MA Group Assignment

Anmol More – 11915043

Sriganesh Balamurugan – 11915001

Raghu Punnamraju – 11915010

Ref : - <https://www.r-bloggers.com/finding-optimal-number-of-clusters/> - <https://www.datanovia.com/en/lessons/determining-the-optimal-number-of-clusters-3-must-know-methods/> - <https://medium.com/codesmart/r-series-k-means-clustering-silhouette-794774b46586>

### Segmentation

## 1. Segment respondents based on the Partworth data (use any unsupervised learning technique).

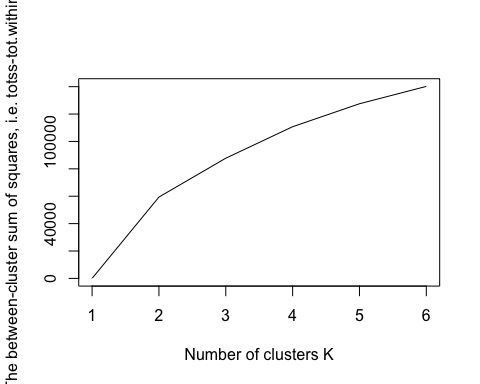
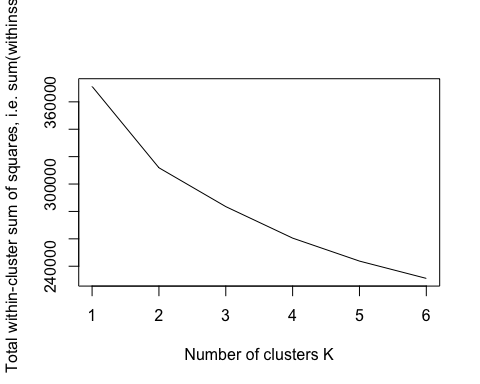
conjoint\_data <- readxl::read\_excel('Beer Partworth Data.xls', sheet = 'Conjoint Data', skip=13, n\_max = 317)

## New names:  
## \* Regular -> Regular...9  
## \* Regular -> Regular...15

#skip the first column which contains respondent IDs, not required for clustering the data  
conjoint\_data <- conjoint\_data[,-c(1)]

# First we apply various clustering algorithms to come up with best K.

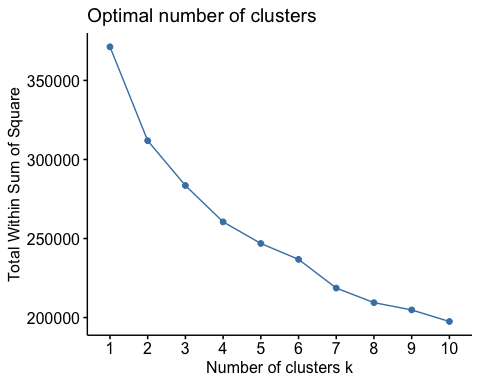
In order to find best K, we looked at within cluster and between cluster Sum of square differences for k.



Looking at within cluster and between cluster differences, we can say that data can be divided in either 2 or 3 clusters.

We try various k means methods like Elbow, Silhouette and find 2 to be most optimal clustering

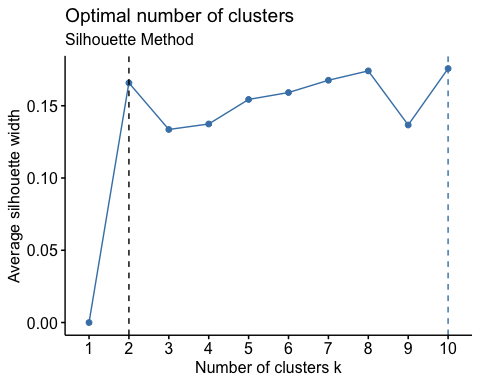
# Elbow Method  
fviz\_nbclust(conjoint\_data, kmeans, method = "wss")



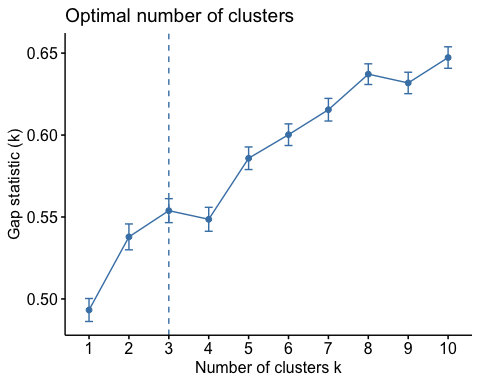
labs(subtitle = "Elbow Method")

## $subtitle  
## [1] "Elbow Method"  
##   
## attr(,"class")  
## [1] "labels"

# Silhouette Method  
fviz\_nbclust(conjoint\_data, kmeans, method = "silhouette") +  
 geom\_vline(xintercept = 2, linetype = 2) +  
 labs(subtitle = "Silhouette Method")



# Gap statistic method doesn't gives a clear k  
fviz\_nbclust(conjoint\_data, kmeans, method = "gap\_stat")



labs(subtitle = "Gap Statistic Method")

## $subtitle  
## [1] "Gap Statistic Method"  
##   
## attr(,"class")  
## [1] "labels"

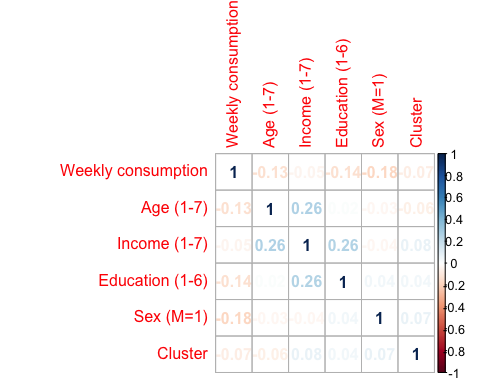
## 2. Use the Descriptors in the Demographic data sheet to perform classification (use any supervised learning technique) based on segments obtained in Step 1 and personify /describe each segment.

# Read demographics data and apply clusters obtained from conjoint data on Demographics data

demographics\_data <- readxl::read\_excel('Beer Partworth Data.xls', sheet = 'Demographics', skip=3, n\_max = 317)  
  
#use two clusters  
demographics\_data$Cluster <- kmeans(conjoint\_data, 2)$cluster  
  
#skip first column containing ID of respondents and clusters from conjoint data  
data <- demographics\_data[,-c(1)]

# Let’s plot various column to understand the distribution

#correlation plot surprisingly doesn't shows much correlation between Age, Income, Education or even with cluster (the target variable)  
corrplot(cor(data), method = "number")



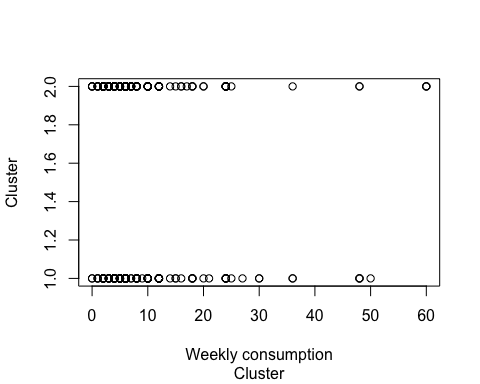
# we see from above correlation plot that, our clusters are not correlated to any variables, to some extent age, income and education are weakly correlated

Let us try to divide two set of people

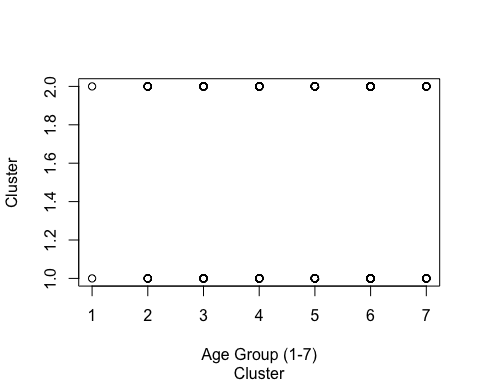
write.csv(data, file = "clustering data.csv", row.names=FALSE)  
names(data) <- c('Weekly\_Consumption', 'Age\_Group', 'Income\_Group', 'Education\_Group', 'Sex', 'Cluster')  
#data <- read.csv("clustering data.csv", colClasses=c('numeric', 'factor', 'factor', 'factor', 'factor', 'factor'))

Plot each variable against no. of clusters defined to identify reltionships

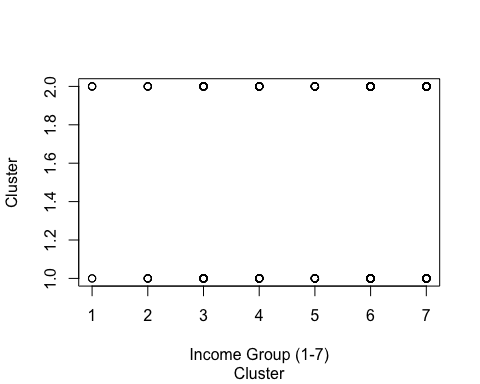
plot(data[,c(1,6)], aes('Weekly consumption','Cluster'))



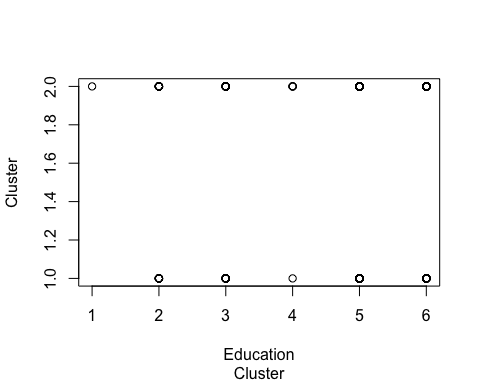
plot(data[,c(2,6)], aes('Age Group (1-7)','Cluster'))



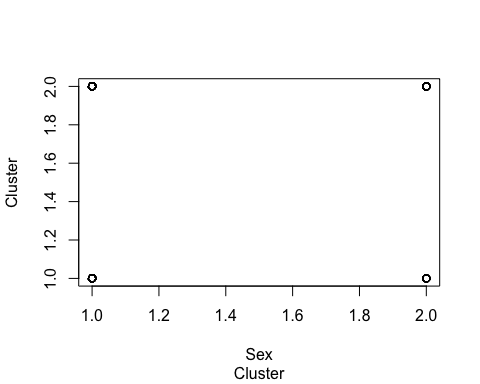
plot(data[,c(3,6)], aes('Income Group (1-7)','Cluster'))



plot(data[,c(4,6)], aes('Education','Cluster'))



plot(data[,c(5,6)], aes('Sex','Cluster'))



check summary of data to understand distribution of each of the variables

describe(data)

## data   
##   
## 6 Variables 317 Observations  
## ---------------------------------------------------------------------------  
## Weekly\_Consumption   
## n missing distinct Info Mean Gmd .05 .10   
## 317 0 27 0.99 9.42 8.941 2 2   
## .25 .50 .75 .90 .95   
## 4 6 12 24 25   
##   
## lowest : 0 1 2 3 4, highest: 30 36 48 50 60  
## ---------------------------------------------------------------------------  
## Age\_Group   
## n missing distinct Info Mean Gmd   
## 317 0 7 0.948 4.77 1.523   
##   
## Value 1 2 3 4 5 6 7  
## Frequency 2 23 28 78 71 92 23  
## Proportion 0.006 0.073 0.088 0.246 0.224 0.290 0.073  
## ---------------------------------------------------------------------------  
## Income\_Group   
## n missing distinct Info Mean Gmd   
## 317 0 7 0.936 5.451 1.686   
##   
## Value 1 2 3 4 5 6 7  
## Frequency 6 10 29 42 34 95 101  
## Proportion 0.019 0.032 0.091 0.132 0.107 0.300 0.319  
## ---------------------------------------------------------------------------  
## Education\_Group   
## n missing distinct Info Mean Gmd   
## 317 0 6 0.909 4.473 1.509   
##   
## Value 1 2 3 4 5 6  
## Frequency 1 40 61 5 126 84  
## Proportion 0.003 0.126 0.192 0.016 0.397 0.265  
## ---------------------------------------------------------------------------  
## Sex   
## n missing distinct Info Mean Gmd   
## 317 0 2 0.317 1.12 0.2117   
##   
## Value 1 2  
## Frequency 279 38  
## Proportion 0.88 0.12  
## ---------------------------------------------------------------------------  
## Cluster   
## n missing distinct Info Mean Gmd   
## 317 0 2 0.739 1.562 0.494   
##   
## Value 1 2  
## Frequency 139 178  
## Proportion 0.438 0.562  
## ---------------------------------------------------------------------------

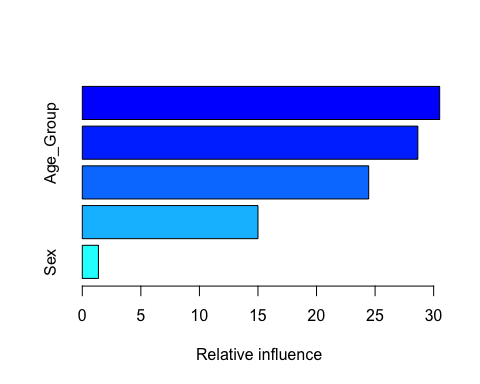
data <- read.csv("clustering data.csv", colClasses=c('numeric', 'factor', 'factor', 'factor', 'factor', 'factor'))  
names(data) <- c('Weekly\_Consumption', 'Age\_Group', 'Income\_Group', 'Education\_Group', 'Sex', 'Cluster')  
random\_forest\_tree = randomForest(Cluster~., data = data)  
summary(random\_forest\_tree)

## Length Class Mode   
## call 3 -none- call   
## type 1 -none- character  
## predicted 317 factor numeric   
## err.rate 1500 -none- numeric   
## confusion 6 -none- numeric   
## votes 634 matrix numeric   
## oob.times 317 -none- numeric   
## classes 2 -none- character  
## importance 5 -none- numeric   
## importanceSD 0 -none- NULL   
## localImportance 0 -none- NULL   
## proximity 0 -none- NULL   
## ntree 1 -none- numeric   
## mtry 1 -none- numeric   
## forest 14 -none- list   
## y 317 factor numeric   
## test 0 -none- NULL   
## inbag 0 -none- NULL   
## terms 3 terms call

library(gbm)

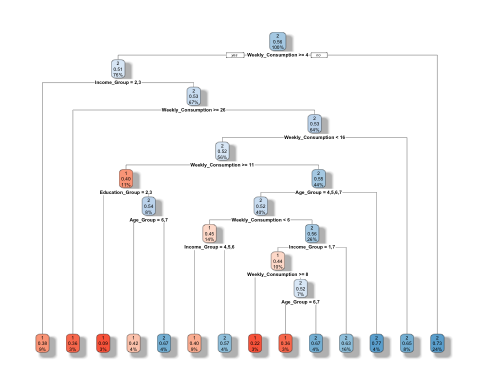
## Loaded gbm 2.1.5

gradient\_boosted\_tree = gbm(Cluster~., data = data, distribution = "gaussian")  
summary(gradient\_boosted\_tree)



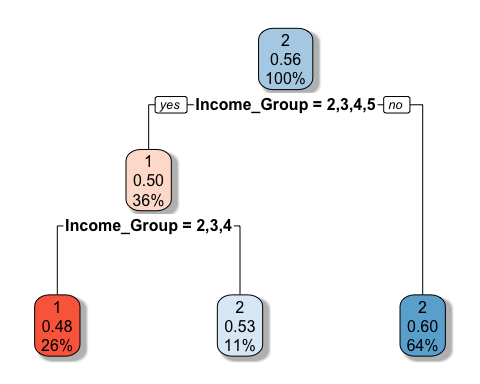
## var rel.inf  
## Income\_Group Income\_Group 30.515456  
## Age\_Group Age\_Group 28.651370  
## Weekly\_Consumption Weekly\_Consumption 24.454245  
## Education\_Group Education\_Group 15.001676  
## Sex Sex 1.377253

# Create a decision tree model  
tree <- rpart(Cluster~., data = data)  
# Visualize the decision tree with rpart.plot  
rpart.plot(tree, box.palette="RdBu", shadow.col="gray")

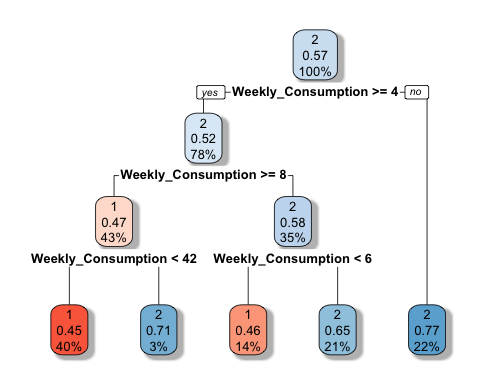


It is evident in below plot that high income group people (75%) fall in cluster 2

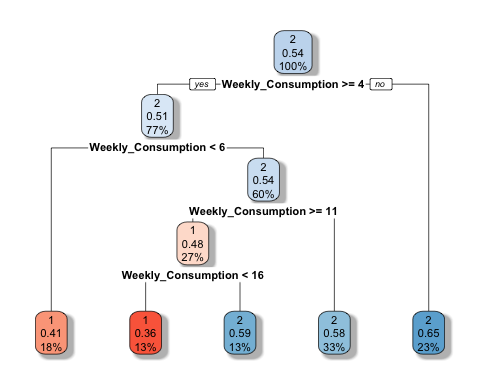
# Create a decision tree to differentiate between different income groups  
tree <- rpart(Cluster~Income\_Group, data = data)  
rpart.plot(tree, box.palette="RdBu", shadow.col="gray")



subset\_data <- data[data$Age\_Group == "1" | data$Age\_Group == "2" | data$Age\_Group == "3" | data$Age\_Group == "4"| data$Age\_Group == "5", ]  
tree <- rpart(Cluster~Weekly\_Consumption, data = subset\_data)  
  
# Visualize the tree in younger age group  
rpart.plot(tree, box.palette="RdBu", shadow.col="gray")



subset\_data <- data[data$Age\_Group == "5" | data$Age\_Group == "6", ]  
tree <- rpart(Cluster~Weekly\_Consumption, data = subset\_data)  
  
# Visualize tree for weekly consumption in older age group  
rpart.plot(tree, box.palette="RdBu", shadow.col="gray")



Finally we can divide the whole data in two groups -

1. Young and heavy drinkers
2. Old and occasional drinkers