### Module 6 Assignment 1

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(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(cluster)

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

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

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

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

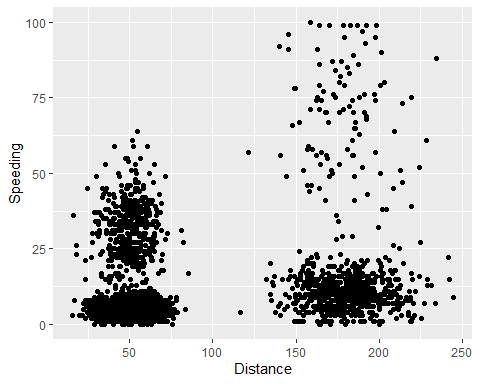
wine <- read\_csv("wineprice.csv")

## 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 <- select(wine,-Year, -FrancePop)  
  
wine2 <- as.data.frame(scale(wine2))

## Task 1

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



Right off the bat, we see that SOME drivers who travel longer distances will spend more/most of their time speeding that anyone who drives the shorter routes. There definitely seems to be a natural clustering of drivers.From looking at these natural clusters, we can see that the conentration of drivers who go shorter distances speed a little less on average than the concentration of drivers who travel longer distances.

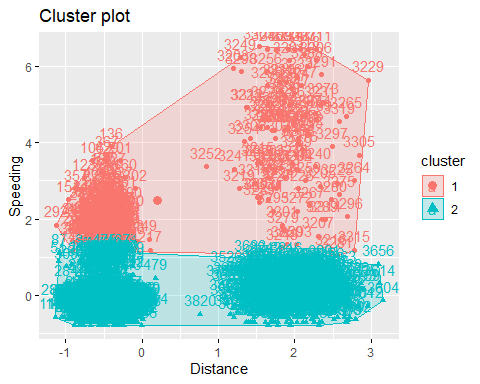
## Task 2

trucks2 <- select(trucks, Distance, Speeding)  
  
trucks2 <- as.data.frame(scale(trucks2))  
  
summary(trucks2)

## 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

## Task 3

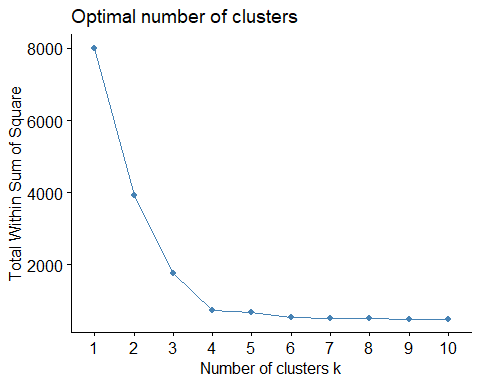
set.seed(1234)  
clusters1 <- kmeans(trucks2, 2)  
fviz\_cluster(clusters1, trucks2)



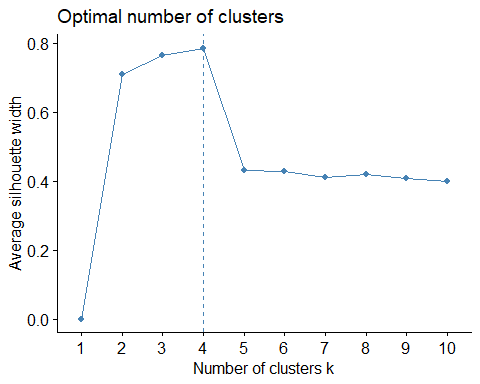
This cluster generated by the k-means clustering method splits the drivers by being speeders more often vs not as often. This kind of goes against the natural clustering - or at least how I was thinking about it, as the first graph makes the clustering look to be naturally more relevant to distance.

## Task 4

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



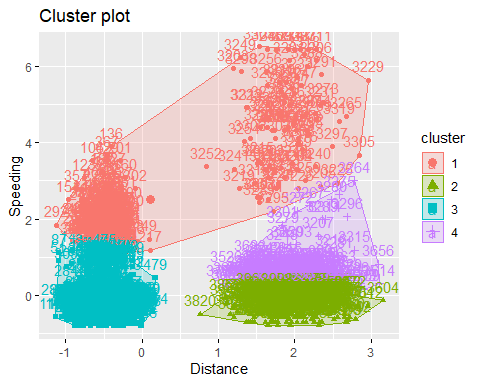
set.seed(123)  
fviz\_nbclust(trucks2, kmeans, method = "silhouette")



Both methods look to suggest 4 to be the optimal number of clusters.

## Task 5

set.seed(1234)  
clusters1 <- kmeans(trucks2, 4)  
fviz\_cluster(clusters1, trucks2)

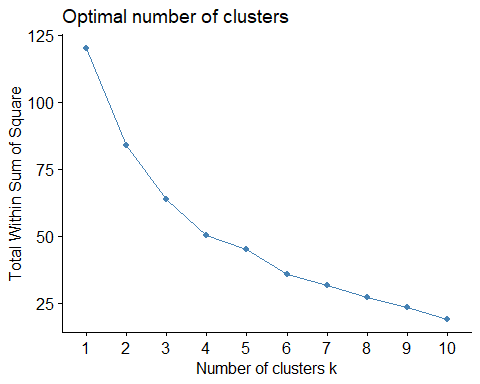


## Task 6

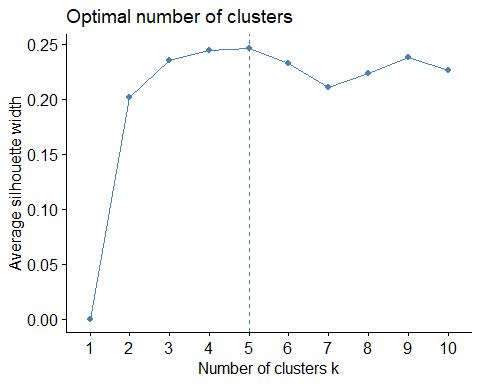
These clusters still relatively divide drivers who drive over the speed limit more often in a single cluster, and then divides those who speed less often into more seperate clusters.

## Task 7

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



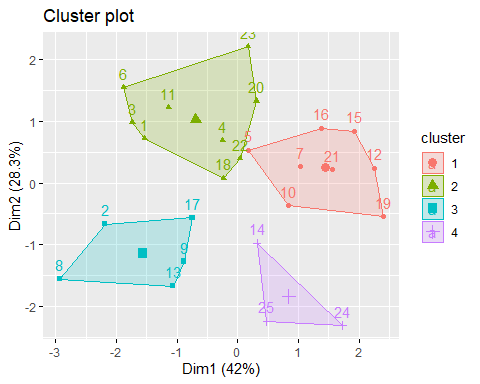
set.seed(123)  
fviz\_nbclust(wine2, kmeans, method = "silhouette")



Both methods suggest that the optimal number of clusters is 4-5. I am going to go with 4 for the remaining tasks.

## Task 8

set.seed(1234)  
clusters1 <- kmeans(wine2, 4)  
fviz\_cluster(clusters1, wine2)

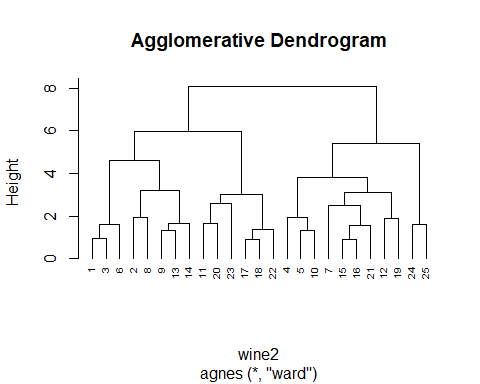


## Task 9

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   
## 0.5666719 0.2920143 0.7196616 0.8112139

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



## Task 10

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

