FML

Deekshitha

2024-04-07

cereal\_1 <- read.csv("C:/Users/deeks/Downloads/Cereals.csv")

head(cereal\_1)

## 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\_1)

## '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 ...

summary(cereal\_1)

## name mfr type calories   
## Length:77 Length:77 Length:77 Min. : 50.0   
## Class :character Class :character Class :character 1st Qu.:100.0   
## Mode :character Mode :character Mode :character Median :110.0   
## Mean :106.9   
## 3rd Qu.:110.0   
## Max. :160.0   
##   
## protein fat sodium fiber   
## Min. :1.000 Min. :0.000 Min. : 0.0 Min. : 0.000   
## 1st Qu.:2.000 1st Qu.:0.000 1st Qu.:130.0 1st Qu.: 1.000   
## Median :3.000 Median :1.000 Median :180.0 Median : 2.000   
## Mean :2.545 Mean :1.013 Mean :159.7 Mean : 2.152   
## 3rd Qu.:3.000 3rd Qu.:2.000 3rd Qu.:210.0 3rd Qu.: 3.000   
## Max. :6.000 Max. :5.000 Max. :320.0 Max. :14.000   
##   
## carbo sugars potass vitamins   
## Min. : 5.0 Min. : 0.000 Min. : 15.00 Min. : 0.00   
## 1st Qu.:12.0 1st Qu.: 3.000 1st Qu.: 42.50 1st Qu.: 25.00   
## Median :14.5 Median : 7.000 Median : 90.00 Median : 25.00   
## Mean :14.8 Mean : 7.026 Mean : 98.67 Mean : 28.25   
## 3rd Qu.:17.0 3rd Qu.:11.000 3rd Qu.:120.00 3rd Qu.: 25.00   
## Max. :23.0 Max. :15.000 Max. :330.00 Max. :100.00   
## NA's :1 NA's :1 NA's :2   
## shelf weight cups rating   
## Min. :1.000 Min. :0.50 Min. :0.250 Min. :18.04   
## 1st Qu.:1.000 1st Qu.:1.00 1st Qu.:0.670 1st Qu.:33.17   
## Median :2.000 Median :1.00 Median :0.750 Median :40.40   
## Mean :2.208 Mean :1.03 Mean :0.821 Mean :42.67   
## 3rd Qu.:3.000 3rd Qu.:1.00 3rd Qu.:1.000 3rd Qu.:50.83   
## Max. :3.000 Max. :1.50 Max. :1.500 Max. :93.70   
##

# Process the data

scaled\_crl <- cereal\_1   
scaled\_crl[ , c(4:16)] <- scale(cereal\_1[ , c(4:16)])   
pre\_processed\_cereal <- na.omit(scaled\_crl)   
head(pre\_processed\_cereal)

## name mfr type calories protein fat  
## 1 100%\_Bran N C -1.8929836 1.3286071 -0.01290349  
## 2 100%\_Natural\_Bran Q C 0.6732089 0.4151897 3.96137277  
## 3 All-Bran K C -1.8929836 1.3286071 -0.01290349  
## 4 All-Bran\_with\_Extra\_Fiber K C -2.9194605 1.3286071 -1.00647256  
## 6 Apple\_Cinnamon\_Cheerios G C 0.1599704 -0.4982277 0.98066557  
## 7 Apple\_Jacks K C 0.1599704 -0.4982277 -1.00647256  
## sodium fiber carbo sugars potass vitamins shelf  
## 1 -0.3539844 3.29284661 -2.5087829 -0.2343906 2.5753685 -0.1453172 0.9515734  
## 2 -1.7257708 -0.06375361 -1.7409943 0.2223705 0.5160205 -1.2642598 0.9515734  
## 3 1.1967306 2.87327158 -1.9969238 -0.4627711 3.1434645 -0.1453172 0.9515734  
## 4 -0.2346986 4.97114672 -1.7409943 -1.6046739 3.2854885 -0.1453172 0.9515734  
## 6 0.2424445 -0.27354112 -1.1011705 0.6791317 -0.4071355 -0.1453172 -1.4507595  
## 7 -0.4136273 -0.48332864 -0.9732057 1.5926539 -0.9752315 -0.1453172 -0.2495930  
## weight cups rating  
## 1 -0.1967771 -2.1100340 1.8321876  
## 2 -0.1967771 0.7690100 -0.6180571  
## 3 -0.1967771 -2.1100340 1.1930986  
## 4 -0.1967771 -1.3795303 3.6333849  
## 6 -0.1967771 -0.3052601 -0.9365625  
## 7 -0.1967771 0.7690100 -0.6756899

# Following the preprocessing and scaling of the data, the final number of observations increased from 77 to 74. Therefore, the number of records with a “NA” value was limited to 3.

# Task 1

library(cluster)

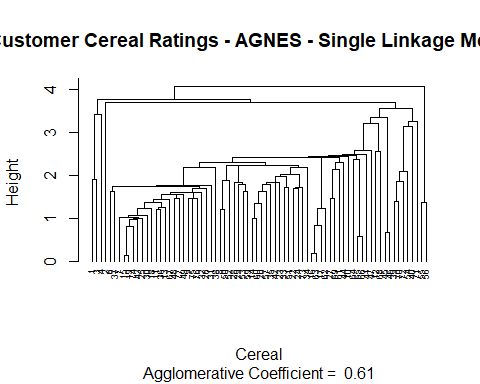
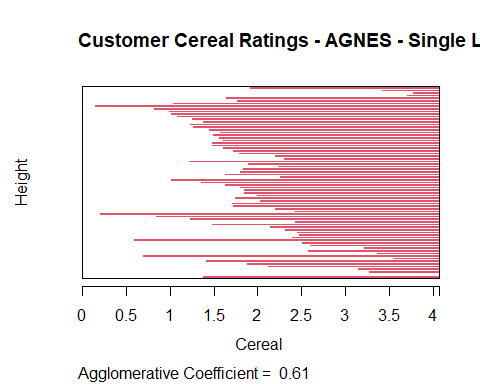
## Warning: package 'cluster' was built under R version 4.3.3

crel\_d\_ecl <- dist(pre\_processed\_cereal[ , c(4:16)], method = "euclidean")   
single\_clust <- agnes(crel\_d\_ecl, method = "single")   
plot(single\_clust, main = "Customer Cereal Ratings - AGNES - Single Linkage Method", xlab = "Cereal", ylab = "Height", cex.axis = 1, cex = 0.56, 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

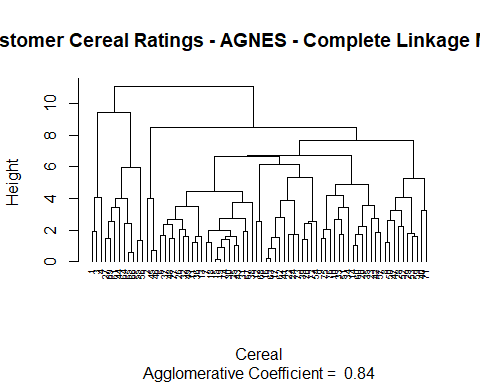
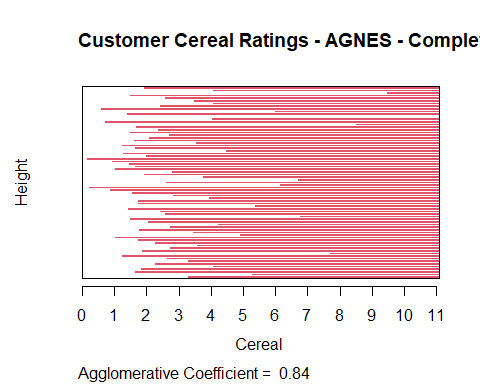


complete\_clust <- agnes(crel\_d\_ecl, method = "complete")   
plot(complete\_clust,   
 main = "Customer Cereal Ratings - AGNES - Complete Linkage Method", xlab = "Cereal", ylab = "Height", cex.axis = 1, cex = 0.56, hang = -1)

## 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

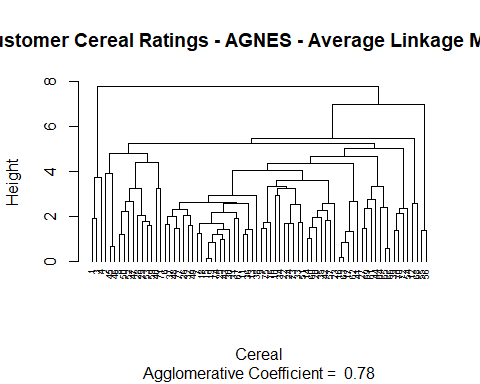
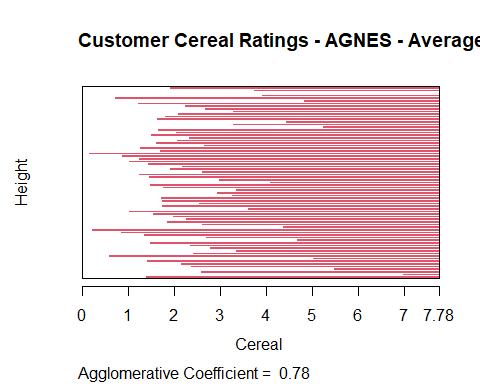


average\_clust <- agnes(crel\_d\_ecl, method = "average")   
plot(average\_clust,   
 main = "Customer Cereal Ratings - AGNES - Average Linkage Method", xlab = "Cereal", ylab = "Height", cex.axis = 1, cex = 0.56, hang = -1)

## 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

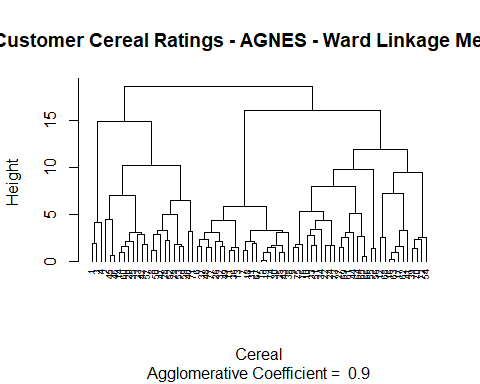
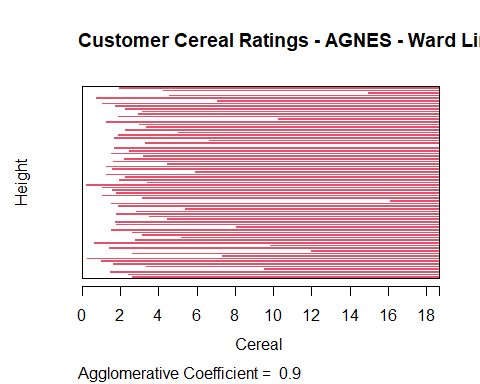


ward\_clust <- agnes(crel\_d\_ecl, method = "ward")   
# Plot the results of the different methods   
plot(ward\_clust,   
 main = "Customer Cereal Ratings - AGNES - Ward Linkage Method", xlab = "Cereal", ylab = "Height", cex.axis = 1, cex = 0.56, hang = -1)

## 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



# The optimal clustering strategy shall be determined by the aggregation coefficient obtained from each method. The value approaching 1.0 indicates a closer clustering structure. Therefore, a choice will be made as to how to achieve a result that is as close as possible to 1.0. The Ward Method: 0.90; Average Linkage: 0.78; Complete Linkage: 0.84; Single Linkage: 0.61. Therefore, the Ward approach will be selected as an optimum clustering model in this particular case.

# Task 2

library(factoextra)

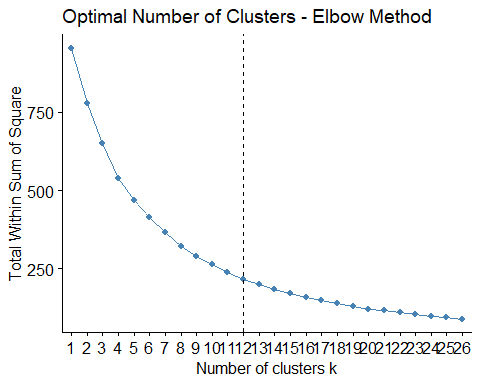
## Warning: package 'factoextra' was built under R version 4.3.2

## Loading required package: ggplot2

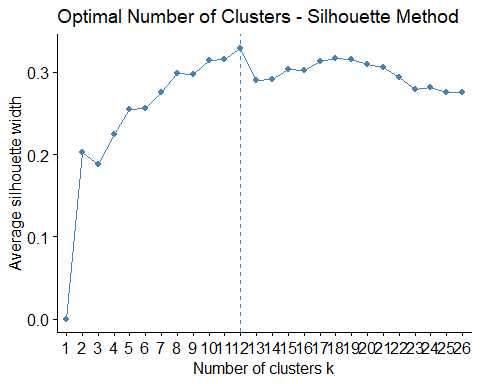
## Warning: package 'ggplot2' was built under R version 4.3.2

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

fviz\_nbclust(pre\_processed\_cereal[ , c(4:16)], hcut, method = "wss", k.max =   
26) +   
 labs(title = "Optimal Number of Clusters - Elbow Method") + geom\_vline(xintercept = 12, linetype = 2)



# Determine the optimal number of clusters for the dataset via the silhouette method   
fviz\_nbclust(pre\_processed\_cereal[ , c(4:16)], hcut,   
 method = "silhouette",   
 k.max = 26) +   
 labs(title = "Optimal Number of Clusters - Silhouette Method")



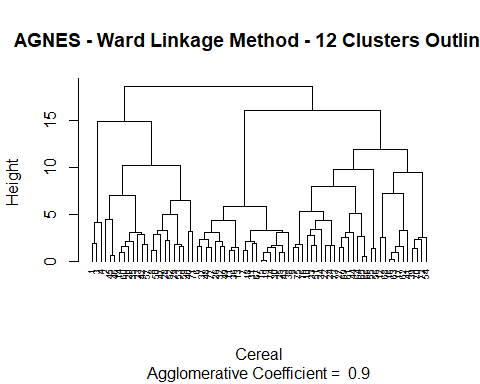
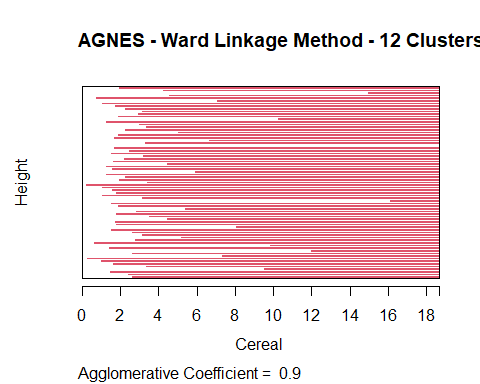
# According to the agreement between the elbow technique and the silhouette approach, this scenario would have 12 clusters. A summary of the 12 hierarchical tree clusters is given here.

plot(ward\_clust,   
 main = "AGNES - Ward Linkage Method - 12 Clusters Outlined", xlab = "Cereal", ylab = "Height", cex.axis = 1, cex = 0.56, 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



# Task 3

# For analysis, divide the tree into 12 clusters.   
clst\_ward\_12 <- cutree(ward\_clust, k = 12)  
  
# Add the assigned cluster to the preprocessed data set   
pre\_processed\_1\_cerl <- cbind(cluster = clst\_ward\_12,   
pre\_processed\_cereal)

# Partition Data: The data set will be divided into two halves in order to test the stability of the clusters. Once 70% has been used for the purpose of drawing up cluster allocations again, another 30% will be allocated according to their nearest centroid.

library(caret)

## Loading required package: lattice

## Warning: package 'lattice' was built under R version 4.3.2

set.seed(708)   
Indx\_crel <- createDataPartition(pre\_processed\_cereal$protein, p=0.3, list = F)   
preprocessed\_crel\_partion\_B <- pre\_processed\_cereal[Indx\_crel, ]  
preprocessed\_crel\_partion\_A <- pre\_processed\_cereal[-Indx\_crel, ]

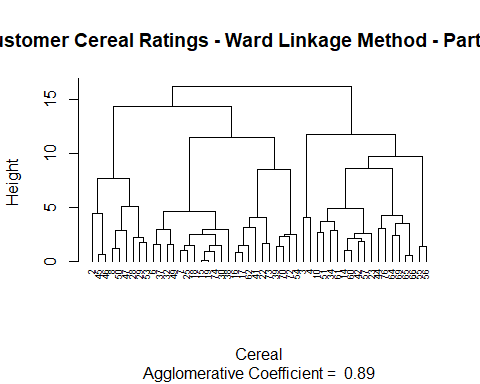
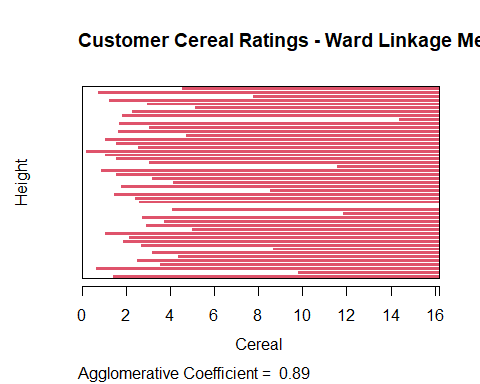
# In order to assess the stability of the clusters, the K value (12) and the ward clustering method shall be used for the duration of this task. Clusters 1 through 12 are then assigned to the closest points in partition B.

crel\_ecl\_A <- dist(preprocessed\_crel\_partion\_A[ , c(4:16)], method = "euclidean")   
Clst\_ward\_A <- agnes(crel\_ecl\_A, method = "ward")   
plot(Clst\_ward\_A,   
 main = "Customer Cereal Ratings - Ward Linkage Method - Partition A", xlab = "Cereal", ylab = "Height",   
cex.axis = 1, cex = 0.56, 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



clst\_ward\_12\_A <- cutree(Clst\_ward\_A, k = 12)  
pre\_processed\_cereal\_A <- cbind(cluster = clst\_ward\_12\_A,   
preprocessed\_crel\_partion\_A)

ctroids\_A\_ward <- aggregate(pre\_processed\_cereal\_A[ , 5:17],   
list(pre\_processed\_cereal\_A$cluster), mean)   
ctroids\_A\_ward <- data.frame(Cluster = ctroids\_A\_ward[ , 1], Centroid =   
rowMeans(ctroids\_A\_ward[ , -c(1:4)]))   
ctroids\_A\_ward <- ctroids\_A\_ward$Centroid

preprocessed\_crel\_partion\_B\_centers <- data.frame(preprocessed\_crel\_partion\_B[, 1:3],  
Center = rowMeans(preprocessed\_crel\_partion\_B[ , 4:16]))

B\_to\_A\_centers <- dist(ctroids\_A\_ward, preprocessed\_crel\_partion\_B\_centers$Center,   
method = "euclidean")

pre\_processed\_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),  
preprocessed\_crel\_partion\_B)

pre\_processed\_cereal\_2 <- rbind(pre\_processed\_cereal\_A, pre\_processed\_cereal\_B)  
pre\_processed\_1\_cerl <-   
pre\_processed\_1\_cerl[order(pre\_processed\_1\_cerl$name), ]   
pre\_processed\_cereal\_2 <-  
pre\_processed\_cereal\_2[order(pre\_processed\_cereal\_2$name), ]

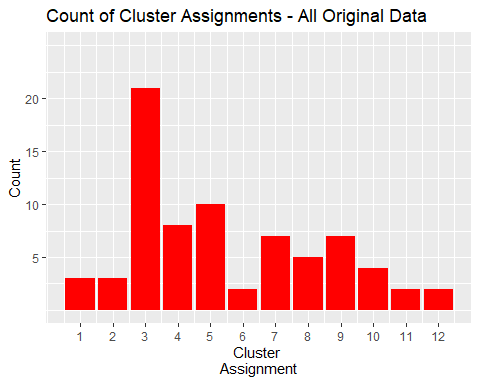
sum(pre\_processed\_1\_cerl$cluster == pre\_processed\_cereal\_2$cluster)

## [1] 20

ggplot(data = pre\_processed\_1\_cerl, aes(pre\_processed\_1\_cerl$cluster)) +   
geom\_bar(fill = "red") +   
labs(title="Count of Cluster Assignments - All Original Data") + labs(x="Cluster   
Assignment", y="Count") + guides(fill=FALSE) +   
scale\_x\_continuous(breaks=c(1:12)) + scale\_y\_continuous(breaks=c(5,10,15,20),   
limits = c(0,25))

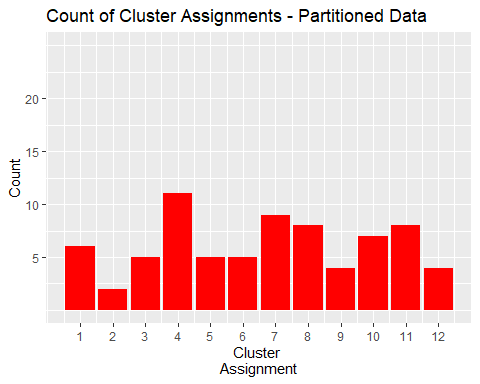
## Warning: The `<scale>` argument of `guides()` cannot be `FALSE`. Use "none" instead as  
## of ggplot2 3.3.4.  
## This warning is displayed once every 8 hours.  
## Call `lifecycle::last\_lifecycle\_warnings()` to see where this warning was  
## generated.

## Warning: Use of `pre\_processed\_1\_cerl$cluster` is discouraged.  
## ℹ Use `cluster` instead.



ggplot(data = pre\_processed\_cereal\_2, aes(pre\_processed\_cereal\_2$cluster)) +   
geom\_bar(fill = "red") +   
labs(title="Count of Cluster Assignments - Partitioned Data") + labs(x="Cluster   
Assignment", y="Count") + guides(fill=FALSE) +   
scale\_x\_continuous(breaks=c(1:12)) +   
scale\_y\_continuous(breaks=c(5,10,15,20), limits = c(0,25))

## Warning: Use of `pre\_processed\_cereal\_2$cluster` is discouraged.  
## ℹ Use `cluster` instead.

 # Assignment Task 4:

#“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?”

Normalizing the data would not be suitable in this situation. It wouldn’t be acceptable because the nutritional information for cereal is scaled and normalized based on the sample of cereal that is being studied. As a result, the collected data set might only contain cereals with extremely high sugar content and extremely low levels of iron, fiber, and other nutrients. It is hard to say how much nutrients the cereal will supply a child once it has been scaled or standardized throughout the sample set. Uninformed viewers would believe that a cereal with an iron score of 0.999 indicates it contains nearly all the iron a child needs for nutrition; nevertheless, it might just be the best of the worst in the sample set, with essentially little nutritional value.

Therefore, converting the data into a ratio to a child’s daily recommended intake of calories, fiber, carbs, etc. would be a more acceptable way to preprocess the data. As a result, analysts would be able to analyze the clusters with greater knowledge and be prevented from letting a few major variables interfere with the distance computations. An analyst examining the clusters could look at the cluster average to find out how much of XX cereal a student should eat each day. This would enable the employees to choose the “healthy” cereal clusters with knowledge.