## Module 6 Clustering Assignment

### Madison West

Load in libraries

options(tidyverse.quiet=TRUE)  
library(tidyverse)  
library(cluster) #algorithms for clustering  
library(factoextra) #visualization

## Warning: package 'factoextra' was built under R version 3.6.2

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

library(dendextend) #viewing clustering dendograms

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

##   
## ---------------------  
## 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

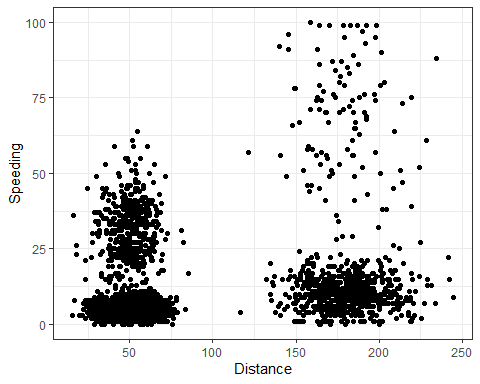
Before beginning the assignment tasks, you should read-in the data for the assignment into a data frame called “trucks”. In this dataset, Driver\_ID is a unique identifer for each delivery driver, Distance is the average mileage driven by each driver in a day, and Speeding is the percentage of the driver’s time in which he is driving at least 5 miles per hour over the speed limit.

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

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

Task 1: Plot the relationship between Distance and Speeding. Describe this relationship. Does there appear to be any natural clustering of drivers?

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



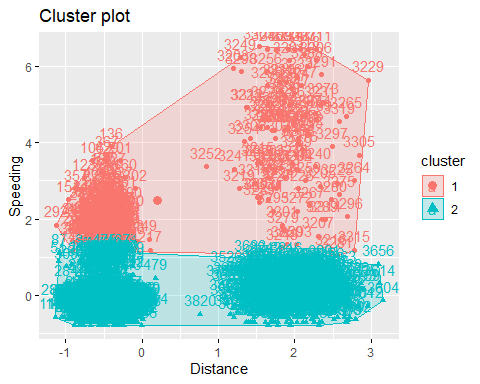
**Based on the plot, it appears that there could be four clusters that roughly correspond to the 4 quadrants (bottom left, bottom right, top-ish left, top right). There could also be 2 clusters on the left and right, or on the top and bottom.**

Task 2: Create a new data frame (called trucks2) that excludes the Driver\_ID variable and includes scaled versions of the Distance and Speeding variables. NOTE: Wrap the scale(trucks2) command in an as.data.frame command to ensure that the resulting object is a data frame. By default, scale converts data frames to lists

trucks2 <- as.data.frame(scale(trucks[,-1]))

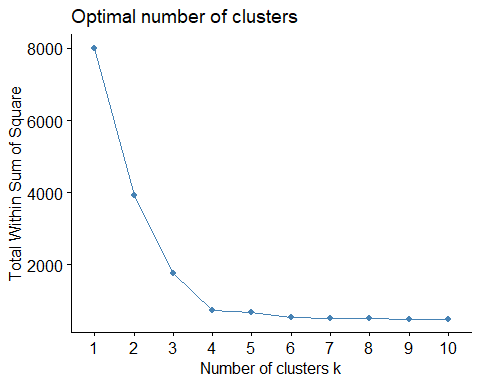
Task 3 Use k-Means clustering with two clusters (k=2) to cluster the trucks2 data frame. Use a random number seed of 1234. Visualize the clusters using the fviz\_cluster function. Comment on the clusters.

set.seed(1234)  
clusters1 <- kmeans(trucks2, 2)  
fviz\_cluster(clusters1, trucks2)

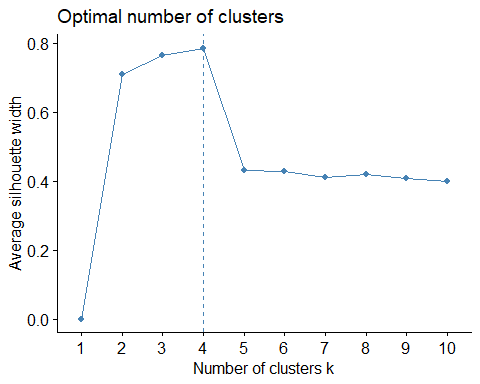


Task 4: Use the two methods from the k-Means lecture to identify the optimal number of clusters. Use a random number seed of 123 for these methods. Is there consensus between these two methods as the optimal number of clusters?

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



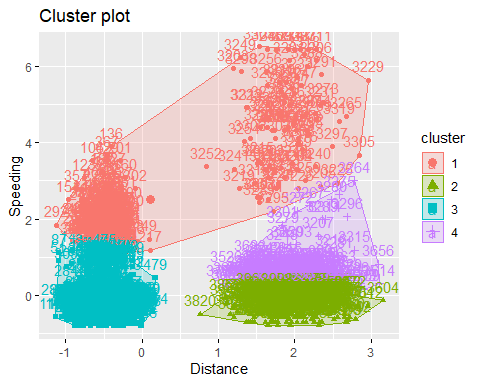
fviz\_nbclust(trucks2, kmeans, method = "silhouette")



**It does appear that these methods are consistent in determining the number of k clusters, which is 4. The bend of the curve appears to be most prominent at k=4 in "silhouette.**

Task 5: Use the optimal number of clusters that you identified in Task 4 to create k-Means clusters. Use a random number seed of 1234. Use the fviz\_cluster function to visualize the clusters.

set.seed(1234)  
clusters2 <- kmeans(trucks2, 4)  
fviz\_cluster(clusters2, trucks2)



Task 6: In words, how would you characterize the clusters you created in Task 5?

**The clusters in task 5 can be characterized as the low speed-short distance cluster, the low speed-long distance cluster, the mid speed-long distance cluster, and the high speed cluster.**

Before starting Task 7, read in the “wineprice.csv” file into a data frame called wine. This is a small dataset containing wine characteristics and the price of wine at auction. WinterRain refers to the amount of rain received in winter AGST refers to the average growing season temperature HarvestRain refers to the amount of rain received in the harvest season Age refers to the age of the wine when sold at auction FrancePop refers to the population of France at the time the wine was sold at auction.

Create a new data frame called wine2 that removes the Year and FrancePop variables and scales the other variables.

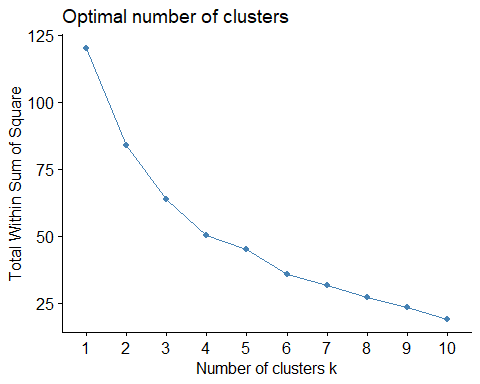
wine <- read\_csv("wineprice.csv")

## 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()  
## )

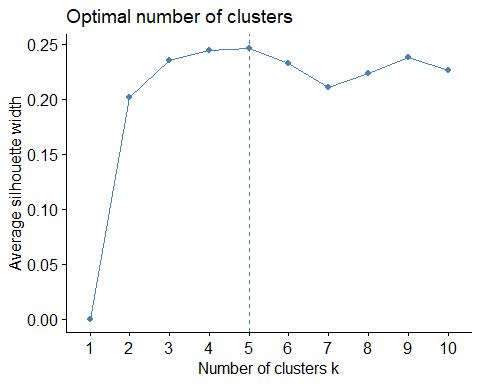
wine <- wine[,-1]  
wine2 <- as.data.frame(scale(wine[,-6]))

Task 7: Use the two methods from Task 4 to determine the optimal number of k-Means clusters for this data. Use a random number seed of 123. Is there consensus between these two methods as the optimal number of clusters?

set.seed(123)  
fviz\_nbclust(wine2, kmeans, method = "wss")



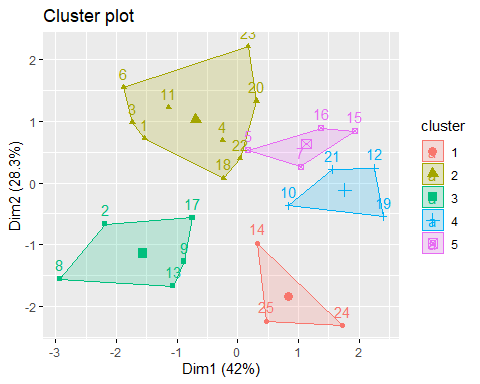
fviz\_nbclust(wine2, kmeans, method = "silhouette")



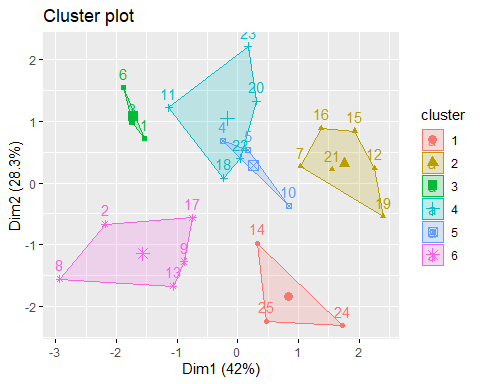
**It appears that the wss method has an optimal k avlue of 5, while silhouette could be either 5 or 6.**

Task 8: Use the optimal number of clusters that you identified in Task 4 to create k-Means clusters. Use a random number seed of 1234. Use the fviz\_cluster function to visualize the clusters.

set.seed(1234)  
clusters3 <- kmeans(wine2, 5)  
fviz\_cluster(clusters3, wine2)



clusters4 <- kmeans(wine2, 6)  
fviz\_cluster(clusters4, wine2)



**Based on the visual of both the 5 clusters and the 6 clusters, it appears that 5 clusters is likely the best choice.**

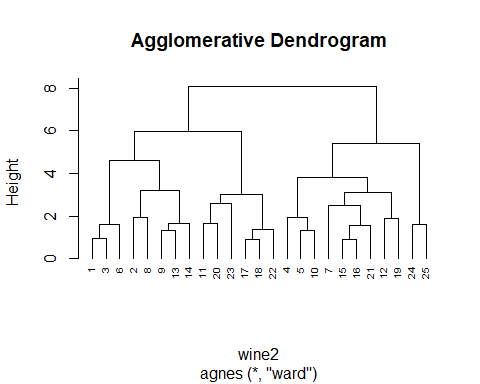
Task 9: Use agglomerative clustering to develop a dendogram for the scaled wine data. Follow the same process from the lecture where we used a custom function to identify the distance metric that maximizes the “agglomerative coefficient”. Plot the dendogram.

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

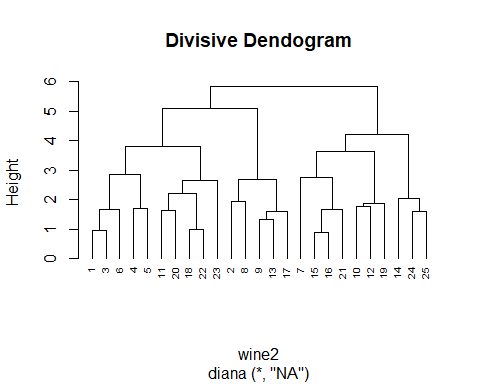
The distance metric that maximizes the agglomerative coefficient is ward (being the highest).

hc = agnes(wine2, method = "ward")  
pltree(hc, cex = 0.6, hang = -1, main = "Agglomerative Dendrogram")

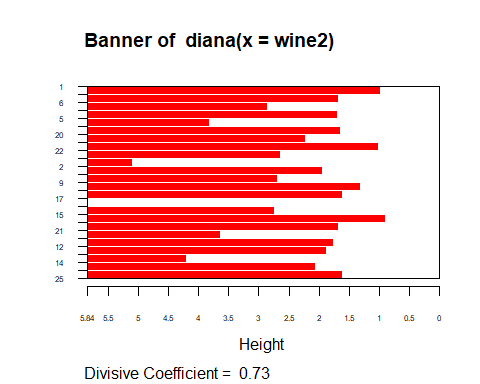


Task 10: Repeat Task 9, but with divisive clustering

hc2 = diana(wine2)  
pltree(hc2, cex = 0.6, hang = -1, main = "Divisive Dendogram")



plot(hc2, cex.axis= 0.5)



rect.hclust(hc2, k = 5, border = 2:6) #border selects colors for the boxes

