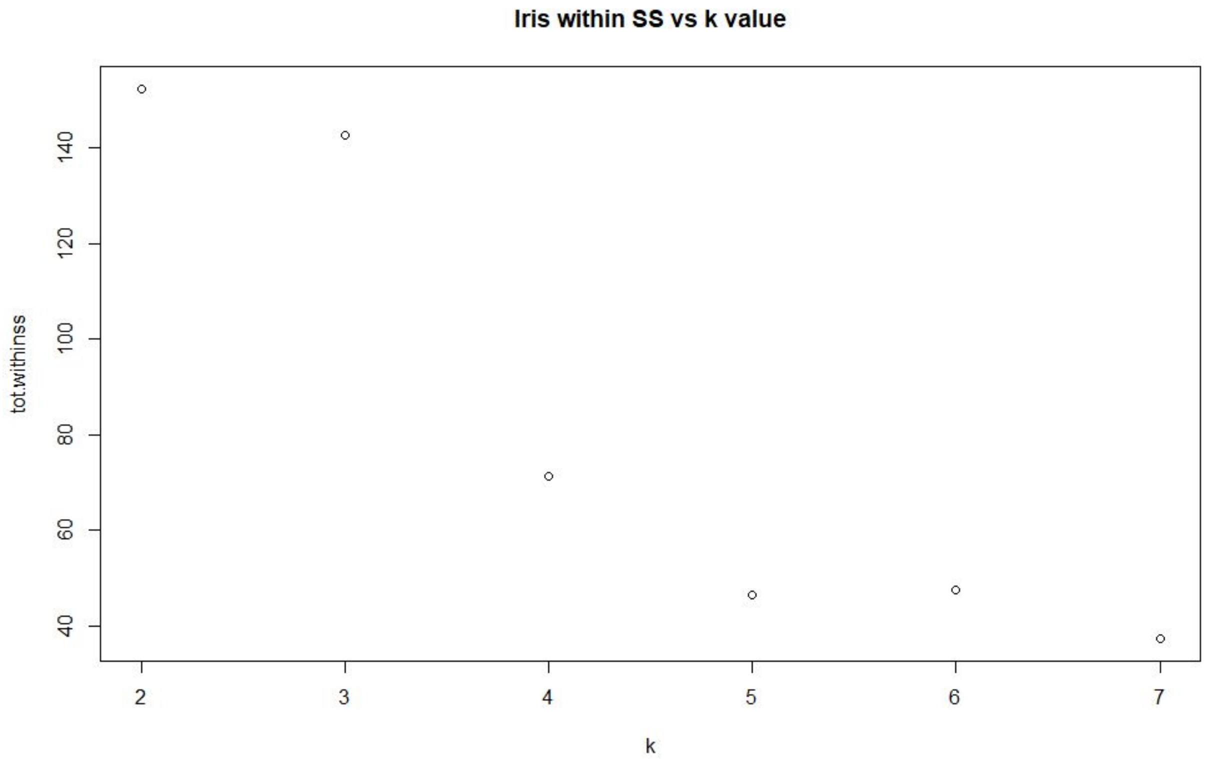
So we go ahead and do what the assignment is asking, we use the kmeans() function to find those clusters, and we're showing the three cluster here because there are three species of flower represented, but I did run some other clusters (2-7 specifically) to be sure there weren't better options available.

I also used different numbers of iterations, starting groups to optimize that 3 cluster solution, once the initial kmeans algorithm had been completed.

```
> kmeans3<-kmeans(as.matrix(iris[1:4]),3,iter.max=10,nstart=1)
> kmeans3
K-means clustering with 3 clusters of sizes 33, 96, 21

Cluster means:
  Sepal.Length Sepal.Width Petal.Length Petal.Width
1     5.175758    3.624242     1.472727   0.2727273
2     6.314583    2.895833     4.973958   1.7031250
3     4.738095    2.904762     1.790476   0.3523810

Clustering vector:
  [1] 1 3 3 3 1 1 1 1 3 3 1 1 3 3 1 1 1 1 1 1 1 1 1 1 3 3 1 1 1 3 3 1 1 1 3 1 1 1 3 1 1
 [42] 3 3 1 1 3 1 3 1 1 2 2 2 2 2 2 2 3 2 2 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
2
 [83] 2 2 2 2 2 2 2 2 2 2 2 3 2 2 2 2 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
2
[124] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2

Within cluster sum of squares by cluster:
[1]   6.432121 118.651875  17.669524
 (between_SS / total_SS =  79.0 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
[6] "betweenss"    "size"         "iter"         "ifault"
```

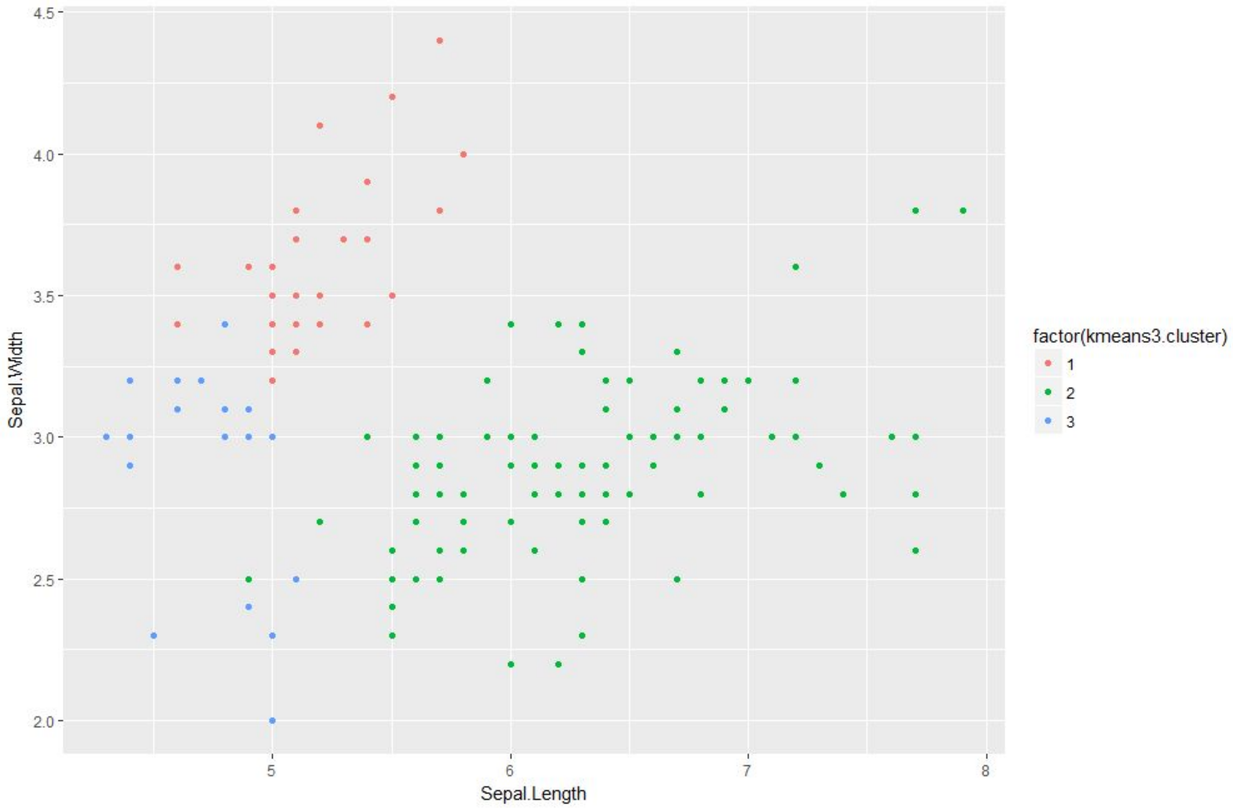
So we've got a way to show the cluster of each item, and a measure of fit of the clusters to the data.

```
> iriswc <- data.frame(iris,kmeans3$cluster)
> kmeans4<-kmeans(as.matrix(iris[1:4]),4,iter.max=10,nstart=1)
> iriswc <- data.frame(iriswc,kmeans4$cluster)
> kmeans5<-kmeans(as.matrix(iris[1:4]),5,iter.max=10,nstart=1)
> iriswc <- data.frame(iriswc,kmeans5$cluster)
> kmeans6<-kmeans(as.matrix(iris[1:4]),6,iter.max=10,nstart=1)
> iriswc <- data.frame(iriswc,kmeans6$cluster)
> kmeans2<-kmeans(as.matrix(iris[1:4]),2,iter.max=10,nstart=1)
> iriswc <- data.frame(iriswc,kmeans2$cluster)
> kmeans7<-kmeans(as.matrix(iris[1:4]),7,iter.max=10,nstart=1)
> iriswc<-data.frame(iriswc,kmeans7$cluster)
> elbow <- data.frame(k=c(2:6), tot.withinss = c(kmeans2$tot.withinss,
kmeans3$tot.withinss, kmeans4$tot.withinss, kmeans5$tot.withinss,
kmeans6$tot.withinss))
```



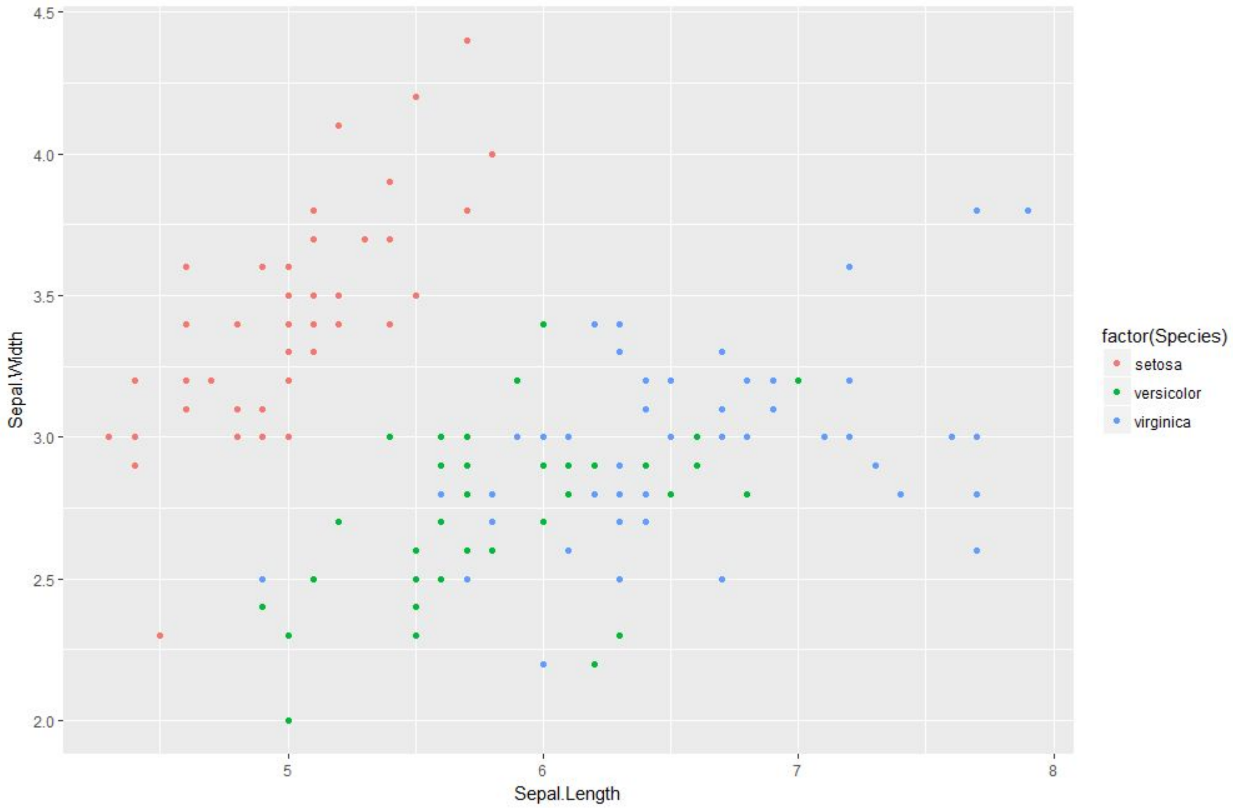
So from the elbow graph, 4 or 5 clusters would probably be ideal, if the elbow is the most important part. If there are, say, three specific species of iris, though, or if there are four light conditions that these irises were found in, then that would be a better indicator of how many clusters to choose. So since we knew already that there were three species, even though we seem to be able to separate into 4 clusters more readily, we should still stick with 3.

```
> ggplot(iriswc,aes(Sepal.Length,Sepal.Width,colour=factor(kmeans3.cluster))) +
geom_point()
```



But to compare the species with k means cluster, since there’s only three clusters, we’d need to compare the k = 3 clustered items.

```
> ggplot(iriswc,aes(Sepal.Length,Sepal.Width,colour=factor(Species))) +
geom_point()
```



Create another column where species = factor so that setosa = 1, versicolor = 2, virginica = 3.

```
> sum(iriswc$kmeans3.cluster == iriswc$fspec)/nrow(iriswc)*100
[1] 52.66667
```

Translation: for k = 3, we’re classifying the species right 52.7% of the time. Which is not very good at all.

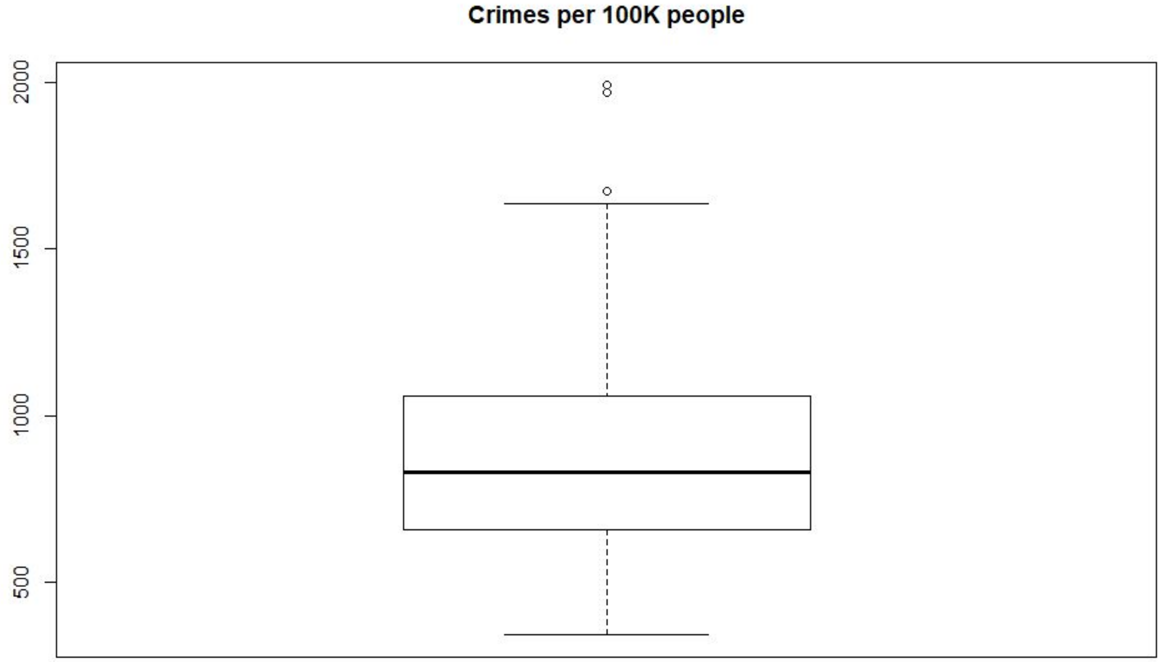
Let’s try some other things to see if we can find something that is good.

```
> kmeans3a<-kmeans(as.matrix(iris[1:4]),3,iter.max=15,nstart=1)
> kmeans3b<-kmeans(as.matrix(iris[1:4]),3,iter.max=20,nstart=1)
> kmeans3c<-kmeans(as.matrix(iris[1:4]),3,iter.max=10,nstart=2)
> kmeans3d<-kmeans(as.matrix(iris[1:4]),3,iter.max=10,nstart=3)
> speciesd <- factor(iriswc$Species, c("virginica","versicolor","setosa"), ordered =
TRUE)
> sum(kmeans3d$cluster == iriswc[,18])/nrow(iriswc)*100
[1] 89.33333
```

I skipped some of the less good ones, because I started from the back of the dataframe. That's really good, and on quick visual inspection of the dataframe, I can already tell you that kmeans3d is definitely the best one that we've got. So. We started with 3 groups. We were shooting for 3 clusters. We had a max of 10 iterations. While it's possible we're going to get better accuracy than that, we're stopping there because it's really good and I've still got a whole nother set of problems to get through.

Question 5.1

```
> boxplot(crimes$Crime, main = "Crimes per 100K people")
```



```
> grubbs.test(crimes$Crime,type=10,opposite=FALSE,two.sided = FALSE)

        Grubbs test for one outlier

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

So that means that at p = .05, we’re showing NO significant outliers, or at least not only one.

Let’s see if we find two:

```
> grubbs.test(crimes$Crime,type=11,opposite=FALSE,two.sided = FALSE)

	Grubbs test for two opposite outliers

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

So checking 2 tailed we’re also not finding any, because p=1 is VERY not significant.

What if we check for 2 on the same side?

```
> grubbs.test(crimes$Crime,type=11,opposite=FALSE,two.sided = TRUE)

	Grubbs test for two opposite outliers

data:  crimes$Crime
G = 4.26880, U = 0.78103, p-value < 2.2e-16
alternative hypothesis: 342 and 1993 are outliers
```

Heyo! We found 2 outliers with a VERY significant p value -- 1993 is DEFINITELY an outlier. Strangely, the box and whisker plot seems to indicate 2 outliers on the very top of the data set, with none on the bottom of the dataset, but I’m showing that there are definitely 2. On the same side. So what we really needed there, was to use type = 20 -- two outliers possible on the same side.

```
> max(crimes$Crime)
[1] 1993
> grubbs.test(crimes$Crime,type=10,opposite=TRUE,two.sided = FALSE)

	Grubbs test for one outlier

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

Just ranging through the alternate possibilities of parameters, opposite and two.sided, to see if we can use one of them as well. This is 100% a fishing expedition and was very frowned upon in my statistics classes, but I’m trying to get a handle on what R’s doing, not practice safe stats.

```
> grubbs.test(crimes$Crime,type=10,opposite=FALSE,two.sided = TRUE)

        Grubbs test for one outlier

data:  crimes$Crime
G = 2.81290, U = 0.82426, p-value = 0.1577
alternative hypothesis: highest value 1993 is an outlier
```

So again, we got a non-significant 1 outlier test - and again we were just playing with parameters.

Let’s actually try looking for 2 on the same side, which I believe from looking at the box and whisker should come back significant:

```
> grubbs.test(crimes$Crime,type=20)
Error in qgrubbs(q, n, type, rev = TRUE) : n must be in range 3-30
> grubbs.test(crimes$Crime,type=20, opposite = FALSE, two.sided = FALSE)
Error in qgrubbs(q, n, type, rev = TRUE) : n must be in range 3-30
```

Uh oh! So instead we’re getting an error that the internet basically says, understand more stats to understand what’s going on here. I *think* it’s really that we’ve got 47 data points, and we’ve got a max of 30 for that qgrubbs argument n, but I’m really not sure.

It does look like I could probably use pmax (rather than max) to tell me the largest couple values of crimes$Crime, but it’s giving me a longer list than I care to replicate. Those two largest values (that I’d consider leaving out or at least exploring, from the boxplot) are 1993 and 1969.

While that’s unfortunate, because as was discussed in a lesson this week following your intuition is important (and I’d really want to examine those two data points!) to understanding the art, not just the science; I’ve only got a finite amount of time and it’s not going to get significant returns to keep banging my head on that problem right now.

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?

Answer:
I’m from Atlanta, lived there my whole life until last year, and so our example problem would be the best one I’d find appropriate.

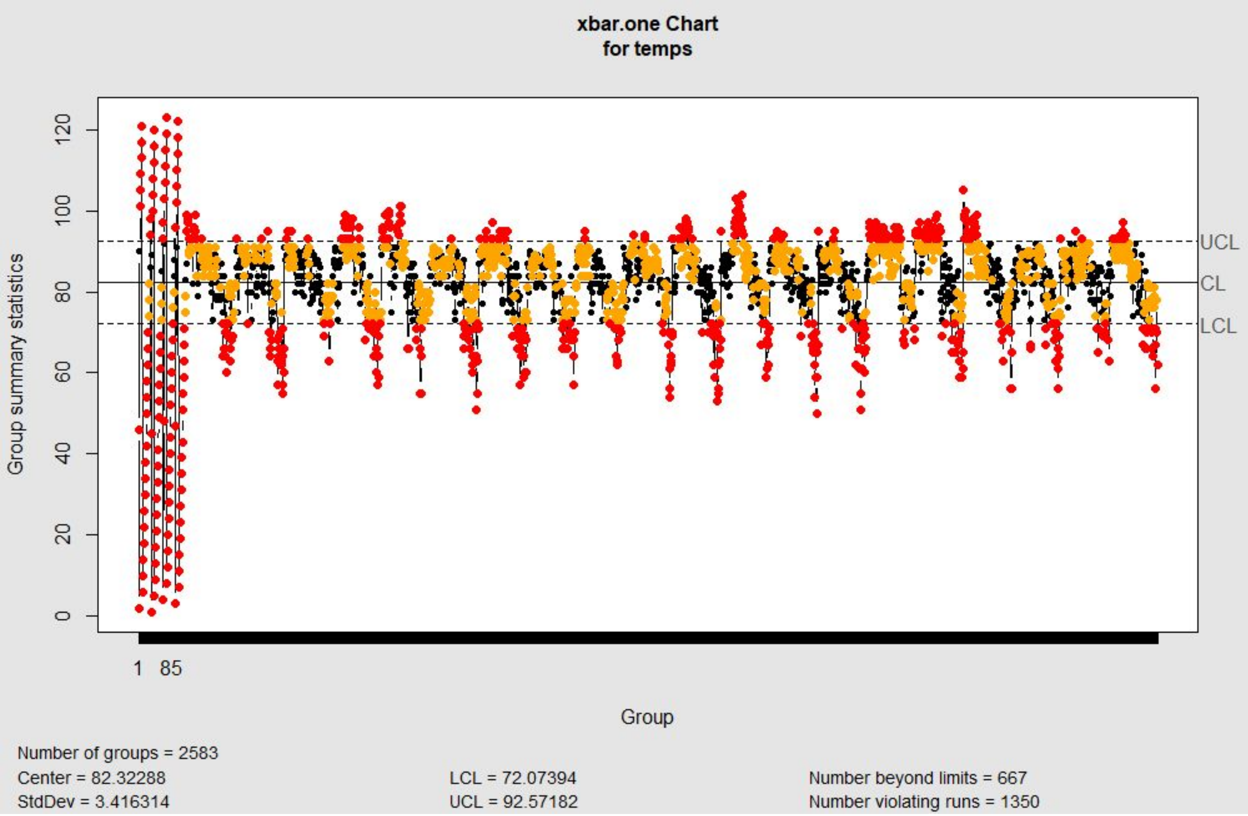
Another example to choose could be choosing pulse rates for patients in the software I work in, maybe. If you choose a critical value of say the 90% of the acceptable range of pulse rates, and thresholds of 85%, that could also be an example .

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.

```
> library(qcc)
  __ _  ___ ___
 / _  |/ __/ __|  Quality Control Charts and
| (_| | (_| (__   Statistical Process Control
 \__  |\___\___|
    |_|          version 2.7
Type 'citation("qcc")' for citing this R package in publications.
> temps<- read.delim("6.2tempsSummer2018.txt", header=T, sep = "\t")
```

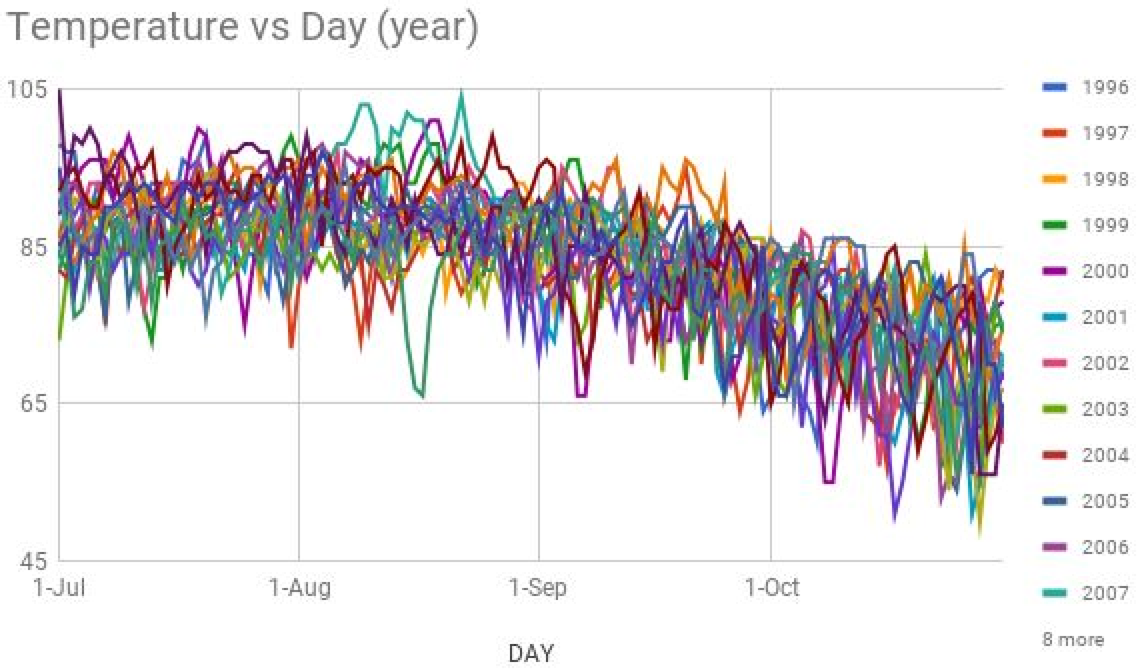
```
> summary(qcc(temps,"xbar.one"))

Call:
qcc(data = temps, type = "xbar.one")

xbar.one chart for temps

Summary of group statistics:
     Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
  1.00000  78.00000  85.00000  82.32288  90.00000 123.00000

Group sample size:  21
Number of groups:  2583
Center of group statistics:  82.32288
Standard deviation:  3.416314

Control limits:
      LCL      UCL
 72.07394 92.57182
```



<u>Question 6.2.2</u>

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

Unclear.


Temperature vs Day (year)
105
85
65
45
1-Jul
1-Aug
1-Sep
1-Oct
DAY
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
8 more