Final Exam - Kerrie Mars

Kerrie Mars

2022-12-08

setwd("/Users/Kyle Mars/Desktop/Kerrie/Machine Learning")  
  
library("tidyverse") #data manipulation

## Warning: package 'tidyverse' was built under R version 4.1.3

## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --  
## v ggplot2 3.3.6 v purrr 0.3.4   
## v tibble 3.1.8 v dplyr 1.0.10  
## v tidyr 1.2.1 v stringr 1.4.1   
## v readr 2.1.3 v forcats 0.5.2

## Warning: package 'ggplot2' was built under R version 4.1.3

## Warning: package 'tibble' was built under R version 4.1.3

## Warning: package 'tidyr' was built under R version 4.1.3

## Warning: package 'readr' was built under R version 4.1.3

## Warning: package 'dplyr' was built under R version 4.1.3

## Warning: package 'stringr' was built under R version 4.1.3

## Warning: package 'forcats' was built under R version 4.1.3

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

library("factoextra") #clustering algorithms and visualization

## Warning: package 'factoextra' was built under R version 4.1.3

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

library("ISLR")

## Warning: package 'ISLR' was built under R version 4.1.3

library("flexclust")

## Warning: package 'flexclust' was built under R version 4.1.3

## Loading required package: grid  
## Loading required package: lattice  
## Loading required package: modeltools

## Warning: package 'modeltools' was built under R version 4.1.1

## Loading required package: stats4

library("scales")

## Warning: package 'scales' was built under R version 4.1.3

##   
## Attaching package: 'scales'  
##   
## The following object is masked from 'package:purrr':  
##   
## discard  
##   
## The following object is masked from 'package:readr':  
##   
## col\_factor

library("ggplot2")  
library("stats")  
library("cluster") # clustering algorithms

## Warning: package 'cluster' was built under R version 4.1.3

library("fpc")

## Warning: package 'fpc' was built under R version 4.1.3

#clear existing data in Environment  
rm(list=ls())  
  
set.seed(1)  
  
#open dataset  
coffee <- read.csv("starbucks\_drinks.csv", header=TRUE)  
head(coffee)

## Beverage\_category Beverage Beverage\_prep Calories  
## 1 Coffee Brewed Coffee Short 3  
## 2 Coffee Brewed Coffee Tall 4  
## 3 Coffee Brewed Coffee Grande 5  
## 4 Coffee Brewed Coffee Venti 5  
## 5 Classic Espresso Drinks CaffÃ¨ Latte Short Nonfat Milk 70  
## 6 Classic Espresso Drinks CaffÃ¨ Latte 2% Milk 100  
## Total.Fat..g. Trans.Fat..g. Saturated.Fat..g. Sodium..mg.  
## 1 0.1 0.0 0.0 0  
## 2 0.1 0.0 0.0 0  
## 3 0.1 0.0 0.0 0  
## 4 0.1 0.0 0.0 0  
## 5 0.1 0.1 0.0 5  
## 6 3.5 2.0 0.1 15  
## Total.Carbohydrates..g. Cholesterol..mg. Dietary.Fibre..g. Sugars..g.  
## 1 5 0 0 0  
## 2 10 0 0 0  
## 3 10 0 0 0  
## 4 10 0 0 0  
## 5 75 10 0 9  
## 6 85 10 0 9  
## Protein..g. Vitamin.A....DV. Vitamin.C....DV. Calcium....DV. Iron....DV.  
## 1 0.3 0% 0% 0% 0%  
## 2 0.5 0% 0% 0% 0%  
## 3 1.0 0% 0% 0% 0%  
## 4 1.0 0% 0% 2% 0%  
## 5 6.0 10% 0% 20% 0%  
## 6 6.0 10% 0% 20% 0%  
## Caffeine..mg.  
## 1 175  
## 2 260  
## 3 330  
## 4 410  
## 5 75  
## 6 75

coffee1 <- coffee  
coffee1[coffee1 == ""] <- NA  
  
#remove non-numerical columns from the dataset  
coffee1 <- coffee1[, -c(1, 2, 3)]  
#examine the dataset  
head(coffee1)

## Calories Total.Fat..g. Trans.Fat..g. Saturated.Fat..g. Sodium..mg.  
## 1 3 0.1 0.0 0.0 0  
## 2 4 0.1 0.0 0.0 0  
## 3 5 0.1 0.0 0.0 0  
## 4 5 0.1 0.0 0.0 0  
## 5 70 0.1 0.1 0.0 5  
## 6 100 3.5 2.0 0.1 15  
## Total.Carbohydrates..g. Cholesterol..mg. Dietary.Fibre..g. Sugars..g.  
## 1 5 0 0 0  
## 2 10 0 0 0  
## 3 10 0 0 0  
## 4 10 0 0 0  
## 5 75 10 0 9  
## 6 85 10 0 9  
## Protein..g. Vitamin.A....DV. Vitamin.C....DV. Calcium....DV. Iron....DV.  
## 1 0.3 0% 0% 0% 0%  
## 2 0.5 0% 0% 0% 0%  
## 3 1.0 0% 0% 0% 0%  
## 4 1.0 0% 0% 2% 0%  
## 5 6.0 10% 0% 20% 0%  
## 6 6.0 10% 0% 20% 0%  
## Caffeine..mg.  
## 1 175  
## 2 260  
## 3 330  
## 4 410  
## 5 75  
## 6 75

tail(coffee1)

## Calories Total.Fat..g. Trans.Fat..g. Saturated.Fat..g. Sodium..mg.  
## 237 350 6 3.0 0.2 15  
## 238 320 3 2 0.4 0.0 0  
## 239 170 0.1 0.1 0.0 0  
## 240 200 3.5 2.0 0.1 10  
## 241 180 1.5 0.2 0.0 0  
## 242 240 0.1 0.1 0.0 5  
## Total.Carbohydrates..g. Cholesterol..mg. Dietary.Fibre..g. Sugars..g.  
## 237 260 70 0 68  
## 238 250 67 1 64  
## 239 160 39 0 38  
## 240 160 39 0 38  
## 241 160 37 1 35  
## 242 230 56 0 55  
## Protein..g. Vitamin.A....DV. Vitamin.C....DV. Calcium....DV. Iron....DV.  
## 237 6 8% 8% 20% 4%  
## 238 5 6% 8% 20% 10%  
## 239 4 6% 0% 10% 0%  
## 240 3 6% 0% 10% 0%  
## 241 3 4% 0% 10% 6%  
## 242 5 8% 0% 15% 0%  
## Caffeine..mg.  
## 237 0  
## 238 0  
## 239 0  
## 240 0  
## 241 0  
## 242 0

summary(coffee1)

## Calories Total.Fat..g. Trans.Fat..g. Saturated.Fat..g.  
## Min. : 0.0 Length:242 Min. :0.000 Min. :0.0000   
## 1st Qu.:120.0 Class :character 1st Qu.:0.100 1st Qu.:0.0000   
## Median :185.0 Mode :character Median :0.500 Median :0.0000   
## Mean :193.9 Mean :1.307 Mean :0.0376   
## 3rd Qu.:260.0 3rd Qu.:2.000 3rd Qu.:0.1000   
## Max. :510.0 Max. :9.000 Max. :0.3000   
## Sodium..mg. Total.Carbohydrates..g. Cholesterol..mg. Dietary.Fibre..g.  
## Min. : 0.000 Min. : 0.0 Min. : 0.00 Min. :0.0000   
## 1st Qu.: 0.000 1st Qu.: 70.0 1st Qu.:21.00 1st Qu.:0.0000   
## Median : 5.000 Median :125.0 Median :34.00 Median :0.0000   
## Mean : 6.364 Mean :128.9 Mean :35.99 Mean :0.8058   
## 3rd Qu.:10.000 3rd Qu.:170.0 3rd Qu.:50.75 3rd Qu.:1.0000   
## Max. :40.000 Max. :340.0 Max. :90.00 Max. :8.0000   
## Sugars..g. Protein..g. Vitamin.A....DV. Vitamin.C....DV.   
## Min. : 0.00 Min. : 0.000 Length:242 Length:242   
## 1st Qu.:18.00 1st Qu.: 3.000 Class :character Class :character   
## Median :32.00 Median : 6.000 Mode :character Mode :character   
## Mean :32.96 Mean : 6.979   
## 3rd Qu.:43.75 3rd Qu.:10.000   
## Max. :84.00 Max. :20.000   
## Calcium....DV. Iron....DV. Caffeine..mg.   
## Length:242 Length:242 Length:242   
## Class :character Class :character Class :character   
## Mode :character Mode :character Mode :character   
##   
##   
##

str(coffee1)

## 'data.frame': 242 obs. of 15 variables:  
## $ Calories : int 3 4 5 5 70 100 70 100 150 110 ...  
## $ Total.Fat..g. : chr "0.1" "0.1" "0.1" "0.1" ...  
## $ Trans.Fat..g. : num 0 0 0 0 0.1 2 0.4 0.2 3 0.5 ...  
## $ Saturated.Fat..g. : num 0 0 0 0 0 0.1 0 0 0.2 0 ...  
## $ Sodium..mg. : int 0 0 0 0 5 15 0 5 25 0 ...  
## $ Total.Carbohydrates..g.: int 5 10 10 10 75 85 65 120 135 105 ...  
## $ Cholesterol..mg. : int 0 0 0 0 10 10 6 15 15 10 ...  
## $ Dietary.Fibre..g. : int 0 0 0 0 0 0 1 0 0 1 ...  
## $ Sugars..g. : int 0 0 0 0 9 9 4 14 14 6 ...  
## $ Protein..g. : num 0.3 0.5 1 1 6 6 5 10 10 8 ...  
## $ Vitamin.A....DV. : chr "0%" "0%" "0%" "0%" ...  
## $ Vitamin.C....DV. : chr "0%" "0%" "0%" "0%" ...  
## $ Calcium....DV. : chr "0%" "0%" "0%" "2%" ...  
## $ Iron....DV. : chr "0%" "0%" "0%" "0%" ...  
## $ Caffeine..mg. : chr "175" "260" "330" "410" ...

#remove rows with varying caffeine content  
coffee1 <- subset(coffee1, Caffeine..mg.!="varies" & Caffeine..mg.!="Varies")  
  
#replace all blanks with 0  
coffee1 <- replace(coffee1, coffee1$Caffeine..mg.==" ", 0)  
coffee1[coffee1 == ' '] <- '0'  
  
#convert percentages to numeric  
coffee1$Total.Fat..g. = as.numeric(coffee1$Total.Fat..g.)

## Warning: NAs introduced by coercion

coffee1$Vitamin.A....DV. = as.numeric(sub("%", "", coffee1$Vitamin.A....DV.))  
coffee1$Vitamin.C....DV. = as.numeric(sub("%", "", coffee1$Vitamin.C....DV.))  
coffee1$Calcium....DV. = as.numeric(sub("%", "", coffee1$Calcium....DV.))  
coffee1$Iron....DV. = as.numeric(sub("%", "", coffee1$Iron....DV.))  
coffee1$Caffeine..mg. = as.numeric(coffee1$Caffeine..mg.)  
  
#remove NA (missing) values  
coffee1 <- na.omit(coffee1)  
  
#convert to row names  
rownames(coffee1) <- coffee1$name  
coffee1$name = NULL  
  
#scale/standardize the data  
coffee\_norm <- scale(coffee1)  
  
#re-examine the scaled data  
head(coffee\_norm)

## Calories Total.Fat..g. Trans.Fat..g. Saturated.Fat..g. Sodium..mg.  
## [1,] -1.9364116 -0.9889702 -0.8292715 -0.5393932 -0.7548729  
## [2,] -1.9266338 -0.9889702 -0.8292715 -0.5393932 -0.7548729  
## [3,] -1.9168560 -0.9889702 -0.8292715 -0.5393932 -0.7548729  
## [4,] -1.9168560 -0.9889702 -0.8292715 -0.5393932 -0.7548729  
## [5,] -1.2812978 -0.9889702 -0.7700185 -0.5393932 -0.1893667  
## [6,] -0.9879633 0.1415429 0.3557904 0.8279059 0.9416455  
## Total.Carbohydrates..g. Cholesterol..mg. Dietary.Fibre..g. Sugars..g.  
## [1,] -1.6394720 -1.779956 -0.5843691 -1.704415  
## [2,] -1.5773039 -1.779956 -0.5843691 -1.704415  
## [3,] -1.5773039 -1.779956 -0.5843691 -1.704415  
## [4,] -1.5773039 -1.779956 -0.5843691 -1.704415  
## [5,] -0.7691174 -1.300965 -0.5843691 -1.252454  
## [6,] -0.6447811 -1.300965 -0.5843691 -1.252454  
## Protein..g. Vitamin.A....DV. Vitamin.C....DV. Calcium....DV. Iron....DV.  
## [1,] -1.4596788 -1.2553296 -0.2480669 -1.4894358 -0.7447436  
## [2,] -1.4180037 -1.2553296 -0.2480669 -1.4894358 -0.7447436  
## [3,] -1.3138161 -1.2553296 -0.2480669 -1.4894358 -0.7447436  
## [4,] -1.3138161 -1.2553296 -0.2480669 -1.3520561 -0.7447436  
## [5,] -0.2719394 -0.0379726 -0.2480669 -0.1156384 -0.7447436  
## [6,] -0.2719394 -0.0379726 -0.2480669 -0.1156384 -0.7447436  
## Caffeine..mg.  
## [1,] 1.3170741  
## [2,] 2.6330828  
## [3,] 3.7168547  
## [4,] 4.9554512  
## [5,] -0.2311715  
## [6,] -0.2311715

tail(coffee\_norm)

## Calories Total.Fat..g. Trans.Fat..g. Saturated.Fat..g. Sodium..mg.  
## [213,] 1.06537847 -0.9557198 -0.7700185 -0.5393932 -0.1893667  
## [214,] 1.45649119 0.9728026 0.9483213 2.1952050 0.9416455  
## [215,] -0.30351602 -0.9889702 -0.7700185 -0.5393932 -0.7548729  
## [216,] -0.01018149 0.1415429 0.3557904 0.8279059 0.3761394  
## [217,] -0.20573785 -0.5234648 -0.7107654 -0.5393932 -0.7548729  
## [218,] 0.38093122 -0.9889702 -0.7700185 -0.5393932 -0.1893667  
## Total.Carbohydrates..g. Cholesterol..mg. Dietary.Fibre..g. Sugars..g.  
## [213,] 1.5311055 1.572978950 -0.5843691 1.76062261  
## [214,] 1.5311055 1.572978950 -0.5843691 1.71040467  
## [215,] 0.2877417 0.088107914 -0.5843691 0.20386642  
## [216,] 0.2877417 0.088107914 -0.5843691 0.20386642  
## [217,] 0.2877417 -0.007690217 0.1079813 0.05321259  
## [218,] 1.1580964 0.902392031 -0.5843691 1.05757143  
## Protein..g. Vitamin.A....DV. Vitamin.C....DV. Calcium....DV. Iron....DV.  
## [213,] -0.2719394 -0.0379726 0.2827658 -0.1156384 -0.3742836  
## [214,] -0.2719394 -0.2814440 0.2827658 -0.1156384 -0.3742836  
## [215,] -0.6886901 -0.5249154 -0.2480669 -0.8025371 -0.7447436  
## [216,] -0.8970654 -0.5249154 -0.2480669 -0.8025371 -0.7447436  
## [217,] -0.8970654 -0.7683868 -0.2480669 -0.8025371 -0.1890536  
## [218,] -0.4803147 -0.2814440 -0.2480669 -0.4590878 -0.7447436  
## Caffeine..mg.  
## [213,] -1.392356  
## [214,] -1.392356  
## [215,] -1.392356  
## [216,] -1.392356  
## [217,] -1.392356  
## [218,] -1.392356

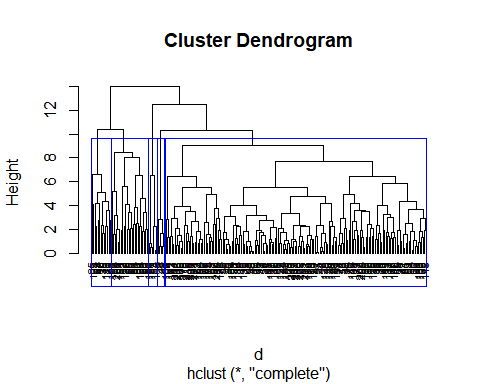
summary(coffee\_norm)

## Calories Total.Fat..g. Trans.Fat..g. Saturated.Fat..g.  
## Min. :-1.9364 Min. :-1.0222 Min. :-0.8293 Min. :-0.5394   
## 1st Qu.:-0.7680 1st Qu.:-0.9557 1st Qu.:-0.7700 1st Qu.:-0.5394   
## Median :-0.1080 Median :-0.1910 Median :-0.5330 Median :-0.5394   
## Mean : 0.0000 Mean : 0.0000 Mean : 0.0000 Mean : 0.0000   
## 3rd Qu.: 0.6743 3rd Qu.: 0.6403 3rd Qu.: 0.3558 3rd Qu.: 0.8279   
## Max. : 3.0209 Max. : 3.9653 Max. : 4.5035 Max. : 3.5625   
## Sodium..mg. Total.Carbohydrates..g. Cholesterol..mg. Dietary.Fibre..g.  
## Min. :-0.7549 Min. :-1.7016 Min. :-1.77996 Min. :-0.5844   
## 1st Qu.:-0.7549 1st Qu.:-0.7069 1st Qu.:-0.77408 1st Qu.:-0.5844   
## Median :-0.1894 Median :-0.1164 Median :-0.05559 Median :-0.5844   
## Mean : 0.0000 Mean : 0.0000 Mean : 0.00000 Mean : 0.0000   
## 3rd Qu.: 0.3761 3rd Qu.: 0.5364 3rd Qu.: 0.73475 3rd Qu.: 0.1080   
## Max. : 3.7692 Max. : 2.5258 Max. : 2.53096 Max. : 4.9544   
## Sugars..g. Protein..g. Vitamin.A....DV. Vitamin.C....DV.   
## Min. :-1.70442 Min. :-1.5222 Min. :-1.2553 Min. :-0.2481   
## 1st Qu.:-0.75027 1st Qu.:-0.6887 1st Qu.:-0.7684 1st Qu.:-0.2481   
## Median :-0.04722 Median :-0.2719 Median :-0.2814 Median :-0.2481   
## Mean : 0.00000 Mean : 0.0000 Mean : 0.0000 Mean : 0.0000   
## 3rd Qu.: 0.50517 3rd Qu.: 0.5616 3rd Qu.: 0.5707 3rd Qu.:-0.2481   
## Max. : 2.51389 Max. : 2.6453 Max. : 4.8315 Max. : 6.3873   
## Calcium....DV. Iron....DV. Caffeine..mg.   
## Min. :-1.4894 Min. :-0.7447 Min. :-1.3924   
## 1st Qu.:-0.8025 1st Qu.:-0.7447 1st Qu.:-0.5989   
## Median :-0.1156 Median :-0.4669 Median :-0.2312   
## Mean : 0.0000 Mean : 0.0000 Mean : 0.0000   
## 3rd Qu.: 0.5713 3rd Qu.: 0.1814 3rd Qu.: 0.8332   
## Max. : 2.6320 Max. : 3.8860 Max. : 4.9555

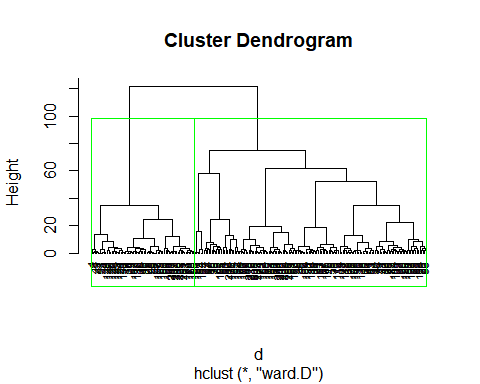
str(coffee\_norm)

## num [1:218, 1:15] -1.94 -1.93 -1.92 -1.92 -1.28 ...  
## - attr(\*, "dimnames")=List of 2  
## ..$ : NULL  
## ..$ : chr [1:15] "Calories" "Total.Fat..g." "Trans.Fat..g." "Saturated.Fat..g." ...  
## - attr(\*, "scaled:center")= Named num [1:15] 201.0413 3.0743 1.3995 0.0394 6.6743 ...  
## ..- attr(\*, "names")= chr [1:15] "Calories" "Total.Fat..g." "Trans.Fat..g." "Saturated.Fat..g." ...  
## - attr(\*, "scaled:scale")= Named num [1:15] 102.2723 3.0075 1.6877 0.0731 8.8416 ...  
## ..- attr(\*, "names")= chr [1:15] "Calories" "Total.Fat..g." "Trans.Fat..g." "Saturated.Fat..g." ...

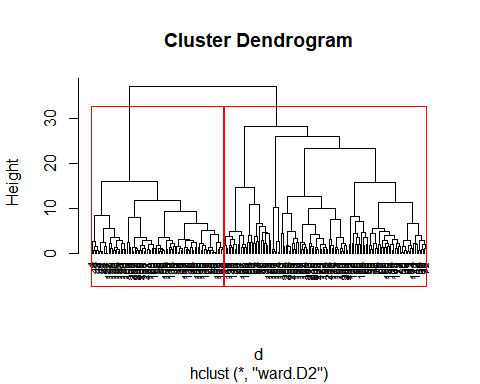
#dissimilarity matrix/ euclidean distance measure  
d <- dist(coffee\_norm, method = "euclidean")  
  
#Hierarchical clustering using Complete Linkage  
hc1 <- hclust(d, method = "complete")  
  
#ward methods  
hc\_fit <- hclust(d, method = "ward.D")  
hc\_ward <- hclust(d, method = "ward.D2")   
  
#plot the obtained dendrograms for hierarchical clustering  
plot(hc1, cex = 0.6, hang = -1)  
rect.hclust(hc1, k = 5, border = "blue")



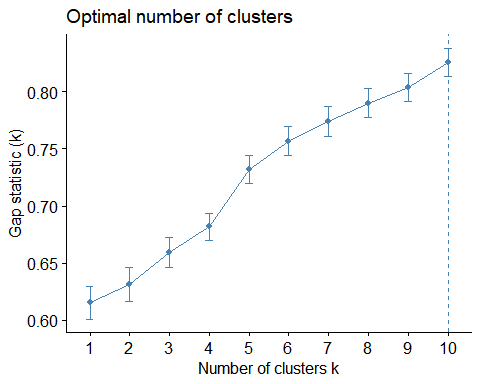
#choose 5 clusters when using complete linkage  
  
plot(hc\_fit, cex = 0.6, hang = -1)  
rect.hclust(hc\_fit, k = 2, border = "green")



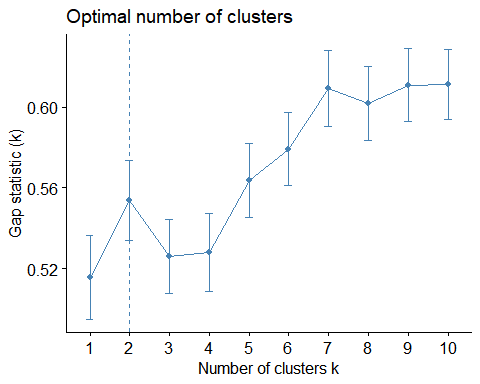
plot(hc\_ward, cex = 0.6, hang = -1)  
rect.hclust(hc\_ward, k = 2, border = "red")



#choose 2 clusters to best break up the data groupings with ward  
  
#calculate gap statistic for each number of clusters (up to 10 clusters) normalized data  
gap\_stat <- clusGap(coffee\_norm, FUN = hcut, nstart = 25, K.max = 10, B = 50)  
  
#produce plot of clusters vs. gap statistic  
fviz\_gap\_stat(gap\_stat)



#optimal number of clusters is 10  
  
#calculate gap statistic for each number of clusters (up to 10 clusters) non-normalized data  
gap\_stat <- clusGap(coffee1, FUN = hcut, nstart = 25, K.max = 10, B = 50)  
  
#produce plot of clusters vs. gap statistic  
fviz\_gap\_stat(gap\_stat)



#optimal number of clusters is 2  
  
#compute with agnes and with different linkage methods  
hc\_single <- agnes(coffee\_norm, method = "single")  
hc\_complete <- agnes(coffee\_norm, method = "complete")  
hc\_average <- agnes(coffee\_norm, method = "average")  
  
#compare Agglomerative coefficients  
print(hc\_single$ac)

## [1] 0.8123786

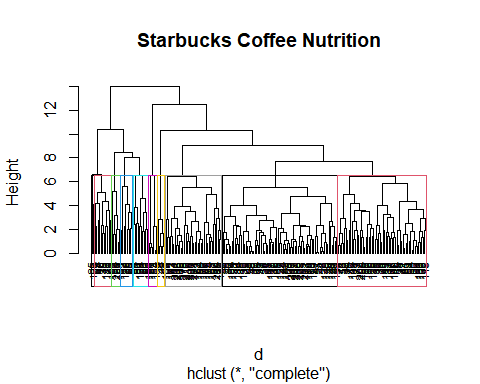
print(hc\_complete$ac)

## [1] 0.9112045

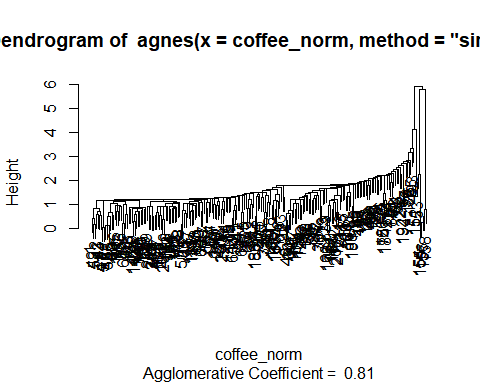
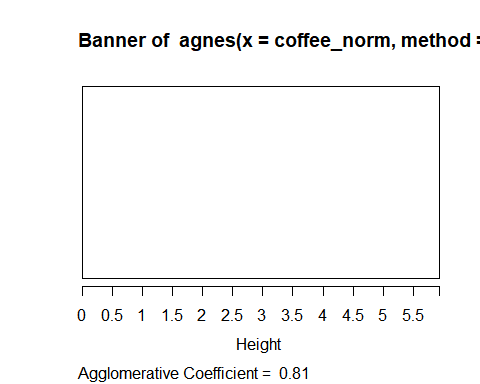
print(hc\_average$ac)

## [1] 0.8764107

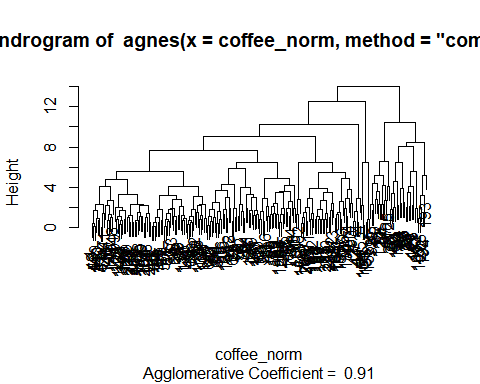
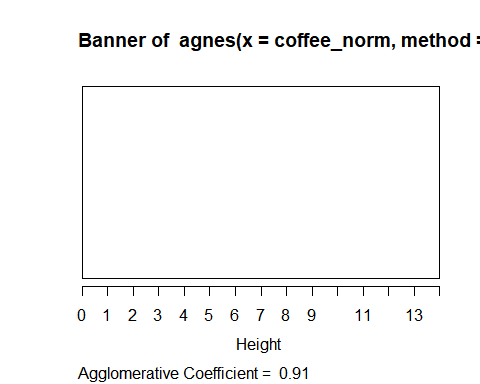
## the best method is "complete" because the Agglomerative coefficient is the closest to 1  
## the second choice would be "average"  
  
#plot the obtained dendrograms for hierarchical clustering with 10 clusters  
plot(hc1, cex = 0.6, hang = -1, main = "Starbucks Coffee Nutrition")  
rect.hclust(hc1, k = 10, border = 1:10)



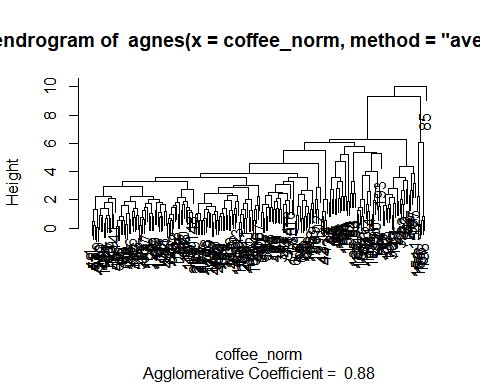
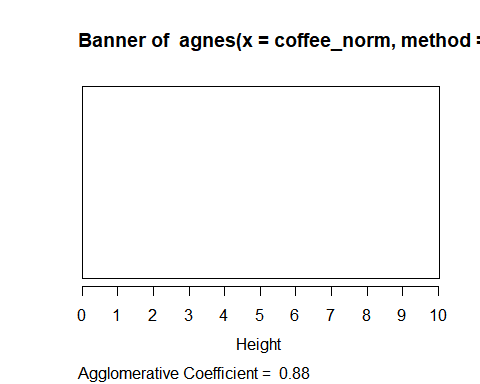
#plot  
plot(hc\_single)



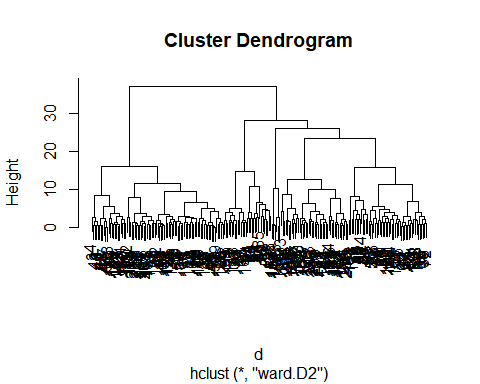
plot(hc\_complete)



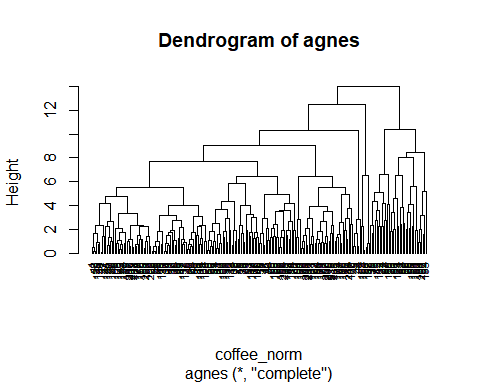
plot(hc\_average)



plot(hc\_ward)



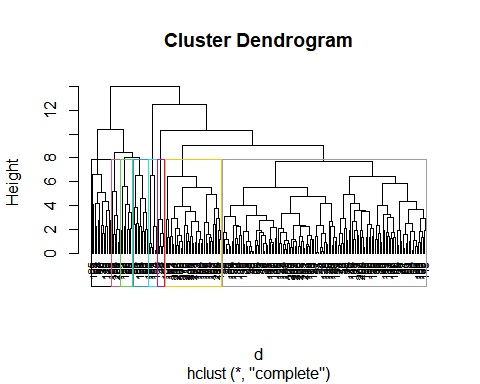
pltree(hc\_complete, cex = 0.6, hang = -1, main = "Dendrogram of agnes")



#define linkage methods  
m <- c( "average", "single", "complete", "ward")  
names(m) <- c( "average", "single", "complete", "ward")  
  
#function to compute agglomerative coefficient  
ac <- function(x) {agnes(coffee1, method = x)$ac}  
  
#calculate agglomerative coefficient for each clustering linkage method  
sapply(m, ac)

## average single complete ward   
## 0.9102861 0.7240627 0.9578418 0.9866696

##Ward method outputs a 0.986 agglomerative coefficient - best method  
##complete outputs a 0.957, so that is the next best option  
##average is 0.91  
##worst method is single, because the agglomerative cofficient is furthest from 1  
  
df <- coffee\_norm  
d <- dist(df, method = "euclidean")  
  
#compute divisive hierarchical clustering  
hc\_complete <- hclust(d, method = "complete")  
  
#plot dendrogram  
plot(hc\_complete, cex = 0.6, hang = -1)  
rect.hclust(hc\_complete, k = 8, border = 1:8) #assuming 8 clusters based on height cutoff



#cluster the data  
set.seed(1)  
  
#run kmeans algorithm using initial value of k=2 to cluster coffee drinks with normalized data  
k2 <- kmeans(coffee\_norm, centers = 2, nstart = 25) # k=2, number of restarts=25  
k2$centers #centroids/output the centers

## Calories Total.Fat..g. Trans.Fat..g. Saturated.Fat..g. Sodium..mg.  
## 1 -0.5616362 -0.4641924 -0.4673124 -0.3115100 -0.3573794  
## 2 0.9688225 0.8007318 0.8061139 0.5373548 0.6164795  
## Total.Carbohydrates..g. Cholesterol..mg. Dietary.Fibre..g. Sugars..g.  
## 1 -0.4375538 -0.4662023 -0.3335175 -0.4227662  
## 2 0.7547802 0.8041989 0.5753178 0.7292716  
## Protein..g. Vitamin.A....DV. Vitamin.C....DV. Calcium....DV. Iron....DV.  
## 1 -0.4795597 -0.3590726 -0.2163323 -0.3774564 -0.3145536  
## 2 0.8272405 0.6194002 0.3731732 0.6511122 0.5426050  
## Caffeine..mg.  
## 1 -0.09710244  
## 2 0.16750171

k2$size #size of each cluster

## [1] 138 80

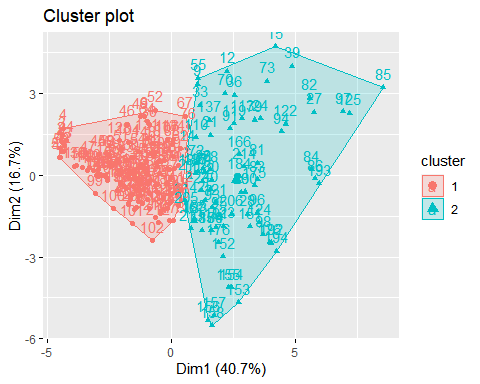
k2$cluster[15] #cluster of 15th observation in data set as example

## [1] 2

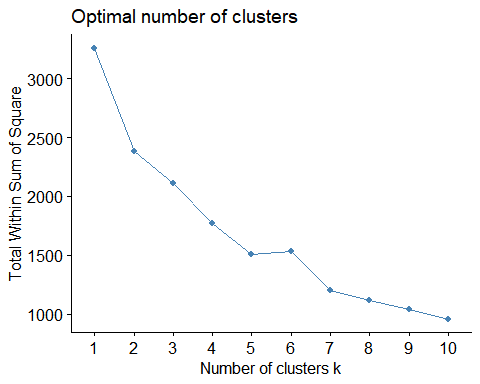
str(k2)

## List of 9  
## $ cluster : int [1:218] 1 1 1 1 1 1 1 1 2 1 ...  
## $ centers : num [1:2, 1:15] -0.562 0.969 -0.464 0.801 -0.467 ...  
## ..- attr(\*, "dimnames")=List of 2  
## .. ..$ : chr [1:2] "1" "2"  
## .. ..$ : chr [1:15] "Calories" "Total.Fat..g." "Trans.Fat..g." "Saturated.Fat..g." ...  
## $ totss : num 3255  
## $ withinss : num [1:2] 873 1506  
## $ tot.withinss: num 2379  
## $ betweenss : num 876  
## $ size : int [1:2] 138 80  
## $ iter : int 1  
## $ ifault : int 0  
## - attr(\*, "class")= chr "kmeans"

fviz\_cluster(k2, data = coffee\_norm) #visualize 2 clusters



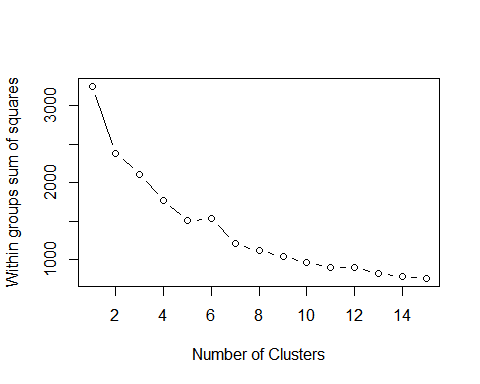
##2 clusters provides no overlapping  
  
#wss method  
set.seed(1)  
fviz\_nbclust(coffee\_norm, kmeans, method = "wss")



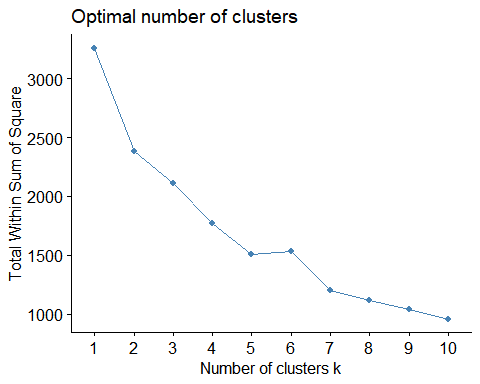
#this method shows that number of clusters should be 2, k = 2 where the curve changes direction  
  
wss <- 0 #fit kmeans model  
for(i in 1:15) wss[i] <- sum(kmeans(coffee\_norm, centers = i)$withinss)  
wss

## [1] 3255.0000 2379.0409 2111.0184 1771.1450 1504.8718 1531.4146 1206.3241  
## [8] 1114.1606 1044.5753 957.8481 891.2077 898.4080 818.3669 784.3172  
## [15] 755.2027

plot(1:15, wss, type = "b", xlab="Number of Clusters", ylab="Within groups sum of squares")



fviz\_nbclust(coffee\_norm, kmeans, method = "wss")



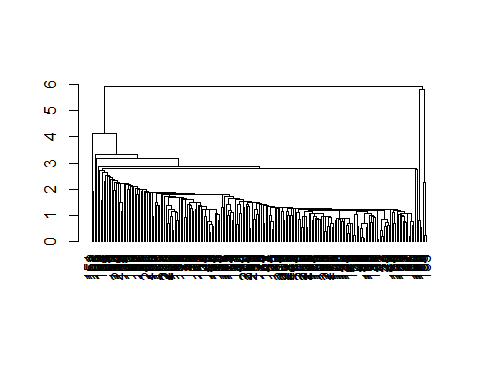
#choose k = 2  
k2 <- kmeans(coffee\_norm, centers = 2)  
k2clust <- k2$cluster  
#store in new data frame  
coffee\_clust <- as.data.frame(cbind(k2$cluster, coffee\_norm))  
head(coffee\_clust)

## V1 Calories Total.Fat..g. Trans.Fat..g. Saturated.Fat..g. Sodium..mg.  
## 1 2 -1.9364116 -0.9889702 -0.8292715 -0.5393932 -0.7548729  
## 2 2 -1.9266338 -0.9889702 -0.8292715 -0.5393932 -0.7548729  
## 3 2 -1.9168560 -0.9889702 -0.8292715 -0.5393932 -0.7548729  
## 4 2 -1.9168560 -0.9889702 -0.8292715 -0.5393932 -0.7548729  
## 5 2 -1.2812978 -0.9889702 -0.7700185 -0.5393932 -0.1893667  
## 6 2 -0.9879633 0.1415429 0.3557904 0.8279059 0.9416455  
## Total.Carbohydrates..g. Cholesterol..mg. Dietary.Fibre..g. Sugars..g.  
## 1 -1.6394720 -1.779956 -0.5843691 -1.704415  
## 2 -1.5773039 -1.779956 -0.5843691 -1.704415  
## 3 -1.5773039 -1.779956 -0.5843691 -1.704415  
## 4 -1.5773039 -1.779956 -0.5843691 -1.704415  
## 5 -0.7691174 -1.300965 -0.5843691 -1.252454  
## 6 -0.6447811 -1.300965 -0.5843691 -1.252454  
## Protein..g. Vitamin.A....DV. Vitamin.C....DV. Calcium....DV. Iron....DV.  
## 1 -1.4596788 -1.2553296 -0.2480669 -1.4894358 -0.7447436  
## 2 -1.4180037 -1.2553296 -0.2480669 -1.4894358 -0.7447436  
## 3 -1.3138161 -1.2553296 -0.2480669 -1.4894358 -0.7447436  
## 4 -1.3138161 -1.2553296 -0.2480669 -1.3520561 -0.7447436  
## 5 -0.2719394 -0.0379726 -0.2480669 -0.1156384 -0.7447436  
## 6 -0.2719394 -0.0379726 -0.2480669 -0.1156384 -0.7447436  
## Caffeine..mg.  
## 1 1.3170741  
## 2 2.6330828  
## 3 3.7168547  
## 4 4.9554512  
## 5 -0.2311715  
## 6 -0.2311715

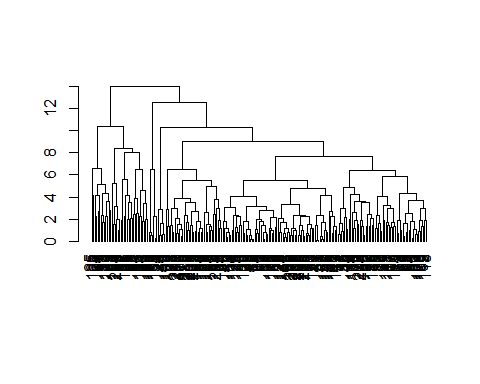
#analyze clusters & stability  
hclust\_stability = clusterboot(coffee\_norm, clustermethod=hclustCBI, method="ward.D2", k=2, count = FALSE)  
hclust\_stability

## \* Cluster stability assessment \*  
## Cluster method: hclust/cutree   
## Full clustering results are given as parameter result  
## of the clusterboot object, which also provides further statistics  
## of the resampling results.  
## Number of resampling runs: 100   
##   
## Number of clusters found in data: 2   
##   
## Clusterwise Jaccard bootstrap (omitting multiple points) mean:  
## [1] 0.5860242 0.5763894  
## dissolved:  
## [1] 16 39  
## recovered:  
## [1] 4 13

clusters = hclust\_stability$results$partition  
  
#stablility  
  
#cluster analysis: single vs. complete  
d <- dist(coffee\_norm, method = "euclidean")  
hc\_single <- hclust(d, method = "single")  
plot(hc\_single, hang = -1, ann = FALSE)



#single is not a good method based on the graph  
  
hc\_complete <- hclust(d, method = "complete")  
plot(hc\_complete, hang = -1, ann = FALSE)



#remove around 10% (5% from head, 5% tail data) to check cluster stability  
dim(coffee\_norm)

## [1] 218 15

#the graphs above depict that the best number of clusters is k = 10  
#even though only a portion of the data was used in these examples  
  
#The height of the cut to the dendrogram controls the number of clusters obtained.   
#It plays the same role as the k in k-means clustering.  
#The function cutree() (one of the options) can be used and it returns a vector   
#containing the cluster number of each observation.  
  
single <- cutree(hc\_single, k = 10)  
single

## [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [26] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [51] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [76] 1 1 1 2 1 1 3 1 1 4 1 1 1 1 1 1 1 1 1 1 1 5 1 1 1  
## [101] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 6  
## [126] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [151] 7 7 8 8 8 9 9 9 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
## [176] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 10 1 1 1 1 1 1 1  
## [201] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

complete <- cutree(hc\_complete, k = 10)  
complete

## [1] 1 2 2 2 1 1 1 3 3 1 3 3 3 3 4 3 1 1 1 3 3 3 5 4 6  
## [26] 5 4 6 1 1 1 3 3 1 3 3 3 3 4 3 1 1 2 2 1 1 1 1 1 1  
## [51] 3 1 1 3 3 3 1 1 1 3 3 3 1 1 1 3 3 1 3 3 3 3 4 3 3  
## [76] 3 3 3 4 3 5 4 5 5 7 5 1 1 1 3 3 3 5 4 6 5 4 6 1 8  
## [101] 8 8 1 1 1 3 3 3 3 3 3 3 4 3 1 1 1 3 3 3 5 4 6 5 7  
## [126] 6 1 1 1 1 1 1 3 1 1 3 3 3 1 1 1 1 1 1 1 1 1 1 1 1  
## [151] 6 6 9 9 9 9 9 9 1 8 1 8 8 8 8 8 8 1 8 1 8 8 8 8 8  
## [176] 8 1 8 1 8 8 8 8 8 8 3 8 3 10 8 10 10 10 10 1 1 8 1 1 8  
## [201] 1 1 8 3 3 10 1 8 1 8 8 8 8 8 1 8 1 8

#In order to adequately understand the coffee data, we would not want to normalize the data because that would skew the actual values of the categories.  
#Instead, we would want to cluster the data by grouping together coffee's with similar nutritional value or caffeine content  
  
#create new data set for example  
sbuxcoffee <- coffee1  
  
fit <- kmeans(sbuxcoffee, 10)  
aggregate(sbuxcoffee, by=list(fit$cluster), FUN=mean)

## Group.1 Calories Total.Fat..g. Trans.Fat..g. Saturated.Fat..g. Sodium..mg.  
## 1 1 290.00000 1.850000 0.6900000 0.01000000 5.000000  
## 2 2 311.48148 5.418519 2.7185185 0.08148148 11.851852  
## 3 3 207.05882 2.920588 1.3088235 0.05000000 6.176471  
## 4 4 189.23077 1.592308 0.9000000 0.03076923 6.153846  
## 5 5 237.00000 4.620000 1.6550000 0.04500000 9.500000  
## 6 6 26.70000 0.050000 0.0000000 0.00000000 0.000000  
## 7 7 83.33333 1.403333 0.4600000 0.01333333 3.333333  
## 8 8 134.11765 1.858824 0.6882353 0.02941176 5.000000  
## 9 9 384.00000 6.260000 3.6900000 0.06500000 10.500000  
## 10 10 150.00000 2.529730 0.9783784 0.02972973 5.675676  
## Total.Carbohydrates..g. Cholesterol..mg. Dietary.Fibre..g. Sugars..g.  
## 1 109.00000 59.70000 5.4000000 44.50000  
## 2 200.37037 57.77778 1.2222222 53.85185  
## 3 145.29412 40.41176 0.6470588 38.26471  
## 4 224.61538 37.07692 0.3076923 35.23077  
## 5 141.50000 38.95000 1.0500000 33.75000  
## 6 8.10000 5.70000 0.0000000 4.60000  
## 7 52.13333 13.63333 0.2666667 12.13333  
## 8 61.17647 25.23529 0.4117647 23.58824  
## 9 293.50000 73.90000 0.9500000 70.30000  
## 10 110.54054 25.29730 0.4324324 22.94595  
## Protein..g. Vitamin.A....DV. Vitamin.C....DV. Calcium....DV. Iron....DV.  
## 1 13.400000 17.400000 57.0000000 10.00000 10.600000  
## 2 10.037037 13.888889 4.4074074 31.11111 13.740741  
## 3 5.882353 8.500000 2.0000000 19.32353 8.676471  
## 4 6.769231 10.923077 0.1538462 20.38462 3.923077  
## 5 11.650000 16.750000 0.4000000 37.75000 14.200000  
## 6 0.720000 0.000000 0.0000000 0.80000 0.600000  
## 7 4.176667 6.166667 0.0000000 13.36667 2.866667  
## 8 4.764706 7.235294 0.8235294 15.58824 4.588235  
## 9 9.550000 11.600000 1.7000000 29.50000 12.700000  
## 10 7.081081 10.621622 0.0000000 22.86486 6.000000  
## Caffeine..mg.  
## 1 4.500000  
## 2 87.407407  
## 3 31.470588  
## 4 112.692308  
## 5 151.000000  
## 6 240.500000  
## 7 88.333333  
## 8 9.117647  
## 9 138.750000  
## 10 98.918919

#Cluster 6 has the highest caffeine in mg, cluster 1 has the lowest  
#Cluster 6 has the lowest calories, cluster 9 has the highest  
#Cluster 6 has the lowest fat, cluster 9 has the highest  
#Cluster 6 has the lowest trans fat, cluster 9 has the highest  
#Cluster 6 has the lowest saturated fat, cluster 2 has the highest  
#Cluster 6 has the lowest sodium, cluster 2 has the highest  
#Cluster 6 has the lowest carbohydrates, cluster 9 has the highest  
#Cluster 6 has the lowest cholesterol, cluster 9 has the highest  
#Cluster 1 has the highest fiber, cluster 6 has the lowest  
#Cluster 6 has the lowest sugar, cluster 9 has the highest  
#Cluster 1 has the highest protein, cluster 6 has the lowest  
  
#The summary above demonstrates that the coffee drinks in cluster 6 are the best choice overall because they have the highest caffeine content and best overall nutritional value.  
#The most unhealthy coffee choices are those in clusters 2 and 9 which have the worst overall nutritional value