MIS-64060 Assignment 5

Antonio Garlisi

2022-11-03

universal.df <- read.csv("C:\\Users\\Andrew Garlisi\\Documents\\Cereals.csv" , header=TRUE, stringsAsFactors=FALSE)  
install.packages("dplyr", repos ="http://cran.us.r-project.org")

## Installing package into 'C:/Users/Andrew Garlisi/AppData/Local/R/win-library/4.2'  
## (as 'lib' is unspecified)

## package 'dplyr' successfully unpacked and MD5 sums checked

## Warning: cannot remove prior installation of package 'dplyr'

## Warning in file.copy(savedcopy, lib, recursive = TRUE):  
## problem copying C:\Users\Andrew Garlisi\AppData\Local\R\win-  
## library\4.2\00LOCK\dplyr\libs\x64\dplyr.dll to C:\Users\Andrew  
## Garlisi\AppData\Local\R\win-library\4.2\dplyr\libs\x64\dplyr.dll: Permission  
## denied

## Warning: restored 'dplyr'

##   
## The downloaded binary packages are in  
## C:\Users\Andrew Garlisi\AppData\Local\Temp\RtmpQbEIs1\downloaded\_packages

library(dplyr)

## Warning: package 'dplyr' was built under R version 4.2.2

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

install.packages("stats", repos ="http://cran.us.r-project.org")

## Warning: package 'stats' is in use and will not be installed

library(stats)  
install.packages("cluster", repos ="http://cran.us.r-project.org")

## Installing package into 'C:/Users/Andrew Garlisi/AppData/Local/R/win-library/4.2'  
## (as 'lib' is unspecified)

## package 'cluster' successfully unpacked and MD5 sums checked

## Warning: cannot remove prior installation of package 'cluster'

## Warning in file.copy(savedcopy, lib, recursive = TRUE):  
## problem copying C:\Users\Andrew Garlisi\AppData\Local\R\win-  
## library\4.2\00LOCK\cluster\libs\x64\cluster.dll to C:\Users\Andrew  
## Garlisi\AppData\Local\R\win-library\4.2\cluster\libs\x64\cluster.dll: Permission  
## denied

## Warning: restored 'cluster'

##   
## The downloaded binary packages are in  
## C:\Users\Andrew Garlisi\AppData\Local\Temp\RtmpQbEIs1\downloaded\_packages

library(cluster)

## Warning: package 'cluster' was built under R version 4.2.2

install.packages("factoextra", repos ="http://cran.us.r-project.org")

## Installing package into 'C:/Users/Andrew Garlisi/AppData/Local/R/win-library/4.2'  
## (as 'lib' is unspecified)

## package 'factoextra' successfully unpacked and MD5 sums checked  
##   
## The downloaded binary packages are in  
## C:\Users\Andrew Garlisi\AppData\Local\Temp\RtmpQbEIs1\downloaded\_packages

library(factoextra)

## Warning: package 'factoextra' was built under R version 4.2.2

## Loading required package: ggplot2

## Warning: package 'ggplot2' was built under R version 4.2.1

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

universal.df <- na.omit(universal.df)

numeric.df <- subset(universal.df, select = -c(2:3))  
rownames(numeric.df) <- numeric.df$name  
numeric.df$name <- NULL  
numeric.df <- scale(numeric.df)

d <- dist(numeric.df, method = "euclidean")  
hc\_single <- agnes(numeric.df, method = "single")  
hc\_complete <- agnes(numeric.df, method = "complete")  
hc\_average <- agnes(numeric.df, method = "average")  
hc\_ward <- agnes(numeric.df, method = "ward")  
  
print(hc\_single$ac)

## [1] 0.6067859

print(hc\_complete$ac)

## [1] 0.8353712

print(hc\_average$ac)

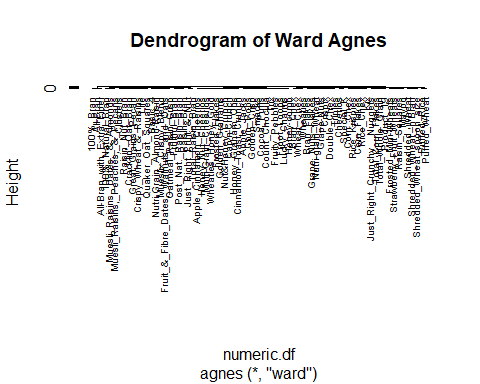
## [1] 0.7766075

print(hc\_ward$ac)

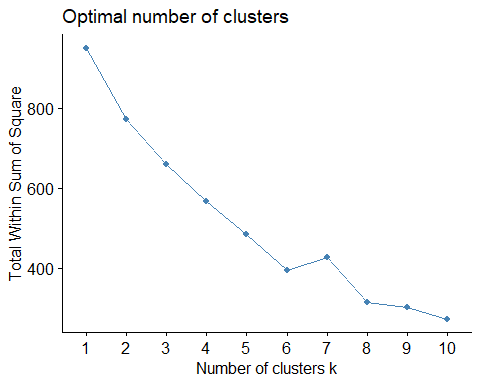
## [1] 0.9046042

#Based on the highest Agglomerative coefficient, it appears that the Ward method is the best

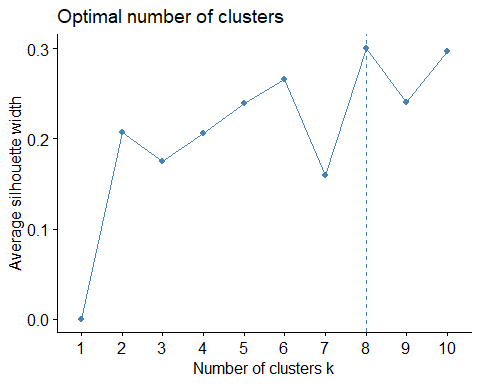
pltree(hc\_ward, cex = 0.6, hang = -1, main = "Dendrogram of Ward Agnes")



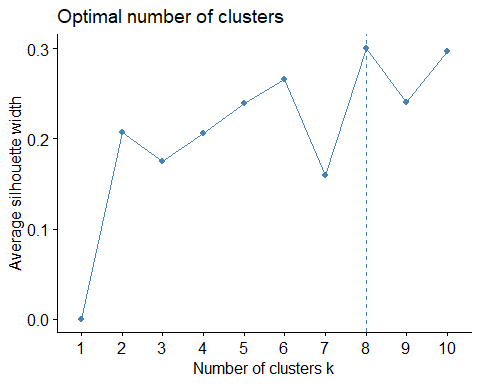
fviz\_nbclust(numeric.df, kmeans, method="wss")



fviz\_nbclust(numeric.df, kmeans, method="silhouette")

 #Elbow method somewhat inconclusive with possible elbow between 6-8

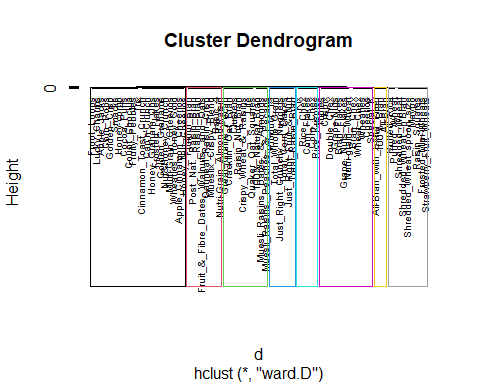
fviz\_nbclust(numeric.df, kmeans, method="silhouette")

 #The Optimal number of clusters was determined to be 8 based on Silhouette

hc\_ward2 <- hclust(d, method = "ward")

## The "ward" method has been renamed to "ward.D"; note new "ward.D2"

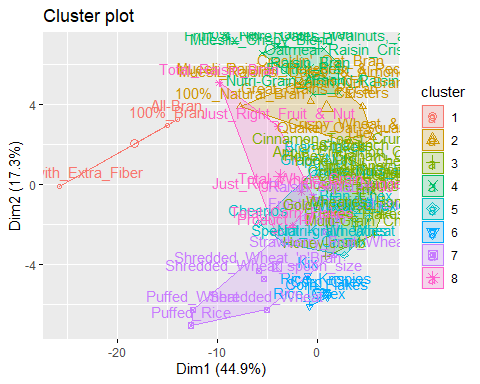
plot(hc\_ward2, cex = 0.6)  
rect.hclust(hc\_ward2, k = 8, border = 1:8)



clustergroups <- cutree(hc\_ward2, k=8)  
table(clustergroups)

## clustergroups  
## 1 2 3 4 5 6 7 8   
## 3 10 21 8 12 5 9 6

Temp <- cbind(as.data.frame(cbind(numeric.df,clustergroups)))  
fviz\_cluster(list(data=d, cluster=clustergroups))



HealthyCereal <- numeric.df  
HealthyCereal <- as.data.frame(cbind(HealthyCereal,clustergroups))  
mean(HealthyCereal[HealthyCereal$clustergroups==1,"rating"])

## [1] 2.242648

mean(HealthyCereal[HealthyCereal$clustergroups==2,"rating"])

## [1] -0.1501906

mean(HealthyCereal[HealthyCereal$clustergroups==3,"rating"])

## [1] -0.9636465

mean(HealthyCereal[HealthyCereal$clustergroups==4,"rating"])

## [1] -0.3691725

mean(HealthyCereal[HealthyCereal$clustergroups==5,"rating"])

## [1] 0.5621701

mean(HealthyCereal[HealthyCereal$clustergroups==6,"rating"])

## [1] -0.03919684

mean(HealthyCereal[HealthyCereal$clustergroups==7,"rating"])

## [1] 1.471215

mean(HealthyCereal[HealthyCereal$clustergroups==8,"rating"])

## [1] -0.3045121

#Based on the average health ratings from the clusters, Cluster 1 would be the healthiest option although it only has 3 cereals included. The next healthiest group would be Cluster 7 so the school could consider adding those 9 cereals if they wanted more variety while still including mainly cereals with above average health ratings. It makes sense to normalize the data so that the various nutritional factors don’t skew the clustering based on their relative scale (ex: calories has a higher range of values than the other nutritional factors).

set.seed(1)  
   
PartA.index <- sample(row.names(numeric.df), 0.6\*dim(numeric.df)[1])   
  
PartB.index <- setdiff(row.names(numeric.df), PartA.index)   
  
PartA.df <- numeric.df[PartA.index]   
  
PartB.df <- numeric.df[PartB.index]

#I didn’t find a method of evaluating clusters stability using partitions that made sense to me in the modules or textbook so I partitioned the data into 60% and 40% partitions but didn’t know how to proceed from there.