Mderangu

2023-12-04

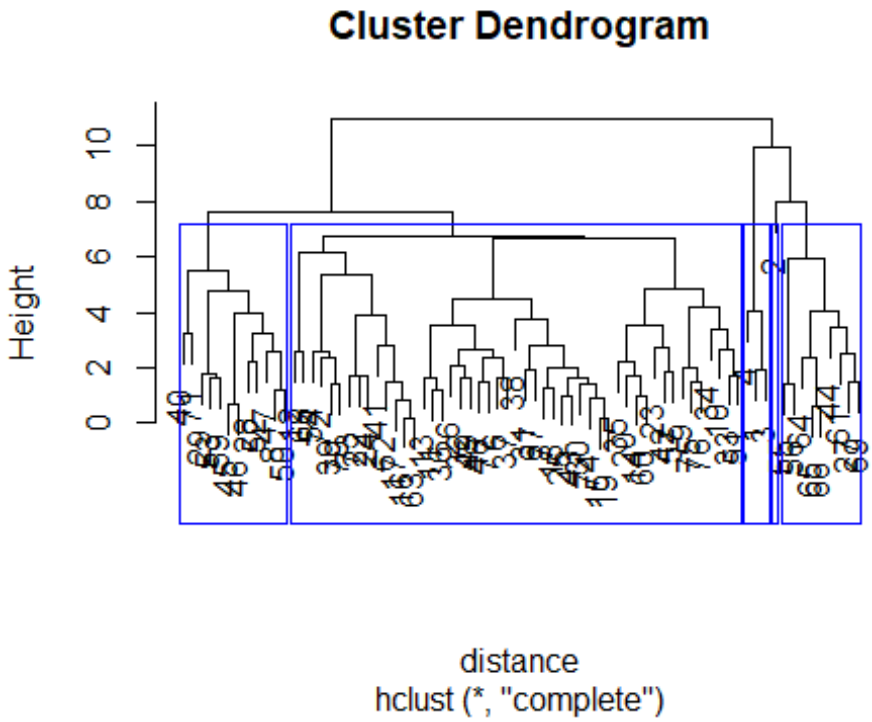
```
#Loading the required packages and reading the cereals file.

library(factoextra)

## Warning: package 'factoextra' was built under R version 4.2.3

## Loading required package: ggplot2

## Warning: package 'ggplot2' was built under R version 4.2.3

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

library(dendextend)

## Warning: package 'dendextend' was built under R version 4.2.3

##
## ---------------------
## 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(cluster)
library(tidyverse)

## Warning: package 'tidyverse' was built under R version 4.2.3
```

```
## Warning: package 'tibble' was built under R version 4.2.3

## Warning: package 'tidyr' was built under R version 4.2.3

## Warning: package 'readr' was built under R version 4.2.3

## Warning: package 'purrr' was built under R version 4.2.3

## Warning: package 'dplyr' was built under R version 4.2.3

## Warning: package 'stringr' was built under R version 4.2.3

## Warning: package 'forcats' was built under R version 4.2.3

## Warning: package 'lubridate' was built under R version 4.2.3

## ── Attaching core tidyverse packages ──────────────────────── tidyverse
2.0.0 ──
## ✓ dplyr     1.1.3     ✓ readr     2.1.4
## ✓ forcats   1.0.0     ✓ stringr   1.5.0
## ✓ lubridate 1.9.2     ✓ tibble    3.2.1
## ✓ purrr     1.0.2     ✓ tidyr     1.3.0

## ── Conflicts ────────────────────────────────────────
tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all
conflicts to become errors

library(readr)
cereals <- read_csv("D:/Manisha/ASS-5/Cereals.csv")

## Rows: 77 Columns: 16
## ── Column specification
──────────────────────────────────────────────────────
## Delimiter: ","
## chr  (3): name, mfr, type
## dbl (13): calories, protein, fat, sodium, fiber, carbo, sugars, potass,
vita...
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this
message.

View(cereals)
numericaldata = data.frame(cereals[,4:16])
spec(cereals)

## cols(
##   name = col_character(),
##   mfr = col_character(),
##   type = col_character(),
```

```
##   calories = col_double(),
##   protein = col_double(),
##   fat = col_double(),
##   sodium = col_double(),
##   fiber = col_double(),
##   carbo = col_double(),
##   sugars = col_double(),
##   potass = col_double(),
##   vitamins = col_double(),
##   shelf = col_double(),
##   weight = col_double(),
##   cups = col_double(),
##   rating = col_double()
## )

#Data prepocessing - Normalize the measurements to ensure that variables with
different scales do not disproportionately influence the clustering.
missing = na.omit(numericaldata)

#normalizing and scaling the data
normalise = scale(missing)

#measuring the distance using the euclidian distance and computing the
dissimilarity matrix
distance = dist(normalise, method = "euclidian")

#Hierarchical clustering is a method of cluster analysis which seeks to build
a hierarchy of clusters.performing hierarchial clustering using complete
linkage and representing in plot.Hierarchical clustering can be divided into
two main types: agglomerative and divisive.
hierarchial_clustering = hclust(distance,method = "complete")
plot(hierarchial_clustering)

#rounding off the decimals
round(hierarchial_clustering$height, 5)

##  [1]  0.14315  0.19616  0.57455  0.69797  0.82806  0.90354  1.00349
1.00405
##  [9]  1.20088  1.20325  1.25377  1.37772  1.40832  1.42074  1.45357
1.46326
## [17]  1.47393  1.51732  1.60758  1.61061  1.61576  1.62451  1.65040
1.68700
## [25]  1.69233  1.72023  1.73046  1.79488  1.83892  1.89651  1.91874
1.98210
## [33]  2.01539  2.04628  2.20301  2.22359  2.33886  2.38139  2.39401
2.52225
## [41]  2.56304  2.57436  2.57921  2.66832  2.68196  2.73397  2.77641
2.78680
## [49]  3.22925  3.23622  3.38498  3.45069  3.51004  3.53516  3.71686
3.86639
```

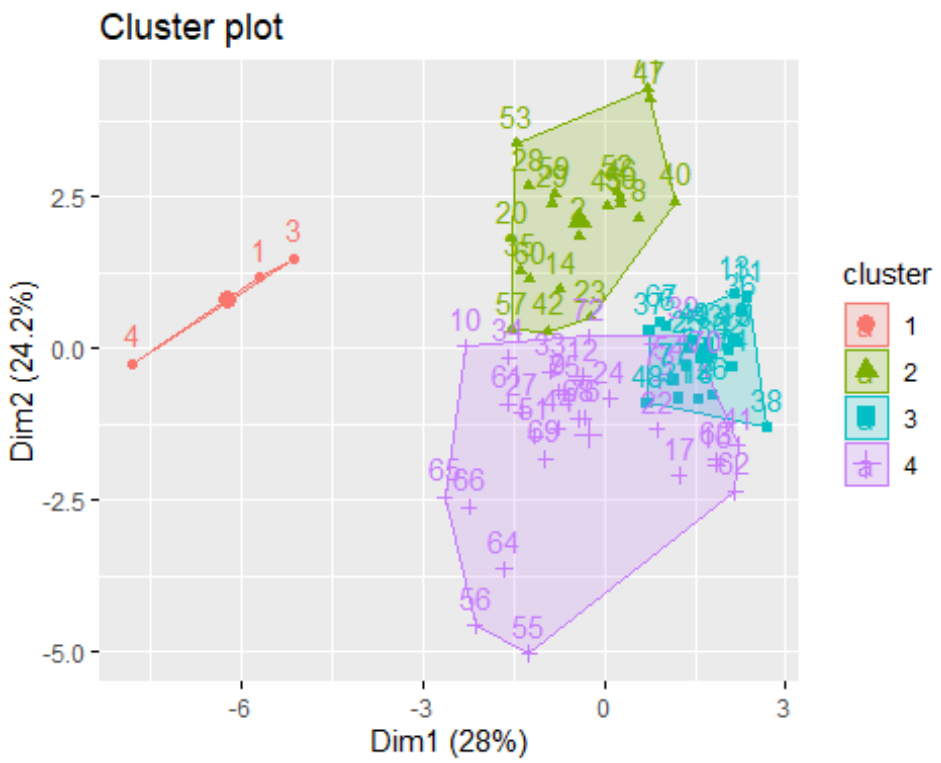
```
## [57]  3.95737  4.00466  4.03105  4.16760  4.45568  4.77888  4.83870
5.34168
## [65]  5.48793  5.91994  6.16858  6.66865  6.73123  7.64963  7.96381
9.97867
## [73] 10.98389
```

```
#Look at the dendrogram plot and observe where the blue rectangles are drawn.
Each rectangle corresponds to a cluster, and the number of rectangles
indicates the specified number of clusters (in this case, 5).

plot(hierarchial_clustering)
rect.hclust(hierarchial_clustering,k = 5, border = "blue")
```




```
#The agnes function in R is part of the cluster package, and it is used to
perform agglomerative hierarchical clustering. Agglomerative hierarchical
clustering is a bottom-up approach where individual data points start as
their own clusters and are successively merged based on their pairwise
similarity until a single cluster containing all the data points is formed.

# Data matrix, data frame, or dissimilarity matrix
# Metric for calculating dissimilarities: "euclidean" or "manhattan"
# Standardize measurements if TRUE
# Clustering method: "average", "single", "complete", "ward"


#performing clustering using AGNES
HCsingle = agnes(normalise, method = "single")
```

```r
HCcomplete = agnes(normalise, method = "complete")
HCaverage = agnes(normalise, method = "average")
HCward = agnes(normalise, method = "ward")

#performing clustering using AGNES
HCsingle = agnes(normalise, method = "single")
HCcomplete = agnes(normalise, method = "complete")
HCaverage = agnes(normalise, method = "average")
HCward = agnes(normalise, method = "ward")

#using the ward method for hierarchial clustering and Ward's method minimizes
the variance within each cluster. It selects the pair of clusters to merge
such that the increase in the total within-cluster variance is minimized.
HC1 <- hclust(distance, method = "ward.D2" )
subgrp <- cutree(HC1, k = 4)
table(subgrp)

## subgrp
##  1  2  3  4
##  3 20 21 30

cereals <- as.data.frame(cbind(normalise,subgrp))

#It is used for visualizing clustering results obtained from various
clustering algorithms. visualising the results on scatterplot
fviz_cluster(list(data = normalise, cluster = subgrp))
```



```r
#choosing the healthy cereal cluster
data <- cereals
data_omit <- na.omit(data)
Clust <- cbind(data_omit, subgrp)
Clust[Clust$subgrp==1,]
```

```
##    calories  protein        fat     sodium    fiber     carbo     sugars
## 1 -1.865915 1.381748  0.0000000 -0.3910227 3.228667 -2.500140 -0.2542051
## 3 -1.865915 1.381748  0.0000000  1.1795987 2.816023 -1.986222 -0.4836096
## 4 -2.873782 1.381748 -0.9932203 -0.2702057 4.879247 -1.729263 -1.6306324
##     potass   vitamins     shelf     weight      cups   rating subgrp
subgrp
## 1 2.560523 -0.1818422 0.9419715 -0.2008324 -2.085658 1.854904      1
1
## 3 3.124867 -0.1818422 0.9419715 -0.2008324 -2.085658 1.215196      1
1
## 4 3.265954 -0.1818422 0.9419715 -0.2008324 -1.364449 3.657844      1
1
```

```r
Clust[Clust$subgrp==2,]
```

```
##      calories    protein        fat      sodium       fiber       carbo
## 2   0.6537514  0.4522084  3.9728810 -1.78041856 -0.07249167 -1.72926320
## 8   1.1576848  0.4522084  0.9932203  0.57551356 -0.07249167  0.84032469
## 14  0.1498180  0.4522084  0.9932203 -0.27020566 -0.07249167 -0.44446926
## 20  0.1498180  0.4522084  1.9864405 -0.27020566  0.75279812 -1.21534562
## 23 -0.3541153 -0.4773310  0.0000000 -0.27020566 -0.07249167 -0.95838683
## 28  0.6537514  0.4522084  0.9932203 -0.02857160  1.16544301 -0.70142805
## 29  0.6537514  0.4522084 -0.9932203  0.93796466  1.16544301 -0.18751047
## 35  0.6537514  0.4522084  1.9864405 -1.05551637  0.34015322 -0.44446926
## 40  1.6616182  0.4522084  0.0000000  0.09224544 -0.07249167  1.35424227
## 42 -0.3541153  1.3817478  0.9932203 -0.14938863 -0.07249167 -0.70142805
## 45  2.1655516  1.3817478  1.9864405 -0.81388230  0.34015322  0.32640711
## 46  2.1655516  1.3817478  1.9864405 -0.14938863  0.34015322  0.32640711
## 47  2.6694849  0.4522084  0.9932203 -0.14938863  0.34015322  0.58336590
## 50  1.6616182  0.4522084  0.9932203  0.69633060  0.34015322  1.61120105
## 52  1.1576848  0.4522084  0.9932203  0.09224544 -0.27881412 -0.31598986
## 53  0.6537514  0.4522084  0.0000000  0.45469653  1.57808790 -0.95838683
## 57 -0.3541153  1.3817478  0.0000000 -0.33061417 -0.07249167 -0.18751047
## 59  0.6537514  0.4522084  0.0000000  0.57551356  1.16544301 -0.18751047
## 60 -0.3541153  0.4522084  0.9932203 -0.27020566  0.13383078 -1.08686623
## 71  1.6616182  0.4522084  0.0000000  0.33387950  0.75279812  0.06944832
##         sugars      potass   vitamins      shelf     weight       cups
## 2   0.20460407  0.51477378 -1.3032024  0.9419715 -0.2008324  0.7567534
## 8   0.20460407  0.02097226 -0.1818422  0.9419715  1.9501886 -0.3038480
## 14 -0.02480049  0.09151534 -0.1818422  0.9419715 -0.2008324 -1.3644493
## 20 -0.02480049  0.86748914 -0.1818422  0.9419715 -0.2008324 -1.3644493
## 23  0.66341318  0.30314456 -0.1818422  0.9419715 -0.2008324 -0.3038480
## 28  0.66341318  1.43183372 -0.1818422  0.9419715  1.4287290 -0.6432404
## 29  1.12222230  1.29074758 -0.1818422  0.9419715  1.9501886 -0.6432404
```

```
## 35 -0.71301417  0.02097226 -0.1818422  0.9419715 -0.2008324 -2.0856582
## 40  0.43400862 -0.04957081  3.1822385  0.9419715  1.7546413 -0.3038480
## 42 -0.25420505 -0.04957081 -0.1818422 -0.2598542 -0.2008324 -0.6432404
## 45  0.89281774  1.00857529 -0.1818422  0.9419715 -0.2008324  0.7567534
## 46  0.89281774  1.00857529 -0.1818422  0.9419715 -0.2008324  0.7567534
## 47  1.35162686  0.86748914 -0.1818422  0.9419715  3.0582904 -0.6432404
## 50 -0.02480049  0.44423070 -0.1818422  0.9419715  1.9501886 -0.6432404
## 52  0.66341318  0.30314456 -0.1818422  0.9419715  1.4287290 -1.3644493
## 53  1.58103142  2.27835060 -0.1818422  0.9419715  1.9501886 -0.6432404
## 57 -0.25420505  0.16205841 -0.1818422  0.9419715 -0.2008324 -1.3644493
## 59  1.12222230  1.99617831 -0.1818422 -0.2598542  1.9501886 -0.3038480
## 60  0.20460407  0.58531685 -0.1818422  0.9419715 -0.2008324 -1.3644493
## 71  1.58103142  1.85509216  3.1822385  0.9419715  3.0582904  0.7567534
##         rating subgrp subgrp
## 2  -0.59771126      2      2
## 8  -0.38002951      2      2
## 14 -0.14048876      2      2
## 20 -0.13702824      2      2
## 23 -0.44147911      2      2
## 28 -0.10366038      2      2
## 29 -0.09664548      2      2
## 35  0.24511896      2      2
## 40 -0.42043579      2      2
## 42  0.21065609      2      2
## 45 -0.37302488      2      2
## 46 -0.58658904      2      2
## 47 -0.85924775      2      2
## 50 -0.11967375      2      2
## 52 -0.84945049      2      2
## 53 -0.32287913      2      2
## 57  0.50878106      2      2
## 59 -0.22179377      2      2
## 60 -0.19014120      2      2
## 71 -0.98185009      2      2
```

```
Clust[Clust$subgrp==3,]
```

```
##      calories    protein        fat     sodium       fiber       carbo
## 6   0.1498180 -0.4773310  0.9932203  0.2130625 -0.27881412 -1.08686623
## 7   0.1498180 -0.4773310 -0.9932203 -0.4514312 -0.48513656 -0.95838683
## 11  0.6537514 -1.4068705  0.9932203  0.6963306 -0.89778146 -0.70142805
## 13  0.6537514 -1.4068705  1.9864405  0.5755136 -0.89778146 -0.44446926
## 15  0.1498180 -1.4068705  0.0000000  0.2130625 -0.89778146 -0.70142805
## 18  0.1498180 -1.4068705 -0.9932203 -0.8742908 -0.48513656 -0.44446926
## 19  0.1498180 -1.4068705  0.0000000  0.2130625 -0.89778146 -0.70142805
## 25  0.1498180 -0.4773310  0.0000000 -0.4514312 -0.48513656 -0.95838683
## 26  0.1498180 -1.4068705 -0.9932203  0.4546965 -0.48513656 -0.18751047
## 30  0.1498180 -1.4068705  0.0000000 -0.3306142 -0.89778146 -0.44446926
## 31 -0.3541153 -0.4773310 -0.9932203 -1.4179675 -0.89778146 -0.95838683
## 32  0.1498180 -1.4068705  0.0000000  1.4212328 -0.89778146  0.06944832
```

```
## 36  0.6537514 -1.4068705  0.9932203  0.6963306 -0.48513656 -0.70142805
## 37  0.1498180  0.4522084  0.0000000  1.0587817 -0.27881412 -0.82990744
## 38  0.1498180 -1.4068705 -0.9932203  0.2130625 -0.89778146 -0.18751047
## 43  0.1498180 -0.4773310  0.0000000  0.2130625 -0.89778146 -0.70142805
## 48 -0.3541153 -0.4773310  0.0000000  0.6963306 -0.07249167  0.06944832
## 49  0.6537514 -0.4773310  0.0000000  0.3338795 -0.89778146  0.06944832
## 67  0.1498180 -0.4773310  0.0000000 -1.1159249 -0.48513656 -1.47230441
## 74  0.1498180 -1.4068705  0.0000000 -0.2702057 -0.89778146 -0.44446926
## 77  0.1498180 -0.4773310  0.0000000  0.4546965 -0.48513656  0.32640711
##        sugars     potass   vitamins      shelf     weight       cups
rating
## 6   0.6634132 -0.4022862 -0.1818422 -1.4616799 -0.2008324 -0.3038480 -
0.9165248
## 7   1.5810314 -0.9666308 -0.1818422 -0.2598542 -0.2008324  0.7567534 -
0.6553998
## 11  1.1222223 -0.8960877 -0.1818422 -0.2598542 -0.2008324 -0.3038480 -
1.7336066
## 13  0.4340086 -0.7550015 -0.1818422 -0.2598542 -0.2008324 -0.3038480 -
1.6067177
## 15  1.3516269 -0.6139154 -0.1818422 -0.2598542 -0.2008324  0.7567534 -
1.3991551
## 18  1.1222223 -1.1077169 -0.1818422 -0.2598542 -0.2008324  0.7567534 -
0.4695120
## 19  1.3516269 -0.4728292 -0.1818422 -0.2598542 -0.2008324  0.7567534 -
1.4233777
## 25  1.3516269 -0.9666308 -0.1818422 -0.2598542 -0.2008324  0.7567534 -
0.7242706
## 26  0.8928177 -1.0371738 -0.1818422 -1.4616799 -0.2008324 -0.3038480 -
0.7792531
## 30  1.1222223 -1.0371738 -0.1818422 -0.2598542 -0.2008324 -0.3038480 -
1.0222542
## 31  1.8104360 -0.8255446 -0.1818422 -1.4616799 -0.2008324  0.2476647 -
0.5073029
## 32  0.4340086 -0.7550015 -0.1818422 -0.2598542 -0.2008324 -0.3038480 -
1.3230814
## 36  0.8928177 -0.7550015 -0.1818422 -0.2598542 -0.2008324  0.7567534 -
1.4608034
## 37  0.6634132 -0.1201139 -0.1818422 -1.4616799 -0.2008324 -0.3038480 -
0.8051733
## 38  0.8928177 -0.8960877 -0.1818422 -1.4616799 -0.2008324  2.1567472 -
0.9711880
## 43  1.1222223 -0.6139154 -0.1818422 -0.2598542 -0.2008324  0.7567534 -
1.1142648
## 48 -0.2542051 -0.1201139 -0.1818422 -1.4616799 -0.2008324  0.7567534 -
0.1614556
## 49  0.4340086 -0.8255446 -0.1818422 -0.2598542 -0.2008324 -0.6432404 -
0.8869714
## 67  1.8104360 -0.8255446 -0.1818422 -0.2598542 -0.2008324 -0.3038480 -
0.7939263
## 74  1.1222223 -1.0371738 -0.1818422 -0.2598542 -0.2008324  0.7567534 -
```

```
1.0416692
## 77  0.2046041 -0.5433723 -0.1818422 -1.4616799 -0.2008324 -0.3038480 -
0.4406694
##    subgrp subgrp
## 6       3      3
## 7       3      3
## 11      3      3
## 13      3      3
## 15      3      3
## 18      3      3
## 19      3      3
## 25      3      3
## 26      3      3
## 30      3      3
## 31      3      3
## 32      3      3
## 36      3      3
## 37      3      3
## 38      3      3
## 43      3      3
## 48      3      3
## 49      3      3
## 67      3      3
## 74      3      3
## 77      3      3

Clust[Clust$subgrp==4,]

##      calories    protein        fat      sodium       fiber       carbo
## 9  -0.8580487 -0.4773310  0.0000000  0.45469653  0.75279812  0.06944832
## 10 -0.8580487  0.4522084 -0.9932203  0.57551356  1.16544301 -0.44446926
## 12  0.1498180  3.2408266  0.9932203  1.54204982 -0.07249167  0.58336590
## 16  0.1498180 -0.4773310 -0.9932203  1.42123279 -0.89778146  1.86815984
## 17 -0.3541153 -0.4773310 -0.9932203  1.54204982 -0.48513656  1.61120105
## 22  0.1498180 -0.4773310 -0.9932203  0.69633060 -0.48513656  1.61120105
## 24 -0.3541153 -0.4773310 -0.9932203  0.33387950 -0.48513656  0.84032469
## 27 -0.3541153  0.4522084 -0.9932203 -1.96164410  0.34015322 -0.18751047
## 33 -0.3541153  0.4522084  0.0000000 -0.27020566  0.34015322  0.06944832
## 34  0.1498180  0.4522084 -0.9932203  0.09224544  0.34015322  0.58336590
## 39  0.1498180 -0.4773310  0.0000000  0.09224544 -0.48513656  0.58336590
## 41  0.1498180 -0.4773310  0.0000000  1.17959872 -0.89778146  1.61120105
## 44 -0.3541153  1.3817478  0.0000000 -1.96164410 -0.89778146  0.32640711
## 51 -0.8580487  0.4522084 -0.9932203  0.09224544  0.34015322  0.84032469
## 54 -0.3541153  0.4522084 -0.9932203  1.90450091 -0.48513656  1.35424227
## 55 -2.8737823 -1.4068705 -0.9932203 -1.96164410 -0.89778146 -0.44446926
## 56 -2.8737823 -0.4773310 -0.9932203 -1.96164410 -0.48513656 -1.21534562
## 61 -0.8580487 -0.4773310 -0.9932203 -1.96164410 -0.07249167  0.06944832
## 62  0.1498180 -1.4068705 -0.9932203  0.93796466 -0.89778146  2.12511863
## 63  0.1498180 -0.4773310 -0.9932203  1.54204982 -0.89778146  1.86815984
## 64 -1.3619821 -0.4773310 -0.9932203 -1.96164410  0.34015322  0.32640711
```

```
## 65 -0.8580487  0.4522084 -0.9932203 -1.96164410  0.75279812  1.09728348
## 66 -0.8580487  0.4522084 -0.9932203 -1.96164410  0.34015322  1.35424227
## 68  0.1498180  3.2408266 -0.9932203  0.81714763 -0.48513656  0.32640711
## 69 -0.8580487 -0.4773310 -0.9932203 -1.78041856  0.34015322  0.06944832
## 70  0.1498180 -0.4773310  0.0000000  0.45469653 -0.89778146  1.61120105
## 72 -0.3541153  0.4522084  0.0000000  0.45469653  0.34015322  0.32640711
## 73  0.1498180 -0.4773310  0.0000000  1.05878169 -0.89778146  1.61120105
## 75 -0.3541153  0.4522084  0.0000000  0.81714763  0.34015322  0.58336590
## 76 -0.3541153  0.4522084  0.0000000  0.45469653  0.34015322  0.58336590
##         sugars      potass   vitamins      shelf     weight        cups
## 9  -0.25420505  0.37368763 -0.1818422 -1.4616799 -0.2008324 -0.64324039
## 10 -0.48360961  1.29074758 -0.1818422  0.9419715 -0.2008324 -0.64324039
## 12 -1.40122785  0.09151534 -0.1818422 -1.4616799 -0.2008324  1.81735475
## 16 -0.94241873 -1.03717383 -0.1818422 -1.4616799 -0.2008324  0.75675340
## 17 -1.17182329 -0.89608768 -0.1818422 -1.4616799 -0.2008324  0.75675340
## 22 -0.94241873 -0.96663076 -0.1818422  0.9419715 -0.2008324  0.75675340
## 24 -0.48360961 -0.26120003 -0.1818422  0.9419715 -0.2008324 -0.30384795
## 27 -0.02480049  0.02097226 -0.1818422 -0.2598542 -0.2008324 -0.09172768
## 33 -0.48360961 -0.19065695 -0.1818422  0.9419715 -0.2008324  0.24766475
## 34 -0.94241873 -0.12011388 -0.1818422  0.9419715 -0.2008324 -2.42505066
## 39 -0.25420505 -0.54337232  3.1822385  0.9419715 -0.2008324  0.75675340
## 41 -0.94241873 -0.82554461 -0.1818422 -0.2598542 -0.2008324  2.87795610
## 44 -0.94241873 -0.04957081 -0.1818422 -0.2598542 -0.2008324  0.75675340
## 51 -1.17182329 -0.12011388 -0.1818422  0.9419715 -0.2008324  0.75675340
## 54 -0.94241873 -0.75500154  3.1822385  0.9419715 -0.2008324  0.75675340
## 55 -1.63063240 -1.17825998 -1.3032024  0.9419715 -3.4599552  0.75675340
## 56 -1.63063240 -0.68445846 -1.3032024  0.9419715 -3.4599552  0.75675340
## 61 -0.25420505  0.16205841 -0.1818422  0.9419715 -0.2008324 -1.36444931
## 62 -1.17182329 -0.96663076 -0.1818422 -1.4616799 -0.2008324  1.30826610
## 63 -0.94241873 -0.89608768 -0.1818422 -1.4616799 -0.2008324  0.75675340
## 64 -1.63063240 -0.04957081 -1.3032024 -1.4616799 -1.3089342  0.75675340
## 65 -1.63063240  0.58531685 -1.3032024 -1.4616799 -0.2008324 -0.64324039
## 66 -1.63063240  0.30314456 -1.3032024 -1.4616799 -0.2008324 -0.64324039
## 68 -0.94241873 -0.61391539 -0.1818422 -1.4616799 -0.2008324  0.75675340
## 69 -0.48360961 -0.12011388 -0.1818422 -0.2598542 -0.2008324  0.75675340
## 70 -0.94241873 -0.89608768  3.1822385  0.9419715 -0.2008324  0.75675340
## 72 -0.94241873  0.16205841  3.1822385  0.9419715 -0.2008324  0.75675340
## 73 -0.94241873 -0.54337232 -0.1818422  0.9419715 -0.2008324 -0.30384795
## 75 -0.94241873  0.23260148 -0.1818422 -1.4616799 -0.2008324 -0.64324039
## 76 -0.94241873  0.16205841 -0.1818422 -1.4616799 -0.2008324  0.75675340
##         rating subgrp subgrp
## 9   0.48087533      4      4
## 10  0.77969576      4      4
## 12  0.59807496      4      4
## 16 -0.06603869      4      4
## 17  0.24879639      4      4
## 22  0.32235640      4      4
## 24  0.13959735      4      4
## 27  1.13821301      4      4
## 33  0.69155685      4      4
```

```
## 34  0.78377123      4      4
## 39 -0.41671824      4      4
## 41 -0.22308231      4      4
## 44  0.88922515      4      4
## 51  1.23068291      4      4
## 54 -0.06186866      4      4
## 55  1.31001152      4      4
## 56  1.47030646      4      4
## 61  0.92358705      4      4
## 62 -0.02656845      4      4
## 63 -0.12909114      4      4
## 64  1.84299757      4      4
## 65  2.28743193      4      4
## 66  2.16834997      4      4
## 68  0.76669214      4      4
## 69  1.21081332      4      4
## 70 -0.25168258      4      4
## 72  0.30548275      4      4
## 73 -0.23269772      4      4
## 75  0.52841741      4      4
## 76  0.65701831      4      4
```

```
#here we calculate the mean rating in order determine the healthy cluster cereals
mean(Clust[Clust$subgrp==1,"rating"])
```

```
## [1] 2.242648
```

```
mean(Clust[Clust$subgrp==2,"rating"])
```

```
## [1] -0.2928786
```

```
mean(Clust[Clust$subgrp==3,"rating"])
```

```
## [1] -0.9636465
```

```
mean(Clust[Clust$subgrp==4,"rating"])
```

```
## [1] 0.6455402
```