Assignment_5_ML

MadhusudhanMasineni

2022-04-07

Load Data Set and Libraries

```r
cereals <- read.csv('/Users/madhusudhanmasineni/Downloads/MSBA/Cereals.csv')
```

#Apply hierarchical clustering to the data using Euclidean distance to the normalized measurements. Use Agnes to compare the clustering from single linkage, complete linkage, average linkage, and Ward. Choose the best method.

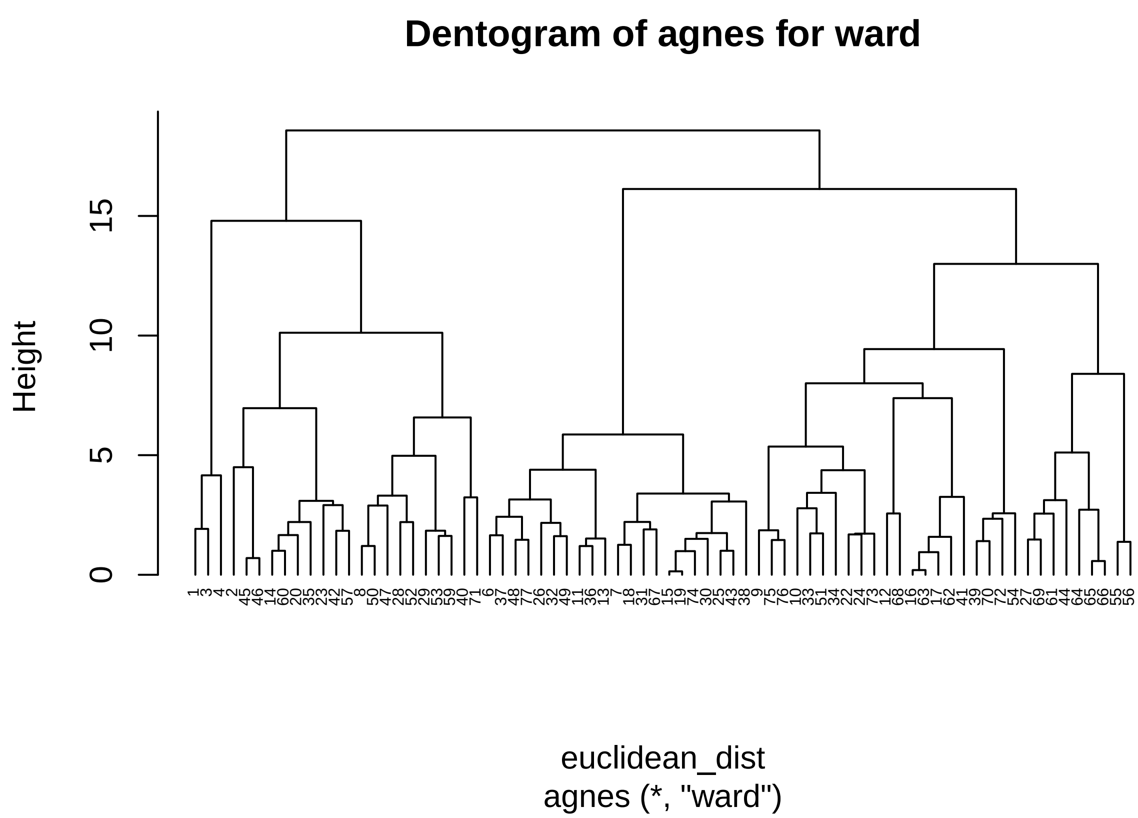
```r
cereals <- na.omit(cereals)
summary(cereals)
```

```
##      name               mfr                type              calories  
##  Length:74          Length:74          Length:74          Min.   : 50  
##  Class :character   Class :character   Class :character   1st Qu.:100  
##  Mode  :character   Mode  :character   Mode  :character   Median :110  
##                                                           Mean   :107  
##                                                           3rd Qu.:110  
##                                                           Max.   :160  
##     protein           fat        sodium          fiber            carbo      
##  Min.   :1.000   Min.   :0   Min.   :  0.0   Min.   : 0.000   Min.   : 5.00  
##  1st Qu.:2.000   1st Qu.:0   1st Qu.:135.0   1st Qu.: 0.250   1st Qu.:12.00  
##  Median :2.500   Median :1   Median :180.0   Median : 2.000   Median :14.50  
##  Mean   :2.514   Mean   :1   Mean   :162.4   Mean   : 2.176   Mean   :14.73  
##  3rd Qu.:3.000   3rd Qu.:1   3rd Qu.:217.5   3rd Qu.: 3.000   3rd Qu.:17.00  
##  Max.   :6.000   Max.   :5   Max.   :320.0   Max.   :14.000   Max.   :23.00  
##      sugars           potass          vitamins          shelf      
##  Min.   : 0.000   Min.   : 15.00   Min.   :  0.00   Min.   :1.000  
##  1st Qu.: 3.000   1st Qu.: 41.25   1st Qu.: 25.00   1st Qu.:1.250  
##  Median : 7.000   Median : 90.00   Median : 25.00   Median :2.000  
##  Mean   : 7.108   Mean   : 98.51   Mean   : 29.05   Mean   :2.216  
##  3rd Qu.:11.000   3rd Qu.:120.00   3rd Qu.: 25.00   3rd Qu.:3.000  
##  Max.   :15.000   Max.   :330.00   Max.   :100.00   Max.   :3.000  
##      weight           cups            rating     
##  Min.   :0.500   Min.   :0.2500   Min.   :18.04  
##  1st Qu.:1.000   1st Qu.:0.6700   1st Qu.:32.45  
##  Median :1.000   Median :0.7500   Median :40.25  
##  Mean   :1.031   Mean   :0.8216   Mean   :42.37  
##  3rd Qu.:1.000   3rd Qu.:1.0000   3rd Qu.:50.52  
##  Max.   :1.500   Max.   :1.5000   Max.   :93.70  
```

```r
cereals <- cereals[,4:16]
cereals <- scale(cereals, center = T, scale = T)
set.seed(64060)

euclidean_dist <- dist(cereals, method = "euclidean")
method <- c( "average", "single", "complete", "ward")
names(method) <- c( "average", "single", "complete", "ward")

values <- function(x) {
  agnes(euclidean_dist, method = x)$ac
}

map_dbl(method, values)
```

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

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

#From the result, the agglomerative coefficient obtained by Ward's method is the largest.
#Let's take a peek at the dendogram.

ward <- agnes(euclidean_dist, method = "ward")
pltree(ward, cex = .5, hang = -1, main = "Dentogram of agnes for ward")
```

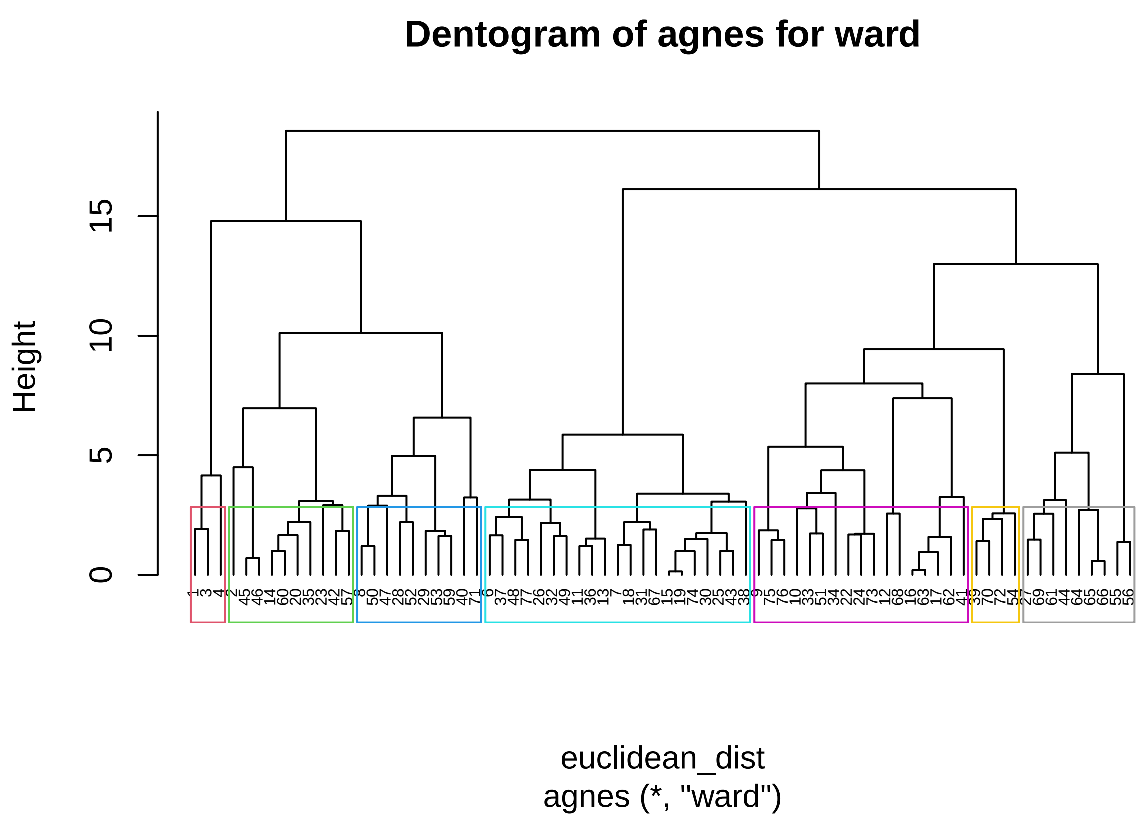


How many clusters would you choose?

```
ward <- agnes(euclidean_dist, method = "ward")
pltree(ward, cex = .5, hang = -1, main = "Dentogram of agnes for ward")
clusters <- NbClust(cereals, distance = "euclidean", min.nc = 5, max.nc = 10, method = "ward.D", index =
clusters$Best.nc
```

```
## Number_clusters     Value_Index
##          7.0000          0.2604
```

```
# The best number of clusters, the best fits is with K=7
rect.hclust(ward, k = 7, border = 2:10)
```




```
cluster_comp <- cutree(ward, k = 7)
temp_var <- cbind(as.data.frame(cbind(cereals,cluster_comp)))
```

Comment on the structure of the clusters and on their stability. Hint: To check stability, partition the data and see how well clusters formed based on one part apply to the other part. To do this: Cluster partition A Use the cluster centroids from A to assign each record in partition B (each record is assigned to the cluster with the closest centroid). Assess how consistent the cluster assignments are compared to the assignments based on all the data

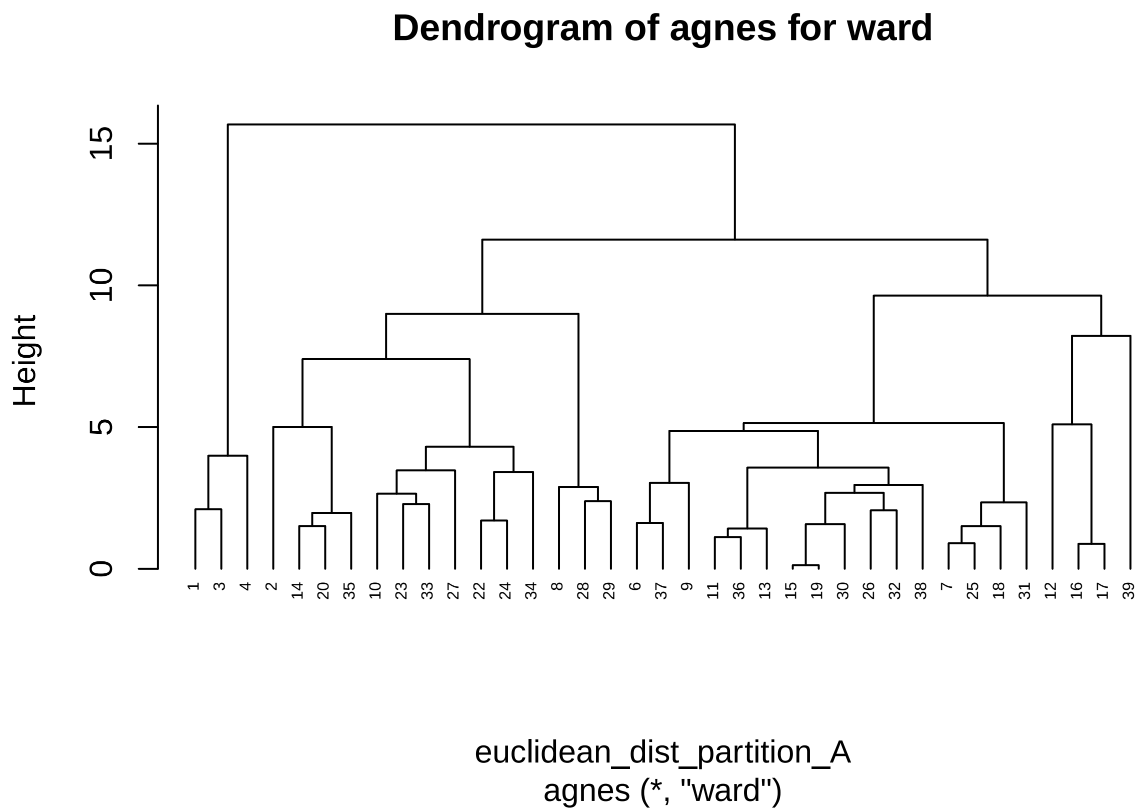
```r
cereals <- read.csv('/Users/madhusudhanmasineni/Downloads/MSBA/Cereals.csv')
sum(is.na(cereals))
```

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

```r
cereals <- na.omit(cereals)
cereals <- cereals[,4:16]
# Creating Partitions for into two data A, B
clust_partition_A <- cereals[1:37,]
clust_partition_B <- cereals[38:74,]
clust_partition_A <- scale(clust_partition_A, center = T, scale = T)
clust_partition_B <- scale(clust_partition_B, center = T, scale = T)
euclidean_dist_partition_A <- dist(clust_partition_A, method = "euclidean")
names(method) <- c( "average", "single", "complete", "ward")
values1 <- function(x) {
  agnes(euclidean_dist_partition_A, method = x)$ac
}
map_dbl(method, values)
```

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

```r
#The agglomerative coefficient obtained by Ward's method is the largest.
#Let's take a peek at the dendogram.
set.seed(64060)
ward_partition_A <- agnes(euclidean_dist_partition_A, method = "ward")
pltree(ward_partition_A, cex = 0.5, hang = -1, main = "Dendrogram of agnes for ward")
```




```r
clust_comp_partition_A <- cutree(ward_partition_A, k = 7)
result<-as.data.frame(cbind(clust_partition_A,clust_comp_partition_A))
#result[result$clust_comp_partition_A==1,]
#center1<-colMeans(result[result$clust_comp_partition_A==1,])
klust <- 1:7
for (i in klust) {
  assign(paste0("center_",i), colMeans(result[result$clust_comp_partition_A==i,]))
}
centroids <- rbind(center_1,center_2,center_3,center_4,center_5,center_6,center_7)
combined <- as.data.frame(rbind(centroids[,-14], clust_partition_B))
temp_var1<-get_dist(combined)
temp_var2<-as.matrix(temp_var1)
data1<-data.frame(data=seq(1,nrow(clust_partition_B),1),clusters=rep(0,nrow(clust_partition_B)))
for(i in 1:nrow(clust_partition_B))
{
  data1[i,2]<-which.min(temp_var2[i+7,1:7])
}
cbind(temp_var$cluster_comp[38:74],data1$clusters)
```

```
##       [,1] [,2]
##  [1,]    4    4
##  [2,]    5    6
##  [3,]    2    2
##  [4,]    3    3
##  [5,]    6    5
```

```
##  [6,]    2    2
##  [7,]    2    2
##  [8,]    4    4
##  [9,]    3    3
## [10,]    3    3
## [11,]    4    4
## [12,]    5    5
## [13,]    4    2
## [14,]    4    4
## [15,]    7    5
## [16,]    6    5
## [17,]    6    5
## [18,]    2    5
## [19,]    4    4
## [20,]    2    2
## [21,]    6    5
## [22,]    5    6
## [23,]    5    6
## [24,]    6    5
## [25,]    6    5
## [26,]    6    5
## [27,]    3    3
## [28,]    5    6
## [29,]    6    5
## [30,]    7    7
## [31,]    4    4
## [32,]    7    5
## [33,]    5    5
## [34,]    3    3
## [35,]    5    5
## [36,]    5    6
## [37,]    3    3
```

```
table(temp_var$cluster_comp[38:74]==data1$clusters)
```

```
##
## FALSE  TRUE
##    17    20
```

```
#We get 17 FALSE and 20 TRUE, indicating that the model is only partly stable.
```

Q) The elementary public schools would like to choose a set of cereals to include in their daily cafeterias. Every day a different cereal is offered, but all cereals should support a healthy diet. For this goal, you are requested to find a cluster of "healthy cereals." Should the data be normalized? If not, how should they be used in the cluster analysis?

A) Normalizing the data would not be suitable in this scenario because the nutritional information for cereal is normalized based on the sample of cereal being evaluated. As a result, the collected data could only contain cereals with extremely high sugar content and very little fiber, iron, and other nutritional data. It's impossible to say how much nourishment the cereal will provide a child once it's been normalized throughout the sample set. We may infer that a cereal with an iron content of 0.999 means it contains virtually all of the nutrional iron a child need; yet, it could simply be the best of the worst in the sample set (having nearly no nutrional value), convert it to a ratio of daily recommended calories, fiber, carbohydrates, and other nutrients for a child. This would allow analysts to make more informed decisions on clusters during review, while also preventing a few larger variables from overriding the distance estimates. When looking at the clusters, an analyst may look at the cluster average to see what percentage of a student's daily needed nutrition would come from XX cereal. This would enable the employees to make well-informed selections regarding which "healthy" cereal clusters to select.