# Clustering

## Brian Fortier

## Module 6 Assignment 1

Libraries:

#install.packages("tidyverse")  
#install.packages("cluster")  
#install.packages("factoextra")  
#install.packages("dendextend")  
library(tidyverse)

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

## v ggplot2 3.1.0 v purrr 0.2.5  
## v tibble 1.4.2 v dplyr 0.7.7  
## v tidyr 0.8.2 v stringr 1.3.1  
## v readr 1.1.1 v forcats 0.3.0

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

library(cluster)  
library(factoextra)

## Welcome! Related Books: `Practical Guide To Cluster Analysis in R` at https://goo.gl/13EFCZ

library(dendextend)

##   
## ---------------------  
## Welcome to dendextend version 1.9.0  
## 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

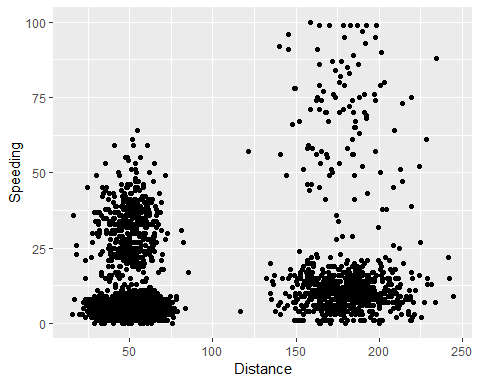
Trucks:

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

Visualization:

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



We do see some natural clustering of this data before scaling the two variables. We see two main clusters between 25-75 and 150-200 in the distance variable with a slight higher variation in the speeding variable. Potentially two more clusters would be forming at the same distance markers but at higher speed values.

Trucks2 Scaling:

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

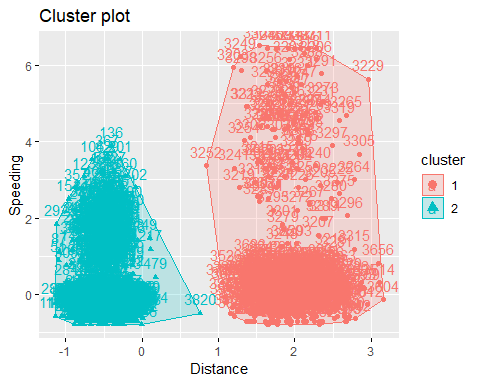
## Distance Speeding   
## Min. :-1.1319 Min. :-0.7821   
## 1st Qu.:-0.5759 1st Qu.:-0.4903   
## Median :-0.4248 Median :-0.3444   
## Mean : 0.0000 Mean : 0.0000   
## 3rd Qu.:-0.1947 3rd Qu.:-0.1255   
## Max. : 3.1560 Max. : 6.5127

Clusters:

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

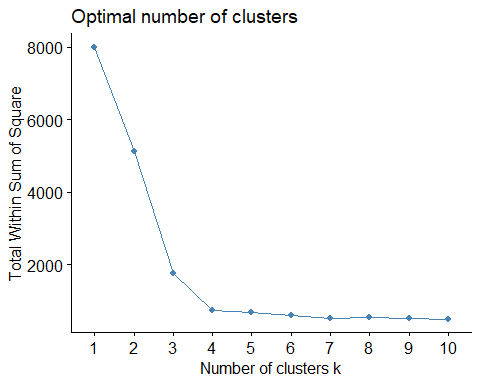
Visualize Clusters:

fviz\_cluster(clusters1, trucks2)



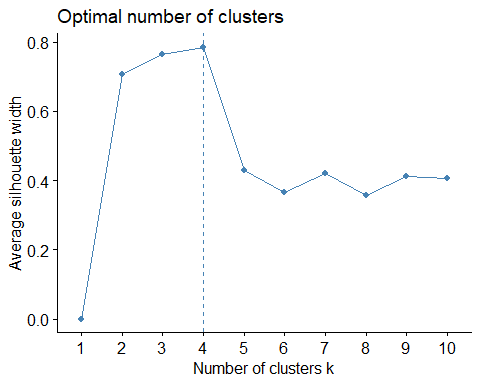
WSS Cluster Method:

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



Silhouette Cluster Method:

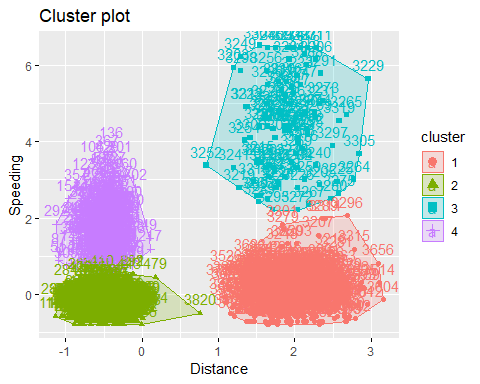
set.seed(123)  
fviz\_nbclust(trucks2, kmeans, method = "silhouette")



There is a consensus that 4 would be the ideal number of clusters for our data. Both the WSS and Silhouette method returned the same amount.

k-Means Cluster with 4 Clusters:

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



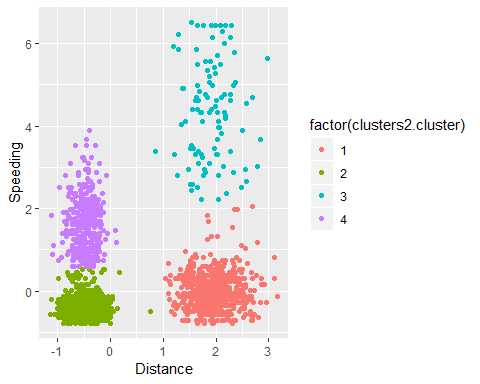
Attaching Clusters to Dataset:

cluster = data.frame(clusters2$cluster)  
trucks2 = bind\_cols(trucks2,cluster)  
str(trucks2)

## 'data.frame': 4000 obs. of 3 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 ...  
## $ clusters2.cluster: int 4 4 4 4 4 2 4 2 4 4 ...

Visualization:

ggplot(trucks2, aes(Distance, Speeding, color = factor(clusters2.cluster)))+geom\_point()



Here we can clearly see our 4 clusters color coded. We have some similarites between them like the green and red clusters both hover around the 0 speeding line while the other two clusters are more spread out above it. Similarly, the green and purple clusters stay near the -0.5 distance marker while the red and blue clusters are more centered over the 2 distance marker.

WinePrice:

wine <- read.csv("wineprice.csv")  
summary(wine)

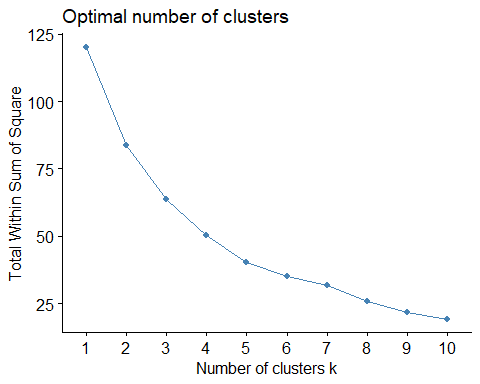
## Year Price WinterRain AGST   
## Min. :1952 Min. :6.205 Min. :376.0 Min. :14.98   
## 1st Qu.:1960 1st Qu.:6.519 1st Qu.:536.0 1st Qu.:16.20   
## Median :1966 Median :7.121 Median :600.0 Median :16.53   
## Mean :1966 Mean :7.067 Mean :605.3 Mean :16.51   
## 3rd Qu.:1972 3rd Qu.:7.495 3rd Qu.:697.0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

Wine Scaling:

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

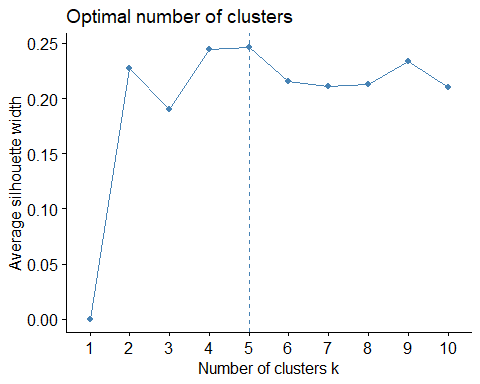
WSS Cluster Method:

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



Silhouette Cluster Method:

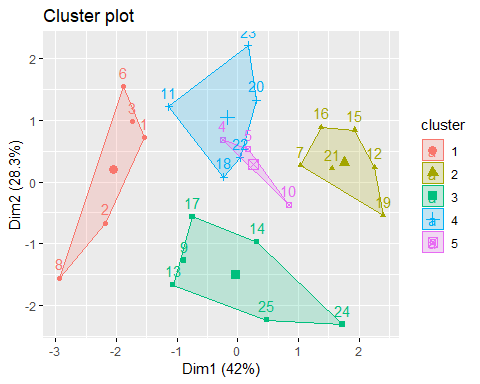
set.seed(123)  
fviz\_nbclust(wine2, kmeans, method = "silhouette")



Seemingly both methods returned the same number of optimal clusters which would be 5.

k-Means Cluster with 5 Clusters:

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



Choosing a Method:

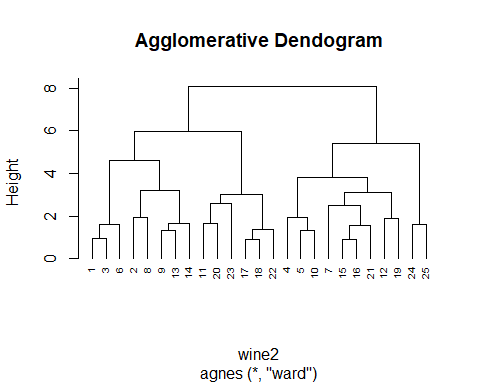
m = c("average", "single", "complete", "ward")  
names(m) = c("average", "single", "complete", "ward")  
  
ac = function(x) {  
 agnes(wine2, method = x)$ac  
}  
map\_dbl(m,ac)

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

Ward’s is the highest so we will use this to develp clusters.

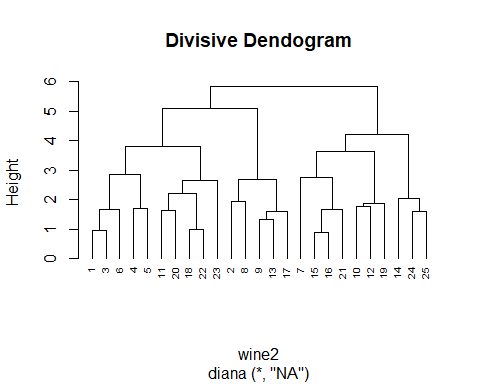
Agglomerative Dendogram:

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



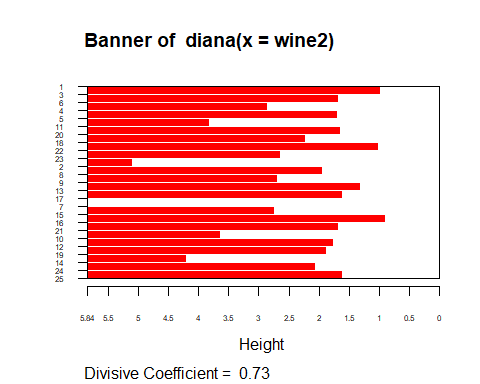
Divisive Clustering:

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

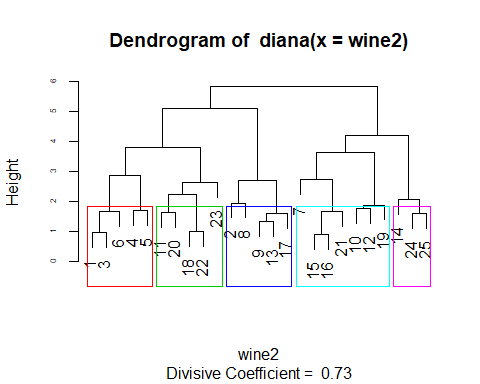


Grouping the Dendogram:

plot(hc2, cex.axis=0.5)



rect.hclust(hc2, k=5, border=2:6)



Cutting the tree into Clusters:

d = dist(wine2, method = "euclidean")  
hc3 = hclust(d, method = "ward.D")  
sub\_group = cutree(hc3, k=5)  
head(sub\_group)

## [1] 1 2 1 3 3 1