ASSIGNMENT_5

2023-11-30

Introduction

The data contains the information about 77 cereal in rows and 16 variables containing observation about different components of cereals like name of cereal, manufacturer, calories, protein, fats, sodium, potassium, fiber and vitamins etc. The Analysis is done by using the Hierarchical Clustering Model, which is an algorithm of unsupervised learning in which the number of clusters is not pre_specified. But select on the basis of comparison between different clusters created by data points of the data. The Analysis is done using R. ## Loading the required packages

Now read the csv file of the data and look at the above few rows and structure. ## R Markdown

```
cereal_data <- read.csv("Cereals.csv")
head(cereal_data)
```

```
##                         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
```

```
str(cereal_data)
```

```
## '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 ...
```

Data Preporcessing

After having a look into the data, it is time to process tha data and do some cleaning using the na.omit function.

```
## 'data.frame':    74 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 50 110 110 130 90 90 120 ...
##  $ protein : int  4 3 4 4 2 2 3 2 3 1 ...
##  $ fat     : int  1 5 1 0 2 0 2 1 0 2 ...
##  $ sodium  : int  130 15 260 140 180 125 210 200 210 220 ...
##  $ fiber   : num  10 2 9 14 1.5 1 2 4 5 0 ...
##  $ carbo   : num  5 8 7 8 10.5 11 18 15 13 12 ...
##  $ sugars  : int  6 8 5 0 10 14 8 6 5 12 ...
##  $ potass  : int  280 135 320 330 70 30 100 125 190 35 ...
##  $ vitamins: int  25 0 25 25 25 25 25 25 25 25 ...
##  $ shelf   : int  3 3 3 3 1 2 3 1 3 2 ...
##  $ weight  : num  1 1 1 1 1 1 1.33 1 1 1 ...
##  $ cups    : num  0.33 1 0.33 0.5 0.75 1 0.75 0.67 0.67 0.75 ...
##  $ rating  : num  68.4 34 59.4 93.7 29.5 ...
##  - attr(*, "na.action")= 'omit' Named int [1:3] 5 21 58
##   ..- attr(*, "names")= chr [1:3] "5" "21" "58"
```

Selecting the relevant columns for the Analysis

Let's consider columns 4 to 13 (calories to potassium) for clustering. Because the rest of the variables includes categorical values and also the last 3 columns cannot be used for clustering and find the best cluster

```
cereal_data2 <- cereal_data[, 4:12]
head(cereal_data2)
```

```
##   calories protein fat sodium fiber carbo sugars potass vitamins
## 1       70       4   1    130  10.0   5.0      6    280       25
## 2      120       3   5     15   2.0   8.0      8    135        0
## 3       70       4   1    260   9.0   7.0      5    320       25
## 4       50       4   0    140  14.0   8.0      0    330       25
## 6      110       2   2    180   1.5  10.5     10     70       25
## 7      110       2   0    125   1.0  11.0     14     30       25
```

Normalization

Normalization of the data is done to make all the observations having values comapreable to each other.

```r
normalized_data <- scale(cereal_data2)
head(normalized_data)
```

```
##     calories    protein        fat     sodium       fiber      carbo     sugars
## 1 -1.8659155  1.3817478  0.0000000 -0.3910227  3.22866747 -2.5001396 -0.2542051
## 2  0.6537514  0.4522084  3.9728810 -1.7804186 -0.07249167 -1.7292632  0.2046041
## 3 -1.8659155  1.3817478  0.0000000  1.1795987  2.81602258 -1.9862220 -0.4836096
## 4 -2.8737823  1.3817478 -0.9932203 -0.2702057  4.87924705 -1.7292632 -1.6306324
## 6  0.1498180 -0.4773310  0.9932203  0.2130625 -0.27881412 -1.0868662  0.6634132
## 7  0.1498180 -0.4773310 -0.9932203 -0.4514312 -0.48513656 -0.9583868  1.5810314
##       potass   vitamins
## 1  2.5605229 -0.1818422
## 2  0.5147738 -1.3032024
## 3  3.1248675 -0.1818422
## 4  3.2659536 -0.1818422
## 6 -0.4022862 -0.1818422
## 7 -0.9666308 -0.1818422
```

Hierarchical Clustering

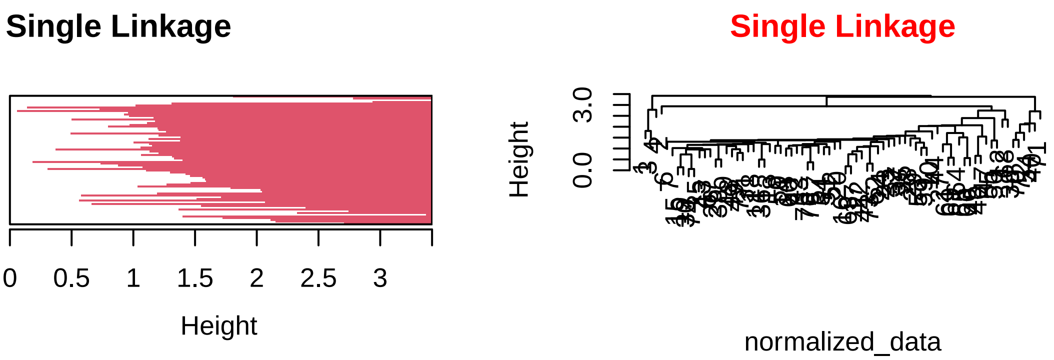
Performing hierarchical clustering using different linkage methods (single, complete, average, and Ward). Clustering is performed using the agnes() function and metric set at euclidean to find Euclidean distance.

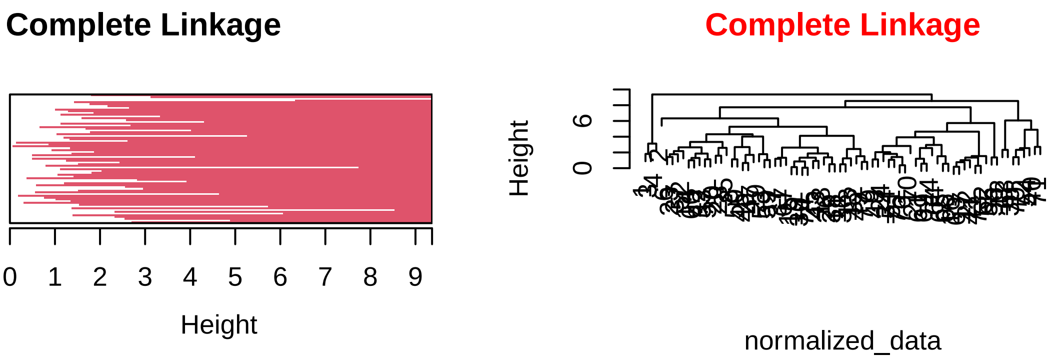
```r
single_linkage <- agnes(normalized_data, method = "single", metric = "euclidean")
complete_linkage <- agnes(normalized_data, method = "complete", metric = "euclidean")
average_linkage <- agnes(normalized_data, method = "average", metric = "euclidean")
ward_linkage <- agnes(normalized_data, method = "ward", metric = "euclidean")
```

dendogram Visualization

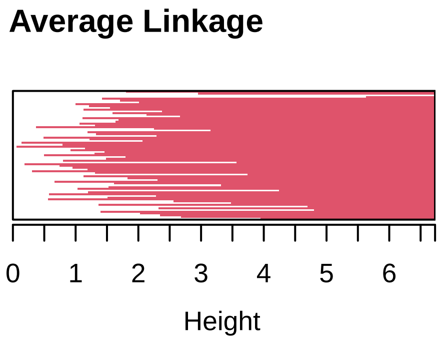
Clustering has been done. To visualize the results of clusters, let's visualize the results so that the comparison between the methods and number of clusters will be decided.

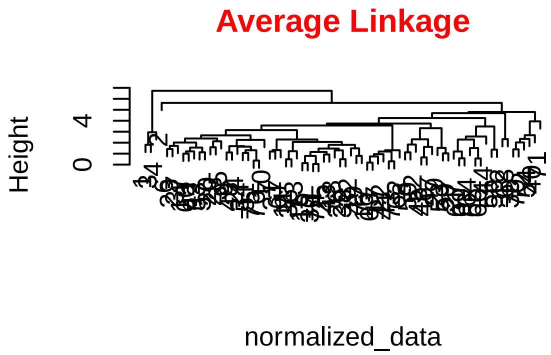
```r
# Plot dendrograms for each method
par(mfrow=c(2,2))
plot(single_linkage, main="Single Linkage", sub="", col.main="red")
plot(complete_linkage, main="Complete Linkage", sub="", col.main="red")
```

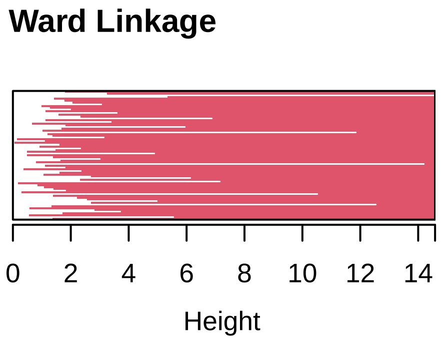


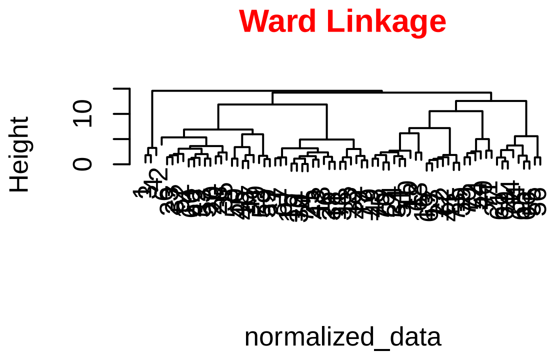


```r
plot(average_linkage, main="Average Linkage", sub="", col.main="red")
plot(ward_linkage, main="Ward Linkage", sub="", col.main="red")
```









## Extracting the information from the Clusters From the visualization we can see that max 3 clusters will be the best to cluster the whole data to compare the clusters setting the number of clusters to 3 in each method

```
# Single Linkage
single_clusters <- cutree(single_linkage, k = 3)  # Adjust 'k' based on the desired number of clusters
single_cluster_sizes <- table(single_clusters)
single_cluster_means <- aggregate(normalized_data, by = list(Cluster = single_clusters), mean)

# Complete Linkage
complete_clusters <- cutree(complete_linkage, k = 3)
complete_cluster_sizes <- table(complete_clusters)
complete_cluster_means <- aggregate(normalized_data, by = list(Cluster = complete_clusters), mean)

# Average Linkage
average_clusters <- cutree(average_linkage, k = 3)
average_cluster_sizes <- table(average_clusters)
average_cluster_means <- aggregate(normalized_data, by = list(Cluster = average_clusters), mean)

# Ward Linkage
ward_clusters <- cutree(ward_linkage, k = 3)
ward_cluster_sizes <- table(ward_clusters)
ward_cluster_means <- aggregate(normalized_data, by = list(Cluster = ward_clusters), mean)
# Printing the above few rows
print(head(single_cluster_means))
```

```
##   Cluster    calories     protein         fat       sodium       fiber
## 1       1 -2.20187108  1.38174776 -0.33107342  0.17279012  3.6413124
## 2       2  0.05678418 -0.07691407  0.03056062 -0.05924053 -0.1550206
## 3       3  0.48577362  0.14236189 -0.16553671  0.55537739 -0.1412658
##         carbo      sugars      potass   vitamins
## 1 -2.07187492 -0.78948236  2.98378133 -0.1818422
## 2  0.01410335  0.05284413 -0.13422250 -0.2853524
## 3  0.88315115 -0.17773687 -0.03781363  3.1822385
```

```
print(head(complete_cluster_means))
```

```
##   Cluster    calories    protein        fat      sodium      fiber
## 1       1 -2.20187108  1.3817478 -0.3310734  0.17279012  3.6413124
## 2       2  0.05383072 -0.1822392  0.0315308 -0.09856876 -0.1510907
## 3       3  0.40178472  0.9169781 -0.1241525  0.71143272 -0.1756529
##           carbo      sugars      potass   vitamins
## 1 -2.0718749196 -0.78948236  2.98378133 -0.1818422
## 2  0.0001102354  0.09172246 -0.13019146 -0.2886384
## 3  0.7760849908 -0.42625847 -0.09366023  2.3412183
```

```
print(head(average_cluster_means))
```

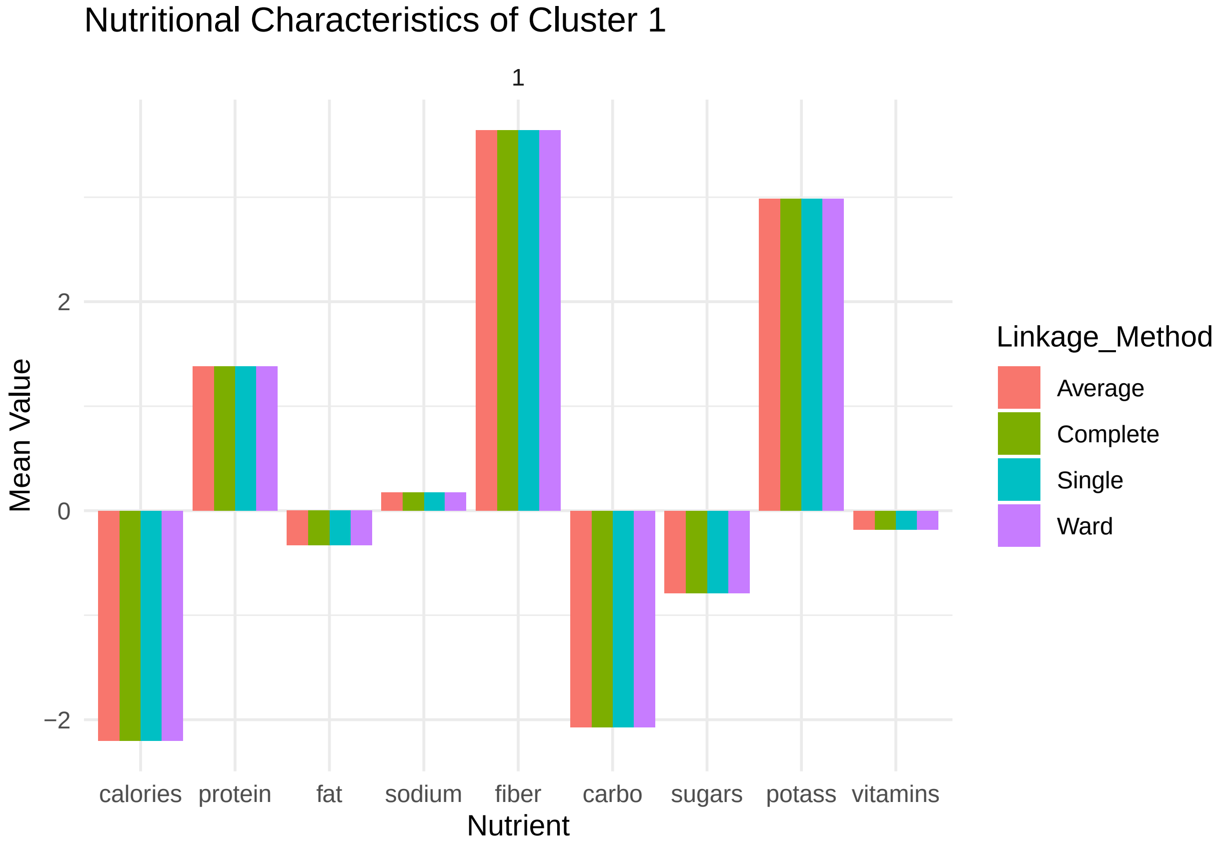
```
##   Cluster   calories     protein         fat      sodium       fiber      carbo
## 1       1 -2.2018711  1.38174776 -0.33107342  0.17279012  3.64131237 -2.0718749
## 2       2  0.6537514  0.45220836  3.97288104 -1.78041856 -0.07249167 -1.7292632
## 3       3  0.0850266 -0.06567788 -0.04256658  0.01802926 -0.15502065  0.1134984
##        sugars     potass    vitamins
## 1 -0.78948236  2.9837813 -0.18184220
## 2  0.20460407  0.5147738 -1.30320244
## 3  0.03091204 -0.1352303  0.02641041
```

```
print(ward_cluster_means)
```

```
##   Cluster   calories    protein        fat       sodium      fiber      carbo
## 1       1 -2.2018711  1.3817478 -0.3310734  0.172790124  3.6413124 -2.0718749
## 2       2  0.4599309 -0.1913189  0.4838765 -0.016180105 -0.1677174 -0.4312919
## 3       3 -0.3541153  0.1036311 -0.5586864  0.003520429 -0.1369674  0.7198753
##       sugars      potass   vitamins
## 1 -0.7894824  2.98378133 -0.1818422
## 2  0.7222349 -0.06404118 -0.2105950
## 3 -0.8062098 -0.20167931  0.2737104
```

Single Linkage

Cluster 1: In comparison to the general mean, this cluster has lower calories, higher protein, lower fat, lower salt, higher fiber, lower carbs, lower sugars, higher potassium, and lower vitamins. Cluster 2: The levels of most nutrients in this cluster are more in line with average. Cluster 3: When compared to the general mean, this cluster has more calories, a little higher protein, less fat, more salt, less fiber, more carbs, less sugar, less potassium, and more vitamins. ### Complete inkage Complete Linkage cluster interpretations are comparable to those of Single Linkage clusters. The specifics of cluster formation may account for the majority of the differences. ### Average Linkage Cluster 1: This cluster has lower calories, higher protein, lower fat, lower salt, greater fiber, lower carbs, lower sugars, higher potassium, and lower vitamins in comparison to the general mean. It is similar to Single and Complete Linkage. Cluster 2: In comparison to the global mean, this cluster has slightly greater salt, slightly higher fat, slightly higher protein, slightly higher calories, lower fiber, slightly higher carbs, slightly higher sugars, lower potassium, and slightly higher vitamins. Cluster 3: The levels of most nutrients in this cluster are more in line with average. ### Ward Linkage Cluster 1: In line with the other techniques, this cluster has lower mean values for calories, protein, fat, salt, fiber, carbs, sugars, potassium, and vitamins than the overall mean. Cluster 2: In comparison to the general mean, this cluster offers slightly more calories, less protein, fat, salt, fiber, carbs, sugars, potassium, and vitamins. It also provides slightly greater fat, protein, and calories. Cluster 3: In comparison to the general mean, this cluster has slightly lower calories, slightly lower protein, fat, and salt as well as greater fiber, slightly higher carbs, slightly higher sugars, slightly higher potassium, and slightly more vitamins. ## Selecting the Best Cluster It appears that Cluster 1 in the "Single Linkage," "Complete Linkage," "Average Linkage," and "Ward Linkage" methods generally matches the criteria of having low calories, high protein, low fat, high fiber, low carbohydrates, low sugars, high potassium, and high vitamins based on the descriptions given for the best cluster of cereals under each linkage method. As a result, the cluster that is designated as "Cluster 1" in all linking techniques appears to meet the predetermined requirements. It should be noted that the cluster labels might change based on how the clustering algorithm is specifically executed ### Visualizing the Cluster 1 with all the variables



The Cluste which represent tha mean values of nutritional characteristics like Low fats, high fiber, low calories, High vitamin can be consider as the best cluster of cereals that will be allowed to use in School's Canteen.

```
# Combine cluster means for Cluster 1 across all linkage methods
all_cluster1 <- rbind(cluster1_single, cluster1_complete, cluster1_average, cluster1_ward)
all_cluster1$Linkage_Method <- c("Single", "Complete", "Average", "Ward")

# Select relevant columns for the summary table
summary_table <- all_cluster1[, c("Linkage_Method", "calories", "protein", "fat", "fiber", "carbo", "su
# Print the summary table
print(summary_table)
```

```
##   Linkage_Method  calories  protein        fat    fiber     carbo     sugars
## 1         Single -2.201871 1.381748 -0.3310734 3.641312 -2.071875 -0.7894824
## 2       Complete -2.201871 1.381748 -0.3310734 3.641312 -2.071875 -0.7894824
## 3        Average -2.201871 1.381748 -0.3310734 3.641312 -2.071875 -0.7894824
## 4           Ward -2.201871 1.381748 -0.3310734 3.641312 -2.071875 -0.7894824
##     potass   vitamins
## 1 2.983781 -0.1818422
## 2 2.983781 -0.1818422
## 3 2.983781 -0.1818422
## 4 2.983781 -0.1818422
```