cereals\_data <- read.csv("C:/Users/rvssu/Downloads/Project2/80-cereals/cereal.csv")  
View(cereals\_data)

cereals\_data[!complete.cases(cereals\_data),]

## [1] name mfr type calories protein fat sodium fiber   
## [9] carbo sugars potass vitamins shelf weight cups rating   
## <0 rows> (or 0-length row.names)

head(cereals\_data)

## 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 -1 25 3 1 0.75 34.38484  
## 6 10 70 25 1 1 0.75 29.50954

colnames(cereals\_data) <- c("Name", "Manufacturer", "Type", "Calories", "Protein", "Fat", "Sodium", "Fibre", "Carbohydrates", "Sugar", "Potassium", "Vitamins", "Shelf", "Weight", "Cups", "Rating")  
  
cereals\_data$Manufacturer\_Name <- cereals\_data$Manufacturer  
cereals\_data$Type <- gsub("H", "Hot", x = cereals\_data$Type)  
cereals\_data$Type <- gsub("C", "Cold", x = cereals\_data$Type)  
cereals\_data$Manufacturer\_Name <- gsub(pattern = "P", replacement = "Post", x = cereals\_data$Manufacturer\_Name)  
cereals\_data$Manufacturer\_Name <- gsub(pattern = "A", replacement = "American Home Food Products", x = cereals\_data$Manufacturer\_Name)  
cereals\_data$Manufacturer\_Name <- gsub(pattern = "G", replacement = "General Mills", x = cereals\_data$Manufacturer\_Name)  
cereals\_data$Manufacturer\_Name <- gsub(pattern = "K", replacement = "Kellogs", x = cereals\_data$Manufacturer\_Name)  
cereals\_data$Manufacturer\_Name <- gsub(pattern = "N", replacement = "Nabisco", x = cereals\_data$Manufacturer\_Name)  
cereals\_data$Manufacturer\_Name <- gsub(pattern = "Q", replacement = "Quaker Oats", x = cereals\_data$Manufacturer\_Name)  
cereals\_data$Manufacturer\_Name <- gsub(pattern = "R", replacement = "Ralston Purina", x = cereals\_data$Manufacturer\_Name)

cereal = cereals\_data

View(cereal)

# Change cereal type and shelf from character to factor  
cereal$Type <- as.factor(cereal$Type)  
cereal$Shelfs <- as.factor(as.character(cereal$Shelf, levels = c("1","2","3"),labels("1","2","3")))  
cereal$Manufacturer <- as.factor(cereal$Manufacturer\_Name)

library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

cereal <- cereal %>% mutate(AverageGroup = case\_when(Rating <= 35 ~ '3',  
 Rating > 35 & Rating <= 54 ~ '2',  
 Rating > 54 ~ '1'))

cereal$AverageGroup <- as.factor(as.character(cereal$AverageGroup, levels = c("1","2","3"),labels("1","2","3")))

# summary

conti = subset(cereal, select = -c(Name, Manufacturer\_Name,Manufacturer,Shelf, Shelfs,Type,AverageGroup))  
  
summary(conti)

## Calories Protein Fat Sodium   
## Min. : 50.0 Min. :1.000 Min. :0.000 Min. : 0.0   
## 1st Qu.:100.0 1st Qu.:2.000 1st Qu.:0.000 1st Qu.:130.0   
## Median :110.0 Median :3.000 Median :1.000 Median :180.0   
## Mean :106.9 Mean :2.545 Mean :1.013 Mean :159.7   
## 3rd Qu.:110.0 3rd Qu.:3.000 3rd Qu.:2.000 3rd Qu.:210.0   
## Max. :160.0 Max. :6.000 Max. :5.000 Max. :320.0   
## Fibre Carbohydrates Sugar Potassium   
## Min. : 0.000 Min. :-1.0 Min. :-1.000 Min. : -1.00   
## 1st Qu.: 1.000 1st Qu.:12.0 1st Qu.: 3.000 1st Qu.: 40.00   
## Median : 2.000 Median :14.0 Median : 7.000 Median : 90.00   
## Mean : 2.152 Mean :14.6 Mean : 6.922 Mean : 96.08   
## 3rd Qu.: 3.000 3rd Qu.:17.0 3rd Qu.:11.000 3rd Qu.:120.00   
## Max. :14.000 Max. :23.0 Max. :15.000 Max. :330.00   
## Vitamins Weight Cups Rating   
## Min. : 0.00 Min. :0.50 Min. :0.250 Min. :18.04   
## 1st Qu.: 25.00 1st Qu.:1.00 1st Qu.:0.670 1st Qu.:33.17   
## Median : 25.00 Median :1.00 Median :0.750 Median :40.40   
## Mean : 28.25 Mean :1.03 Mean :0.821 Mean :42.67   
## 3rd Qu.: 25.00 3rd Qu.:1.00 3rd Qu.:1.000 3rd Qu.:50.83   
## Max. :100.00 Max. :1.50 Max. :1.500 Max. :93.70

cate = subset(cereal, select = c(Type,Shelfs,Manufacturer,AverageGroup))  
summary(cate)

## Type Shelfs Manufacturer AverageGroup  
## Cold:74 1:20 American Home Food Products: 1 1:14   
## Hot : 3 2:21 General Mills :22 2:40   
## 3:36 Kellogs :23 3:23   
## Nabisco : 6   
## Post : 9   
## Quaker Oats : 8   
## Ralston Purina : 8

# regression

mod1 <- lm(Rating ~ Fibre + Sugar + Calories + Sodium,data = cereal)  
#mod1 <- lm(rating ~ mfr + type + sugars + fiber + calories,data = cereals\_data)  
  
summary(mod1)

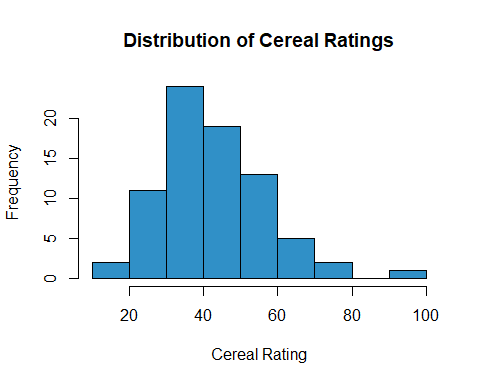
##   
## Call:  
## lm(formula = Rating ~ Fibre + Sugar + Calories + Sodium, data = cereal)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -15.7637 -1.9589 0.4252 2.3289 9.6502   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 68.710582 3.173502 21.651 < 2e-16 \*\*\*  
## Fibre 2.582691 0.215831 11.966 < 2e-16 \*\*\*  
## Sugar -1.850487 0.134264 -13.782 < 2e-16 \*\*\*  
## Calories -0.109422 0.033038 -3.312 0.00145 \*\*   
## Sodium -0.044453 0.006171 -7.204 4.64e-10 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 4.284 on 72 degrees of freedom  
## Multiple R-squared: 0.9119, Adjusted R-squared: 0.907   
## F-statistic: 186.3 on 4 and 72 DF, p-value: < 2.2e-16

Each of the variables is assumed,and there is 0.9 for R-squared,which can be use to predict some scores for other cereals.

let us check distribution for target variable ratings,

# ratings

hist(cereal$Rating,col="#3090C7", main = "Distribution of Cereal Ratings", xlab = "Cereal Rating")

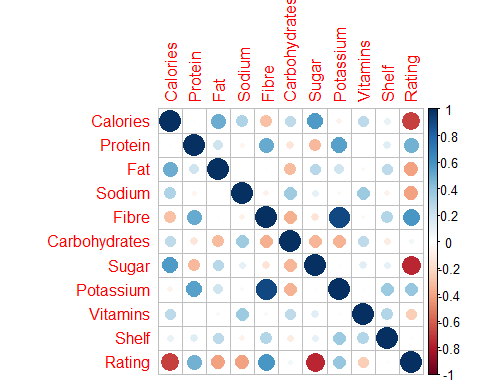


# Correlation

library(corrplot)

## corrplot 0.84 loaded

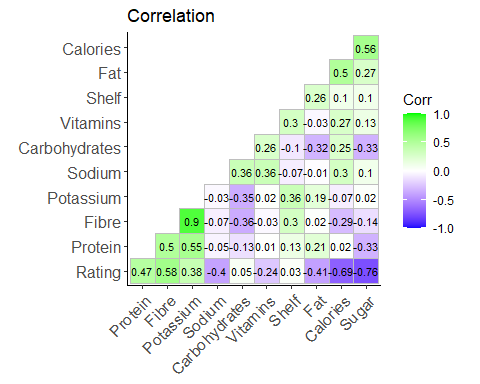
corDat <- subset(cereal, select = -c(Manufacturer\_Name,Type,Name,Manufacturer,Cups,Weight,Shelfs,AverageGroup))  
corrplot(cor(corDat), method= "circle")



library(ggcorrplot)

## Loading required package: ggplot2

ggcorrplot(cor(corDat), hc.order = TRUE,   
 type = "lower",   
 lab = TRUE,   
 lab\_size = 3,   
 method="square",   
 colors = c("blue", "white", "green"),   
 title="Correlation",   
 ggtheme=theme\_classic)



Sugars has been negatively correlated rating.

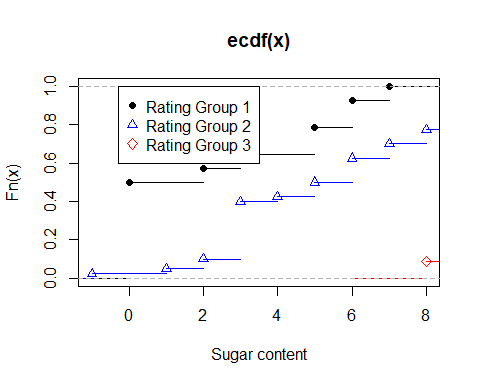
Let’s see if sugar, negatively correlated with rating, has also an influence on the shelving.

# ecdf

#"Name", "Manufacturer", "Type", "Calories", "Protein", "Fat", "Sodium", "Fibre", "Carbohydrates", "Sugar", "Potassium", "Vitamins", "Shelf", "Weight", "Cups", "Rating"

bp1 <- subset(cereal, select = Sugar, subset = AverageGroup == "1", drop = T)  
bp2 <- subset(cereal, select = Sugar, subset = AverageGroup == "2", drop = T)  
bp3 <- subset(cereal, select = Sugar, subset = AverageGroup == "3", drop = T)

plot.ecdf(bp1, xlab = "Sugar content")  
plot.ecdf(bp2, col = "blue",pch = 2, add = TRUE)  
plot.ecdf(bp3, col = "red",pch = 5, add = TRUE)  
legend(-0.3, 1, legend = c("Rating Group 1", "Rating Group 2", "Rating Group 3"), col = c("black", "blue", "Red"), pch = c(19, 2,5))



#abline(v = 250, lty = 0.6)

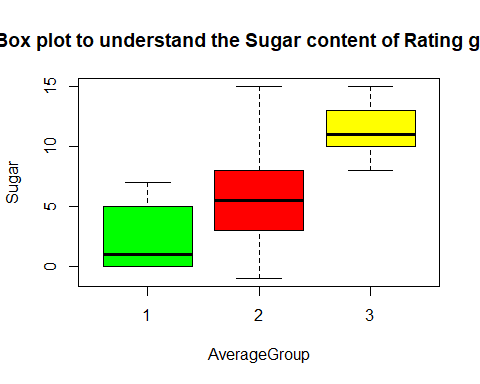
We can see shelf 2 contains more sugars

Performing boxplot to understand average ratings of shelf

# cereal$avggrp = if(cereal$Rating <= 35) '3'  
# cereal$avggrp = (cereal$Rating>35 & cereal$Rating<=54 <- 2  
# cereal$avggrp = (cereal$Rating>54 & data$Rating<=29) <- 1

# Boxplot

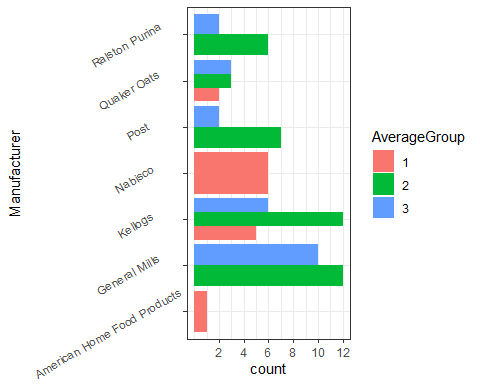
boxplot(Sugar~AverageGroup,data = cereal, main="Box plot to understand the Sugar content of Rating group",col = c("green", "red","yellow"))



The second row of cereal on the shelf had lower scores showing that customers are not interested in sugars.

manufacturer vs rating

library(ggplot2)  
ggplot(cereal, aes(Manufacturer,fill = AverageGroup),ggtheme=theme\_classic)+  
 geom\_bar(position = 'dodge')+  
 theme\_bw()+  
 theme(axis.text.y = element\_text(angle = 30, hjust = 0.75))+  
 coord\_flip()+  
 scale\_y\_continuous(breaks=c(2,4,6,8,10,12,14))



#ggplot(data=cereal,aes(Manufacturer\_Name,y = ..Count..))  
#geom\_col(aes(group = Shelfs),stat\_identity(position = "dodge"))

Let us concentrate on count values.

Most of manufacturers have placed more number of cereals on 3rd shelf. General Mills = 9, Kellogs = 12, Post = 6 and Quaker = 4.

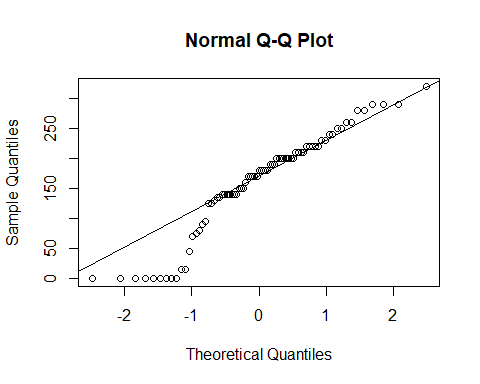
Exception is Nabisco - it has done opposite - More on shelf 1 and least on shelf 3.

The neutral one is Ralston - 4 on shelf 1 and 4 on shelf 3.

# QQ plot Normality

On performing normality on all the variables, Sodium seems to fit the normality graph,

a = cereal$Sodium  
qqnorm(a)  
qqline(a)



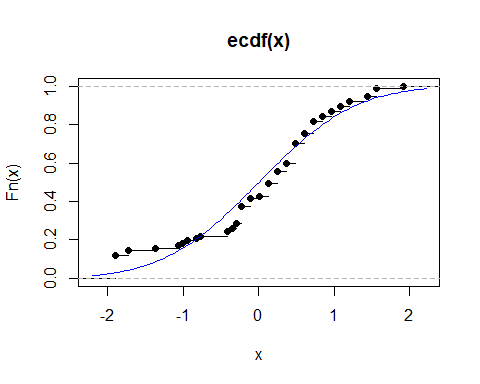
mean(a)

## [1] 159.6753

sd(a)

## [1] 83.8323

x <-(a-159.6753)/83.8323   
 plot.ecdf(x) # adjust x sample mean and standard deviation   
curve(pnorm(x), col = "blue", add = TRUE) # impose normal cdf

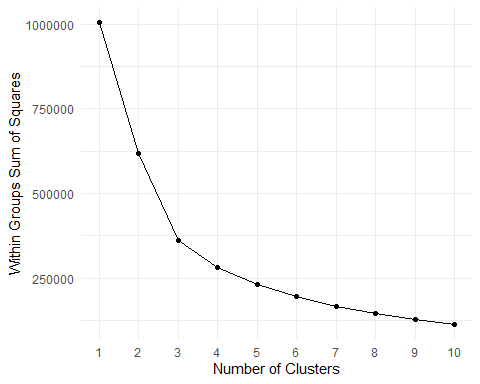


# K means clustering

kmeans\_data = corDat  
kmeans\_data <- kmeans\_data[complete.cases(kmeans\_data), ]

wss <- 0  
for (i in 1:10) {  
 kmeans.output <- kmeans(x = kmeans\_data[], centers = i, nstart = 20)  
 wss[i] <- kmeans.output$tot.withinss  
}

library(dplyr)  
library(tidyr)  
data.frame(Clusters = 1:10, WSS = wss) %>%   
 ggplot(aes(x = Clusters, y = WSS)) +  
 geom\_point() +  
 geom\_line() +  
 scale\_x\_continuous(name = "Number of Clusters", breaks = 1:10, minor\_breaks = NULL) +  
 scale\_y\_continuous(name = "Within Groups Sum of Squares") +  
 theme\_minimal() #+



#labs(title = "Scree Plot")

kmeans.output <- kmeans(x = kmeans\_data, centers =3 , nstart = 20)  
  
kmeans.output

## K-means clustering with 3 clusters of sizes 47, 14, 16  
##   
## Cluster means:  
## Calories Protein Fat Sodium Fibre Carbohydrates Sugar  
## 1 109.7872 2.297872 0.9787234 201.9149 1.223404 15.81915 7.042553  
## 2 112.1429 3.357143 1.4285714 172.5000 5.607143 12.03571 9.142857  
## 3 93.7500 2.562500 0.7500000 24.3750 1.856250 13.25000 4.625000  
## Potassium Vitamins Shelf Rating  
## 1 65.93617 32.97872 2.063830 37.79694  
## 2 217.14286 30.35714 2.928571 46.01532  
## 3 78.68750 12.50000 2.000000 54.03678  
##   
## Clustering vector:  
## 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26   
## 2 3 2 2 1 1 1 1 1 2 1 1 1 1 1 1 1 3 1 2 3 1 1 1 1 1   
## 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52   
## 3 2 2 1 3 1 1 1 3 1 1 1 1 1 1 1 1 3 2 2 2 1 1 1 1 1   
## 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77   
## 2 1 3 3 1 3 2 2 3 1 1 3 3 3 3 1 3 1 2 1 1 1 1 1 1   
##   
## Within cluster sum of squares by cluster:  
## [1] 200182.68 100744.60 60403.75  
## (between\_SS / total\_SS = 64.1 %)  
##   
## Available components:  
##   
## [1] "cluster" "centers" "totss" "withinss" "tot.withinss"  
## [6] "betweenss" "size" "iter" "ifault"

dune\_dist <- dist(corDat)  
cmd <- cmdscale(dune\_dist)

library(vegan3d)

## Loading required package: vegan

## Loading required package: permute

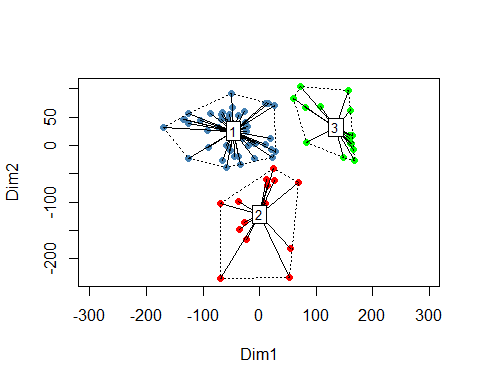
## Loading required package: lattice

## This is vegan 2.5-6

groups <- levels(factor(kmeans.output$cluster))  
ordiplot(cmd, type = "n")

## species scores not available

cols <- c("steelblue", "red", "green", "orange")  
for(i in seq\_along(groups)){  
 points(cmd[factor(kmeans.output$cluster) == groups[i], ], col = cols[i], pch = 16)  
}  
ordispider(cmd, factor(kmeans.output$cluster), label = TRUE)  
ordihull(cmd, factor(kmeans.output$cluster), lty = "dotted")



dd <- cbind(cereal, cluster = kmeans.output$cluster)

dd$Cluster = as.factor(dd$cluster)

# manufacturer vs cluster

library(ggplot2)  
ggplot(dd, aes(Cluster,fill = Manufacturer))+  
 geom\_bar(position = 'dodge')+  
 theme\_minimal()

