FML ASSIGNMENT-5

Sai Harshitha Akula

2023-12-02

#Installing the required packages.

```
#install.packages("purrr")
#It provides a functional programming approach to iterating, mapping, and manipulating data structures.

#install.packages("GGally")
#It provides enhanced functionalities for exploring and visualizing relationships within multivariate d

#install.packages("pvclust")
#The pvclust package is particularly useful when we want to assess the reliability of the clusters obta

#install.packages("fpc")
# It provides a range of functions and tools that are particularly useful for working with clustering a
```

#Loading the required libraries.

```
library(dplyr)
```

```
##
## 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
```

```
library(purrr)
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(pvclust)
```

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

```
library(fpc)
```

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

```
library(cluster)
```

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

#Importing the dataset.

```
Cereals <- read.csv("C:/Users/saiha/OneDrive/Documents/R PROGRAMMING/Cereals.csv")
Cereals1<-read.csv("C:/Users/saiha/OneDrive/Documents/R PROGRAMMING/Cereals.csv")
```

#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 ...
```

```
sum(is.na(Cereals))
```

```
## [1] 4
```

#Removing any instances of missing values within the dataset.

```
Cereals <- na.omit(Cereals)
Cereals1<-na.omit(Cereals1)
sum(is.na(Cereals))
```

```
## [1] 0
```

#Transforming the cereal names into row names to facilitate the subsequent visualization of clusters.

```
rownames(Cereals) <- Cereals$name
rownames(Cereals1) <- Cereals1$name
```

#Excluding the “name” column as it no longer contributes any meaningful information.

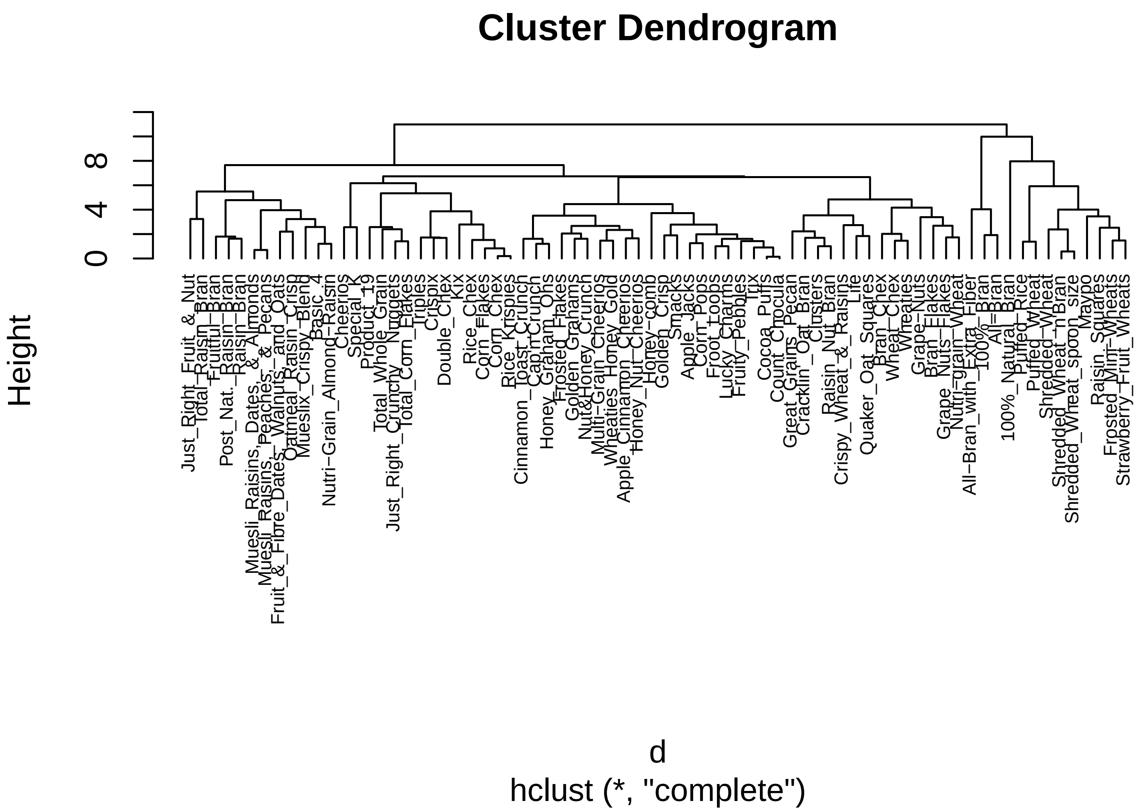
```
Cereals$name = NULL
Cereals1$name = NULL
```

#Before applying any distance measure, it is essential to normalize the data as variables with broader ranges can disproportionately impact the distance calculation.

```
Cereals <- scale(Cereals[,3:15])
```

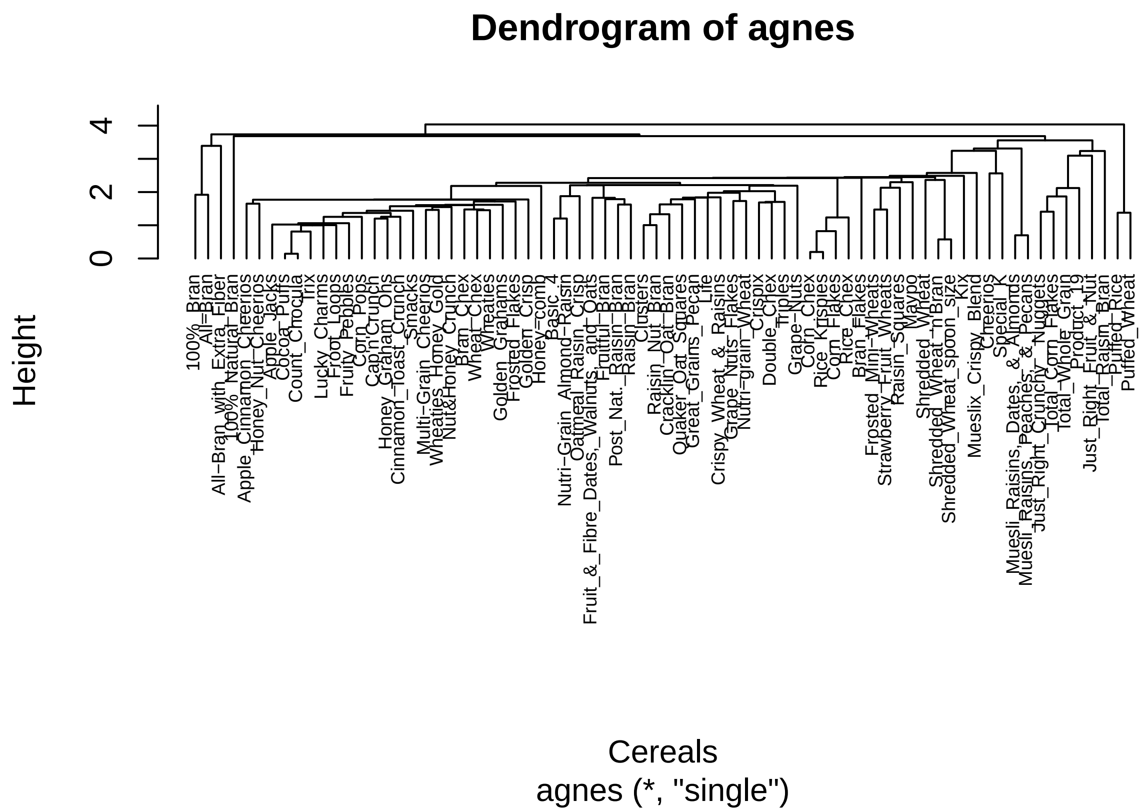
#Intending to employ the Euclidean distance metric for conducting hierarchical clustering on the dataset.

```
# computing the Euclidean distance matrix for the dataset using the "euclidean" method for distance cal
d <- dist(Cereals, method = "euclidean")
# initiating hierarchical clustering using the "complete" linkage method based on the dissimilarity mat
HC_complete <- hclust(d, method = "complete" )
# Plotting the obtained dendrogram.
plot(HC_complete, cex = 0.6, hang = -1)
```



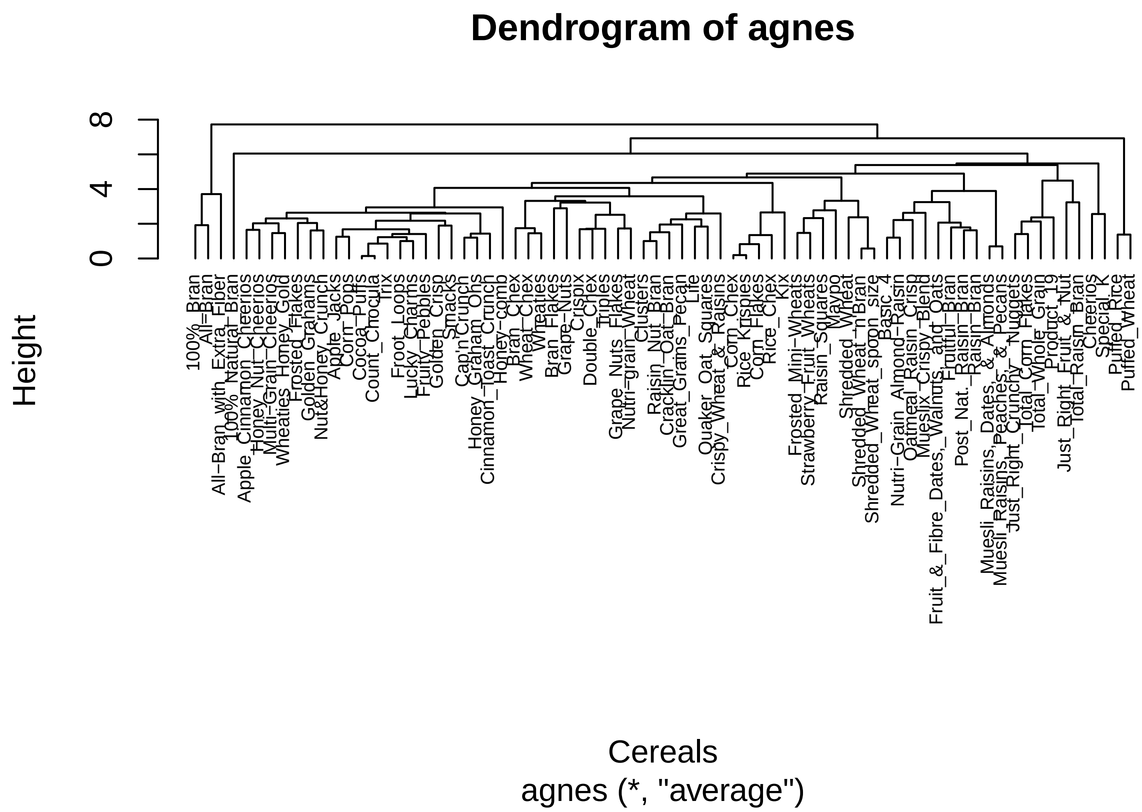

```r
#Using the agnes function from the cluster package to perform agglomerative hierarchical clustering wit
library(cluster)
HC_single <- agnes(Cereals, method = "single")

#Generating a dendrogram plot for the hierarchical clustering result obtained from the agnes function.
pltree(HC_single, cex = 0.6, hang = -1, main = "Dendrogram of agnes")
```



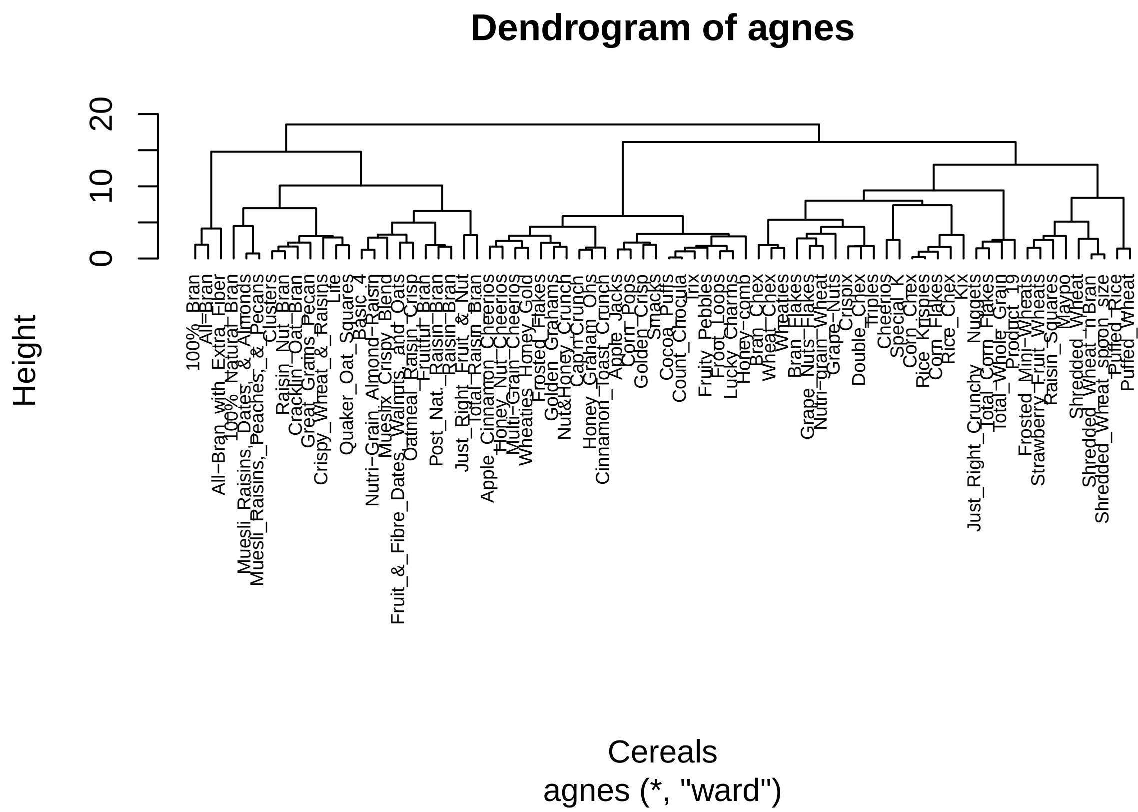

```r
#Using the agnes function from the cluster package to perform agglomerative hierarchical clustering wit
HC_average <- agnes(Cereals, method = "average")

#Generating a dendrogram plot.
pltree(HC_average, cex = 0.6, hang = -1, main = "Dendrogram of agnes")
```




```r
#Using the agnes function from the cluster package to perform agglomerative hierarchical clustering wit
HC_Ward <- agnes(Cereals, method = "ward")

#Generating a dendrogram plot.
pltree(HC_Ward, cex = 0.6, hang = -1, main = "Dendrogram of agnes")
```



#Computing the Agnes coefficient for each methodology.

```r
library(purrr)
# Representing different linkage methods used in hierarchical clustering.
m <- c( "average", "single", "complete", "ward")
names(m) <- c( "average", "single", "complete", "ward")
# Function for computing the coefficient
ac <- function(x) {
  agnes(Cereals, method = x)$ac
}
map_dbl(m, ac)
```

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

#So, from the above values we can see that “Ward” emerges as the most favorable linkage method, exhibiting a compelling agglomerative coefficient of 0.9046042.

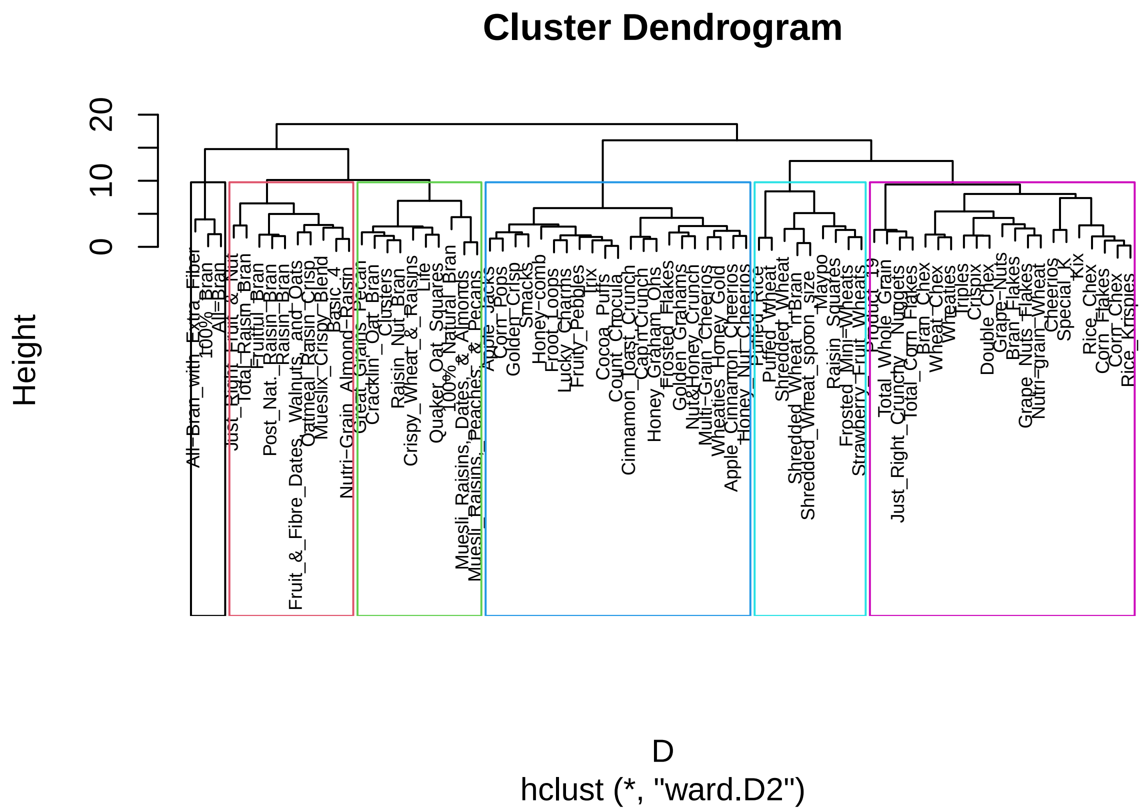
#Segment the dendrogram using the “cutree()” function to identify distinct sub-groups, commonly referred to as clusters.

```r
#Creating the distance matrix
D <- dist(Cereals, method = "euclidean")

# Using Ward method for Hierarchical clustering
HC_Ward_cluster <- hclust(D, method = "ward.D2" )
```

```r
#Creating a plot to visualize hierarchical clustering.
plot(HC_Ward_cluster, cex=0.6 )

#Drawing rectangles around clusters in a hierarchical clustering dendrogram.
rect.hclust(HC_Ward_cluster,k=6,border = 1:6)
```

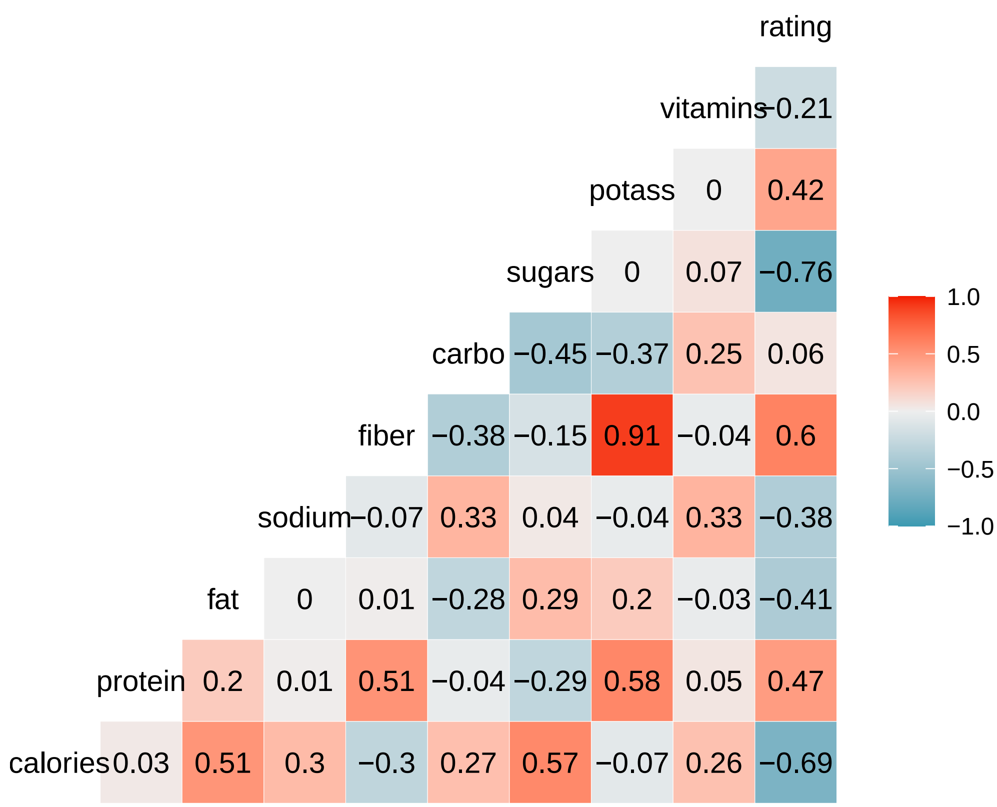


#Let's explore the distribution of data records across various clusters and understand the allocation patterns.

```r
# Cutting the tree into 6 groups.
sub_group <- cutree(HC_Ward_cluster, k = 6)
# Creating a frequency table, summarizing the counts of unique values.
table(sub_group)
```

```
## sub_group
##  1  2  3  4  5  6
##  3 10 21 10 21  9
```

```r
#install.packages("GGally")
library(GGally)
library(dplyr)
Cereals1 %>%
select(calories, protein, fat, sodium, fiber, carbo, sugars, potass,vitamins,rating) %>%
ggcorr(palette = "RdBu", label = TRUE, label_round =  2)
```



#The pvclust() function from the pvclust package is designed for assessing the statistical significance of clusters generated through hierarchical clustering using multiscale bootstrap resampling. In this method, clusters with strong support from the data are assigned larger p-values. It's important to note that when using pvclust, the grouping is applied to columns rather than rows. Therefore, we have to transpose the data before employing this method for more accurate results.

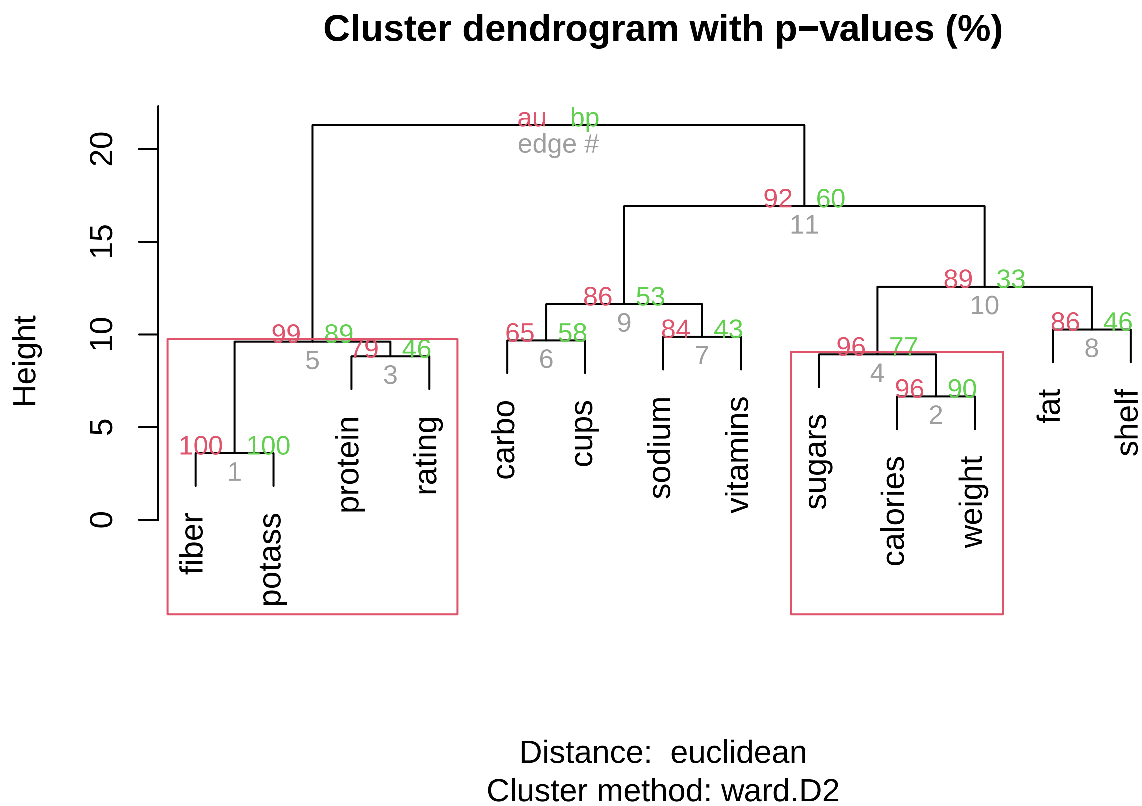
```
# Using Ward Hierarchical Clustering with Bootstrapped p values
library(pvclust)
```

```
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.
```

```
#Creating a dendogram with p values
plot(fit.pv)
```

```r
# Adding rectangles around groups highly supported by the data
pvrect(fit.pv, alpha=.95)
```



#The stability of each cluster in the original clustering is expressed through the average Jaccard coefficient across all bootstrap iterations. Clusters with a stability rating below 0.6 should be deemed unstable. When stability ratings fall between 0.6 and 0.75, the cluster can discern a pattern in the data, but there is not a robust consensus on the grouping of points. Notably stable clusters exhibit exceptional stability when their ratings surpass 0.85.

#1.The optimal strategy involves maximizing the Jaccard bootstrap for each cluster, emphasizing robustness in cluster-wise assessments.

#2.Minimizing the dissolution of clusters is advised to uphold their integrity effectively.

#3.Striving to augment the number of recovered clusters while maintaining proximity to the original configuration as closely as feasible.

#Running clusterboot() function

```r
#install.packages("fpc")
library(fpc)
library(cluster)
Kbestp<-6
clusterb_hclust <- clusterboot(Cereals,clustermethod=hclustCBI,method="ward.D2", k=Kbestp)
```

```
## 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
```

```r
summary(clusterb_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
```

```r
groups<-clusterb_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
```

#Representing the mean values obtained through bootstrapping.

```r
clusterb_hclust$bootmean
```

```
## [1] 0.9187551 0.5391111 0.8874789 0.6495864 0.6323878 0.7160858
```

#Determining the frequency with which each cluster is disassembled by default during the execution of clusterboot(), which, by default, conducts 100 bootstrap iterations.

```r
clusterb_hclust$bootbrd
```

```
## [1]  9 55  0 41 32 36
```

#Retrieving the identified clusters from the results generated by the hclust() function.

```r
groups <- cutree(HC_Ward_cluster, k = 6)
print_cluster <- function(labels, k) {
for(i in 1:k) {
print(paste("cluster", i))
print(Cereals1[labels==i,c("mfr","calories","protein","fat","sodium","fiber","carbo","sugars","potass",
                "vitamins","rating")])
}
}
print_cluster(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 have opted to select clusters based on both statistical values and nutritional richness to formulate a health-conscious diet. It's crucial to note that this approach is entirely subjective, as there is no explicit mention of a defined metric or scale for constructing a healthy diet.

– Regarding the necessity of normalization, my stance is negative. Normalizing the data tends to diminish its magnitude, posing significant challenges for comprehensive analysis and decision-making.

– The clusters representing different levels of cereal diets showcase varying degrees of richness, adequacy, and deficiencies in nutrients. After segregating the data into six distinct groups, a thorough examination of these clusters is conducted, taking into account all relevant factors and variables.

– Despite Cluster 1 offering nutritionally sound guidelines for crafting a balanced diet, it does present limitations due to its relatively constrained options. On the contrary, Clusters 2 and 3 are discouraged for inclusion in a health-conscious meal plan due to their subpar ratings and elevated levels of fat and sugar content.

– Clusters 4 and 5 stand out for maintaining a well-balanced nutritional profile and receiving favorable ratings for consumer satisfaction. Consequently, these clusters emerge as optimal choices, particularly suitable for implementation in primary public schools' cafeterias.