Assignment 1

Jordan Whitaker

6/21/2020

#install.packages(c("tidyverse", "cluster", "factoextra", "dendextend"))  
options(tidyverse.quiet = TRUE)  
library(tidyverse)

library(cluster)

## Warning: package 'cluster' was built under R version 3.6.3

library(factoextra)

## Warning: package 'factoextra' was built under R version 3.6.3

## Welcome! Want to learn more? See two factoextra-related books at https://goo.gl/ve3WBa

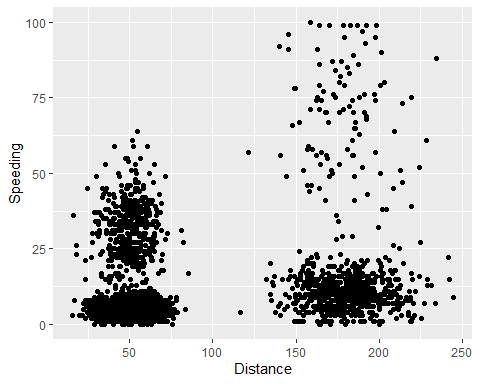
library(dendextend)

trucks = read\_csv("trucks.csv")

## Parsed with column specification:  
## cols(  
## Driver\_ID = col\_double(),  
## Distance = col\_double(),  
## Speeding = col\_double()  
## )

Task 1: Looking at Relationship between Distance and Speeding There definitely seems to be a strong grouping of points in three places. There are also some scatter points that could be grouped into a few more clusters.

ggplot(trucks, aes(Distance, Speeding)) +   
 geom\_point()

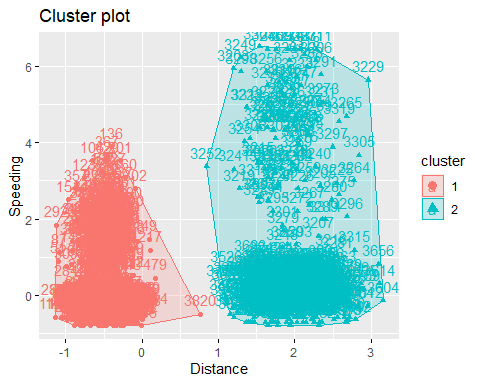


Task 2:

trucks2 = trucks %>% select(Distance, Speeding)  
  
trucks2 = as.data.frame(scale(trucks2))

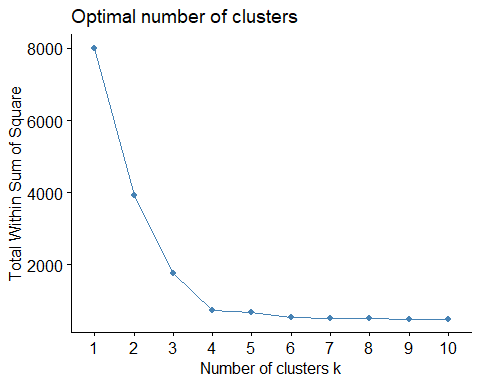
Task 3: The below cluster plot, with 2 clusters, does fully embody all of the points. However, these two clusters might not tell the entire story of the data. I believe that there is still a need to increase the number of clusters. Diagnostic measures will tell..

set.seed(64)  
clusters = kmeans(trucks2, 2)  
  
fviz\_cluster(clusters, trucks2)

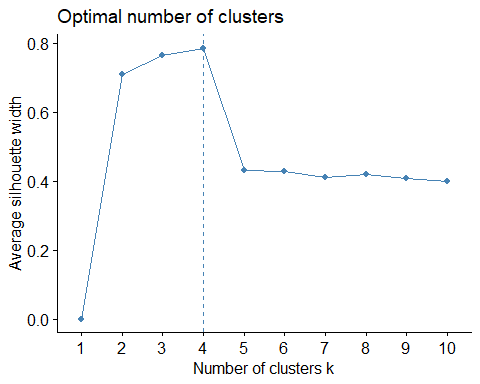


Task 4: The two diagnostic measures to identify the optimal K value seem to agree on 4 clusters.

set.seed(64)  
fviz\_nbclust(trucks2, kmeans, method ="wss")

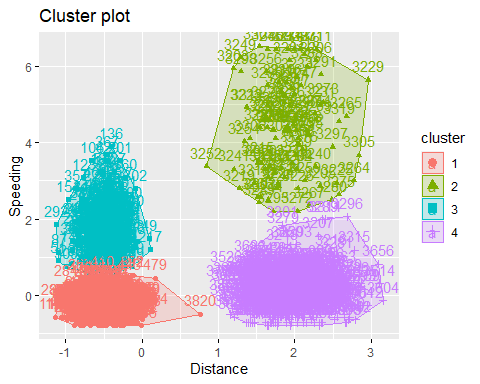


fviz\_nbclust(trucks2, kmeans, method ="silhouette")



Task 5:

set.seed(64)  
clusters\_new = kmeans(trucks2, 4)  
  
fviz\_cluster(clusters\_new, trucks2)



Task 6: I would describe the above cluster plot as beautiful. These four clusters seem to fully describe the data and there would strong evidence to back any decisions being made around this data.

bball = read\_csv("kenpom20.csv")

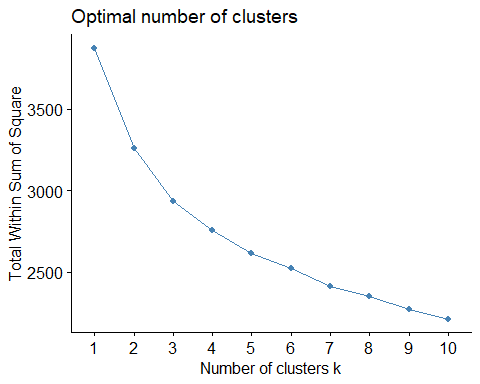
## Parsed with column specification:  
## cols(  
## TeamName = col\_character(),  
## AdjTempo = col\_double(),  
## AdjOE = col\_double(),  
## AdjDE = col\_double(),  
## eFGPct = col\_double(),  
## TOPct = col\_double(),  
## ORPct = col\_double(),  
## FTRate = col\_double(),  
## eFGPctD = col\_double(),  
## TOPctD = col\_double(),  
## ORPctD = col\_double(),  
## FTRateD = col\_double()  
## )

Task 7:

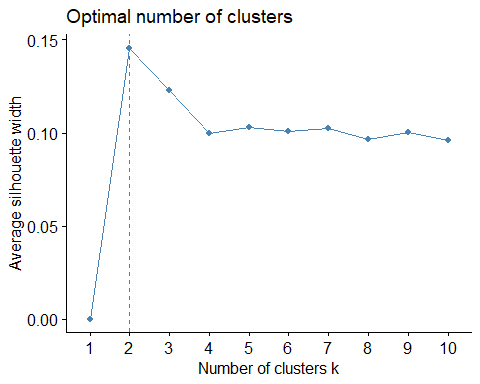
bball2 = bball %>% select(AdjTempo:FTRateD)  
  
bball2 = as.data.frame(scale(bball2))

There does not seem to be consensus between the two methods. With the WSS method, I would have picked 4 clusters, while the silhouette method identified 2 as the optimal number.

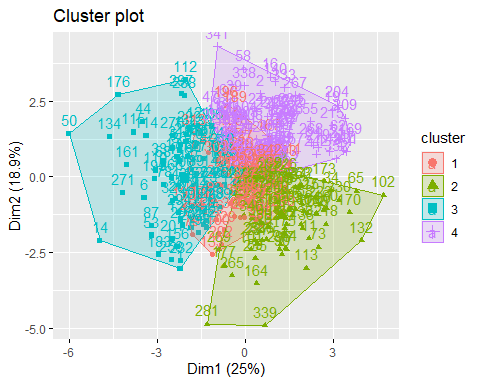
set.seed(123)  
fviz\_nbclust(bball2, kmeans, method ="wss")



fviz\_nbclust(bball2, kmeans, method ="silhouette")

 Task 8:

set.seed(1234)  
clusters\_bball = kmeans(bball2, 4)  
  
fviz\_cluster(clusters\_bball, bball2)



Task 9: The only patterns that are somewhat clear is that teams who do not score a lot of points are likely to give up a lot of points and teams who do score a lot of points are not giving up a lot of points. Those groups are identified in clusters 3 and 2 respectively.

bball2 = bball2 %>% mutate(clusternum = clusters\_bball$cluster)  
  
  
ggplot(bball2, aes(AdjOE, AdjDE, color = factor(clusternum))) +  
 geom\_point()

