FML_Assignment5

Namrah

2023-12-01

```
library(cluster)
library(caret)

## Loading required package: ggplot2

## Loading required package: lattice

library(dendextend)

##
## ---------------------
## Welcome to dendextend version 1.17.1
## Type citation('dendextend') for how to cite the package.
##
## Type browseVignettes(package = 'dendextend') for the package vignette.
## The github page is: https://github.com/talgalili/dendextend/
##
## Suggestions and bug-reports can be submitted at:
https://github.com/talgalili/dendextend/issues
## You may ask questions at stackoverflow, use the r and dendextend tags:
##   https://stackoverflow.com/questions/tagged/dendextend
##
##  To suppress this message use:
suppressPackageStartupMessages(library(dendextend))
## ---------------------

##
## Attaching package: 'dendextend'

## The following object is masked from 'package:stats':
##
##     cutree

library(factoextra)

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

c<-read.csv("/Users/Namrah/Downloads/Cereals.csv")
c<- na.omit(c)
head(c)

##                          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
## 6   Apple_Cinnamon_Cheerios   G    C      110       2   2    180   1.5
10.5
## 7               Apple_Jacks   K    C      110       2   0    125   1.0
11.0
##   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
## 6     10     70       25     1      1 0.75 29.50954
## 7     14     30       25     2      1 1.00 33.17409
```

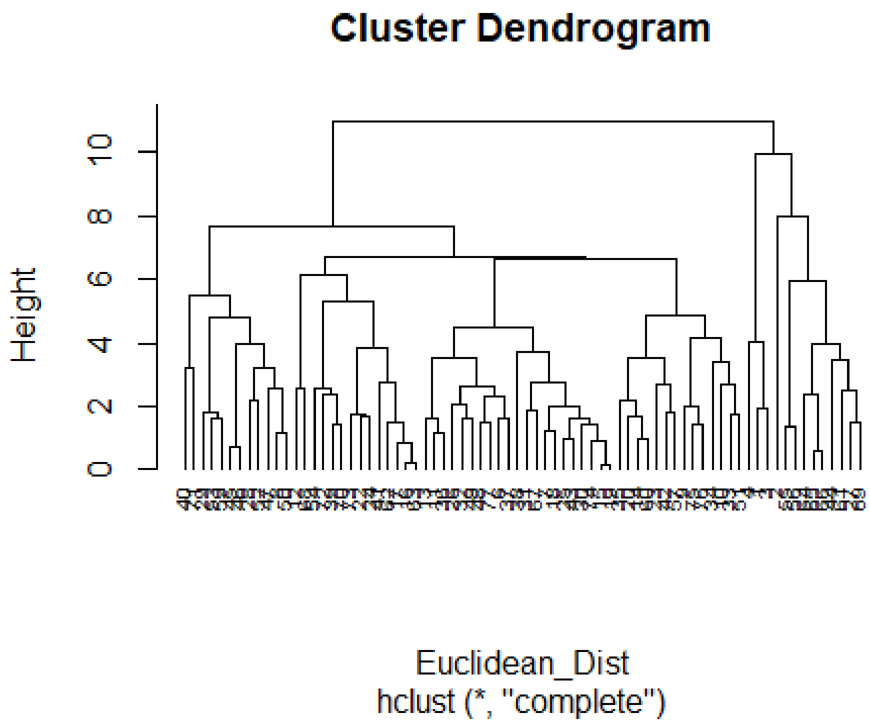
#Normalizing the dataset

```
c<- c[4:16]
c <- scale(c,center = TRUE,scale = TRUE)
head(c)
```

```
##     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      shelf     weight       cups     rating
## 1  2.5605229 -0.1818422  0.9419715 -0.2008324 -2.0856582  1.8549038
## 2  0.5147738 -1.3032024  0.9419715 -0.2008324  0.7567534 -0.5977113
## 3  3.1248675 -0.1818422  0.9419715 -0.2008324 -2.0856582  1.2151965
## 4  3.2659536 -0.1818422  0.9419715 -0.2008324 -1.3644493  3.6578436
## 6 -0.4022862 -0.1818422 -1.4616799 -0.2008324 -0.3038480 -0.9165248
## 7 -0.9666308 -0.1818422 -0.2598542 -0.2008324  0.7567534 -0.6553998
```

#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

```
Euclidean_Dist <- dist(c, method = "euclidean")
# Hierarchical clustering using Complete Linkage
hc1 <- hclust(Euclidean_Dist, method = "complete" )
# Plot the obtained dendrogram
plot(hc1, cex = 0.6, hang = -1)
```



```
round(hc1$height, 3)

##  [1]  0.143  0.196  0.575  0.698  0.828  0.904  1.003  1.004  1.201  1.203
## [11]  1.254  1.378  1.408  1.421  1.454  1.463  1.474  1.517  1.608  1.611
## [21]  1.616  1.625  1.650  1.687  1.692  1.720  1.730  1.795  1.839  1.897
## [31]  1.919  1.982  2.015  2.046  2.203  2.224  2.339  2.381  2.394  2.522
## [41]  2.563  2.574  2.579  2.668  2.682  2.734  2.776  2.787  3.229  3.236
## [51]  3.385  3.451  3.510  3.535  3.717  3.866  3.957  4.005  4.031  4.168
## [61]  4.456  4.779  4.839  5.342  5.488  5.920  6.169  6.669  6.731  7.650
## [71]  7.964  9.979 10.984
```

Compute with agnes and with different linkage methods

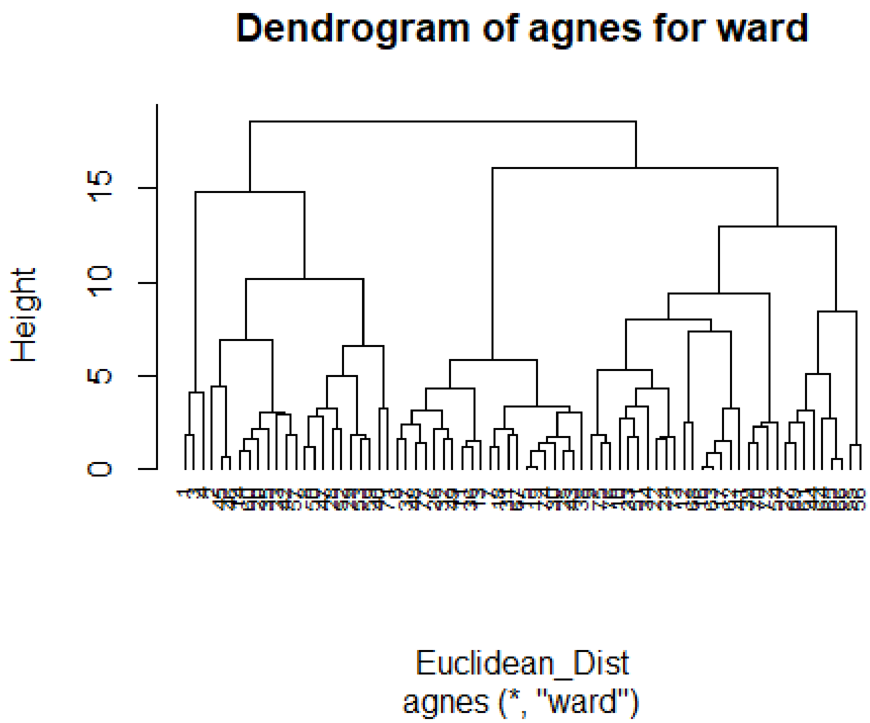
```
hc_single <- agnes(c, method = "single")
print(hc_single$ac)

## [1] 0.6067859
```

```r
hc_complete <- agnes(c, method = "complete")
print(hc_complete$ac)

## [1] 0.8353712

hc_average <- agnes(c, method = "average")
print(hc_average$ac)

## [1] 0.7766075

hc_ward <- agnes(c, method = "ward")
print(hc_ward$ac)

## [1] 0.9046042
```

#The agglomerative coefficient obtained by Ward's method is the largest.

#visualizing the dendrogram

```r
hc_Ward <- agnes(Euclidean_Dist, method = "ward")
pltree(hc_Ward, cex = 0.6, hang = -1, main = "Dendrogram of agnes for ward")
```

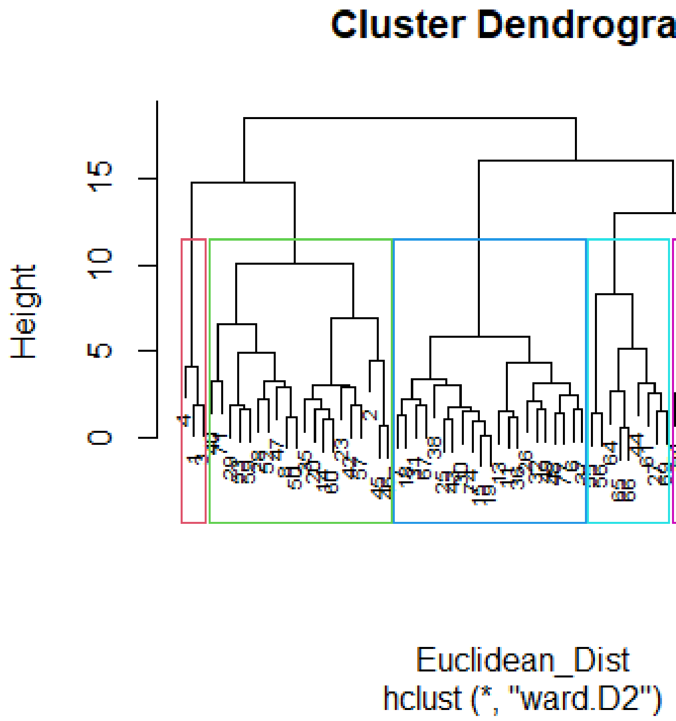


#How many clusters would you choose?

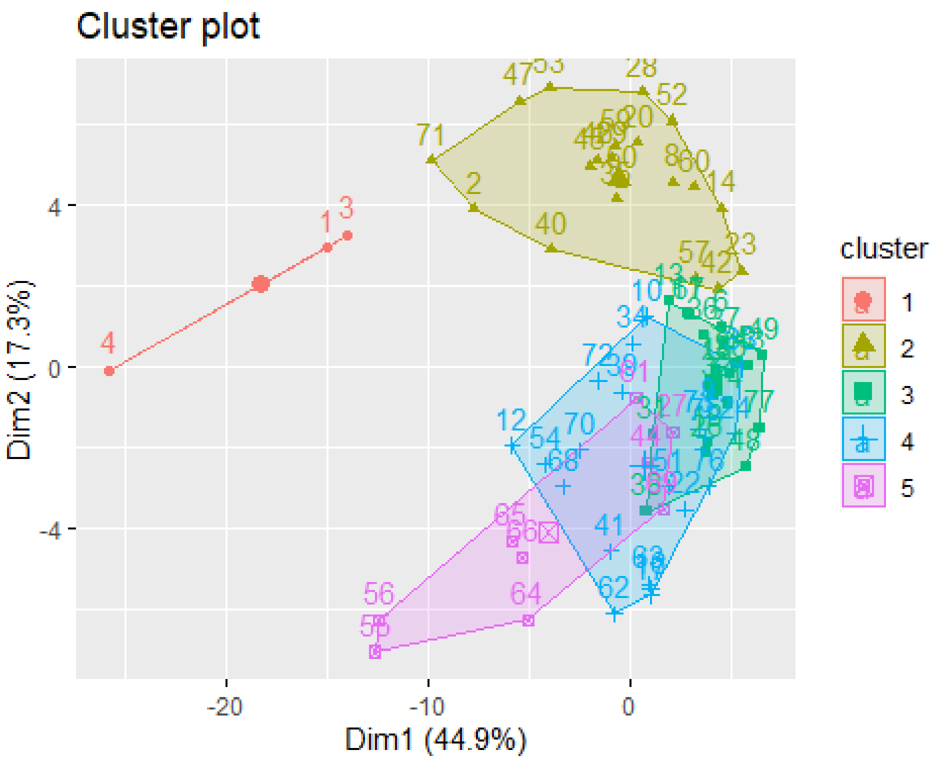
```r
#The largest difference in height can be used to determine the k value, hence
K =5 is the best option.
hc_Ward <- hclust(Euclidean_Dist,method = "ward.D2")
clust_comp <- cutree(hc_Ward, k=5)
table(clust_comp)
```

```
## clust_comp
##  1  2  3  4  5
##  3 20 21 21  9
```

```
plot(hc_Ward,cex=0.6)
rect.hclust(hc_Ward, k = 5, border = 2:10,)
```



```
Temp <- cbind(as.data.frame(cbind(c,clust_comp)))
#Visualizing the clusters in Scatter plot
fviz_cluster(list(data=Euclidean_Dist, cluster = clust_comp))
```



#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?

```
Healthy_cereal <- na.omit(read.csv("/Users/Namrah/Downloads/Cereals.csv"))
Healthy_cereal<- cbind(Healthy_cereal,clust_comp)
mean(Healthy_cereal[Healthy_cereal$clust_comp==1,"rating"])

## [1] 73.84446

mean(Healthy_cereal[Healthy_cereal$clust_comp==2,"rating"])

## [1] 38.26161

mean(Healthy_cereal[Healthy_cereal$clust_comp==3,"rating"])

## [1] 28.84825

mean(Healthy_cereal[Healthy_cereal$clust_comp==4,"rating"])

## [1] 46.46513

mean(Healthy_cereal[Healthy_cereal$clust_comp==5,"rating"])

## [1] 63.0184
```

It is clear that Cluster1 has maximum rating (73.84446), therefore we’ll choose it as a healthy cereal.