# Mod6Ass1

## BAN502

### Jesse Chroman

options(tidyverse.quiet=TRUE)  
  
library(tidyverse)  
library(cluster)   
library(factoextra)

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

library(dendextend)

##   
## ---------------------  
## Welcome to dendextend version 1.13.4  
## Type citation('dendextend') for how to cite the package.  
##   
## Type browseVignettes(package = 'dendextend') for the package vignette.  
## The github page is: https://github.com/talgalili/dendextend/  
##   
## Suggestions and bug-reports can be submitted at: https://github.com/talgalili/dendextend/issues  
## Or contact: <tal.galili@gmail.com>  
##   
## To suppress this message use: suppressPackageStartupMessages(library(dendextend))  
## ---------------------

##   
## Attaching package: 'dendextend'

## The following object is masked from 'package:stats':  
##   
## cutree

library(readr)  
trucks <- read\_csv("trucks.csv")

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

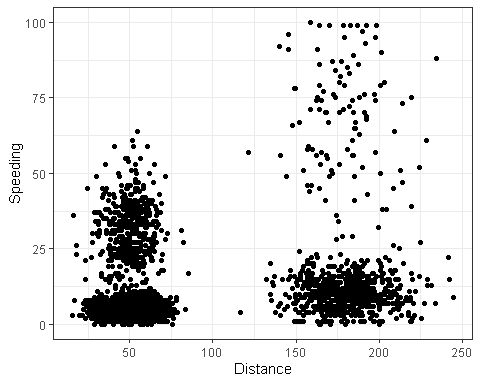
View(trucks)  
str(trucks)

## Classes 'spec\_tbl\_df', 'tbl\_df', 'tbl' and 'data.frame': 4000 obs. of 3 variables:  
## $ Driver\_ID: num 3.42e+09 3.42e+09 3.42e+09 3.42e+09 3.42e+09 ...  
## $ Distance : num 71.2 52.5 64.5 55.7 54.6 ...  
## $ Speeding : num 28 25 27 22 25 10 20 8 34 19 ...  
## - attr(\*, "spec")=  
## .. cols(  
## .. Driver\_ID = col\_double(),  
## .. Distance = col\_double(),  
## .. Speeding = col\_double()  
## .. )

summary(trucks)

## Driver\_ID Distance Speeding   
## Min. :3.423e+09 Min. : 15.52 Min. : 0.00   
## 1st Qu.:3.423e+09 1st Qu.: 45.25 1st Qu.: 4.00   
## Median :3.423e+09 Median : 53.33 Median : 6.00   
## Mean :3.423e+09 Mean : 76.04 Mean : 10.72   
## 3rd Qu.:3.423e+09 3rd Qu.: 65.63 3rd Qu.: 9.00   
## Max. :3.423e+09 Max. :244.79 Max. :100.00

ggplot(trucks, aes(x = Distance, y = Speeding)) +   
 geom\_point() +  
 theme\_bw()



Based on the scatterplot above, there does appear to be natural clustering in the data points.

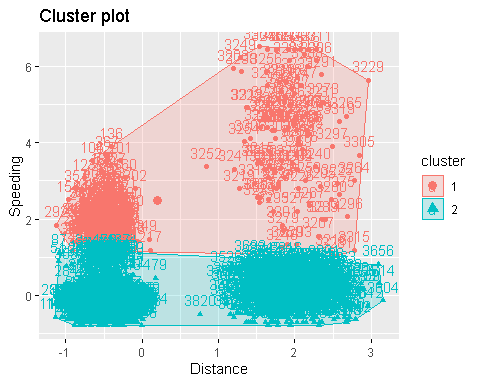
trucks2 = trucks %>% select("Distance","Speeding")  
trucks2\_scaled = as.data.frame(scale(trucks2))   
summary(trucks2\_scaled)

## Distance Speeding   
## Min. :-1.1319 Min. :-0.7821   
## 1st Qu.:-0.5759 1st Qu.:-0.4903   
## Median :-0.4248 Median :-0.3444   
## Mean : 0.0000 Mean : 0.0000   
## 3rd Qu.:-0.1947 3rd Qu.:-0.1255   
## Max. : 3.1560 Max. : 6.5127

str(trucks2\_scaled)

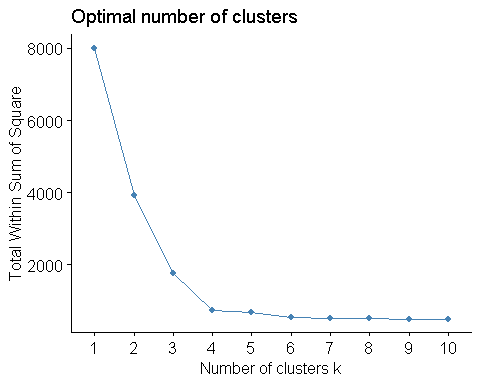
## 'data.frame': 4000 obs. of 2 variables:  
## $ Distance: num -0.0898 -0.4397 -0.2151 -0.3806 -0.4014 ...  
## $ Speeding: num 1.26 1.042 1.188 0.823 1.042 ...

set.seed(1234)  
clustered1 = kmeans(trucks2\_scaled, 2)  
fviz\_cluster(clustered1, trucks2\_scaled)

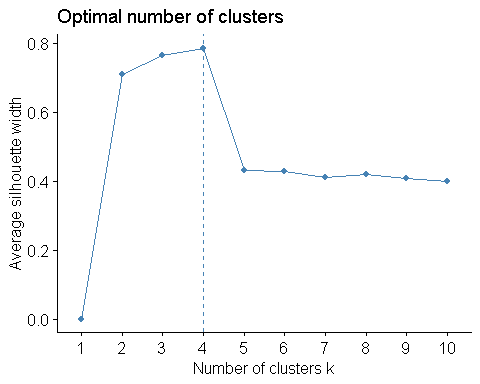


The plot shows two distinct clusters deliniated by Speeding at around a measure of 1.5 or 1.6. Two might not be the optimal amount of clusters seeing as there seems to be clusters which might be determined by daily distance driven.

set.seed(123)  
fviz\_nbclust(trucks2\_scaled, kmeans, method = "wss")

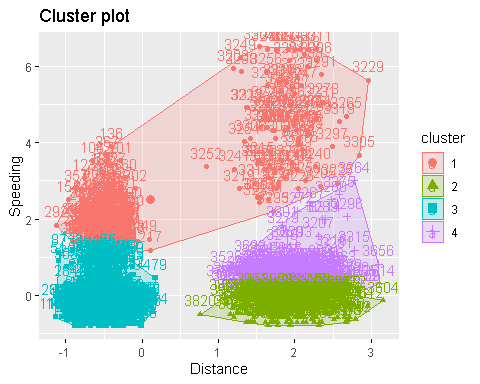


set.seed(123)  
fviz\_nbclust(trucks2\_scaled, kmeans, method = "silhouette")



Both methods indicate that 4 is the appropriate number of clustered to be used on this dataset, so yes, there is a consensus.

set.seed(1234)  
clustered2 = kmeans(trucks2\_scaled, 4)  
fviz\_cluster(clustered2, trucks2\_scaled )



### Task 6

There are 4 total clusters. Clusters 2, 3, and 4 consist of people on the lower end of the speeding spectrum however only cluster 3 consists of people who driver shorter distances. Clusters 2 and 4 consist of drivers driving longer distances. Cluster 1 spans both drivers covering short and long distances but consists of the people who speed moreso than others.

wineprice <- read\_csv("wineprice.csv")

## Parsed with column specification:  
## cols(  
## Year = col\_double(),  
## Price = col\_double(),  
## WinterRain = col\_double(),  
## AGST = col\_double(),  
## HarvestRain = col\_double(),  
## Age = col\_double(),  
## FrancePop = col\_double()  
## )

View(wineprice)  
str(wineprice)

## Classes 'spec\_tbl\_df', 'tbl\_df', 'tbl' and 'data.frame': 25 obs. of 7 variables:  
## $ Year : num 1952 1953 1955 1957 1958 ...  
## $ Price : num 7.5 8.04 7.69 6.98 6.78 ...  
## $ WinterRain : num 600 690 502 420 582 485 763 830 697 608 ...  
## $ AGST : num 17.1 16.7 17.1 16.1 16.4 ...  
## $ HarvestRain: num 160 80 130 110 187 187 290 38 52 155 ...  
## $ Age : num 31 30 28 26 25 24 23 22 21 20 ...  
## $ FrancePop : num 43184 43495 44218 45152 45654 ...  
## - attr(\*, "spec")=  
## .. cols(  
## .. Year = col\_double(),  
## .. Price = col\_double(),  
## .. WinterRain = col\_double(),  
## .. AGST = col\_double(),  
## .. HarvestRain = col\_double(),  
## .. Age = col\_double(),  
## .. FrancePop = col\_double()  
## .. )

summary(wineprice)

## Year Price WinterRain AGST HarvestRain   
## Min. :1952 Min. :6.205 Min. :376.0 Min. :14.98 Min. : 38.0   
## 1st Qu.:1960 1st Qu.:6.519 1st Qu.:536.0 1st Qu.:16.20 1st Qu.: 89.0   
## Median :1966 Median :7.121 Median :600.0 Median :16.53 Median :130.0   
## Mean :1966 Mean :7.067 Mean :605.3 Mean :16.51 Mean :148.6   
## 3rd Qu.:1972 3rd Qu.:7.495 3rd Qu.:697.0 3rd Qu.:17.07 3rd Qu.:187.0   
## Max. :1978 Max. :8.494 Max. :830.0 Max. :17.65 Max. :292.0   
## Age FrancePop   
## Min. : 5.0 Min. :43184   
## 1st Qu.:11.0 1st Qu.:46584   
## Median :17.0 Median :50255   
## Mean :17.2 Mean :49694   
## 3rd Qu.:23.0 3rd Qu.:52894   
## Max. :31.0 Max. :54602

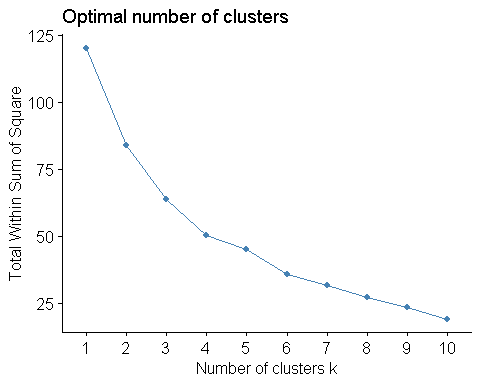
wine1 = wineprice %>% select("Price","WinterRain", "AGST", "HarvestRain", "Age")  
Wine2 = as.data.frame(scale(wine1))   
summary(Wine2)

## Price WinterRain AGST HarvestRain   
## Min. :-1.32596 Min. :-1.73332 Min. :-2.25947 Min. :-1.4856   
## 1st Qu.:-0.84329 1st Qu.:-0.52375 1st Qu.:-0.45801 1st Qu.:-0.8003   
## Median : 0.08284 Median :-0.03992 Median : 0.03548 Median :-0.2494   
## Mean : 0.00000 Mean : 0.00000 Mean : 0.00000 Mean : 0.0000   
## 3rd Qu.: 0.65777 3rd Qu.: 0.69339 3rd Qu.: 0.82524 3rd Qu.: 0.5165   
## Max. : 2.19343 Max. : 1.69885 Max. : 1.68888 Max. : 1.9275   
## Age   
## Min. :-1.586   
## 1st Qu.:-0.806   
## Median :-0.026   
## Mean : 0.000   
## 3rd Qu.: 0.754   
## Max. : 1.794

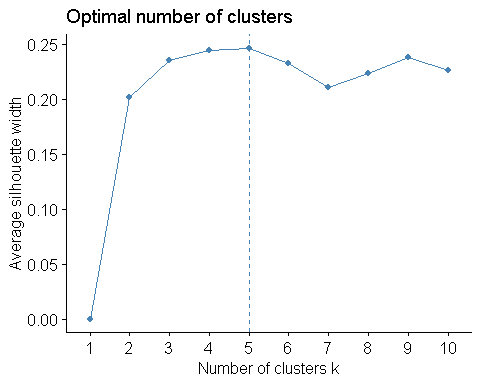
str(Wine2)

## 'data.frame': 25 obs. of 5 variables:  
## $ Price : num 0.658 1.495 0.951 -0.127 -0.446 ...  
## $ WinterRain : num -0.0399 0.6405 -0.7808 -1.4007 -0.176 ...  
## $ AGST : num 0.899 0.332 0.949 -0.557 -0.137 ...  
## $ HarvestRain: num 0.154 -0.921 -0.249 -0.518 0.517 ...  
## $ Age : num 1.79 1.66 1.4 1.14 1.01 ...

set.seed(123)  
fviz\_nbclust(Wine2, kmeans, method = "wss")

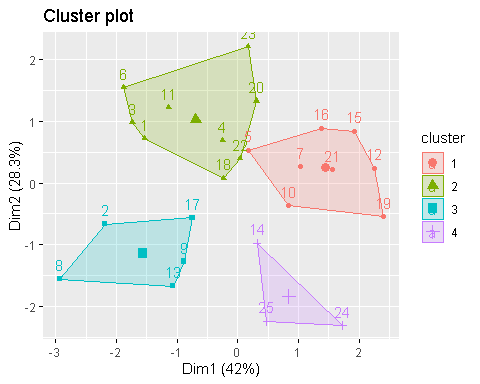


set.seed(123)  
fviz\_nbclust(Wine2, kmeans, method = "silhouette")

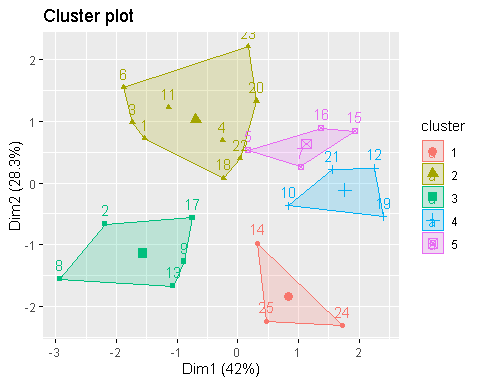


I don’t believe there is a consensus here as the silhouette method clearly indicates 5 clusters, while I believe that the WSS method would indicate that 4 is the appropriate number of clusters. WSS indicates 4 as this seems like the point after which the slope of the line starts to diminish.

set.seed(1234)  
clustered3 = kmeans(Wine2, 4)  
fviz\_cluster(clustered3, Wine2 )



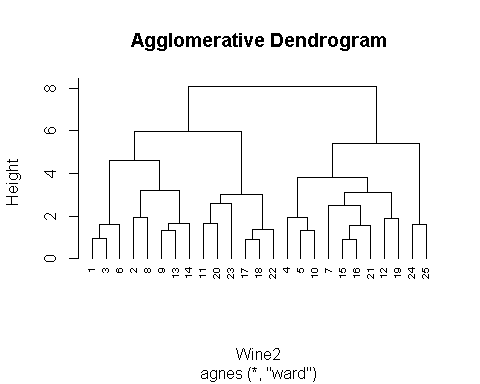
set.seed(1234)  
clustered4 = kmeans(Wine2, 5)  
fviz\_cluster(clustered4, Wine2 )



m = c( "average", "single", "complete", "ward")  
names(m) = c( "average", "single", "complete", "ward")  
  
ac = function(x) {  
 agnes(Wine2, method = x)$ac  
}  
map\_dbl(m, ac)

## average single complete ward   
## 0.5666719 0.2920143 0.7196616 0.8112139

hc = agnes(Wine2, method = "ward")   
pltree(hc, cex = 0.6, hang = -1, main = "Agglomerative Dendrogram")



hc2 = diana(Wine2)  
pltree(hc2, cex = 0.6, hang = -1, main = "Divisive Dendogram")

