HW7

Enguang Fan

3/19/2020

City crime data

0. Inspect the help file of the **all.us.city.crime.1970** data in the **cluster.datasets** package to familiarize yourself with it. We will be clustering cities based on the crime rate variables in columns 5-10.

```
library(cluster.datasets)
set.seed(5)
data(all.us.city.crime.1970)
help(all.us.city.crime.1970)
```

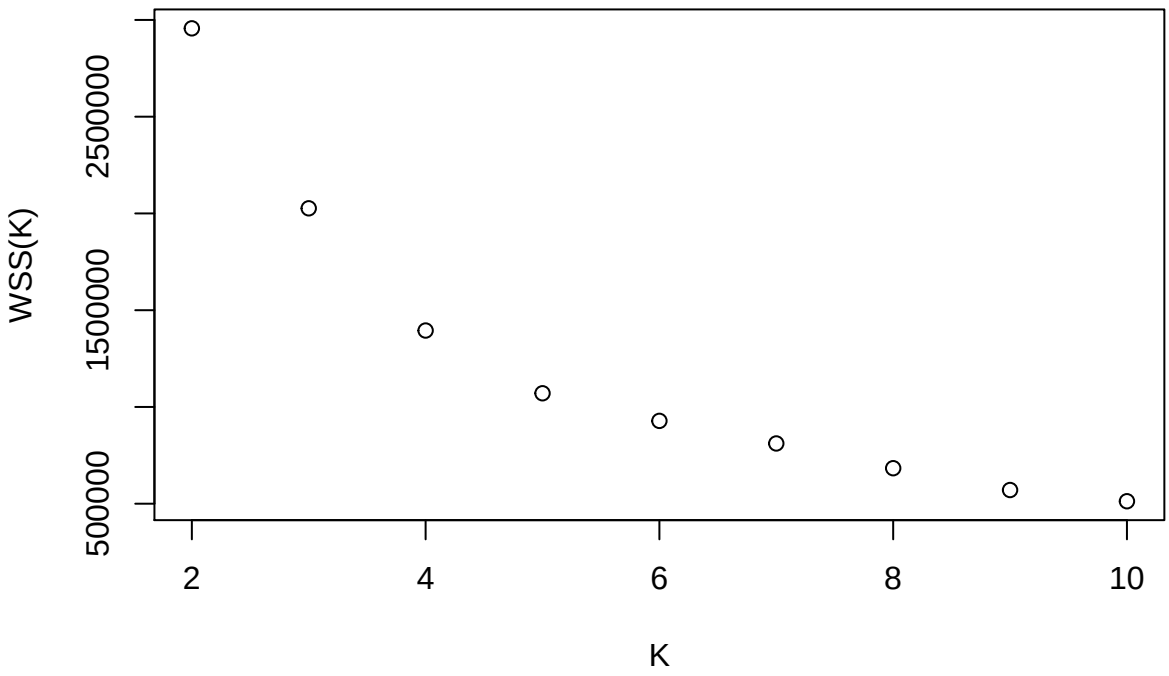
```
## starting httpd help server ... done
```

```
#df <- data(all.us.city.crime.1970)
```

1. For K=2,3,4,...,10 find the within cluster sum of squares and plot this against K. Based on this, what would be a reasonable number of clusters?

```
## save WSS(K) for K=2,10)
WSS=NULL
for(K in 2:10){
 kms=kmeans(all.us.city.crime.1970[,5:10],centers=K,nstart=10)
 wssk=sum(kms$withinss)
 WSS=c(WSS,wssk)
}

plot(c(2:10),WSS,xlab="K",ylab="WSS(K)")
```



There is an elbow when K = 4 or 5, so I'd use K = 4

2. Find the K-means solution for the number K you found in 1. Which cities go in which clusters?

```
set.seed(5)
cityclusters=kmeans(all.us.city.crime.1970[,5:10],centers=4)
city = all.us.city.crime.1970[,1]
```

```
print("cities in cluster 1")
```

```
## [1] "cities in cluster 1"
```

```
city[cityclusters$cluster==1]
```

```
## [1] "Buffalo"      "Chicago"      "Cincinnatti"  "Milwaukee"    "Paterson"
## [6] "Philadelphia" "Pittsburgh"   "San Diego"
```

```
print("cities in cluster 2")
```

```
## [1] "cities in cluster 2"
```

```
city[cityclusters$cluster==2]
```

```
## [1] "Detroit"       "Los Angeles"   "Miami"         "New York"
## [5] "San Francisco"
```

```
print("cities in cluster 3")
```

```
## [1] "cities in cluster 3"
```

```
city[cityclusters$cluster==3]
```

```
## [1] "Baltimore"   "Boston"      "Cleveland"   "Houston"     "Minneapolis"
## [6] "Newark"      "St Louis"    "Washington"
```

```
print("cities in cluster 4")
```

```
## [1] "cities in cluster 4"
```

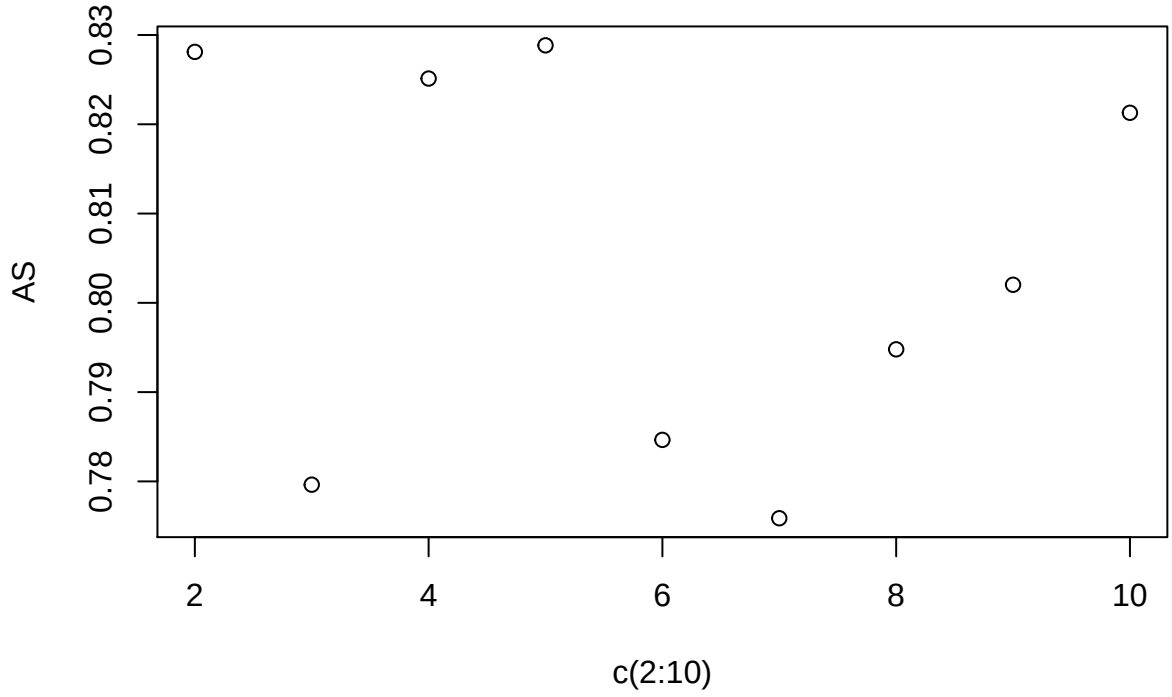
```
city[cityclusters$cluster==4]
```

```
## [1] "Anaheim" "Dallas"  "Seattle"
```

3. Now find K based on maximizing the average silhouette. What is the optimal K based on this criterion?

```
set.seed(5)
#use the function from lecture notes
avsil=function(x,kmobject){
## x is the raw data
## kmobject is the resulting list after running kmeans
 n=nrow(x)
 K=length(kmobject$size)
 sils=c(1:n)
for(i in 1:n){
 distvals=c(1:K)
for(k in 1:K){
 distvals[k]=sum((x[i,]-kmobject$centers[k,])^2)
 }
 distvals=sort(distvals)
 sils[i]=(distvals[2]-distvals[1])/distvals[2]
 }
return(mean(sils))
}

AS=c(1:9)
for(k in 2:10){
 km=kmeans(all.us.city.crime.1970[,5:10],centers=k,nstart=10)
 AS[k-1]=avsil(all.us.city.crime.1970[,5:10],km)
}
plot(c(2:10),AS)
```



The optimal K = 5

4. Repeat 1 and 2, but after standardizing each variable and replacing them with z-scores. Variables with higher variances could dominate the euclidean distances used in creating clusters. Did K change? What about the cluster memberships?

```
set.seed(5)
#standardized the data
df = all.us.city.crime.1970
df1 = all.us.city.crime.1970
#head(df,5)
dim(df)
```

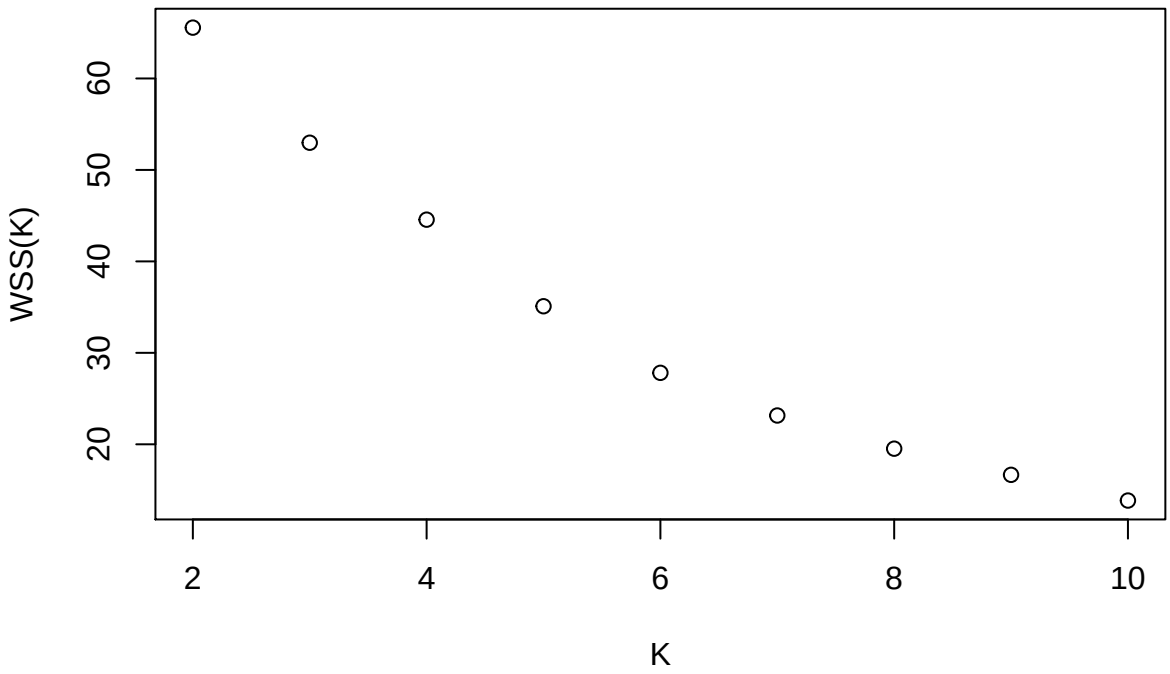
```
## [1] 24 10
```

```
for (i in 5:10) {
  df[,i] = (df1[,i] - mean(df1[,i])) / sd(df1[,i])
}
#df[,5] = (df1[,5] - mean(df1[,5])) / sd(df1[,5])
```

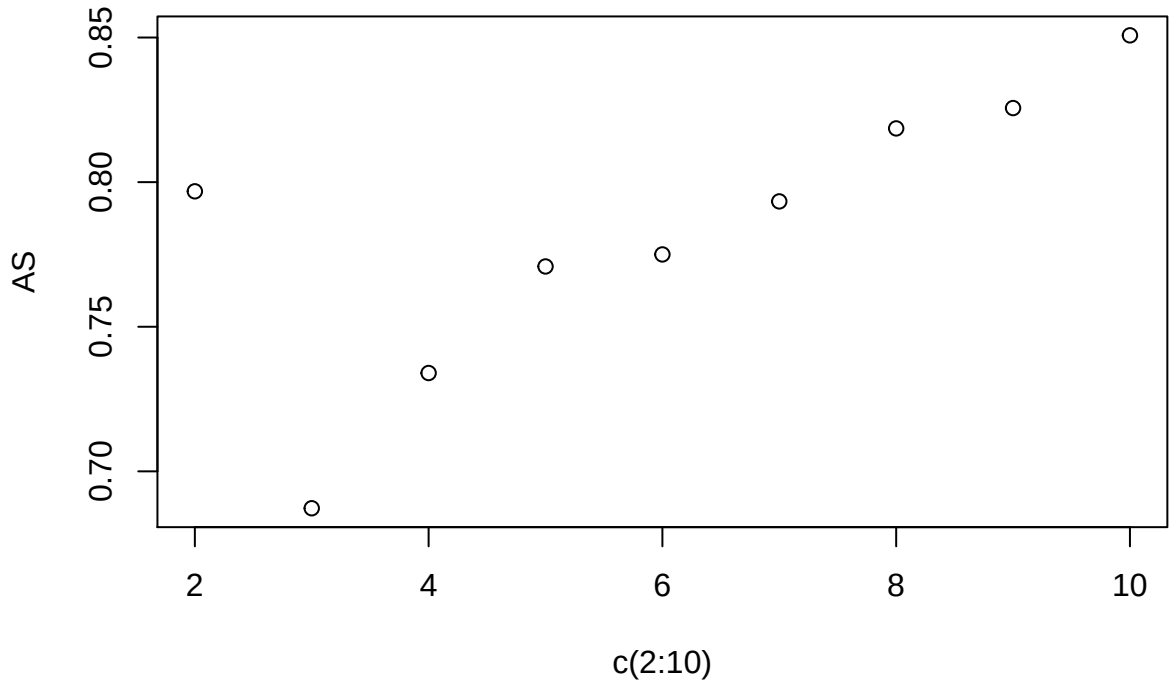
```
#use WSS against K plot
## save WSS(K) for K=2,10)
WSS=NULL
for(K in 2:10){
 kms=kmeans(df[,5:10],centers=K,nstart=10)
```

```
 wssk=sum(kms$withinss)
 WSS=c(WSS,wssk)
}

plot(c(2:10),WSS,xlab="K",ylab="WSS(K)")
```




```
#by maximizing the average silhouette
AS=c(1:9)
for(k in 2:10){
 km=kmeans(df[,5:10],centers=k,nstart=10)
 AS[k-1]=avsil(df[,5:10],km)
}
plot(c(2:10),AS)
```



We use K = 10

```
set.seed(5)
cityclusters=kmeans(df[,5:10],centers=10)
city = all.us.city.crime.1970[,1]


city[cityclusters$cluster==1]
```

```
## [1] "Dallas"
```

```
city[cityclusters$cluster==2]
```

```
## [1] "Los Angeles"   "San Francisco"
```

```
city[cityclusters$cluster==3]
```

```
## [1] "Baltimore" "Miami"
```

```
city[cityclusters$cluster==4]
```

```
## [1] "Anaheim" "Seattle"
```

```
city[cityclusters$cluster==5]
```

```
## [1] "Detroit"  "New York"
```

```
city[cityclusters$cluster==6]
```

```
## [1] "Cleveland"
```

```
city[cityclusters$cluster==7]
```

```
## [1] "Boston"
```

```
city[cityclusters$cluster==8]
```

```
## [1] "Buffalo"      "Cincinnatti"  "Minneapolis"  "Philadelphia" "Pittsburgh"
## [6] "San Diego"
```

```
city[cityclusters$cluster==9]
```

```
## [1] "Chicago"    "Houston"    "Newark"     "St Louis"   "Washington"
```

```
city[cityclusters$cluster==10]
```

```
## [1] "Milwaukee" "Paterson"
```

5. Using these standardized data inspect the cluster means and use them to describe differences between clusters.

```
set.seed(5)
km=kmeans(df[,5:10],centers=10,nstart=10)
km
```

```
## K-means clustering with 10 clusters of sizes 1, 2, 2, 2, 2, 1, 1, 6, 5, 2
##
## Cluster means:
##         murder        rape     robbery    assault      burglary  car.theft
## 1   1.82589438  1.62066415 -0.40309969  1.3978368  0.5488114091 -0.4591519
## 2  -0.06689165  2.11629864  0.27791925  1.0301483  1.5565603790  1.2141520
## 3   1.03310442  0.25198542  1.21957508  2.0540909  0.5969947271  0.2745963
## 4  -1.11733835 -0.02993511 -0.83469195 -0.7384798  0.9804109171 -0.9736705
## 5   0.67634893  0.21106148  2.12479780  0.6205713  1.2100505602  0.7734556
## 6   1.05292417 -0.40734486  0.05651803 -0.5197284 -0.9992058283  2.3639892
## 7  -0.94887048 -0.76201908 -0.79545629 -0.8641455 -0.5317251262  1.3617964
## 8  -0.74736970 -0.65440425 -0.75902317 -0.7493398 -0.8362026888 -0.8707668
## 9   0.77544768  0.26380790  0.47690010  0.1746908  0.0005467611  0.2907030
## 10 -1.18670747 -1.47136753 -1.13176194 -1.1620197 -1.3457156471 -1.0363075
##
## Clustering vector:
```

```
##  [1]  4  3  7  8  9  8  6  1  5  9  2  3 10  8  5  9 10  8  8  9  2  8  4  9
##
## Within cluster sum of squares by cluster:
##  [1] 0.0000000 1.2413656 2.3643808 0.7293629 1.4559164 0.0000000 0.0000000
##  [8] 3.0625267 5.0090458 0.3733957
##  (between_SS / total_SS =  89.7 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
## [6] "betweenss"    "size"         "iter"         "ifault"
```

So the 1st cluster means has the biggest murder rate 2nd cluster means has the biggest rape rate 3rd cluster means has the biggest assult rate 4th cluster means has the second biggest burglary rate 5th cluster means has the robbery murder rate

```
cities above representing cities with serious crimes

6th cluster means has the biggest car theft rate
7th cluster means has decrease in all other crime but increase in car theft, representing cities witl
8th cluster means has decrease in all crime rates
9th cluster means has mild increase in most crime rate
10th cluster means has the lowest rape and murder rate, representing good cities.
```