## Module 6 – Clustering Assignment

# Lea Shipley

library(tidyverse)

## -- Attaching packages ------------------------------------------ tidyverse 1.2.1 --

## v ggplot2 3.2.1 v purrr 0.3.3  
## v tibble 2.1.3 v dplyr 0.8.3  
## v tidyr 1.0.0 v stringr 1.4.0  
## v readr 1.3.1 v forcats 0.4.0

## -- Conflicts --------------------------------------------- tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(cluster)  
library(factoextra)

## 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)

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

##   
## ---------------------  
## Welcome to dendextend version 1.13.3  
## 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'

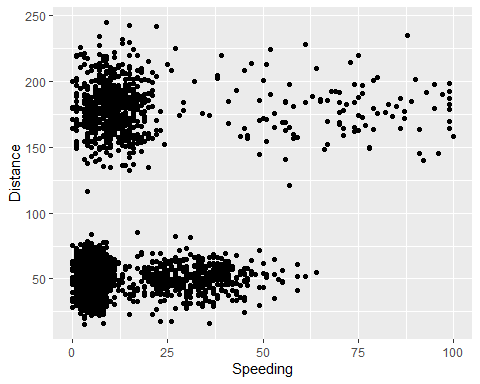
## The following object is masked from 'package:stats':  
##   
## cutree

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

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

**Task 1**

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



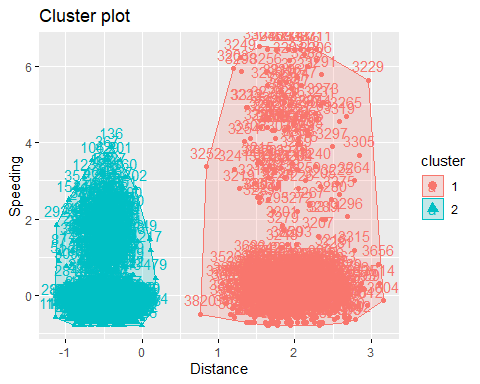
There does appear to be natural clustering of drivers.

**Task 2**

trucks2 <- trucks %>% select(-Driver\_ID) %>%  
as.data.frame(scale(trucks2))

**Task 3**

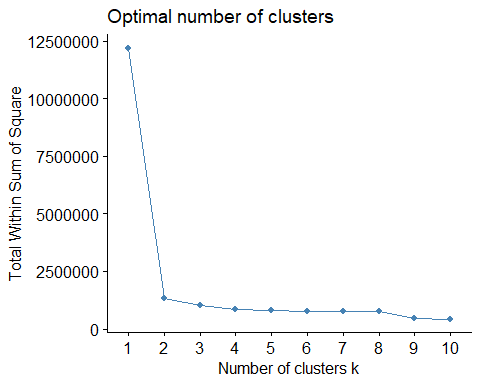
set.seed(1234)  
clusters1 <- kmeans(trucks2, 2)  
  
fviz\_cluster(clusters1, trucks2)



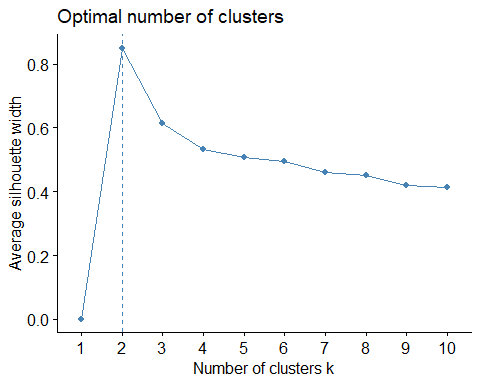
The two clusters are clearly defined with no overlap.

**Task 4**

set.seed(123)  
fviz\_nbclust(trucks2, kmeans, method = "wss") #minimize within-cluster variation



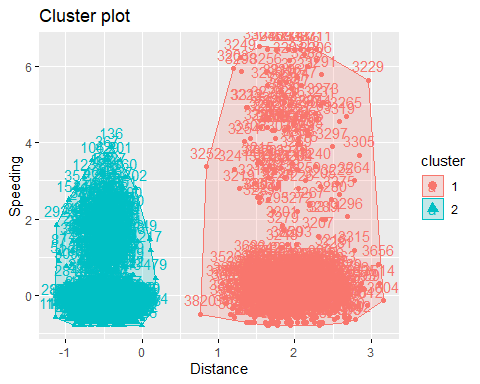
set.seed(123)  
fviz\_nbclust(trucks2, kmeans, method = "silhouette") #maximize how well points sit in their clusters



There is consensus between the two methods on the optimal number (2) of clusters.

**Task 5**

set.seed(1234)  
clusters2 <- kmeans(trucks2, 2)  
   
fviz\_cluster(clusters2, trucks2)



**Task 6**

The two clusters are clearly defined with no overlap. Cluster 1 drivers are spending more time speeding versus the Cluster 2 drivers who are not driving as many miles.

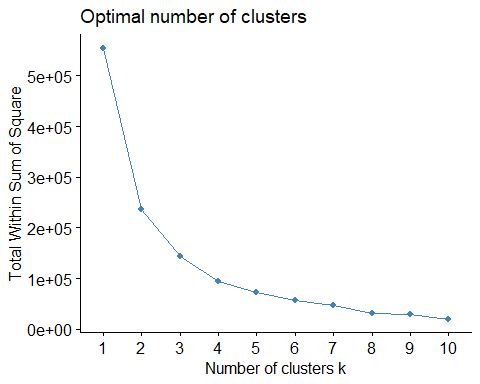
wine <- read\_csv("wineprice.csv")

## 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()  
## )

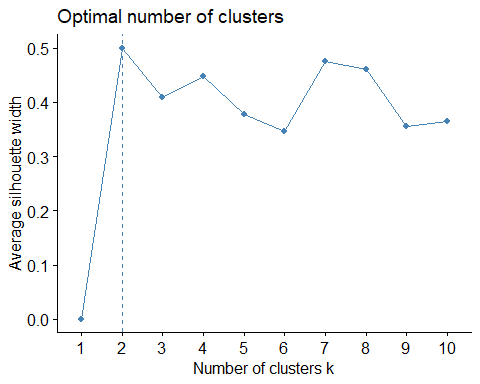
wine2 <- wine %>% select(-c(Year, FrancePop)) %>%  
as.data.frame(scale(wine2))

**Task 7**

set.seed(123)  
fviz\_nbclust(wine2, kmeans, method = "wss") #minimize within-cluster variation



set.seed(123)  
fviz\_nbclust(wine2, kmeans, method = "silhouette") #maximize how well points sit in their clusters



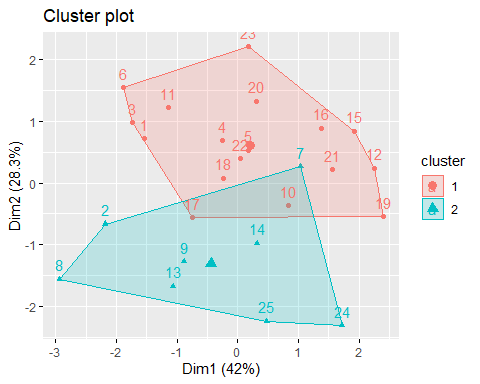
There is consensus between the two methods. The “within-cluster variation” method looks like 2 or possibly 3 could be the optimal number of clusters, while the silhouette method shows 2 as the optimal number.

**Task 8**

set.seed(1234)  
clusters3 <- kmeans(wine2, 2)  
clusters3

## K-means clustering with 2 clusters of sizes 17, 8  
##   
## Cluster means:  
## Price WinterRain AGST HarvestRain Age  
## 1 6.953647 531.4706 16.57550 171.2941 17.05882  
## 2 7.308575 762.1250 16.36874 100.2500 17.50000  
##   
## Clustering vector:  
## [1] 1 2 1 1 1 1 2 2 2 1 1 1 2 2 1 1 1 1 1 1 1 1 1 2 2  
##   
## Within cluster sum of squares by cluster:  
## [1] 168100.75 69322.51  
## (between\_SS / total\_SS = 57.2 %)  
##   
## Available components:  
##   
## [1] "cluster" "centers" "totss" "withinss"   
## [5] "tot.withinss" "betweenss" "size" "iter"   
## [9] "ifault"

fviz\_cluster(clusters3, wine2)

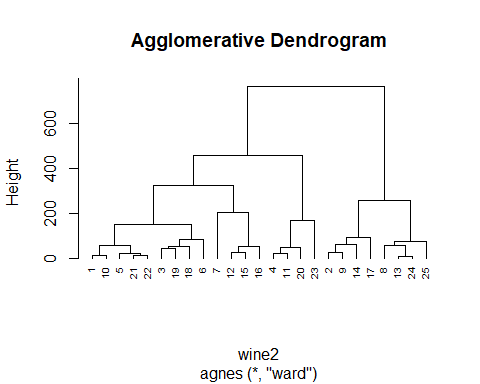


**Task 9**

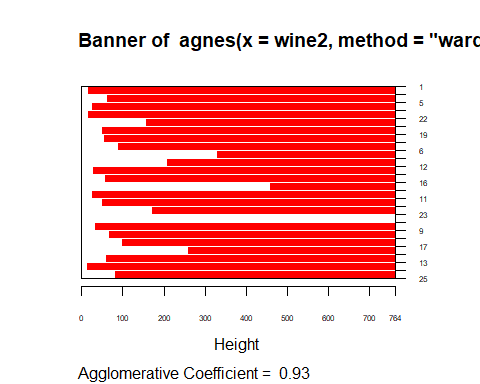
m = c( "average", "single", "complete", "ward")  
names(m) = c( "average", "single", "complete", "ward")  
  
ac = function(x) {  
 agnes(wine2, method = x)$ac  
}  
map\_dbl(m, ac)

## average single complete ward   
## 0.8125463 0.7384114 0.8973199 0.9330872

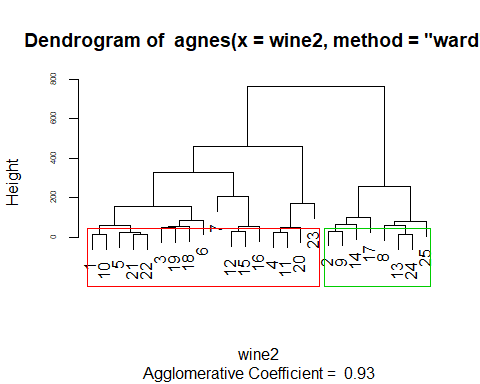
hc = agnes(wine2, method = "ward")  
pltree(hc, cex = 0.6, hang = -1, main = "Agglomerative Dendrogram")



plot(hc, cex.axis= 0.5)

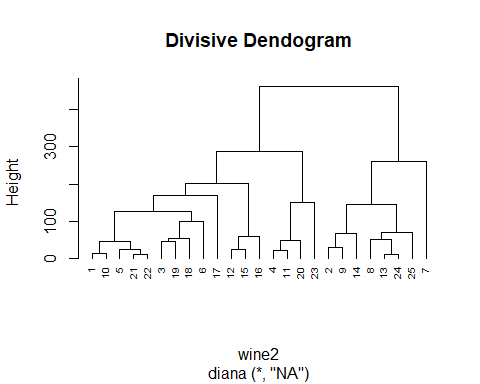


rect.hclust(hc, k = 2, border = 2:6) #border selects colors for the boxes

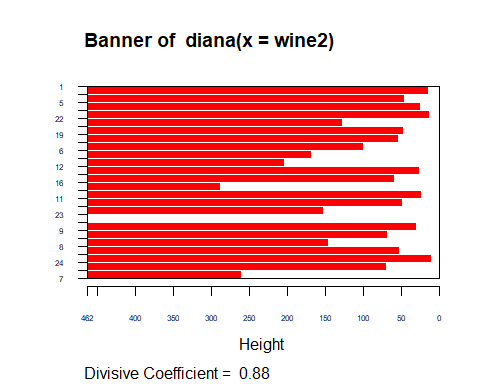


**Task 10**

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



plot(hc2, cex.axis= 0.5)



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

