FML ASSIGNMENT 5

Sri Chandana

2023-12-03

Cereals <- read.csv("C:/Users/srich/OneDrive/Desktop/Cereals.csv")  
Cereals\_1<-read.csv("C:/Users/srich/OneDrive/Desktop/Cereals.csv")  
# Displays the structure of the dataset  
str(Cereals)

## 'data.frame': 77 obs. of 16 variables:  
## $ name : chr "100%\_Bran" "100%\_Natural\_Bran" "All-Bran" "All-Bran\_with\_Extra\_Fiber" ...  
## $ mfr : chr "N" "Q" "K" "K" ...  
## $ type : chr "C" "C" "C" "C" ...  
## $ calories: int 70 120 70 50 110 110 110 130 90 90 ...  
## $ protein : int 4 3 4 4 2 2 2 3 2 3 ...  
## $ fat : int 1 5 1 0 2 2 0 2 1 0 ...  
## $ sodium : int 130 15 260 140 200 180 125 210 200 210 ...  
## $ fiber : num 10 2 9 14 1 1.5 1 2 4 5 ...  
## $ carbo : num 5 8 7 8 14 10.5 11 18 15 13 ...  
## $ sugars : int 6 8 5 0 8 10 14 8 6 5 ...  
## $ potass : int 280 135 320 330 NA 70 30 100 125 190 ...  
## $ vitamins: int 25 0 25 25 25 25 25 25 25 25 ...  
## $ shelf : int 3 3 3 3 3 1 2 3 1 3 ...  
## $ weight : num 1 1 1 1 1 1 1 1.33 1 1 ...  
## $ cups : num 0.33 1 0.33 0.5 0.75 0.75 1 0.75 0.67 0.67 ...  
## $ rating : num 68.4 34 59.4 93.7 34.4 ...

# Displays the first 6 rows of the "Cereals" dataset  
head(Cereals)

## name mfr type calories protein fat sodium fiber carbo  
## 1 100%\_Bran N C 70 4 1 130 10.0 5.0  
## 2 100%\_Natural\_Bran Q C 120 3 5 15 2.0 8.0  
## 3 All-Bran K C 70 4 1 260 9.0 7.0  
## 4 All-Bran\_with\_Extra\_Fiber K C 50 4 0 140 14.0 8.0  
## 5 Almond\_Delight R C 110 2 2 200 1.0 14.0  
## 6 Apple\_Cinnamon\_Cheerios G C 110 2 2 180 1.5 10.5  
## sugars potass vitamins shelf weight cups rating  
## 1 6 280 25 3 1 0.33 68.40297  
## 2 8 135 0 3 1 1.00 33.98368  
## 3 5 320 25 3 1 0.33 59.42551  
## 4 0 330 25 3 1 0.50 93.70491  
## 5 8 NA 25 3 1 0.75 34.38484  
## 6 10 70 25 1 1 0.75 29.50954

# Calculates and returns the total count of missing values in the "Cereals" dataset   
sum(is.na(Cereals))

## [1] 4

# Removes rows with missing values from the "Cereals" dataset  
Cereals <- na.omit(Cereals)  
# Removes rows with missing values from the "Cereals\_1" dataset  
Cereals\_1 <-na.omit(Cereals\_1)  
# Calculates the sum of missing values in the "Cereals" data and print the result  
sum(is.na(Cereals))

## [1] 0

# Convert the names of the cereals to row names   
rownames(Cereals) <- Cereals$name  
rownames(Cereals\_1) <- Cereals\_1$name

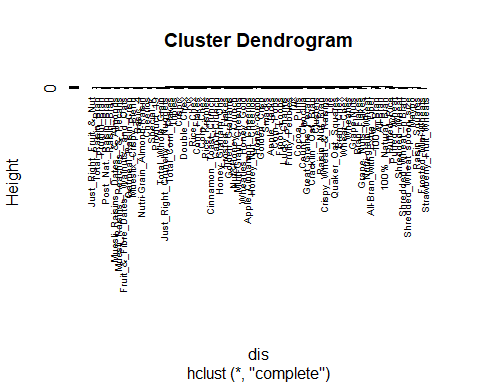
# Sets the "name" variable in the "Cereals" dataset to NULL  
# Sets the "name" variable in the "Cereals\_1" dataset to NULL  
Cereals$name = NULL  
Cereals\_1$name = NULL

# Scales the variables in columns 3 to 15 of the "Cereals" dataset  
Cereals <- scale(Cereals[,3:15])

# Calculates Euclidean distance matrix for the "Cereals" dataset  
dis <- dist(Cereals, method = "euclidean")

# Performs hierarchical clustering using complete linkage method  
hc\_comp <- hclust(dis, method = "complete" )

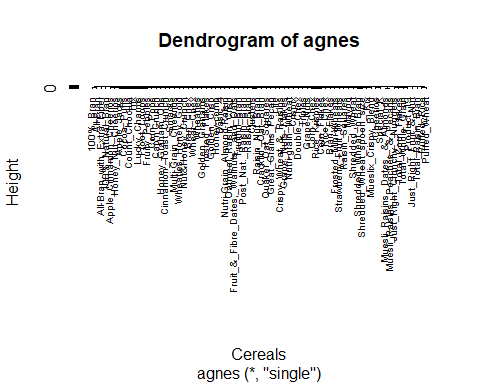
# Plotting hierarchical clustering dendrogram  
plot(hc\_comp, cex = 0.6, hang = -1)



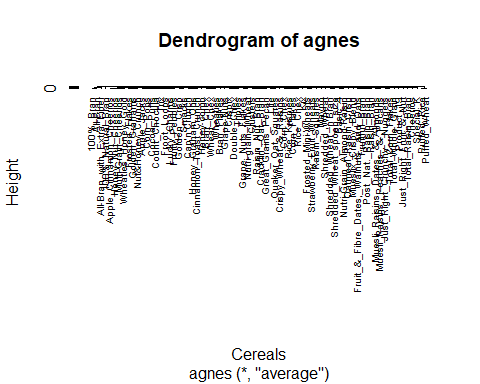
# Loads the specified library  
library(cluster)

## Warning: package 'cluster' was built under R version 4.3.2

# Performs hierarchical clustering using the agnes function with the "single" method  
hc\_single1 <- agnes(Cereals, method = "single")  
# Visualizes the dendrogram using the pltree function  
pltree(hc\_single1, cex = 0.6, hang = -1, main = "Dendrogram of agnes")



# Performs hierarchical clustering  
hc\_avg <- agnes(Cereals, method = "average")  
# Plots the dendrogram  
pltree(hc\_avg, cex = 0.6, hang = -1, main = "Dendrogram of agnes")



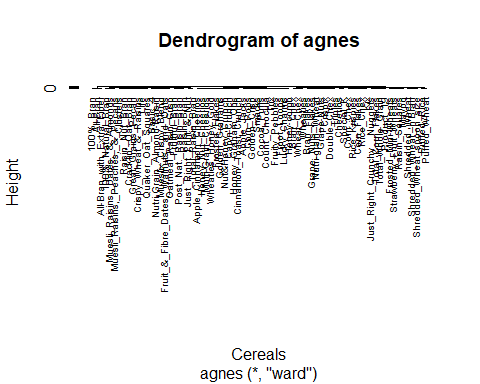
# Defines a vector "m" with clustering method names  
m <- c( "average", "single", "complete", "ward")  
# Assigns names to the vector elements  
names(m) <- c( "average", "single", "complete", "ward")  
# Defines a function "ac" that takes a clustering method as an argument and returns the agglomerative coefficient  
ac <- function(x) {  
 agnes(Cereals, method = x)$ac  
}  
#install.packages("purrr")  
library(purrr)

## Warning: package 'purrr' was built under R version 4.3.2

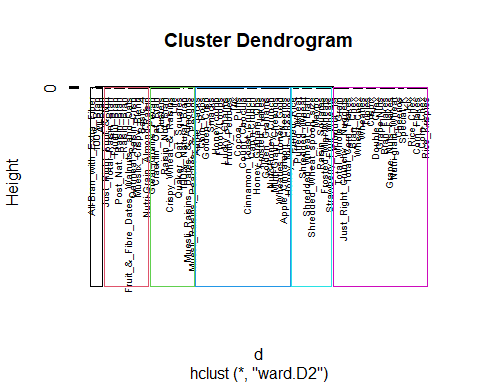
map\_dbl(m, ac)

## average single complete ward   
## 0.7766075 0.6067859 0.8353712 0.9046042

### Based on the obtained coefficients, we can see that the ward linkage method has the highest coefficient.So, Ward is the best linking method with an agglomerative coefficient of 0.9046042.   
### Ward linkage method is the best method for clustering the cereal data based on Euclidean distance to the normalized measurements  
  
  
  
# Performs hierarchical clustering on the "Cereals" dataset using the agnes function with the "ward" method  
hc\_wards <- agnes(Cereals, method = "ward")  
# Plots the dendrogram based on the hierarchical clustering results  
# Adjusting the text size (cex), hang distance, and adding title to the plot.  
pltree(hc\_wards, cex = 0.6, hang = -1, main = "Dendrogram of agnes")



# Calculates the Euclidean distance matrix for the "Cereals" dataset and store it in "d"  
d <- dist(Cereals, method = "euclidean")  
# Performs hierarchical clustering using the Ward's method on the distance matrix "d"  
hc\_ward\_clust <- hclust(d, method = "ward.D2" )  
# Plots the hierarchical clustering dendrogram with reduced text size  
plot(hc\_ward\_clust, cex=0.6 )  
# Highlights clusters by drawing rectangles around them where "k=6" specifies 6 clusters  
rect.hclust(hc\_ward\_clust,k=6,border = 1:6)



### 6 Clusters appear to be a good number to group the data using the Ward linkage

# Using hierarchical clustering with the Ward's method (hc\_ward\_clust) to create clusters, and then assigning each observation to a subgroup using cutree with k = 6.  
# Displays the table of counts for each subgroup.  
sub\_group <- cutree(hc\_ward\_clust, k = 6)  
table(sub\_group)

## sub\_group  
## 1 2 3 4 5 6   
## 3 10 21 10 21 9

#install.packages("GGally")  
# Load the required library  
library(GGally)

## Warning: package 'GGally' was built under R version 4.3.2

## Loading required package: ggplot2

## Warning: package 'ggplot2' was built under R version 4.3.2

## Registered S3 method overwritten by 'GGally':  
## method from   
## +.gg ggplot2

library(dplyr)

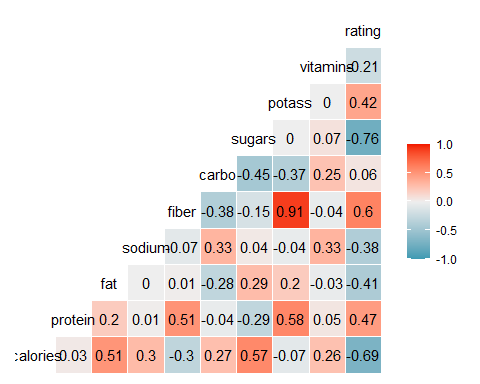
## Warning: package 'dplyr' was built under R version 4.3.2

##   
## 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

# Using the Cereals\_1 data frame and selecting specific columns related to nutrition and rating  
# Adjusting the data frame name and column names based on actual data  
Cereals\_1 %>%   
 select(calories, protein, fat, sodium, fiber, carbo, sugars, potass,vitamins,rating) %>%   
# Create a correlation matrix plot using ggcorr function from GGally  
 ggcorr(palette = "RdBu", label = TRUE, label\_round = 2)



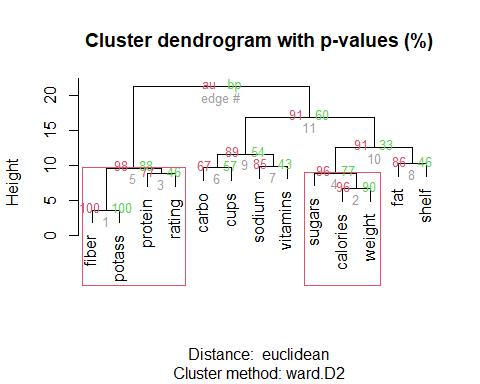
#install.packages("pvclust")  
# Load the required library  
library(pvclust)

## Warning: package 'pvclust' was built under R version 4.3.2

# Performs hierarchical clustering on the "Cereals" data using the Ward.D2 method with Euclidean distance as the dissimilarity measure  
fit.pv <- pvclust(Cereals, method.hclust="ward.D2",  
 method.dist="euclidean")

## Bootstrap (r = 0.5)... Done.  
## Bootstrap (r = 0.59)... Done.  
## Bootstrap (r = 0.69)... Done.  
## Bootstrap (r = 0.8)... Done.  
## Bootstrap (r = 0.89)... Done.  
## Bootstrap (r = 1.0)... Done.  
## Bootstrap (r = 1.09)... Done.  
## Bootstrap (r = 1.19)... Done.  
## Bootstrap (r = 1.3)... Done.  
## Bootstrap (r = 1.39)... Done.

# Plots the results of the fit.pv model using the plot function  
plot(fit.pv)  
pvrect(fit.pv, alpha=.95)



### The effectiveness of the initial clustering is evaluated through the cluster stability, represented by the mean Jaccard coefficient across all bootstrap iterations for each cluster.If a clusters stability rating is below 0.6, it is said to be unstable.  
### Stability rating between 0.6 and 0.75 suggests that a cluster identifies a pattern in the data,but there is not a strong consensus on which points should be grouped together. Exceptionally stable clusters are characterized by stability ratings exceeding 0.85.  
### The goal is to maximize the Jaccard bootstrap for each cluster.Where efforts should be made to minimize the dissolution of clusters.While aiming to maintain proximity to the original cluster count, increasing the number of recovered clusters is advisable.

#install.packages("fpc")  
# Load the required library  
library(fpc)

## Warning: package 'fpc' was built under R version 4.3.2

# Set the number of clusters to 6  
Kbest\_p<-6  
# Perform hierarchical clustering using the Ward.D2 method and the hclustCBI function on the "Cereals" dataset with cluster bootstrapping  
cboot\_hclust <- clusterboot(Cereals,clustermethod=hclustCBI,method="ward.D2", k=Kbest\_p)

## boot 1   
## boot 2   
## boot 3   
## boot 4   
## boot 5   
## boot 6   
## boot 7   
## boot 8   
## boot 9   
## boot 10   
## boot 11   
## boot 12   
## boot 13   
## boot 14   
## boot 15   
## boot 16   
## boot 17   
## boot 18   
## boot 19   
## boot 20   
## boot 21   
## boot 22   
## boot 23   
## boot 24   
## boot 25   
## boot 26   
## boot 27   
## boot 28   
## boot 29   
## boot 30   
## boot 31   
## boot 32   
## boot 33   
## boot 34   
## boot 35   
## boot 36   
## boot 37   
## boot 38   
## boot 39   
## boot 40   
## boot 41   
## boot 42   
## boot 43   
## boot 44   
## boot 45   
## boot 46   
## boot 47   
## boot 48   
## boot 49   
## boot 50   
## boot 51   
## boot 52   
## boot 53   
## boot 54   
## boot 55   
## boot 56   
## boot 57   
## boot 58   
## boot 59   
## boot 60   
## boot 61   
## boot 62   
## boot 63   
## boot 64   
## boot 65   
## boot 66   
## boot 67   
## boot 68   
## boot 69   
## boot 70   
## boot 71   
## boot 72   
## boot 73   
## boot 74   
## boot 75   
## boot 76   
## boot 77   
## boot 78   
## boot 79   
## boot 80   
## boot 81   
## boot 82   
## boot 83   
## boot 84   
## boot 85   
## boot 86   
## boot 87   
## boot 88   
## boot 89   
## boot 90   
## boot 91   
## boot 92   
## boot 93   
## boot 94   
## boot 95   
## boot 96   
## boot 97   
## boot 98   
## boot 99   
## boot 100

# Summarizing the results of hierarchical clustering using cboot\_hclust  
summary(cboot\_hclust$result)

## Length Class Mode   
## result 7 hclust list   
## noise 1 -none- logical   
## nc 1 -none- numeric   
## clusterlist 6 -none- list   
## partition 74 -none- numeric   
## clustermethod 1 -none- character  
## nccl 1 -none- numeric

# Extracting the cluster assignments from the hierarchical clustering results stored in cboot\_hclust$result$partition and displaying the first few rows as a data frame using head  
groups<-cboot\_hclust$result$partition  
head(data.frame(groups))

## groups  
## 100%\_Bran 1  
## 100%\_Natural\_Bran 2  
## All-Bran 1  
## All-Bran\_with\_Extra\_Fiber 1  
## Apple\_Cinnamon\_Cheerios 3  
## Apple\_Jacks 3

# Extracting the bootstrap means from the "cboot\_hclust" object  
cboot\_hclust$bootmean

## [1] 0.8775349 0.5342264 0.9100573 0.6506041 0.6107279 0.6840468

# Count of how many times each cluster was dissolved and by default clusterboot() runs 100 bootstrap iterations   
# Accessing the "bootbrd" column in the "cboot\_hclust" data frame  
cboot\_hclust$bootbrd

## [1] 13 56 0 40 26 39

### The results suggest that clusters 1 and 3 exhibit a high degree of stability. While clusters 4 and 5 show indications of identifying a pattern, there is a notable absence of consensus regarding the optimal grouping of specific points.Clusters 2 and 5 are currently displaying instability.  
  
  
# Assign cluster labels using cutree() based on hierarchical clustering (hc\_ward\_clust) with k=6  
groups <- cutree(hc\_ward\_clust, k = 6)  
# Define a function to print details of each cluster  
print\_clusters <- function(labels, k) {  
for(i in 1:k) {  
print(paste("cluster", i))  
print(Cereals\_1[labels==i,c("mfr","calories","protein","fat","sodium","fiber","carbo","sugars","potass",  
 "vitamins","rating")])  
}  
}  
# Call the function to print cluster details for the assigned labels  
print\_clusters(groups, 6)

## [1] "cluster 1"  
## mfr calories protein fat sodium fiber carbo sugars  
## 100%\_Bran N 70 4 1 130 10 5 6  
## All-Bran K 70 4 1 260 9 7 5  
## All-Bran\_with\_Extra\_Fiber K 50 4 0 140 14 8 0  
## potass vitamins rating  
## 100%\_Bran 280 25 68.40297  
## All-Bran 320 25 59.42551  
## All-Bran\_with\_Extra\_Fiber 330 25 93.70491  
## [1] "cluster 2"  
## mfr calories protein fat sodium fiber carbo  
## 100%\_Natural\_Bran Q 120 3 5 15 2.0 8.0  
## Clusters G 110 3 2 140 2.0 13.0  
## Cracklin'\_Oat\_Bran K 110 3 3 140 4.0 10.0  
## Crispy\_Wheat\_&\_Raisins G 100 2 1 140 2.0 11.0  
## Great\_Grains\_Pecan P 120 3 3 75 3.0 13.0  
## Life Q 100 4 2 150 2.0 12.0  
## Muesli\_Raisins,\_Dates,\_&\_Almonds R 150 4 3 95 3.0 16.0  
## Muesli\_Raisins,\_Peaches,\_&\_Pecans R 150 4 3 150 3.0 16.0  
## Quaker\_Oat\_Squares Q 100 4 1 135 2.0 14.0  
## Raisin\_Nut\_Bran G 100 3 2 140 2.5 10.5  
## sugars potass vitamins rating  
## 100%\_Natural\_Bran 8 135 0 33.98368  
## Clusters 7 105 25 40.40021  
## Cracklin'\_Oat\_Bran 7 160 25 40.44877  
## Crispy\_Wheat\_&\_Raisins 10 120 25 36.17620  
## Great\_Grains\_Pecan 4 100 25 45.81172  
## Life 6 95 25 45.32807  
## Muesli\_Raisins,\_Dates,\_&\_Almonds 11 170 25 37.13686  
## Muesli\_Raisins,\_Peaches,\_&\_Pecans 11 170 25 34.13976  
## Quaker\_Oat\_Squares 6 110 25 49.51187  
## Raisin\_Nut\_Bran 8 140 25 39.70340  
## [1] "cluster 3"  
## mfr calories protein fat sodium fiber carbo sugars  
## Apple\_Cinnamon\_Cheerios G 110 2 2 180 1.5 10.5 10  
## Apple\_Jacks K 110 2 0 125 1.0 11.0 14  
## Cap'n'Crunch Q 120 1 2 220 0.0 12.0 12  
## Cinnamon\_Toast\_Crunch G 120 1 3 210 0.0 13.0 9  
## Cocoa\_Puffs G 110 1 1 180 0.0 12.0 13  
## Corn\_Pops K 110 1 0 90 1.0 13.0 12  
## Count\_Chocula G 110 1 1 180 0.0 12.0 13  
## Froot\_Loops K 110 2 1 125 1.0 11.0 13  
## Frosted\_Flakes K 110 1 0 200 1.0 14.0 11  
## Fruity\_Pebbles P 110 1 1 135 0.0 13.0 12  
## Golden\_Crisp P 100 2 0 45 0.0 11.0 15  
## Golden\_Grahams G 110 1 1 280 0.0 15.0 9  
## Honey\_Graham\_Ohs Q 120 1 2 220 1.0 12.0 11  
## Honey\_Nut\_Cheerios G 110 3 1 250 1.5 11.5 10  
## Honey-comb P 110 1 0 180 0.0 14.0 11  
## Lucky\_Charms G 110 2 1 180 0.0 12.0 12  
## Multi-Grain\_Cheerios G 100 2 1 220 2.0 15.0 6  
## Nut&Honey\_Crunch K 120 2 1 190 0.0 15.0 9  
## Smacks K 110 2 1 70 1.0 9.0 15  
## Trix G 110 1 1 140 0.0 13.0 12  
## Wheaties\_Honey\_Gold G 110 2 1 200 1.0 16.0 8  
## potass vitamins rating  
## Apple\_Cinnamon\_Cheerios 70 25 29.50954  
## Apple\_Jacks 30 25 33.17409  
## Cap'n'Crunch 35 25 18.04285  
## Cinnamon\_Toast\_Crunch 45 25 19.82357  
## Cocoa\_Puffs 55 25 22.73645  
## Corn\_Pops 20 25 35.78279  
## Count\_Chocula 65 25 22.39651  
## Froot\_Loops 30 25 32.20758  
## Frosted\_Flakes 25 25 31.43597  
## Fruity\_Pebbles 25 25 28.02576  
## Golden\_Crisp 40 25 35.25244  
## Golden\_Grahams 45 25 23.80404  
## Honey\_Graham\_Ohs 45 25 21.87129  
## Honey\_Nut\_Cheerios 90 25 31.07222  
## Honey-comb 35 25 28.74241  
## Lucky\_Charms 55 25 26.73451  
## Multi-Grain\_Cheerios 90 25 40.10596  
## Nut&Honey\_Crunch 40 25 29.92429  
## Smacks 40 25 31.23005  
## Trix 25 25 27.75330  
## Wheaties\_Honey\_Gold 60 25 36.18756  
## [1] "cluster 4"  
## mfr calories protein fat sodium fiber  
## Basic\_4 G 130 3 2 210 2.0  
## Fruit\_&\_Fibre\_Dates,\_Walnuts,\_and\_Oats P 120 3 2 160 5.0  
## Fruitful\_Bran K 120 3 0 240 5.0  
## Just\_Right\_Fruit\_&\_Nut K 140 3 1 170 2.0  
## Mueslix\_Crispy\_Blend K 160 3 2 150 3.0  
## Nutri-Grain\_Almond-Raisin K 140 3 2 220 3.0  
## Oatmeal\_Raisin\_Crisp G 130 3 2 170 1.5  
## Post\_Nat.\_Raisin\_Bran P 120 3 1 200 6.0  
## Raisin\_Bran K 120 3 1 210 5.0  
## Total\_Raisin\_Bran G 140 3 1 190 4.0  
## carbo sugars potass vitamins rating  
## Basic\_4 18.0 8 100 25 37.03856  
## Fruit\_&\_Fibre\_Dates,\_Walnuts,\_and\_Oats 12.0 10 200 25 40.91705  
## Fruitful\_Bran 14.0 12 190 25 41.01549  
## Just\_Right\_Fruit\_&\_Nut 20.0 9 95 100 36.47151  
## Mueslix\_Crispy\_Blend 17.0 13 160 25 30.31335  
## Nutri-Grain\_Almond-Raisin 21.0 7 130 25 40.69232  
## Oatmeal\_Raisin\_Crisp 13.5 10 120 25 30.45084  
## Post\_Nat.\_Raisin\_Bran 11.0 14 260 25 37.84059  
## Raisin\_Bran 14.0 12 240 25 39.25920  
## Total\_Raisin\_Bran 15.0 14 230 100 28.59278  
## [1] "cluster 5"  
## mfr calories protein fat sodium fiber carbo sugars  
## Bran\_Chex R 90 2 1 200 4 15 6  
## Bran\_Flakes P 90 3 0 210 5 13 5  
## Cheerios G 110 6 2 290 2 17 1  
## Corn\_Chex R 110 2 0 280 0 22 3  
## Corn\_Flakes K 100 2 0 290 1 21 2  
## Crispix K 110 2 0 220 1 21 3  
## Double\_Chex R 100 2 0 190 1 18 5  
## Grape\_Nuts\_Flakes P 100 3 1 140 3 15 5  
## Grape-Nuts P 110 3 0 170 3 17 3  
## Just\_Right\_Crunchy\_\_Nuggets K 110 2 1 170 1 17 6  
## Kix G 110 2 1 260 0 21 3  
## Nutri-grain\_Wheat K 90 3 0 170 3 18 2  
## Product\_19 K 100 3 0 320 1 20 3  
## Rice\_Chex R 110 1 0 240 0 23 2  
## Rice\_Krispies K 110 2 0 290 0 22 3  
## Special\_K K 110 6 0 230 1 16 3  
## Total\_Corn\_Flakes G 110 2 1 200 0 21 3  
## Total\_Whole\_Grain G 100 3 1 200 3 16 3  
## Triples G 110 2 1 250 0 21 3  
## Wheat\_Chex R 100 3 1 230 3 17 3  
## Wheaties G 100 3 1 200 3 17 3  
## potass vitamins rating  
## Bran\_Chex 125 25 49.12025  
## Bran\_Flakes 190 25 53.31381  
## Cheerios 105 25 50.76500  
## Corn\_Chex 25 25 41.44502  
## Corn\_Flakes 35 25 45.86332  
## Crispix 30 25 46.89564  
## Double\_Chex 80 25 44.33086  
## Grape\_Nuts\_Flakes 85 25 52.07690  
## Grape-Nuts 90 25 53.37101  
## Just\_Right\_Crunchy\_\_Nuggets 60 100 36.52368  
## Kix 40 25 39.24111  
## Nutri-grain\_Wheat 90 25 59.64284  
## Product\_19 45 100 41.50354  
## Rice\_Chex 30 25 41.99893  
## Rice\_Krispies 35 25 40.56016  
## Special\_K 55 25 53.13132  
## Total\_Corn\_Flakes 35 100 38.83975  
## Total\_Whole\_Grain 110 100 46.65884  
## Triples 60 25 39.10617  
## Wheat\_Chex 115 25 49.78744  
## Wheaties 110 25 51.59219  
## [1] "cluster 6"  
## mfr calories protein fat sodium fiber carbo sugars  
## Frosted\_Mini-Wheats K 100 3 0 0 3 14 7  
## Maypo A 100 4 1 0 0 16 3  
## Puffed\_Rice Q 50 1 0 0 0 13 0  
## Puffed\_Wheat Q 50 2 0 0 1 10 0  
## Raisin\_Squares K 90 2 0 0 2 15 6  
## Shredded\_Wheat N 80 2 0 0 3 16 0  
## Shredded\_Wheat\_'n'Bran N 90 3 0 0 4 19 0  
## Shredded\_Wheat\_spoon\_size N 90 3 0 0 3 20 0  
## Strawberry\_Fruit\_Wheats N 90 2 0 15 3 15 5  
## potass vitamins rating  
## Frosted\_Mini-Wheats 100 25 58.34514  
## Maypo 95 25 54.85092  
## Puffed\_Rice 15 0 60.75611  
## Puffed\_Wheat 50 0 63.00565  
## Raisin\_Squares 110 25 55.33314  
## Shredded\_Wheat 95 0 68.23588  
## Shredded\_Wheat\_'n'Bran 140 0 74.47295  
## Shredded\_Wheat\_spoon\_size 120 0 72.80179  
## Strawberry\_Fruit\_Wheats 90 25 59.36399

### I opted to select clusters based on statistical values and nutritional richness with the goal of forming a healthy diet. This approach is inherently subjective,as it lacks a defined measure or scale for constructing a healthy diet.  
  
### To determine whether normalization was necessary,I concluded that it was not. Normalizing the data would diminish its magnitude, making analysis and decision-making more challenging.  
  
### The cereal diet levels within the clusters exhibit variations in richness, adequacy, and deficiencies in nutrients. After sorting the data into six groups, we will delve into these clusters considering all factors and variables.  
  
### While Cluster 1 provides nutritionally consistent guidelines for crafting a balanced diet, the options are somewhat limited. Clusters 2 and 3 are not advisable for a healthy meal due to their poor ratings and elevated levels of fat and sugar.  
  
### Clusters 4 and 5 showcase well-balanced nutritional values and receive high ratings for consumer satisfaction.Therefore,Clusters 4 and 5 are the optimal choices for primary public schools aiming to implement this in their cafeterias.