Assignment5

Asif Sayyad

2024-04-05

```
#install.packages("tidyverse")
#install.packages("factoextra")
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.0     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(factoextra)
```

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

```
cereals=read.csv("C:/Users/chand/Downloads/Cereals.csv")
head(cereals)
```

```
##                        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
```

```
nrow(cereals)
```

```
## [1] 77
```

```r
cereals[ , c(4:16)] <- scale(cereals[ , c(4:16)])
cereals<-na.omit(cereals)
nrow(cereals)
```

```
## [1] 74
```

they are only three NA values. after removing those missing values they are only 74 rows

```r
cereal_df <- dist(cereals, method ="euclidean")
```

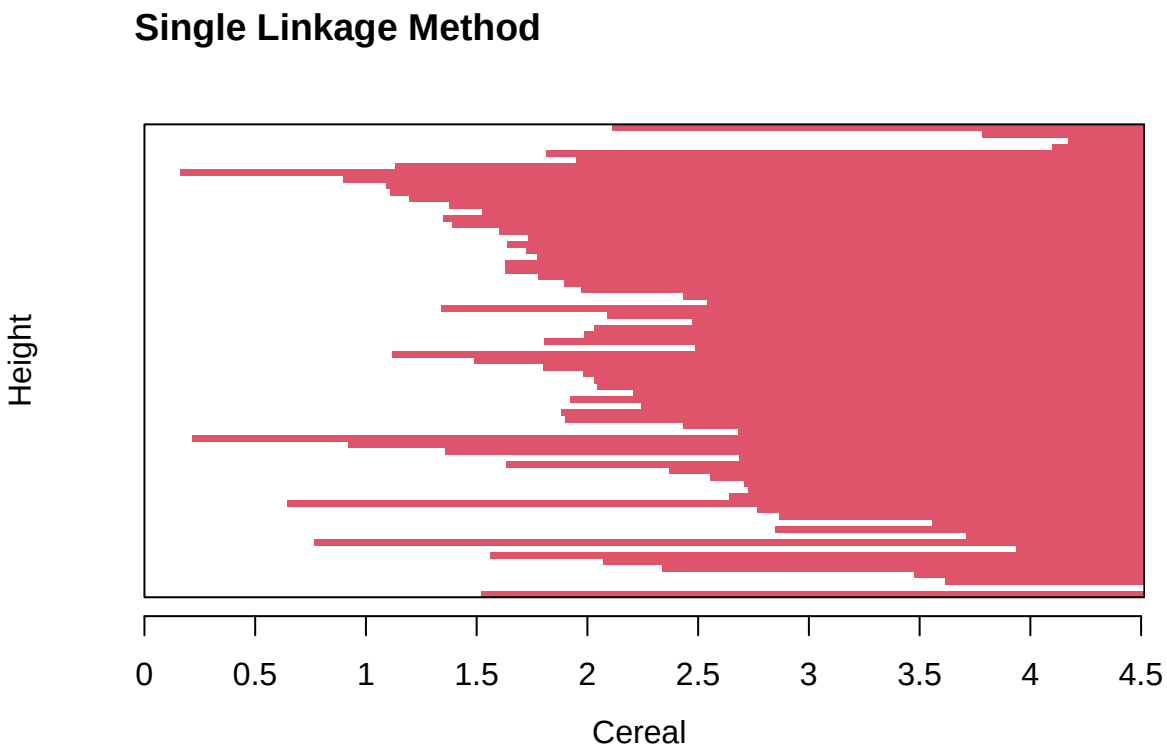
```
## Warning in dist(cereals, method = "euclidean"): NAs introduced by coercion
```

```r
# Perform hierarchical clustering via the single linkage method
#install.packages("cluster")
library(cluster)
single_df <-agnes(cereal_df, method = "single")
plot(single_df,
     main = "Single Linkage Method",
     xlab = "Cereal",
     ylab = "Height",
     cex.axis = 1,
     cex = 0.55,
 hang = -1)
```

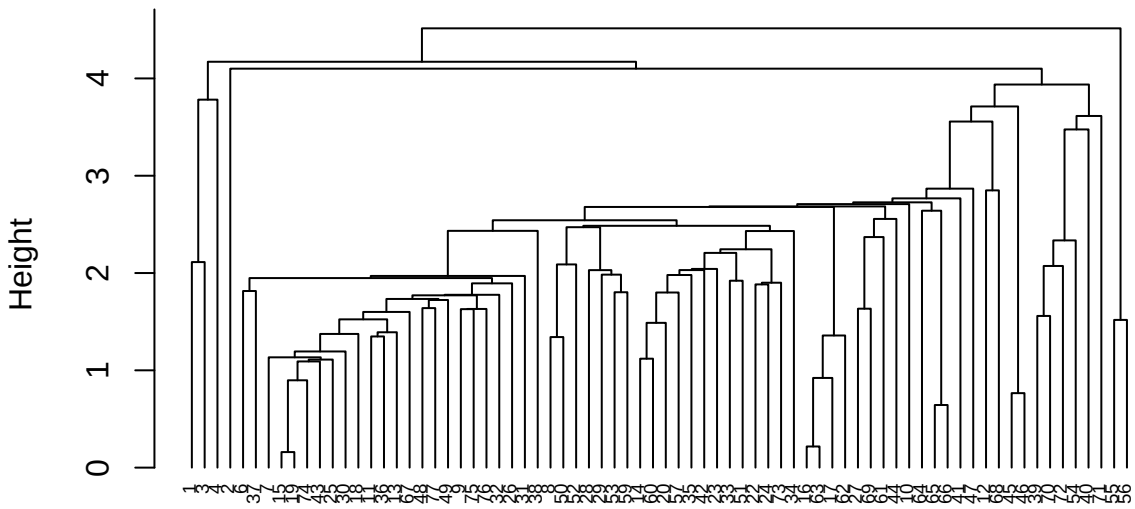
```
## Warning in plot.window(xlim, ylim, log = log, ...): "hang" is not a graphical
## parameter

## Warning in title(main = main, sub = sub, xlab = xlab, ylab = ylab, ...): "hang"
## is not a graphical parameter

## Warning in axis(1, at = at.vals, labels = lab.vals, ...): "hang" is not a
## graphical parameter
```



Agglomerative Coefficient = 0.61



Cereal
Agglomerative Coefficient = 0.61

```
cereal_df <- dist(cereals, method ="euclidean")
```

```
## Warning in dist(cereals, method = "euclidean"): NAs introduced by coercion
```

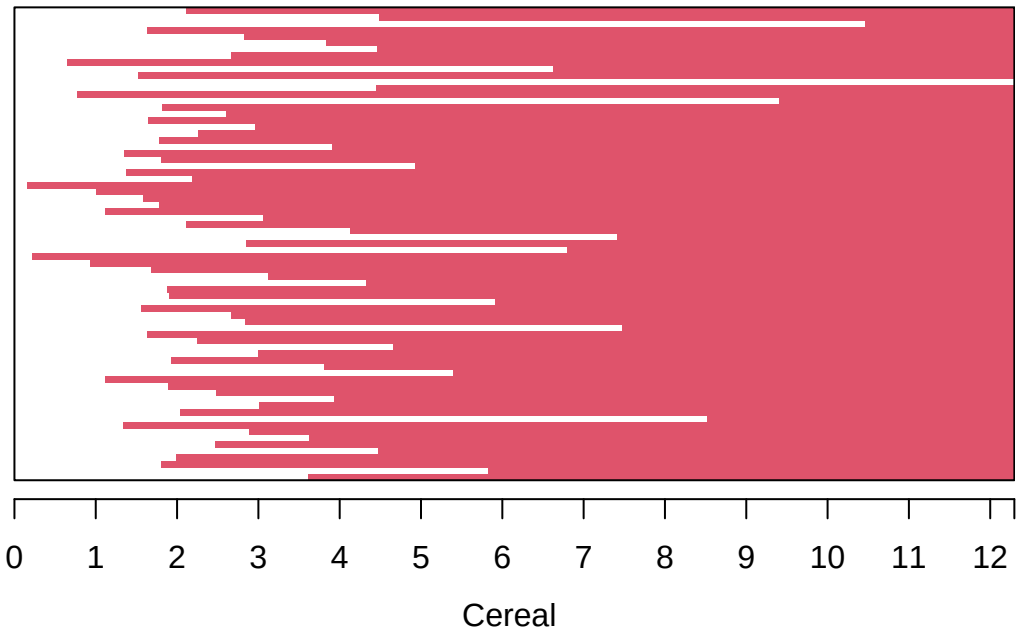
```
# Perform hierarchical clustering via the complete linkage method
#install.packages("cluster")
library(cluster)
complete_df <-agnes(cereal_df, method = "complete")
plot(complete_df,
     main = "complete Linkage Method",
     xlab = "Cereal",
     cex.axis = 1,
     cex = 0.55,
 hang = -1)
```

```
## Warning in plot.window(xlim, ylim, log = log, ...): "hang" is not a graphical
## parameter
```

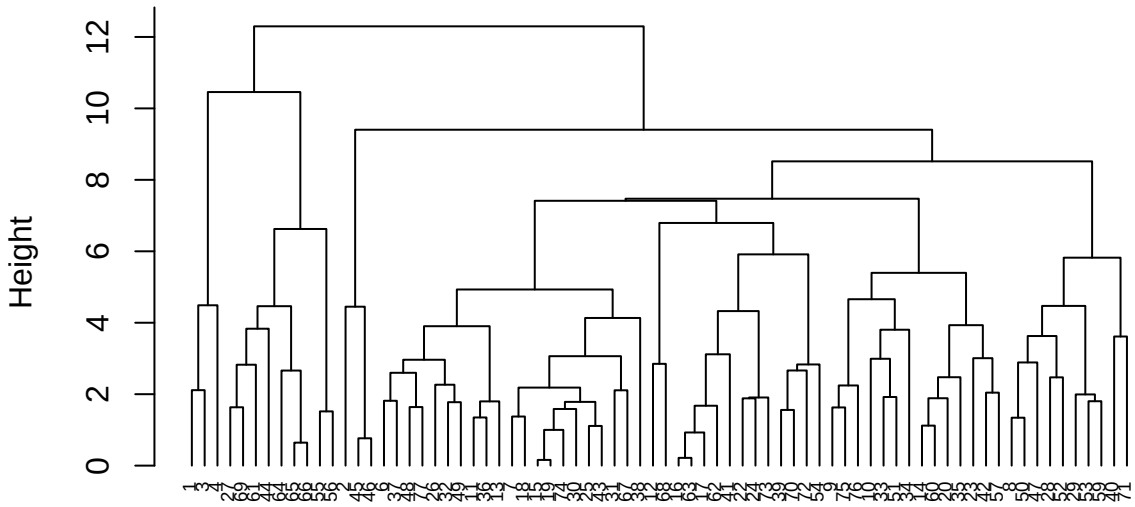
```
## Warning in title(main = main, sub = sub, xlab = xlab, ylab = ylab, ...): "hang"
## is not a graphical parameter
```

```
## Warning in axis(1, at = at.vals, labels = lab.vals, ...): "hang" is not a
## graphical parameter
```

complete Linkage Method



Agglomerative Coefficient = 0.84




```
cereal_df <- dist(cereals, method ="euclidean")
```

```
## Warning in dist(cereals, method = "euclidean"): NAs introduced by coercion
```

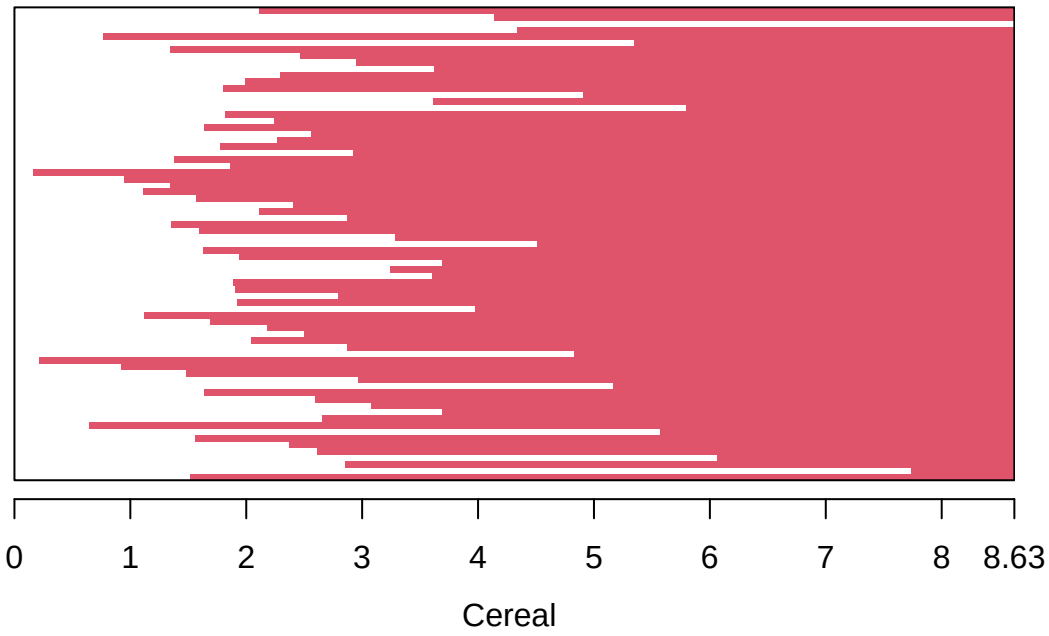
```
# Perform hierarchical clustering via the average linkage method
#install.packages("cluster")
library(cluster)
average_df <-agnes(cereal_df, method = "average")
plot(average_df,
     main = "average Linkage Method",
     xlab = "Cereal",
     cex.axis = 1,
     cex = 0.55,
 hang = -1)
```

```
## Warning in plot.window(xlim, ylim, log = log, ...): "hang" is not a graphical
## parameter
```

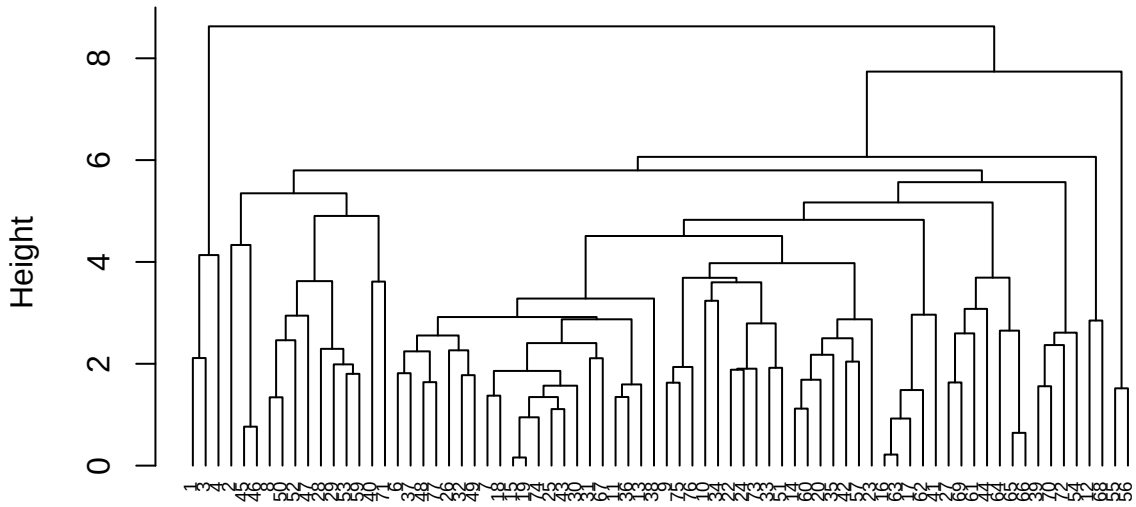
```
## Warning in title(main = main, sub = sub, xlab = xlab, ylab = ylab, ...): "hang"
## is not a graphical parameter
```

```
## Warning in axis(1, at = at.vals, labels = lab.vals, ...): "hang" is not a
## graphical parameter
```

average Linkage Method



Agglomerative Coefficient = 0.78




```
cereal_df <- dist(cereals, method ="euclidean")
```

```
## Warning in dist(cereals, method = "euclidean"): NAs introduced by coercion
```

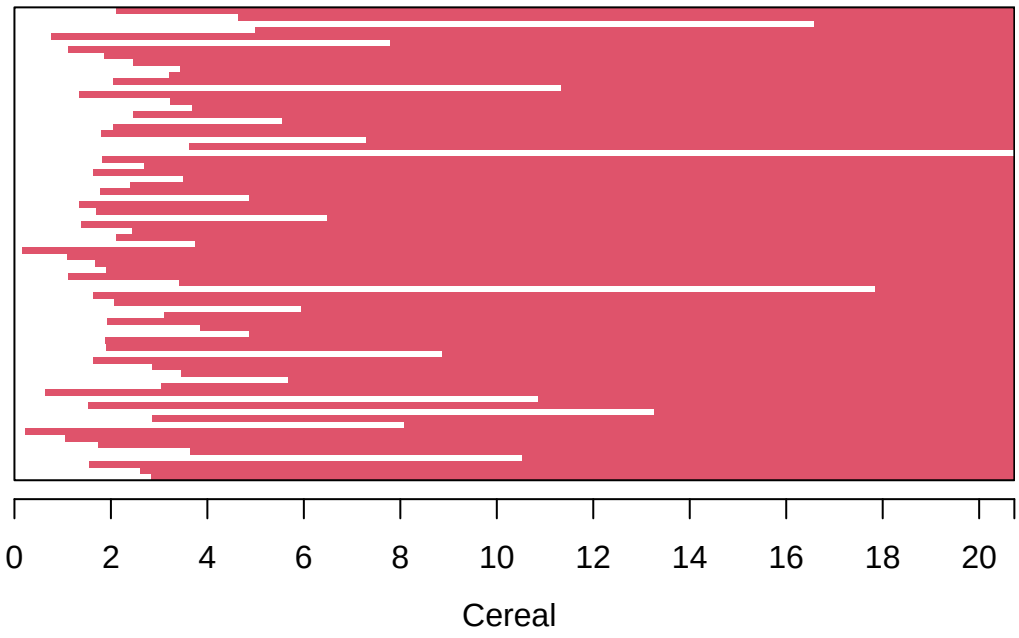
```
# Perform hierarchical clustering via the ward linkage method
#install.packages("cluster")
library(cluster)
ward_df <-agnes(cereal_df, method = "ward")
plot(ward_df,
     main = "ward Linkage Method",
     xlab = "Cereal",
     cex.axis = 1,
     cex = 0.55,
 hang = -1)
```

```
## Warning in plot.window(xlim, ylim, log = log, ...): "hang" is not a graphical
## parameter
```

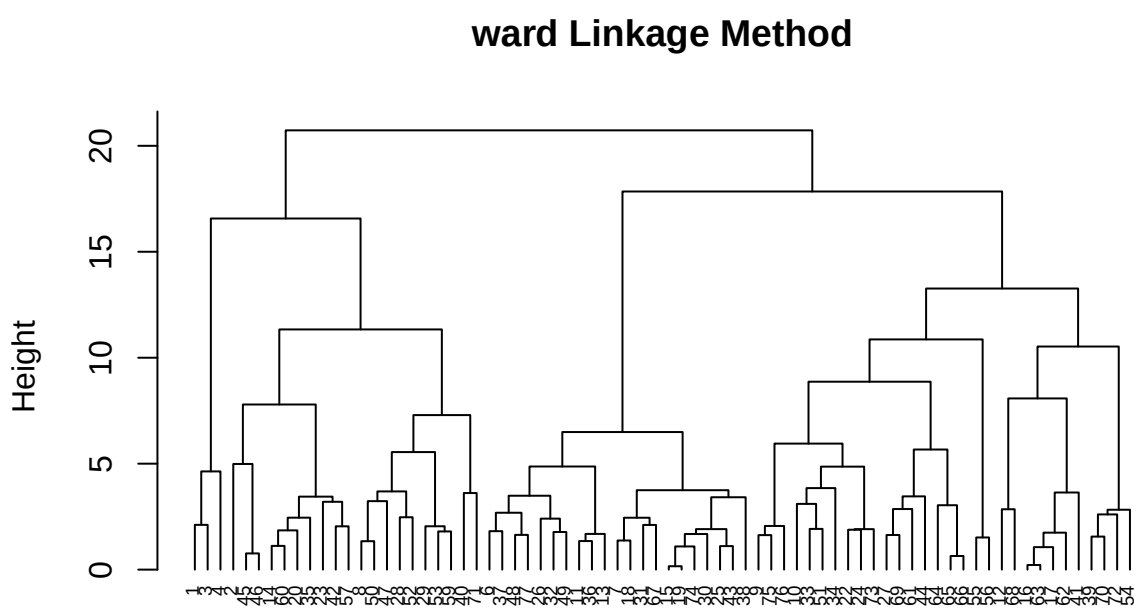
```
## Warning in title(main = main, sub = sub, xlab = xlab, ylab = ylab, ...): "hang"
## is not a graphical parameter
```

```
## Warning in axis(1, at = at.vals, labels = lab.vals, ...): "hang" is not a
## graphical parameter
```

ward Linkage Method



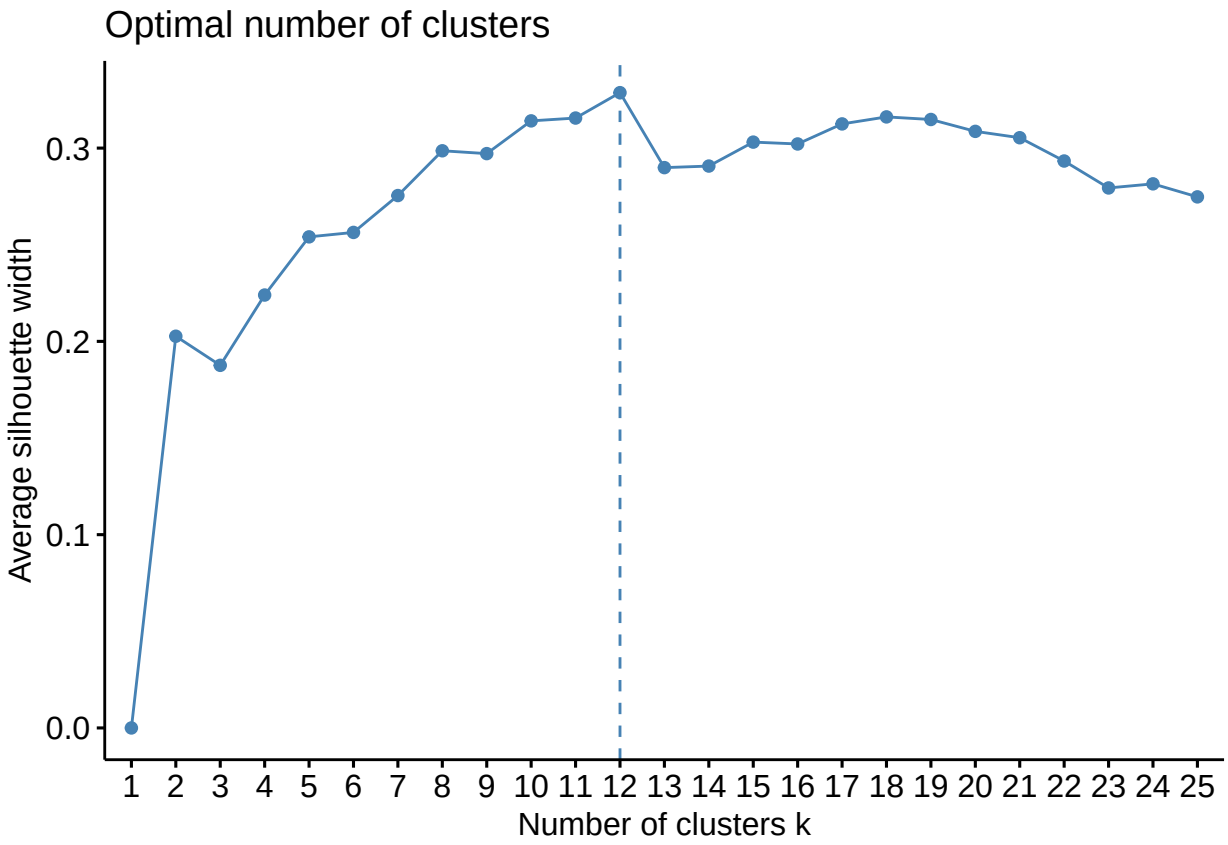
Agglomerative Coefficient = 0.9



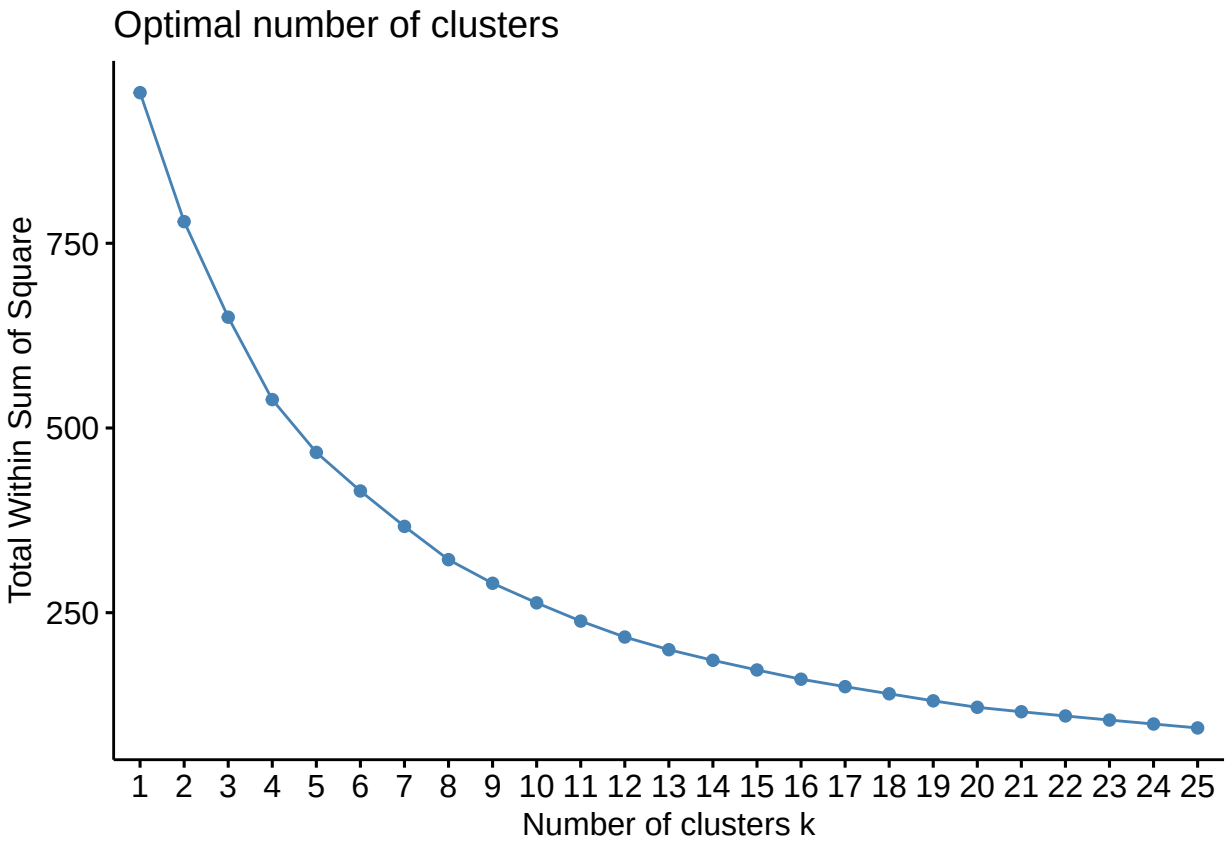
In order to choose the best clustering we use the agglomerative coefficient. That can be acheived by using agnes method from cluster library. single linkage method has 0.73 complete linkage mehod has 0.92 average linkage method has 0.88 ward linkage method has 0.96 The coefficent which is closer to 1.0 is best cluster. so, ward linkage method is choosen for best cluster.

#to find how many clusters we use the elbow and silhouette methods

```
library(factoextra)
fviz_nbclust(cereals[ , c(4:16)],hcut,method="silhouette",k.max=25)
```




```
fviz_nbclust(cereals[ , c(4:16)],hcut,method="wss",k.max=25)
```



Based on the plot of two methods, appropriate number of cluster are 12.

```
#divide the ward tree with 12 clusters
ward_clusters <- cutree(ward_df, k = 12)
cereals_df <- cbind(cluster = ward_clusters,
cereals)
#partition the data
set.seed(124)

# Split the data into 70% partition A and 30% partition B
#install.packages("caret")
library(caret)
```

```
## Loading required package: lattice

##
## Attaching package: 'caret'

## The following object is masked from 'package:purrr':
##
##     lift
```

```
Index <- createDataPartition(cereals$protein, p=0.3, list =F)
cereals_PartitionB <- cereals[Index, ]
cereal_PartitionA <- cereals[-Index,]
```

```r
#clustering with partition data. for this we are using 12 clusters and the ward linkage method to find
cereal_df <- dist(cereal_PartitionA, method ="euclidean")
```

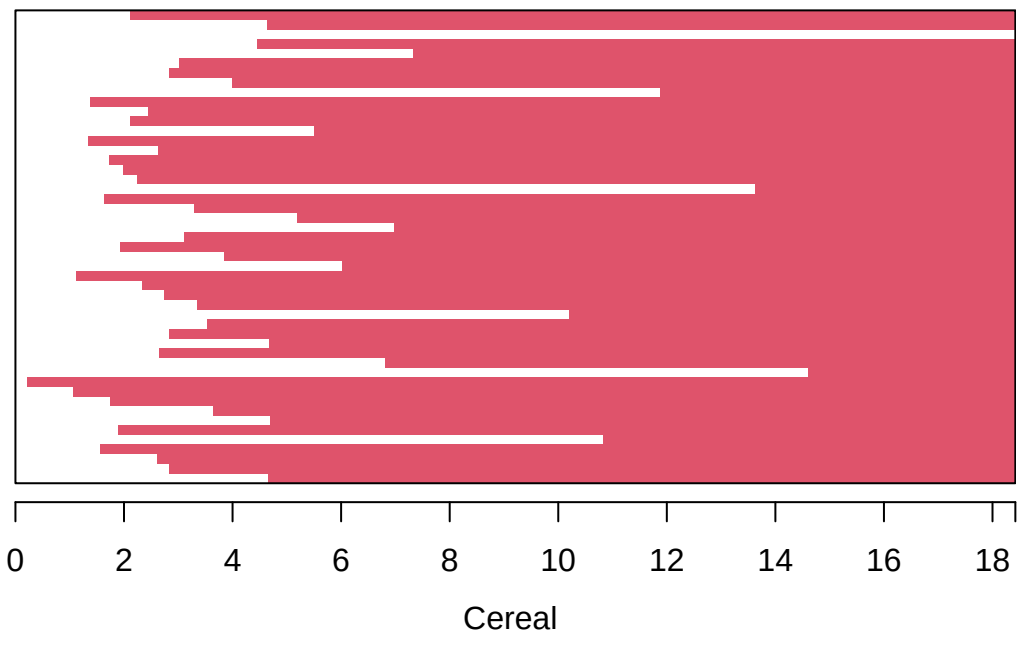
```
## Warning in dist(cereal_PartitionA, method = "euclidean"): NAs introduced by
## coercion
```

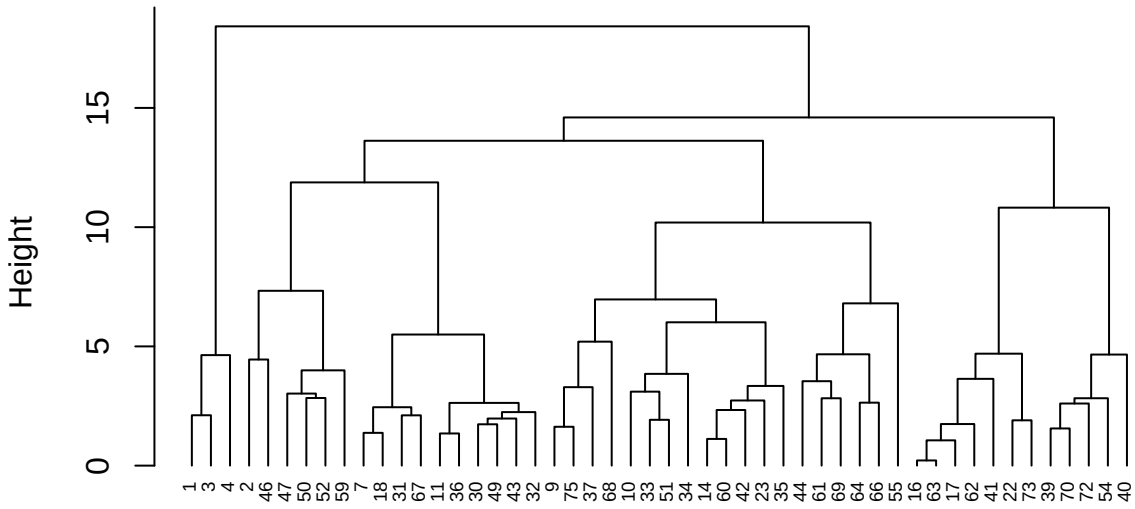
```r
# Perform hierarchical clustering via the ward linkage method
#install.packages("cluster")
ward_df1 <-agnes(cereal_df, method = "ward")
plot(ward_df1,
     main = "ward Linkage Method",
     xlab = "Cereal",
     cex.axis = 1,
     cex = 0.55,
 hang = -1)
```

```
## Warning in plot.window(xlim, ylim, log = log, ...): "hang" is not a graphical
## parameter

## Warning in title(main = main, sub = sub, xlab = xlab, ylab = ylab, ...): "hang"
## is not a graphical parameter

## Warning in axis(1, at = at.vals, labels = lab.vals, ...): "hang" is not a
## graphical parameter
```






```
#continue the process again to find the centriod in partition B
ward_clusters1 <- cutree(ward_df1, k = 12)
cereals_A <- cbind(cluster = ward_clusters1,
cereal_PartitionA)
#find the centroids in partition B
ward_Centroids_A <- aggregate(cereals_A[ , 5:17],
list(cereals_A$cluster), mean)
ward_Centroids_A <- data.frame(Cluster = ward_Centroids_A[ , 1], Centroid =
rowMeans(ward_Centroids_A[ , -c(1:4)]))
ward_Centroids_A <- ward_Centroids_A$Centroid
# Calculate Centers of Partition B
cereal_B_centers <-
data.frame(cereals_PartitionB[, 1:3], Center =
rowMeans(cereals_PartitionB[ , 4:16]))
#find distance between the center of partition A and partition B
centers <- dist(ward_Centroids_A,
cereal_B_centers$Center, method = "euclidean")
cereal_B <- cbind(cluster =
c(4,8,7,3,5,6,7,11,11,10,8,5,10,1,10,1,4,12,12,7,7,1,4,9),
cereals_PartitionB)

# Combine partitions A and B
cereal_1 <- rbind(cereals_A, cereal_B)
cereals_df <-cereals_df[order(cereals_df$name), ]
cereal_1 <-cereal_1[order(cereal_1$name), ]
#to see the stability of the clusters
```

```r
sum(cereals_df$cluster == cereal_1$cluster)
```

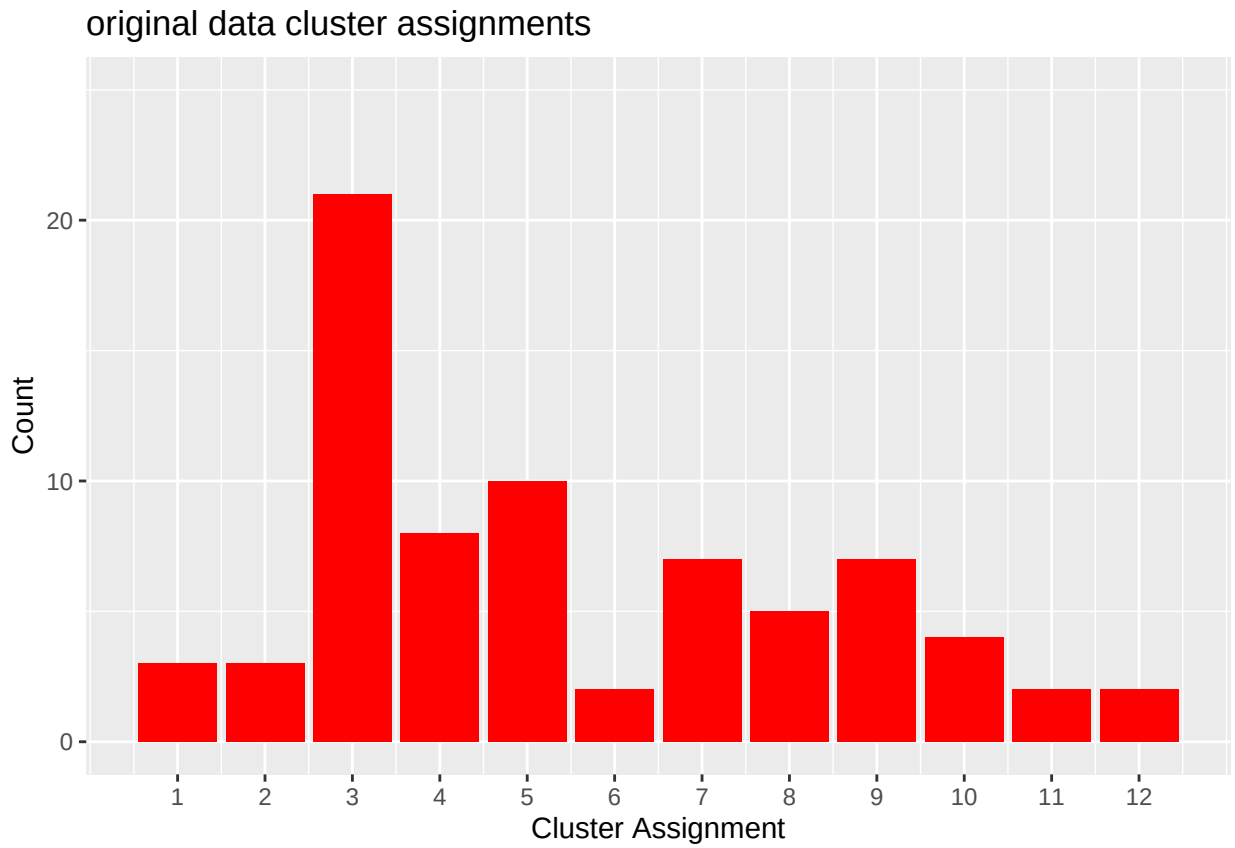
```
## [1] 28
```

From the above results, we can conclude that the clusters are not stable. to visualise thev clusters assignment and their differneces use ggplot

```r
ggplot(data = cereals_df, aes(cereals_df$cluster)) +
 geom_bar(fill = "red") +
 labs(title="original data cluster assignments") +
 labs(x="Cluster Assignment", y="Count") +
 guides(fill=FALSE) +
 scale_x_continuous(breaks=c(1:12)) +
 scale_y_continuous(breaks=c(0,10,20), limits = c(0,25))
```

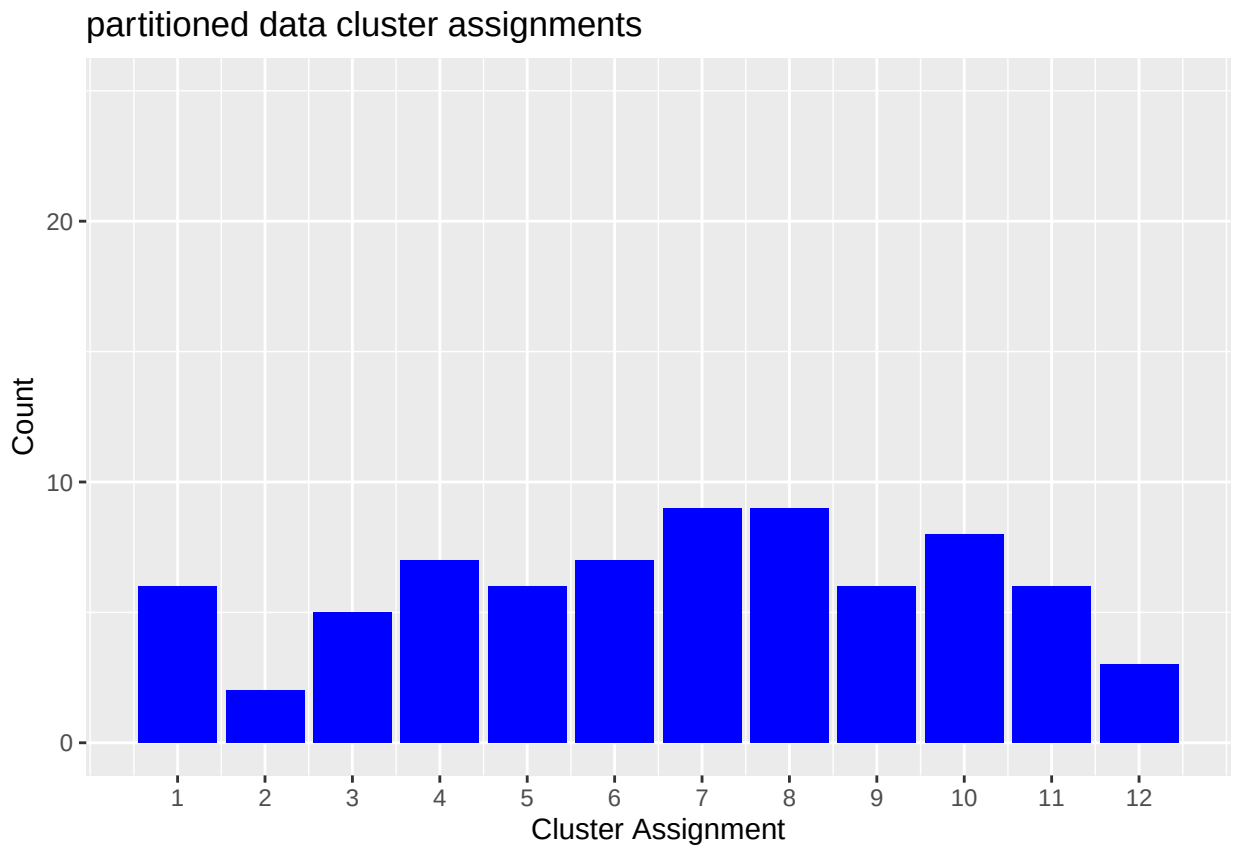
```
## Warning: The '<scale>' argument of 'guides()' cannot be 'FALSE'. Use "none" instead as
## of ggplot2 3.3.4.
## This warning is displayed once every 8 hours.
## Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was
## generated.

## Warning: Use of 'cereals_df$cluster' is discouraged.
## i Use 'cluster' instead.
```



```
ggplot(data = cereal_1, aes(cereal_1$cluster)) +
 geom_bar(fill = "blue") +
 labs(title="partitioned data cluster assignments") +
 labs(x="Cluster Assignment", y="Count") +
 guides(fill=FALSE) +
 scale_x_continuous(breaks=c(1:12)) +
 scale_y_continuous(breaks=c(0,10,20), limits = c(0,25))
```

```
## Warning: Use of `cereal_1$cluster` is discouraged.
## i Use `cluster` instead.
```



It is evident that Cluster 3 drastically reduces when partitioned data are used. Many of the other clusters subsequently expanded in size. The graph shows that when the data is divided, the clusters seem to be distributed more evenly throughout the 12 clusters.

Assignment D: This is not the situation where normalizing the data would be suitable. The reason for this is that the specific cereal sample under investigation determines how to scale or normalize nutrition data related to cereal. Consequently, cereals with an exceptionally high sugar content but low levels of fiber, iron, and other nutrients may be the only ones in the data set. Estimating the cereal's nutritional value for a child is impossible when the data inside the sample set is scaled or normalized. A child may receive practically all of the iron they require from cereal with an iron score of 0.999, but this cereal may actually be the best of the worst in the sample set, offering little to no iron. This is what an untrained observer may think. Because of this, preprocessing the data as a ratio to a child's daily recommended intake of calories, fiber, carbs, and other nutrients would be a better approach. This would prevent a few larger factors from overriding distance computations and enable analysts to make more informed conclusions about clusters when reviewing them. An analyst analyzing the clusters might determine, from the cluster averages, the proportion of a student's

daily nutritional needs that would come from XX cereal. The personnel would be able to choose “healthy” cereal clusters with greater knowledge as a result.