Machine Learning - Final Exam

Tanmoy Kumar

12/10/2020

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

Load the Data

```
rm(list = ls())

library(caret)

## Loading required package: lattice

## Loading required package: ggplot2

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

library(ISLR)
library(tidyverse)

## -- Attaching packages --------------------------------------- tidyverse
1.3.0 --

## v tibble  3.0.4     v purrr   0.3.4
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.0
```

```
## -- Conflicts ------------------------------------------
tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## x purrr::lift()   masks caret::lift()
```

```r
library(NbClust)
library(factoextra)
```

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

```r
library(ISLR)
library(ggplot2)
library(e1071)

set.seed(123)
BathSoapDF<-read.csv("BathSoap.csv")

# Data Structure
colnames(BathSoapDF)
```

```
##  [1] "Member.id"            "SEC"                  "FEH"
##  [4] "MT"                   "SEX"                  "AGE"
##  [7] "EDU"                  "HS"                   "CHILD"
## [10] "CS"                   "Affluence.Index"      "No..of.Brands"
## [13] "Brand.Runs"           "Total.Volume"         "No..of..Trans"
## [16] "Value"                "Trans...Brand.Runs"   "Vol.Tran"
## [19] "Avg..Price"           "Pur.Vol.No.Promo...." "Pur.Vol.Promo.6.."
## [22] "Pur.Vol.Other.Promo.." "Br..Cd..57..144"     "Br..Cd..55"
## [25] "Br..Cd..272"          "Br..Cd..286"          "Br..Cd..24"
## [28] "Br..Cd..481"          "Br..Cd..352"          "Br..Cd..5"
## [31] "Others.999"           "Pr.Cat.1"             "Pr.Cat.2"
## [34] "Pr.Cat.3"             "Pr.Cat.4"             "PropCat.5"
## [37] "PropCat.6"            "PropCat.7"            "PropCat.8"
## [40] "PropCat.9"            "PropCat.10"           "PropCat.11"
## [43] "PropCat.12"           "PropCat.13"           "PropCat.14"
## [46] "PropCat.15"
```

```r
str(BathSoapDF)
```

```
## 'data.frame':    600 obs. of  46 variables:
##  $ Member.id            : int  1010010 1010020 1014020 1014030 1014190
1017020 1017110 1017160 1017360 1017460 ...
##  $ SEC                  : int  4 3 2 4 4 4 4 4 4 1 ...
##  $ FEH                  : int  3 2 3 0 1 3 2 3 3 3 ...
##  $ MT                   : int  10 10 10 0 10 10 10 10 10 5 ...
##  $ SEX                  : int  1 2 2 0 2 2 2 2 2 1 ...
##  $ AGE                  : int  4 2 4 4 3 3 4 2 4 4 ...
##  $ EDU                  : int  4 4 5 0 4 4 1 4 4 7 ...
##  $ HS                   : int  2 4 6 0 4 5 3 5 6 3 ...
```

```
##  $ CHILD                : int  4 2 4 5 3 2 2 3 4 4 ...
##  $ CS                   : int  1 1 1 0 1 1 1 0 1 1 ...
##  $ Affluence.Index      : int  2 19 23 0 10 13 11 0 17 6 ...
##  $ No..of.Brands        : int  3 5 5 2 3 3 4 3 2 4 ...
##  $ Brand.Runs           : int  17 25 37 4 6 26 17 8 12 13 ...
##  $ Total.Volume         : int  8025 13975 23100 1500 8300 18175 9950 9300
26490 7455 ...
##  $ No..of..Trans        : int  24 40 63 4 13 41 26 25 27 18 ...
##  $ Value                : num  818 1682 1950 114 591 ...
##  $ Trans...Brand.Runs   : num  1.41 1.6 1.7 1 2.17 1.58 1.53 3.13 2.25
1.38 ...
##  $ Vol.Tran             : num  334 349 367 375 638 ...
##  $ Avg..Price           : num  10.19 12.03 8.44 7.6 7.12 ...
##  $ Pur.Vol.No.Promo.... : chr  "100%" "89%" "94%" "100%" ...
##  $ Pur.Vol.Promo.6..    : chr  "0%" "10%" "2%" "0%" ...
##  $ Pur.Vol.Other.Promo..: chr  "0%" "2%" "4%" "0%" ...
##  $ Br..Cd..57..144      : chr  "38%" "2%" "3%" "40%" ...
##  $ Br..Cd..55           : chr  "13%" "8%" "55%" "60%" ...
##  $ Br..Cd..272          : chr  "0%" "0%" "0%" "0%" ...
##  $ Br..Cd..286          : chr  "0%" "0%" "3%" "0%" ...
##  $ Br..Cd..24           : chr  "0%" "0%" "0%" "0%" ...
##  $ Br..Cd..481          : chr  "0%" "6%" "0%" "0%" ...
##  $ Br..Cd..352          : chr  "0%" "0%" "0%" "0%" ...
##  $ Br..Cd..5            : chr  "0%" "14%" "2%" "0%" ...
##  $ Others.999           : chr  "49.2%" "69.9%" "37.9%" "0.0%" ...
##  $ Pr.Cat.1             : chr  "23%" "29%" "12%" "0%" ...
##  $ Pr.Cat.2             : chr  "56%" "55%" "32%" "40%" ...
##  $ Pr.Cat.3             : chr  "13%" "9%" "56%" "60%" ...
##  $ Pr.Cat.4             : chr  "7%" "6%" "0%" "0%" ...
##  $ PropCat.5            : chr  "50%" "46%" "24%" "40%" ...
##  $ PropCat.6            : chr  "0%" "35%" "12%" "0%" ...
##  $ PropCat.7            : chr  "0%" "3%" "3%" "0%" ...
##  $ PropCat.8            : chr  "0%" "2%" "1%" "0%" ...
##  $ PropCat.9            : chr  "0%" "1%" "1%" "0%" ...
##  $ PropCat.10           : chr  "0%" "0%" "0%" "0%" ...
##  $ PropCat.11           : chr  "0%" "6%" "0%" "0%" ...
##  $ PropCat.12           : chr  "3%" "0%" "2%" "0%" ...
##  $ PropCat.13           : chr  "0%" "0%" "0%" "0%" ...
##  $ PropCat.14           : chr  "13%" "8%" "56%" "60%" ...
##  $ PropCat.15           : chr  "34%" "0%" "0%" "0%" ...
```

```r
colMeans(is.na(BathSoapDF))
```

```
##             Member.id                   SEC                   FEH
##                     0                     0                     0
##                    MT                   SEX                   AGE
##                     0                     0                     0
##                   EDU                    HS                 CHILD
##                     0                     0                     0
##                    CS       Affluence.Index         No..of.Brands
```

```
##                        0                      0                      0
##             Brand.Runs           Total.Volume         No..of..Trans
##                      0                      0                      0
##                  Value     Trans...Brand.Runs               Vol.Tran
##                      0                      0                      0
##             Avg..Price  Pur.Vol.No.Promo....     Pur.Vol.Promo.6..
##                      0                      0                      0
## Pur.Vol.Other.Promo..        Br..Cd..57..144            Br..Cd..55
##                      0                      0                      0
##             Br..Cd..272            Br..Cd..286            Br..Cd..24
##                      0                      0                      0
##             Br..Cd..481            Br..Cd..352             Br..Cd..5
##                      0                      0                      0
##             Others.999               Pr.Cat.1               Pr.Cat.2
##                      0                      0                      0
##               Pr.Cat.3               Pr.Cat.4              PropCat.5
##                      0                      0                      0
##              PropCat.6              PropCat.7              PropCat.8
##                      0                      0                      0
##              PropCat.9             PropCat.10             PropCat.11
##                      0                      0                      0
##             PropCat.12             PropCat.13             PropCat.14
##                      0                      0                      0
##             PropCat.15
##                      0

#No missing records present

#Sample Data
head(BathSoapDF)

##   Member.id SEC FEH MT SEX AGE EDU HS CHILD CS Affluence.Index
No..of.Brands
## 1   1010010   4   3 10   1   4   4  2     4  1               2
3
## 2   1010020   3   2 10   2   2   4  4     2  1              19
5
## 3   1014020   2   3 10   2   4   5  6     4  1              23
5
## 4   1014030   4   0  0   0   4   0  0     5  0               0
2
## 5   1014190   4   1 10   2   3   4  4     3  1              10
3
## 6   1017020   4   3 10   2   3   4  5     2  1              13
3
##   Brand.Runs Total.Volume No..of..Trans  Value Trans...Brand.Runs Vol.Tran
## 1         17         8025            24  818.0               1.41   334.38
## 2         25        13975            40 1681.5               1.60   349.38
## 3         37        23100            63 1950.0               1.70   366.67
## 4          4         1500             4  114.0               1.00   375.00
```

```
## 5          6        8300             13  591.0               2.17   638.46
## 6         26       18175             41 1705.5               1.58   443.29
##   Avg..Price Pur.Vol.No.Promo.... Pur.Vol.Promo.6.. Pur.Vol.Other.Promo..
## 1      10.19                 100%                0%                    0%
## 2      12.03                  89%               10%                    2%
## 3       8.44                  94%                2%                    4%
## 4       7.60                 100%                0%                    0%
## 5       7.12                  61%               14%                   24%
## 6       9.38                 100%                0%                    0%
##   Br..Cd..57..144 Br..Cd..55 Br..Cd..272 Br..Cd..286 Br..Cd..24
Br..Cd..481
## 1             38%        13%          0%          0%         0%
0%
## 2              2%         8%          0%          0%         0%
6%
## 3              3%        55%          0%          3%         0%
0%
## 4             40%        60%          0%          0%         0%
0%
## 5              5%        14%          0%          0%         0%
0%
## 6              8%         7%          0%          0%         0%
0%
##   Br..Cd..352 Br..Cd..5 Others.999 Pr.Cat.1 Pr.Cat.2 Pr.Cat.3 Pr.Cat.4
## 1          0%        0%      49.2%      23%      56%      13%       7%
## 2          0%       14%      69.9%      29%      55%       9%       6%
## 3          0%        2%      37.9%      12%      32%      56%       0%
## 4          0%        0%       0.0%       0%      40%      60%       0%
## 5          0%        0%      80.7%       0%       5%      14%      81%
## 6          0%        0%      85.7%      22%      45%       7%      27%
##   PropCat.5 PropCat.6 PropCat.7 PropCat.8 PropCat.9 PropCat.10 PropCat.11
## 1       50%        0%        0%        0%        0%         0%         0%
## 2       46%       35%        3%        2%        1%         0%         6%
## 3       24%       12%        3%        1%        1%         0%         0%
## 4       40%        0%        0%        0%        0%         0%         0%
## 5       81%        0%        0%        5%        0%         0%         0%
## 6       49%       10%        0%        1%        7%         0%         0%
##   PropCat.12 PropCat.13 PropCat.14 PropCat.15
## 1         3%         0%        13%        34%
## 2         0%         0%         8%         0%
## 3         2%         0%        56%         0%
## 4         0%         0%        60%         0%
## 5         0%         0%        14%         0%
## 6         0%         0%         7%        27%
```

Converting the character values to numeric ones

```
#creating temporary DF with % value column
a<-BathSoapDF[20:46] %>% mutate_each(funs(as.numeric(gsub("%", "",., fixed =
TRUE))/100))
```

```
## Warning: `funs()` is deprecated as of dplyr 0.8.0.
## Please use a list of either functions or lambdas:
##
##   # Simple named list:
##   list(mean = mean, median = median)
##
##   # Auto named with `tibble::lst()`:
##   tibble::lst(mean, median)
##
##   # Using lambdas
##   list(~ mean(., trim = .2), ~ median(., na.rm = TRUE))
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_warnings()` to see where this warning was generated.

## Warning: `mutate_each_()` is deprecated as of dplyr 0.7.0.
## Please use `across()` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_warnings()` to see where this warning was generated.
```

```
#Adding column of numeric values to the
BathSoapDF<-cbind(BathSoapDF[1:19], a)

#validating data conversion
str(BathSoapDF)
```

```
## 'data.frame':    600 obs. of  46 variables:
##  $ Member.id            : int  1010010 1010020 1014020 1014030 1014190
1017020 1017110 1017160 1017360 1017460 ...
##  $ SEC                  : int  4 3 2 4 4 4 4 4 4 1 ...
##  $ FEH                  : int  3 2 3 0 1 3 2 3 3 3 ...
##  $ MT                   : int  10 10 10 0 10 10 10 10 10 5 ...
##  $ SEX                  : int  1 2 2 0 2 2 2 2 2 1 ...
##  $ AGE                  : int  4 2 4 4 3 3 4 2 4 4 ...
##  $ EDU                  : int  4 4 5 0 4 4 1 4 4 7 ...
##  $ HS                   : int  2 4 6 0 4 5 3 5 6 3 ...
##  $ CHILD                : int  4 2 4 5 3 2 2 3 4 4 ...
##  $ CS                   : int  1 1 1 0 1 1 1 0 1 1 ...
##  $ Affluence.Index      : int  2 19 23 0 10 13 11 0 17 6 ...
##  $ No..of.Brands        : int  3 5 5 2 3 3 4 3 2 4 ...
##  $ Brand.Runs           : int  17 25 37 4 6 26 17 8 12 13 ...
##  $ Total.Volume         : int  8025 13975 23100 1500 8300 18175 9950 9300
26490 7455 ...
##  $ No..of..Trans        : int  24 40 63 4 13 41 26 25 27 18 ...
##  $ Value                : num  818 1682 1950 114 591 ...
##  $ Trans...Brand.Runs   : num  1.41 1.6 1.7 1 2.17 1.58 1.53 3.13 2.25
1.38 ...
##  $ Vol.Tran             : num  334 349 367 375 638 ...
##  $ Avg..Price           : num  10.19 12.03 8.44 7.6 7.12 ...
##  $ Pur.Vol.No.Promo.... : num  1 0.89 0.94 1 0.61 1 0.98 0.94 0.9 1 ...
##  $ Pur.Vol.Promo.6..    : num  0 0.1 0.02 0 0.14 0 0.02 0 0.1 0 ...
```

```
##  $ Pur.Vol.Other.Promo..: num  0 0.02 0.04 0 0.24 0 0 0.06 0 0 ...
##  $ Br..Cd..57..144      : num  0.38 0.02 0.03 0.4 0.05 0.08 0.45 0.04 0.39
0.07 ...
##  $ Br..Cd..55           : num  0.13 0.08 0.55 0.6 0.14 0.07 0.05 0.79 0
0.12 ...
##  $ Br..Cd..272          : num  0 0 0 0 0 0 0.01 0 0 0 ...
##  $ Br..Cd..286          : num  0 0 0.03 0 0 0 0 0 0 0 ...
##  $ Br..Cd..24           : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Br..Cd..481          : num  0 0.06 0 0 0 0 0 0 0 0 ...
##  $ Br..Cd..352          : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Br..Cd..5            : num  0 0.14 0.02 0 0 0 0 0 0 0.4 ...
##  $ Others.999           : num  0.492 0.699 0.379 0 0.807 0.857 0.495 0.167
0.615 0.41 ...
##  $ Pr.Cat.1             : num  0.23 0.29 0.12 0 0 0.22 0.07 0.04 0.11 0.61
...
##  $ Pr.Cat.2             : num  0.56 0.55 0.32 0.4 0.05 0.45 0.66 0.04 0.89
0.1 ...
##  $ Pr.Cat.3             : num  0.13 0.09 0.56 0.6 0.14 0.07 0.05 0.9 0
0.12 ...
##  $ Pr.Cat.4             : num  0.07 0.06 0 0 0.81 0.27 0.23 0.02 0 0.17
...
##  $ PropCat.5            : num  0.5 0.46 0.24 0.4 0.81 0.49 0.82 0.06 0.7
0.24 ...
##  $ PropCat.6            : num  0 0.35 0.12 0 0 0.1 0 0 0.28 0.46 ...
##  $ PropCat.7            : num  0 0.03 0.03 0 0 0 0.02 0 0 0.15 ...
##  $ PropCat.8            : num  0 0.02 0.01 0 0.05 0.01 0.01 0 0 0 ...
##  $ PropCat.9            : num  0 0.01 0.01 0 0 0.07 0 0 0.02 0 ...
##  $ PropCat.10           : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ PropCat.11           : num  0 0.06 0 0 0 0 0 0 0 0 ...
##  $ PropCat.12           : num  0.03 0 0.02 0 0 0 0 0.01 0 0 ...
##  $ PropCat.13           : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ PropCat.14           : num  0.13 0.08 0.56 0.6 0.14 0.07 0.05 0.9 0
0.12 ...
##  $ PropCat.15           : num  0.34 0 0 0 0 0.27 0.1 0.03 0 0.03 ...
```

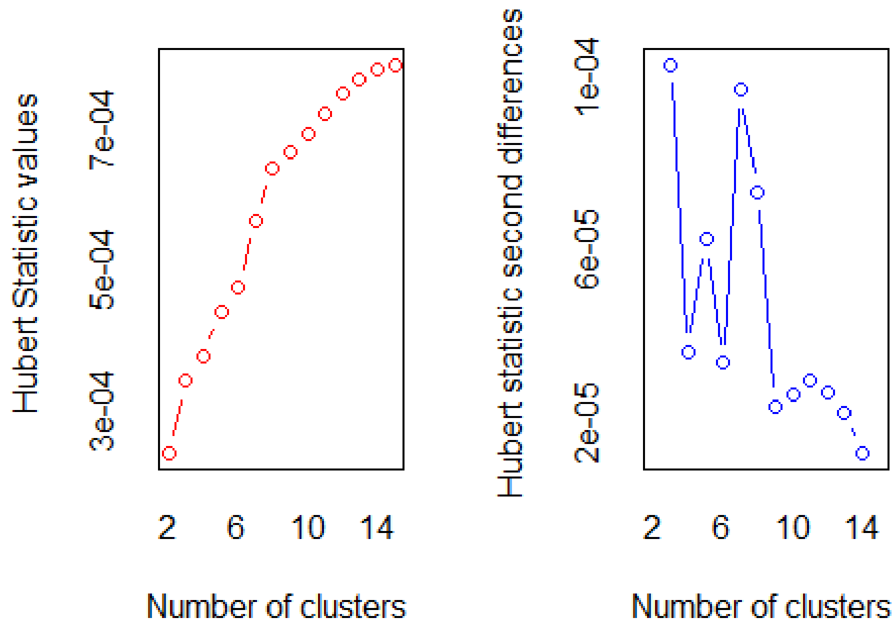
```
head(BathSoapDF)
```

```
##   Member.id SEC FEH MT SEX AGE EDU HS CHILD CS Affluence.Index
No..of.Brands
## 1   1010010   4   3 10   1   4   4  2     4  1               2
3
## 2   1010020   3   2 10   2   2   4  4     2  1              19
5
## 3   1014020   2   3 10   2   4   5  6     4  1              23
5
## 4   1014030   4   0  0   0   4   0  0     5  0               0
2
## 5   1014190   4   1 10   2   3   4  4     3  1              10
3
## 6   1017020   4   3 10   2   3   4  5     2  1              13
```

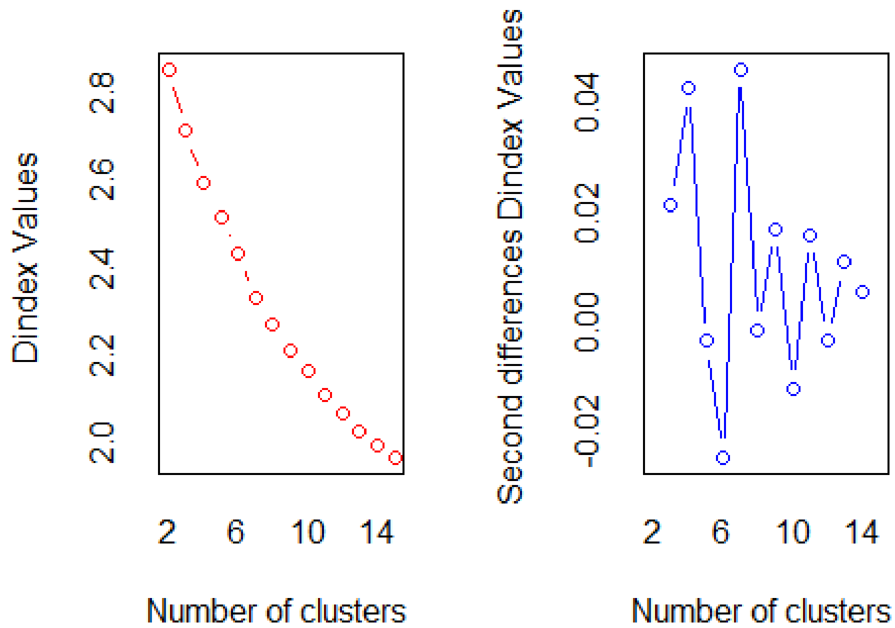
```
3
##   Brand.Runs Total.Volume No..of..Trans  Value Trans...Brand.Runs Vol.Tran
## 1         17         8025            24  818.0               1.41   334.38
## 2         25        13975            40 1681.5               1.60   349.38
## 3         37        23100            63 1950.0               1.70   366.67
## 4          4         1500             4  114.0               1.00   375.00
## 5          6         8300            13  591.0               2.17   638.46
## 6         26        18175            41 1705.5               1.58   443.29
##   Avg..Price Pur.Vol.No.Promo.... Pur.Vol.Promo.6.. Pur.Vol.Other.Promo..
## 1      10.19                 1.00              0.00                  0.00
## 2      12.03                 0.89              0.10                  0.02
## 3       8.44                 0.94              0.02                  0.04
## 4       7.60                 1.00              0.00                  0.00
## 5       7.12                 0.61              0.14                  0.24
## 6       9.38                 1.00              0.00                  0.00
##   Br..Cd..57..144 Br..Cd..55 Br..Cd..272 Br..Cd..286 Br..Cd..24
Br..Cd..481
## 1            0.38       0.13           0        0.00          0
0.00
## 2            0.02       0.08           0        0.00          0
0.06
## 3            0.03       0.55           0        0.03          0
0.00
## 4            0.40       0.60           0        0.00          0
0.00
## 5            0.05       0.14           0        0.00          0
0.00
## 6            0.08       0.07           0        0.00          0
0.00
##   Br..Cd..352 Br..Cd..5 Others.999 Pr.Cat.1 Pr.Cat.2 Pr.Cat.3 Pr.Cat.4
## 1           0      0.00      0.492     0.23     0.56     0.13     0.07
## 2           0      0.14      0.699     0.29     0.55     0.09     0.06
## 3           0      0.02      0.379     0.12     0.32     0.56     0.00
## 4           0      0.00      0.000     0.00     0.40     0.60     0.00
## 5           0      0.00      0.807     0.00     0.05     0.14     0.81
## 6           0      0.00      0.857     0.22     0.45     0.07     0.27
##   PropCat.5 PropCat.6 PropCat.7 PropCat.8 PropCat.9 PropCat.10 PropCat.11
## 1      0.50      0.00      0.00      0.00      0.00          0       0.00
## 2      0.46      0.35      0.03      0.02      0.01          0       0.06
## 3      0.24      0.12      0.03      0.01      0.01          0       0.00
## 4      0.40      0.00      0.00      0.00      0.00          0       0.00
## 5      0.81      0.00      0.00      0.05      0.00          0       0.00
## 6      0.49      0.10      0.00      0.01      0.07          0       0.00
##   PropCat.12 PropCat.13 PropCat.14 PropCat.15
## 1       0.03          0       0.13       0.34
## 2       0.00          0       0.08       0.00
## 3       0.02          0       0.56       0.00
## 4       0.00          0       0.60       0.00
## 5       0.00          0       0.14       0.00
## 6       0.00          0       0.07       0.27
```

Kmeans clustering

```
#The variables that describe purchase behavior (including brand loyalty)
BrandLoyality<-
select(BathSoapDF,"Br..Cd..57..144","Br..Cd..55","Br..Cd..272","Br..Cd..286",
"Br..Cd..24","Br..Cd..481","Br..Cd..352","Br..Cd..5")
BathSoapDF$Loyality <- apply(BrandLoyality,MARGIN = 1,FUN = max)
BathSoapDF$LoyalBrand_1_8 <- max.col(BrandLoyality)
Brand_Other <- select(BathSoapDF,"Loyality","Others.999")
BathSoapDF$Brand_1_Other_2_Loyal <- max.col(Brand_Other)

#1.Use k-means clustering to identify clusters of households based on:
# a.The variables that describe purchase behavior (including brand loyalty)
#Purchase behavior (volume, frequency, susceptibility to discounts, and brand
loyalty)

BSa <-
select(BathSoapDF,"No..of.Brands","Brand.Runs","Total.Volume","No..of..Trans"
,"Value",
              "Trans...Brand.Runs","Avg..Price",

"Pur.Vol.No.Promo....","Pur.Vol.Promo.6..","Pur.Vol.Other.Promo..","Brand_1_O
ther_2_Loyal","LoyalBrand_1_8")

#scaling the data set
BSaScale <- scale(BSa)

# Finding the Best Number Of Clusters Comparing with Most of the methods
NbClust(data = BSaScale,distance = "euclidean",min.nc = 2, max.nc = 15,
method = "kmeans")
```




```
## *** : The Hubert index is a graphical method of determining the number of clusters.
##                In the plot of Hubert index, we seek a significant knee that corresponds to a
##                significant increase of the value of the measure i.e the significant peak in Hubert
##                index second differences plot.
##
```



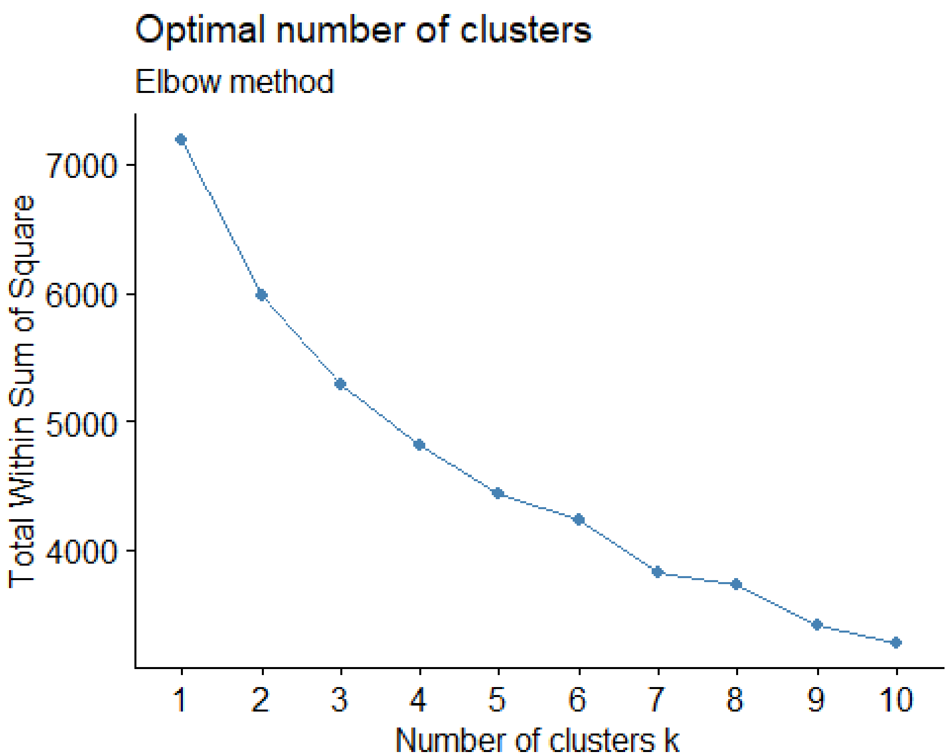

```
## *** : The D index is a graphical method of determining the number of clusters.
##                  In the plot of D index, we seek a significant knee (the significant peak in Dindex
##                  second differences plot) that corresponds to a significant increase of the value of
##                  the measure.
##
## *******************************************************************
## * Among all indices:
## * 5 proposed 2 as the best number of clusters
## * 5 proposed 3 as the best number of clusters
## * 1 proposed 4 as the best number of clusters
## * 1 proposed 6 as the best number of clusters
## * 3 proposed 7 as the best number of clusters
## * 1 proposed 9 as the best number of clusters
## * 1 proposed 10 as the best number of clusters
## * 5 proposed 14 as the best number of clusters
## * 1 proposed 15 as the best number of clusters
##
##                     ***** Conclusion *****
##
## * According to the majority rule, the best number of clusters is  2
##
##
## *******************************************************************
```

```
## $All.index
##        KL       CH Hartigan     CCC    Scott      Marriot    TrCovW
TraceW
## 2  1.3245 121.6933  78.6948 -4.9109  629.945 6.643862e+27 412380.92
5972.578
## 3  1.4398 108.0203  55.4463 -6.8143 1148.336 6.300557e+27 294451.52
5278.010
## 4  1.2230  97.0211  44.4244 -7.1779 1515.290 6.076402e+27 224896.07
4829.473
## 5  1.7472  89.1459  31.0086 -6.4935 1914.282 4.882769e+27 192260.27
4494.466
## 6  0.3197  81.0986  57.4932 -6.4605 2491.526 2.686618e+27 181217.54
4271.838
## 7  1.4986  83.5647  42.5980 -1.1425 3207.911 1.108058e+27 154812.99
3894.856
## 8  1.0320  82.7179  41.3446  2.5855 3270.846 1.303145e+27 133955.69
3633.821
## 9  1.5613  82.4615  29.9749  6.7576 3590.680 9.678197e+26 112685.74
3396.606
## 10 1.2645  80.2114  25.2248  6.9942 3966.501 6.386759e+26 100733.77
3232.649
## 11 1.5472  77.6672  18.9463  8.7804 4231.248 4.970907e+26  95564.06
3100.108
## 12 0.5722  74.4735  26.4804  9.5960 4393.121 4.516953e+26  88472.59
3003.495
## 13 1.5015  73.4234  19.9002 11.8635 4652.514 3.440446e+26  79521.40
2874.062
## 14 2.6844  71.4819  11.4981 13.0854 4842.020 2.909476e+26  73969.14
2779.822
## 15 0.8947  68.3829  11.6376 12.9766 4978.479 2.660538e+26  71439.89
2726.328
##     Friedman  Rubin Cindex     DB Silhouette   Duda  Pseudot2   Beale
Ratkowsky
## 2   280.8452 1.2035 0.2065 2.2140     0.1746 1.1228  -41.5750 -0.8907
0.2558
## 3  1620.7853 1.3619 0.2201 1.8938     0.1796 1.0782  -25.8818 -0.5900
0.2723
## 4  1624.8484 1.4884 0.2080 1.9148     0.1467 1.2475  -52.1792 -1.6132
0.2737
## 5  1638.6195 1.5993 0.1991 1.8538     0.1422 0.7927   46.5528  2.1259
0.2608
## 6  1631.1230 1.6826 0.1937 1.8033     0.1392 1.8671 -129.1055 -3.7676
0.2548
## 7  1641.8136 1.8455 0.1787 1.6898     0.1544 2.1572  -86.9041 -4.3497
0.2502
## 8  2048.1705 1.9781 0.2193 1.5967     0.1511 1.2564  -33.8741 -1.6551
0.2466
## 9  2053.9962 2.1162 0.2106 1.5611     0.1611 1.6296  -56.7947 -3.1199
0.2397
## 10 2049.2589 2.2236 0.2119 1.4885     0.1683 1.3734  -40.2373 -2.1939
```

```
0.2340
## 11 2023.2354 2.3186 0.2078 1.5303     0.1568 1.4987  -46.9214 -2.6897
0.2270
## 12 2036.1861 2.3932 0.2064 1.5591     0.1470 1.7624  -43.6911 -3.4918
0.2198
## 13 2260.5658 2.5010 0.2018 1.5183     0.1508 1.2883  -24.3929 -1.8073
0.2143
## 14 2271.9805 2.5858 0.1974 1.4873     0.1547 1.7073  -55.5135 -3.3427
0.2087
## 15 2457.6753 2.6365 0.1959 1.5051     0.1477 1.4138  -26.0510 -2.3523
0.2028
##         Ball Ptbiserial    Frey McClain   Dunn Hubert SDindex Dindex
SDbw
## 2  2986.2889     0.2452 -0.1731  0.8004 0.0370  2e-04  1.7279 2.8561
0.9338
## 3  1759.3366     0.3357  0.6283  1.1013 0.0428  3e-04  1.7394 2.7148
1.0791
## 4  1207.3683     0.3284  0.0232  1.7667 0.0285  4e-04  1.8106 2.5949
1.0257
## 5   898.8932     0.3508  1.0079  1.9687 0.0316  4e-04  1.7972 2.5176
1.0050
## 6   711.9730     0.3202 -0.1460  2.7340 0.0428  5e-04  1.8088 2.4369
0.9427
## 7   556.4080     0.3721  0.3222  2.7058 0.0227  6e-04  1.8445 2.3315
0.9061
## 8   454.2276     0.3667  0.0529  3.0682 0.0556  6e-04  1.8726 2.2721
0.9648
## 9   377.4007     0.3762  0.1878  3.2353 0.0556  7e-04  1.7937 2.2112
0.9161
## 10  323.2650     0.3756  0.7317  3.4636 0.0575  7e-04  1.7705 2.1674
0.8440
## 11  281.8280     0.3554  2.4689  4.0600 0.0560  7e-04  1.8375 2.1115
0.7970
## 12  250.2912     0.3247  0.0779  4.9774 0.0560  8e-04  1.8708 2.0715
0.7570
## 13  221.0817     0.3262  0.1674  5.1436 0.0560  8e-04  1.7150 2.0282
0.7080
## 14  198.5587     0.3234  0.8617  5.4489 0.0338  8e-04  1.7142 1.9957
0.6969
## 15  181.7552     0.3114 -1.3717  5.9696 0.0537  8e-04  1.8484 1.9688
0.6773
##
## $All.CriticalValues
##    CritValue_Duda CritValue_PseudoT2 Fvalue_Beale
## 2          0.8776            52.9942       1.0000
## 3          0.8745            51.2440       1.0000
## 4          0.8650            41.0445       1.0000
## 5          0.8619            28.5210       0.0129
## 6          0.8456            50.7520       1.0000
## 7          0.8415            30.5234       1.0000
```

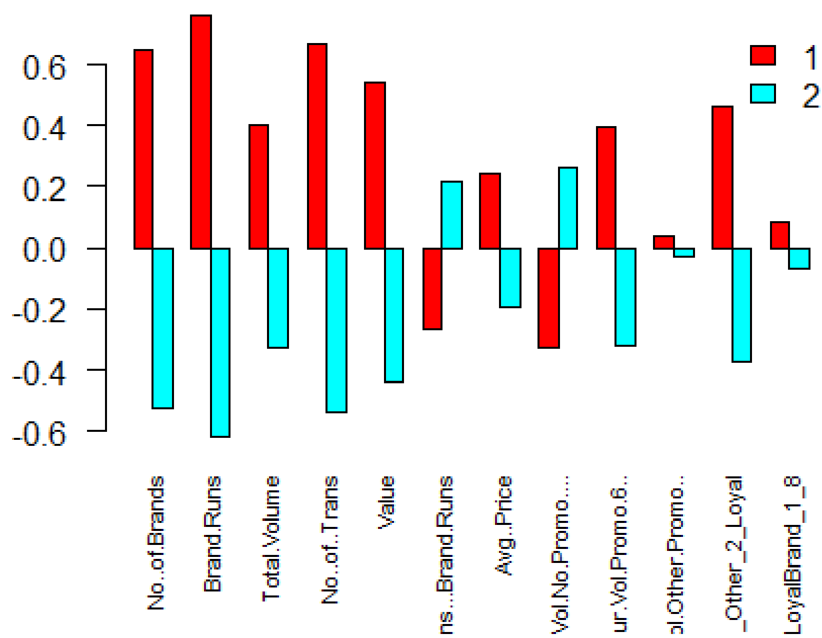
```
## 8          0.8436            30.7752       1.0000
## 9          0.8147            33.4329       1.0000
## 10         0.8109            34.5094       1.0000
## 11         0.8202            30.9171       1.0000
## 12         0.8125            23.3120       1.0000
## 13         0.8154            24.6722       1.0000
## 14         0.8101            31.4077       1.0000
## 15         0.7893            23.7638       1.0000
##
## $Best.nc
##                      KL       CH Hartigan     CCC    Scott      Marriot
## Number_clusters 14.0000   2.0000   6.0000 14.0000   7.0000 7.000000e+00
## Value_Index      2.6844 121.6933  26.4846 13.0854 716.3854 1.773647e+27
##                    TrCovW   TraceW Friedman   Rubin Cindex      DB
Silhouette
## Number_clusters      3.0   3.0000     3.00 14.0000 7.0000 14.0000
3.0000
## Value_Index     117929.4 246.0316  1339.94 -0.0341 0.1787  1.4873
0.1796
##                    Duda PseudoT2   Beale Ratkowsky     Ball PtBiserial Frey
## Number_clusters 2.0000    2.000  2.0000    4.0000    3.000     9.0000    1
## Value_Index     1.1228  -41.575 -0.8907    0.2737 1226.952     0.3762   NA
##                 McClain    Dunn Hubert SDindex Dindex    SDbw
## Number_clusters  2.0000 10.0000      0 14.0000      0 15.0000
## Value_Index      0.8004  0.0575      0  1.7142      0  0.6773
##
## $Best.partition
##   [1] 1 2 2 1 1 2 1 1 2 1 2 2 1 1 1 2 2 1 1 2 1 1 1 1 1 2 1 1 1 1 1 1 1 1
1 2 1
##  [38] 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 2 1 1 2 2 1 1 2 2 1 1 2 1 2 2 1 1 1
2 1 1
##  [75] 1 1 1 1 2 1 1 2 1 2 1 2 2 2 1 1 2 2 1 2 2 2 1 2 1 1 1 1 2 2 1 2 2 1
2 2 1
## [112] 1 1 2 2 2 2 1 2 2 2 2 1 2 2 1 2 1 1 2 2 2 1 2 1 2 1 2 1 1 1 1 1 1 2
1 1 2
## [149] 2 1 1 1 1 1 2 2 2 1 1 2 1 1 2 2 1 1 2 1 2 1 2 2 1 1 2 1 1 1 1 1 1 2
2 2 2
## [186] 1 2 2 1 2 1 2 1 1 1 2 2 2 2 1 1 1 2 1 1 1 1 2 1 2 1 1 2 1 2 1 1 2 1
2 1 2
## [223] 1 1 2 1 1 2 2 1 1 2 1 1 1 1 2 1 1 1 2 2 1 1 1 1 1 1 1 1 1 1 2 1 2 1 1
1 2 2
## [260] 2 1 1 2 1 2 1 1 1 2 2 2 2 1 2 1 1 2 2 1 1 2 1 2 2 2 2 2 1 2 2 2 1 1
1 1 1
## [297] 1 2 2 2 1 1 1 2 2 1 2 1 1 1 2 2 2 2 2 2 1 1 2 1 2 2 2 2 1 1 2 1 1 1
2 1 2
## [334] 2 1 2 2 2 1 1 1 1 1 1 2 2 2 1 2 2 1 1 2 1 2 2 2 2 2 2 1 1 2 2 2 2 2
2 2 1
## [371] 1 1 1 1 1 1 1 1 2 2 1 1 2 2 2 1 1 2 2 2 2 2 1 1 2 1 2 1 1 1 1 2 1 2
1 1 1
## [408] 1 2 2 1 1 2 1 1 2 2 1 1 1 2 1 1 1 1 2 2 1 2 1 2 1 1 1 1 1 2 2 1 1 1
```

```
1 2 2
## [445] 1 1 1 2 1 1 1 1 1 1 1 2 1 2 2 2 2 2 2 1 2 2 2 1 1 2 2 1 2 2 2 2 2 2
1 2 1
## [482] 2 2 1 1 1 2 2 2 2 1 2 1 1 2 2 1 2 2 2 1 2 1 1 1 1 1 1 2 2 1 1 1 2 2
1 1 2
## [519] 1 2 1 1 1 2 1 1 2 1 2 1 2 1 2 1 2 1 1 1 1 1 2 2 2 2 2 1 1 1 2 2 1 1
1 2 2
## [556] 1 1 2 2 1 2 1 2 2 2 2 2 2 1 2 1 2 1 2 2 1 2 1 1 2 2 1 2 2 1 2 2 2 1
2 1 1
## [593] 1 1 1 1 2 2 1 1

# According to the majority rule, the best number of clusters is  2

# Elbow method Approache

fviz_nbclust(BSaScale, kmeans, method = "wss")+labs(subtitle = "Elbow
method")
```

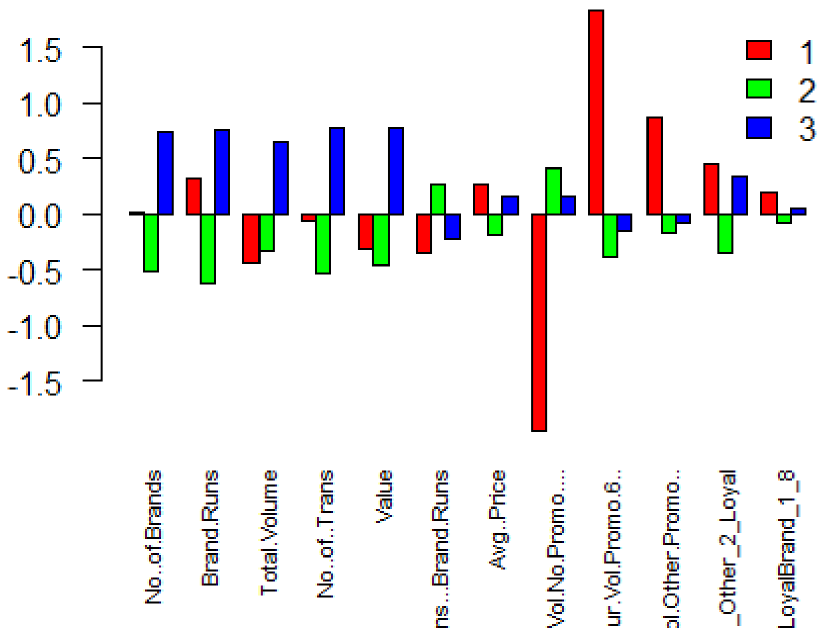



```
#lets check the results with 2 clusters

BSaK <- kmeans(BSaScale,centers = 2,nstart = 25)

library(cluster)
BSaK$size

## [1] 269 331
```

```
barplot(BSaK$centers,beside = TRUE,col = rainbow(2),cex.names = .7,las=2,
        args.legend = list(x ='topright', bty='n'),legend.text =
(1:nrow(BSaK$centers)))
```



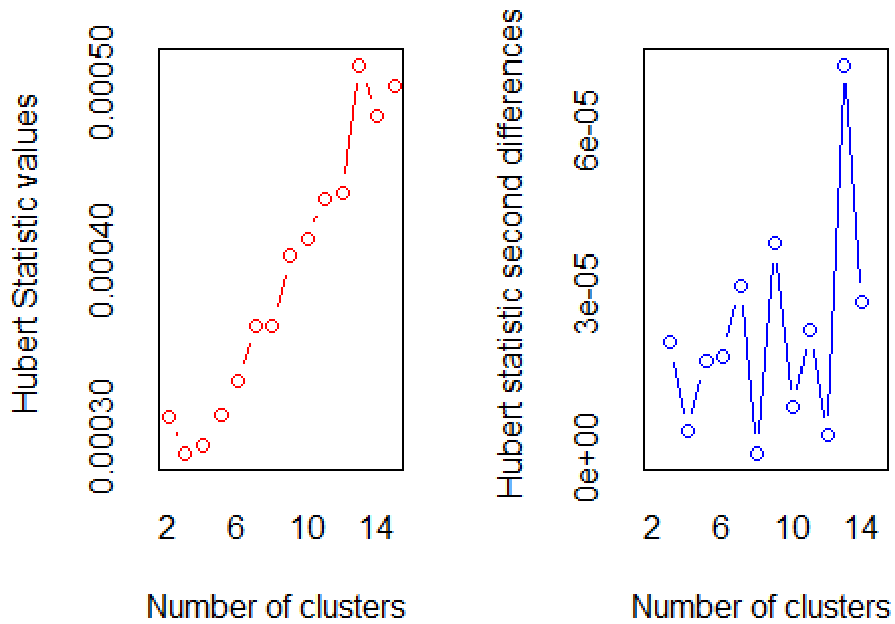

```
#lets check the results with 3 clusters

BSaK <- kmeans(BSaScale,centers = 3,nstart = 25)

library(cluster)
BSaK$size

## [1]  81 302 217

barplot(BSaK$centers,beside = TRUE,col = rainbow(3),cex.names = .7,las=2,
        args.legend = list(x ='topright', bty='n'),legend.text =
(1:nrow(BSaK$centers)))
```



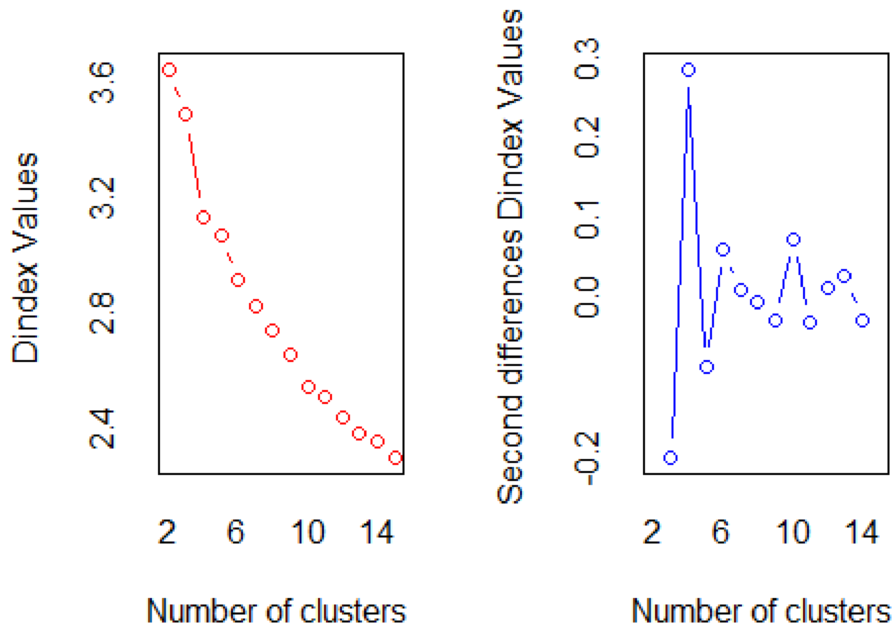

```
# from the above two clustering results, it reflects that the best cluster
count would be 2. Cluster-2 is loyal to the brands as the "No. of Brands"
purchased is on lower side. It also reflects that the customers are not
looking for promotions as they are loyal to the brands. Cluster-1 is not
loyal to brands and frequently changes to different brands.
```

#b.The variables that describe the basis for purchase

```
#Basis of purchase (price, selling proposition)
BSb <- select(BathSoapDF,"Pr.Cat.1","Pr.Cat.2","Pr.Cat.3","Pr.Cat.4",

"PropCat.5","PropCat.6","PropCat.7","PropCat.8","PropCat.9","PropCat.10",

"PropCat.11","PropCat.12","PropCat.13","PropCat.14","PropCat.15",

"Pur.Vol.No.Promo....","Pur.Vol.Promo.6..","Pur.Vol.Other.Promo..")
#BSb
#Scaling the Data
BSbScale <- scale(BSb)

# Finding the Best Number Of Clusters Comparing with Most of the methods
NbClust(data = BSbScale,distance = "euclidean",min.nc = 2, max.nc = 15,
method = "kmeans")
```




```
## *** : The Hubert index is a graphical method of determining the number of clusters.
##                   In the plot of Hubert index, we seek a significant knee that corresponds to a
##                   significant increase of the value of the measure i.e the significant peak in Hubert
##                   index second differences plot.
##
```



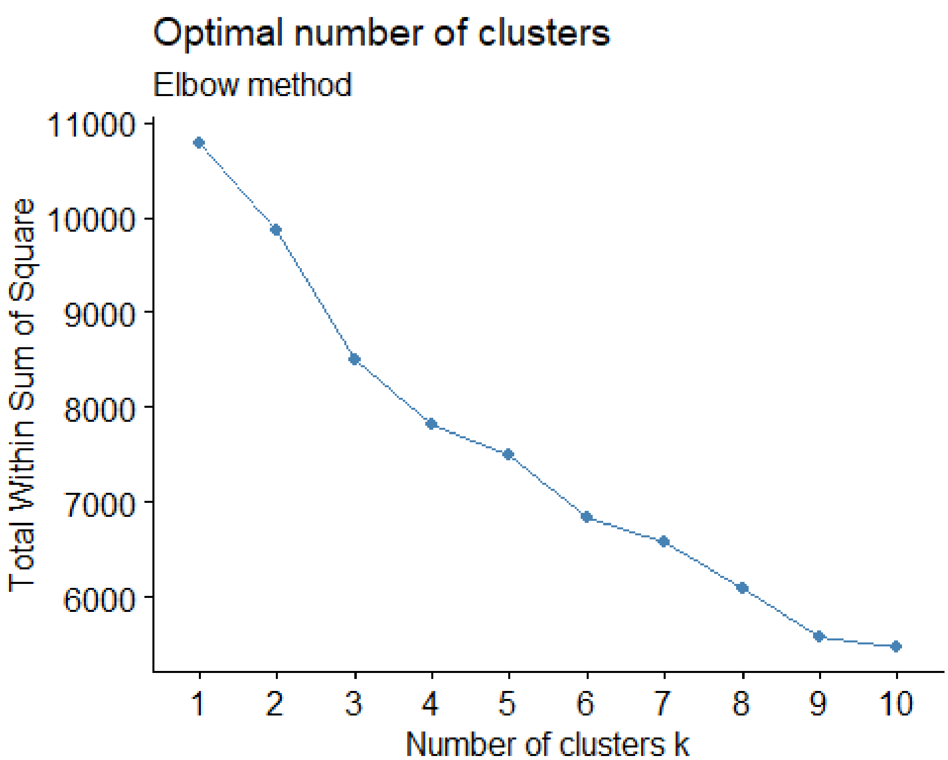

```
## *** : The D index is a graphical method of determining the number of
clusters.
##                 In the plot of D index, we seek a significant knee (the
significant peak in Dindex
##                 second differences plot) that corresponds to a significant
increase of the value of
##                 the measure.
##
## *******************************************************************
## * Among all indices:
## * 4 proposed 2 as the best number of clusters
## * 1 proposed 3 as the best number of clusters
## * 6 proposed 4 as the best number of clusters
## * 2 proposed 6 as the best number of clusters
## * 2 proposed 10 as the best number of clusters
## * 1 proposed 11 as the best number of clusters
## * 1 proposed 12 as the best number of clusters
## * 6 proposed 13 as the best number of clusters
##
##                    ***** Conclusion *****
##
## * According to the majority rule, the best number of clusters is  4
##
##
## *******************************************************************
```

```
## $All.index
##         KL      CH Hartigan     CCC     Scott      Marriot    TrCovW
TraceW
## 2  0.2965 55.3667  53.0274 -8.7999  630.7445 6.819258e+36 498313.33
9868.326
## 3  0.3665 56.5571  96.4249 -7.3074 1444.3222 3.953962e+36 426083.94
9064.533
## 4  5.4313 75.8091  29.3513  6.4321 2496.7547 1.216561e+36 285242.24
7804.056
## 5  0.3407 66.8823  57.2734  6.7894 3031.0413 7.802331e+35 263711.62
7437.767
## 6  1.7304 69.9930  37.3615 16.3637 4108.1212 1.866253e+35 227184.96
6784.687
## 7  0.9979 68.1083  37.1870 22.2624 4711.4427 9.293202e+34 206321.54
6383.196
## 8  0.9898 67.2383  37.7379 28.8713 5407.1454 3.807000e+34 190510.06
6006.527
## 9  0.7410 67.1875  50.1464 36.2256 5781.3992 2.582224e+34 166934.28
5646.578
## 10 1.8443 70.2424  30.5843 47.5442 6508.4173 9.490208e+33 145329.22
5204.938
## 11 0.6974 69.4358  42.6349 54.0049 6784.7676 7.244894e+33 127861.81
4948.423
## 12 0.7734 71.4470  57.0200 64.3995 7383.9834 3.176013e+33 112623.18
4614.408
## 13 2.1071 76.4656 -18.6824 79.3419 8219.6312 9.258582e+32  96147.72
4206.493
## 14 0.9888 66.7861  29.2534 70.8544 7902.0137 1.823111e+33  99237.05
4344.774
## 15 0.4141 67.0863  70.5464 75.8040 8270.3814 1.132677e+33  90552.36
4138.194
##     Friedman  Rubin Cindex     DB Silhouette   Duda  Pseudot2   Beale
Ratkowsky
## 2   2456.081 1.0926 0.2323 3.0147     0.1862 1.2784  -48.5655 -2.7012
0.1548
## 3   7853.092 1.1895 0.2187 2.5493     0.1668 2.2587 -249.0964 -6.9058
0.1932
## 4  16346.042 1.3816 0.2028 2.1424     0.2099 1.2667  -32.4246 -2.6078
0.2220
## 5  16805.180 1.4496 0.1904 2.1674     0.2091 0.3630  112.3142 21.5141
0.2210
## 6  19921.901 1.5892 0.1876 1.7951     0.2176 3.4768 -101.8698 -8.7825
0.2197
## 7  19958.433 1.6891 0.1713 1.8083     0.2220 1.7821  -52.6632 -5.4114
0.2210
## 8  20336.638 1.7950 0.1687 1.7009     0.2277 3.2584 -187.1376 -8.5434
0.2175
## 9  19974.427 1.9095 0.1543 1.6657     0.2289 3.3396  -37.8302 -8.1671
0.2206
## 10 20613.236 2.0715 0.1460 1.5888     0.2526 1.3959  -32.6185 -3.4943
```

```
0.2188
## 11 20459.026 2.1789 0.1796 1.5148     0.2579 2.5327  -74.4358 -7.4494
0.2166
## 12 19961.773 2.3366 0.1782 1.4621     0.2693 2.6224  -78.5708 -7.5845
0.2141
## 13 21055.662 2.5632 0.1728 1.3697     0.2693 3.4932  -87.0747 -8.7401
0.2145
## 14 22411.438 2.4816 0.1717 1.5334     0.2624 0.4892  111.7206 12.8606
0.2025
## 15 22803.090 2.6055 0.1635 1.4668     0.2639 3.4462  -31.9421 -8.6388
0.1998
##         Ball Ptbiserial    Frey McClain   Dunn Hubert SDindex Dindex
SDbw
## 2  4934.1631     0.2995  0.6969  0.4476 0.0450  3e-04  1.4709 3.6496
1.4504
## 3  3021.5111     0.3296  0.0442  0.8819 0.0371  3e-04  1.3549 3.4931
1.3563
## 4  1951.0140     0.3883  0.2435  1.1886 0.0369  3e-04  1.2596 3.1377
1.1492
## 5  1487.5534     0.4096 -0.4025  1.5299 0.0369  3e-04  1.3437 3.0689
1.3291
## 6  1130.7812     0.4263  0.2423  1.5059 0.0369  3e-04  1.1851 2.9157
1.0926
## 7   911.8851     0.4416  0.4476  1.9811 0.0369  4e-04  1.2221 2.8244
1.1749
## 8   750.8159     0.4381  0.2235  2.1147 0.0351  4e-04  1.1904 2.7428
1.1384
## 9   627.3975     0.4410  0.0309  2.6121 0.0351  4e-04  1.2842 2.6574
1.2643
## 10  520.4938     0.4557 -0.0430  2.7055 0.0334  4e-04  1.1620 2.5444
1.1114
## 11  449.8566     0.4677  0.2074  2.7018 0.0425  4e-04  1.2899 2.5062
1.1844
## 12  384.5340     0.4669  0.1846  2.8027 0.0452  4e-04  1.2130 2.4376
1.1144
## 13  323.5764     0.4662  1.9080  2.9286 0.0452  5e-04  1.5073 2.3817
1.2505
## 14  310.3410     0.4486  0.0722  3.2062 0.0452  5e-04  1.3388 2.3544
1.1693
## 15  275.8796     0.4543  3.2220  3.2966 0.0431  5e-04  1.3324 2.2993
1.1798
##
## $All.CriticalValues
##    CritValue_Duda CritValue_PseudoT2 Fvalue_Beale
## 2          0.8778            31.0535            1
## 3          0.8725            65.3217            1
## 4          0.8695            23.1180            1
## 5          0.8241            13.6625            0
## 6          0.8466            25.9096            1
## 7          0.8474            21.6170            1
```

```
## 8          0.8458           49.2083           1
## 9          0.7040           22.7003           1
## 10         0.8435           21.3433           1
## 11         0.8400           23.4208           1
## 12         0.8241           27.1115           1
## 13         0.8201           26.7658           1
## 14         0.8426           19.9831           0
## 15         0.7998           11.2628           1
##
## $Best.nc
##                     KL      CH Hartigan     CCC   Scott      Marriot
TrCovW
## Number_clusters 4.0000 13.0000  13.0000 13.0000    6.00 4.000000e+00
4.0
## Value_Index     5.4313 76.4656  75.7024 79.3419 1077.08 2.301073e+36
140841.7
##                   TraceW Friedman   Rubin Cindex      DB Silhouette   Duda
## Number_clusters   4.0000     4.00 13.0000 10.000 13.0000    13.0000 2.0000
## Value_Index     894.1887  8492.95 -0.3082  0.146  1.3697     0.2693 1.2784
##                 PseudoT2   Beale Ratkowsky     Ball PtBiserial Frey
McClain
## Number_clusters   2.0000  2.0000     4.000    3.000    11.0000    1
2.0000
## Value_Index     -48.5655 -2.7012     0.222 1912.652     0.4677   NA
0.4476
##                    Dunn Hubert SDindex Dindex   SDbw
## Number_clusters 12.0000      0  10.000      0 6.0000
## Value_Index      0.0452      0   1.162      0 1.0926
##
## $Best.partition
##   [1] 4 4 2 2 3 4 4 2 4 1 3 4 4 4 4 4 4 4 4 3 2 2 2 2 2 1 4 4 4 2 2 4 2 2
2 4 4
##  [38] 4 2 2 4 2 2 2 4 4 4 4 2 1 2 4 2 3 2 4 2 4 2 4 4 2 2 4 4 2 2 4 4 4 4
4 2 4
##  [75] 4 1 4 2 4 4 4 2 2 1 3 4 3 4 1 2 1 4 2 1 4 4 2 4 2 4 4 4 4 3 4 4 1 4
4 3 2
## [112] 4 4 3 3 1 4 2 4 4 3 1 4 1 4 2 4 4 4 1 4 4 1 4 2 1 4 4 4 3 4 2 4 2 1
2 2 3
## [149] 4 3 4 4 2 2 4 1 1 2 4 2 2 2 4 4 4 4 4 4 4 4 4 4 4 2 4 4 4 2 2 4 4 4
4 3 4
## [186] 4 4 4 4 3 4 1 4 4 4 4 4 4 1 4 4 4 4 4 2 4 4 4 2 4 4 3 3 4 4 1 4 3 2
1 4 4
## [223] 2 1 4 4 4 4 1 3 2 4 2 4 2 2 2 2 2 1 4 4 3 4 2 4 2 1 1 4 4 3 4 4 4 4
4 3 4
## [260] 4 1 4 4 4 4 4 4 4 4 1 4 4 4 4 4 3 4 1 4 3 4 4 4 4 4 4 2 4 4 1 4 1 4
3 4 4
## [297] 2 4 4 1 4 4 3 4 1 4 4 4 3 1 1 3 4 1 4 4 4 4 4 1 2 3 4 4 4 2 4 2 4 4
4 4 4
## [334] 4 4 3 4 4 4 4 4 4 4 1 4 3 3 1 4 4 4 1 1 4 1 4 1 4 1 3 1 4 4 1 1 3 3
4 1 4
```

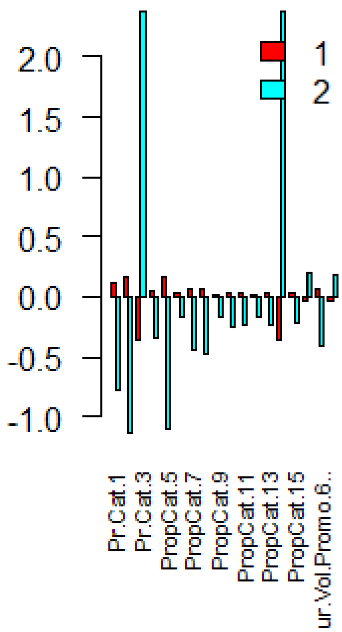
```
## [371] 4 1 4 4 2 1 4 1 4 4 1 1 3 1 4 1 4 4 4 4 1 1 1 1 3 1 1 3 1 4 4 1 4 1
3 1 1
## [408] 4 4 4 4 1 1 4 1 3 4 1 4 4 4 4 1 4 4 4 3 1 1 4 4 4 4 1 1 4 3 1 4 1 4
1 4 1
## [445] 1 4 4 4 4 4 1 4 4 4 4 4 4 1 3 1 3 3 3 1 3 2 4 4 1 1 3 1 1 4 4 4 4 3
1 1 1
## [482] 3 4 1 1 4 1 4 1 1 3 4 1 4 1 1 1 1 4 4 2 3 4 4 1 1 4 4 4 3 1 4 4 3 3
4 1 3
## [519] 1 4 4 3 4 4 4 4 4 4 4 4 3 2 4 4 4 4 2 4 4 4 4 3 4 1 3 4 4 4 4 1 4 4
1 4 4
## [556] 4 4 4 3 1 3 4 1 4 3 3 4 3 4 1 4 4 1 3 4 2 1 1 1 3 3 1 4 3 4 3 4 4 4
4 4 1
## [593] 3 4 4 3 1 4 3 4

# According to the majority rule, the best number of clusters is  2

# Traditional Approches
fviz_nbclust(BSbScale, kmeans, method = "wss")+labs(subtitle = "Elbow
method")
```




```
# 2,9 are the best cluster from 3 approches
BSbK <- kmeans(BSbScale,centers = 2,nstart = 25)


# Allow us to represent the cluster solution into 2 dimensions
#'2D-Representation of the Cluster solution - Purchase Behavior')
BSaK$size
```

```
## [1]  81 302 217
```

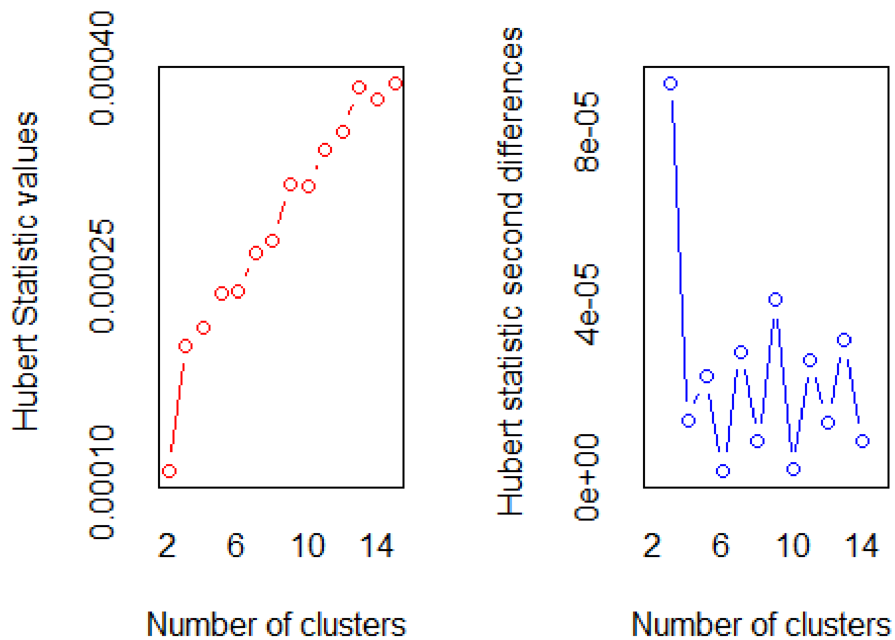
```
#dev.off()
barplot(BSbK$centers,beside = TRUE,col = rainbow(2),cex.names = .7,las=2,
        args.legend = list(x ='topright', bty='n'),legend.text =
(1:nrow(BSbK$centers)))
#from the bar plot its clear that the Cluster-2 customers purchases under the
product proposition category 3 and 14.
```



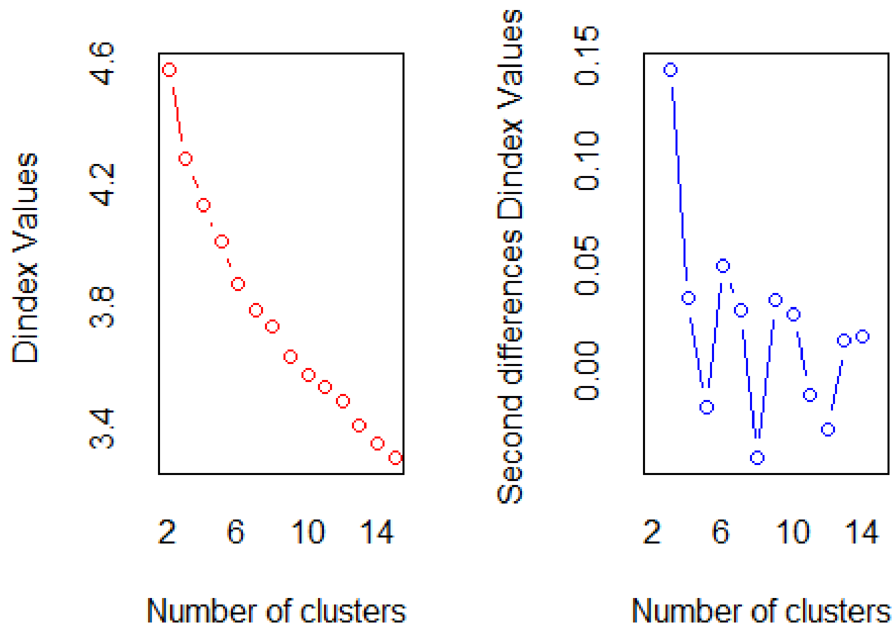
c.The variables that describe both purchase behavior and basis of purchase

```
BSc <-
select(BathSoapDF,"No..of.Brands","Brand.Runs","Total.Volume","No..of..Trans"
,"Value",
              "Trans...Brand.Runs","Avg..Price",

"Pur.Vol.No.Promo....","Pur.Vol.Promo.6..","Pur.Vol.Other.Promo..","Brand_1_O
ther_2_Loyal","LoyalBrand_1_8",
              "Pr.Cat.1","Pr.Cat.2","Pr.Cat.3","Pr.Cat.4",

"PropCat.5","PropCat.6","PropCat.7","PropCat.8","PropCat.9","PropCat.10",

"PropCat.11","PropCat.12","PropCat.13","PropCat.14","PropCat.15",

"Pur.Vol.No.Promo....","Pur.Vol.Promo.6..","Pur.Vol.Other.Promo..")

#scaling the data set
```

```r
BScScale <- scale(BSc)

NbClust(data = BScScale,distance = "euclidean",min.nc = 2, max.nc = 15,
method = "kmeans")
```




```
## *** : The Hubert index is a graphical method of determining the number of
clusters.
##                 In the plot of Hubert index, we seek a significant knee
that corresponds to a
##                 significant increase of the value of the measure i.e the
significant peak in Hubert
##                 index second differences plot.
##
```




```
## *** : The D index is a graphical method of determining the number of
clusters.
##                 In the plot of D index, we seek a significant knee (the
significant peak in Dindex
##                 second differences plot) that corresponds to a significant
increase of the value of
##                 the measure.
##
## *******************************************************************
## * Among all indices:
## * 4 proposed 2 as the best number of clusters
## * 7 proposed 3 as the best number of clusters
## * 1 proposed 4 as the best number of clusters
## * 1 proposed 11 as the best number of clusters
## * 6 proposed 13 as the best number of clusters
## * 4 proposed 15 as the best number of clusters
##
##                    ***** Conclusion *****
##
## * According to the majority rule, the best number of clusters is  3
##
##
## *******************************************************************
```

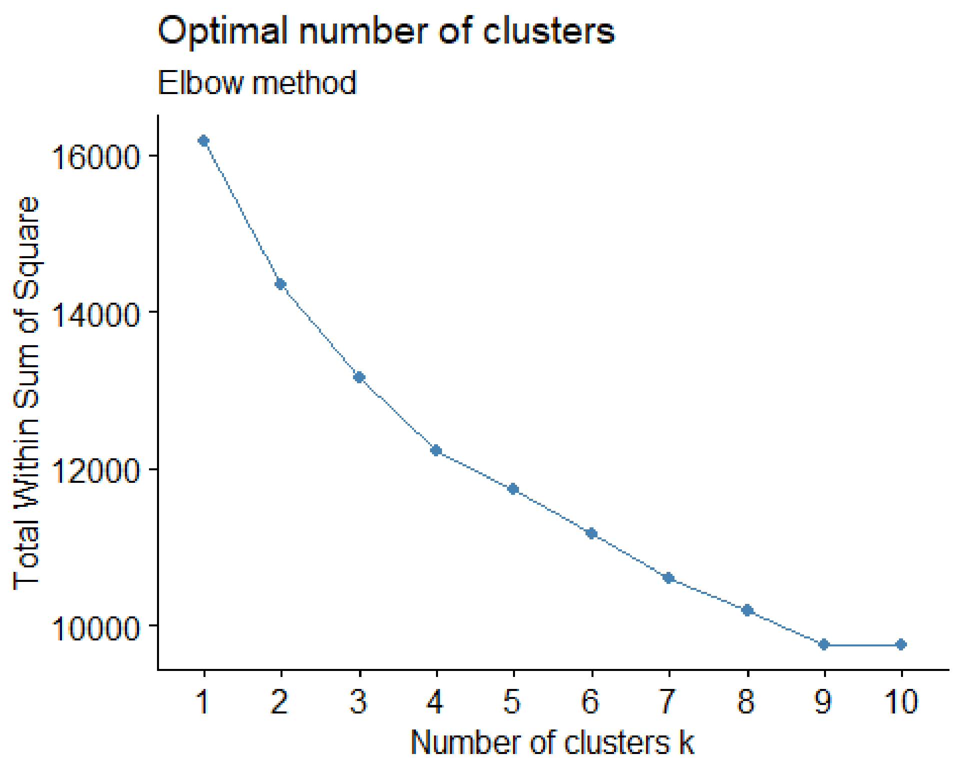
```
## $All.index
##         KL       CH Hartigan     CCC     Scott      Marriot    TrCovW
TraceW
```

```
## 2  0.7661 65.3654  64.7677 -5.6338  772.3769 5.268605e+57 539106.8
14579.377
## 3  1.5011 68.4916  45.1633 -0.1319 1946.9655 1.673719e+57 431654.2
13154.634
## 4  1.3175 64.0623  35.3155  3.1821 2496.2444 1.191183e+57 353455.4
12229.470
## 5  1.0094 59.6224  34.2018  5.7072 3005.8596 7.960254e+56 299964.1
11545.359
## 6  1.2569 57.1838  28.2555 10.4075 3861.6394 2.753312e+56 271462.9
10917.782
## 7  2.1125 54.5372  16.1820 14.4719 4448.9880 1.408030e+56 246995.0
10422.025
## 8  0.3858 50.2486  33.0193 15.7646 5173.5452 5.497228e+55 246909.7
10145.179
## 9  1.5353 50.4619  23.2772 22.7030 5713.8576 2.827207e+55 216479.9
9609.217
## 10 1.3087 49.1248  18.7417 27.2549 6302.8808 1.307746e+55 204044.0
9245.088
## 11 1.0997 47.4104  17.3209 30.7699 6569.8670 1.014046e+55 189715.0
8960.454
## 12 0.4313 45.8645  36.5029 34.1382 6838.6083 7.711047e+54 176164.7
8704.479
## 13 4.6229 47.6132  10.4426 43.9038 7764.1532 1.935111e+54 158869.3
8195.693
## 14 0.4808 45.4582  18.3591 45.4823 8049.1563 1.395671e+54 154157.6
8052.442
## 15 0.5612 44.7685  31.4402 49.9215 8483.8234 7.764026e+53 147395.9
7807.827
##     Friedman  Rubin Cindex     DB Silhouette   Duda  Pseudot2    Beale
## 2   4692.499 1.1093 0.3242 3.0066     0.1021 1.1921  -46.0855  -3.0475
## 3  13878.719 1.2295 0.3059 2.6517     0.1353 4.6182 -141.0234 -14.6641
## 4  15103.851 1.3225 0.3018 2.4174     0.1257 1.8051 -131.5762  -8.3888
## 5  16748.706 1.4008 0.2538 2.2527     0.1238 0.9748    5.5077   0.4883
## 6  20111.558 1.4813 0.2488 2.1007     0.1375 1.2356  -32.9906  -3.5978
## 7  20225.202 1.5518 0.2351 2.0335     0.1418 2.8387 -121.7729 -11.9945
## 8  20883.096 1.5942 0.2361 2.0797     0.1280 1.0899   -9.3226  -1.5537
## 9  20314.540 1.6831 0.2209 1.9699     0.1396 1.4404  -21.7097  -5.6687
## 10 20727.743 1.7494 0.2156 1.9958     0.1481 1.4388  -27.1429  -5.6475
## 11 20875.159 1.8049 0.2728 1.9158     0.1505 1.5369  -37.3809  -6.4768
## 12 21227.015 1.8580 0.2683 1.8897     0.1500 1.4324  -24.4511  -5.5899
## 13 21907.753 1.9734 0.2616 1.7147     0.1577 1.0999  -10.7128  -1.7060
## 14 22012.431 2.0085 0.2038 1.7896     0.1328 2.9028 -106.8472 -12.2864
## 15 22174.851 2.0714 0.1973 1.7638     0.1511 0.7695   24.2577   5.6257
##    Ratkowsky      Ball Ptbiserial    Frey McClain   Dunn Hubert SDindex
Dindex
## 2     0.1859 7289.6883     0.1586 -0.4231  0.8921 0.0761  1e-04  1.2252
4.5815
## 3     0.2085 4384.8780     0.2771  0.4616  1.1736 0.0909  2e-04  1.2480
4.2858
## 4     0.2165 3057.3675     0.2828  0.1295  1.9362 0.0606  2e-04  1.2236
```

```
4.1332
## 5     0.2148 2309.0717     0.3039  0.2900  2.5347 0.0809  2e-04  1.1929
4.0148
## 6     0.2094 1819.6303     0.3035 -0.0114  2.9447 0.0646  2e-04  1.1442
3.8796
## 7     0.2098 1488.8607     0.3358 -2.5055  3.3473 0.0619  3e-04  1.1645
3.7946
## 8     0.1993 1268.1474     0.3060 -0.0226  3.9700 0.0728  3e-04  1.1536
3.7382
## 9     0.2015 1067.6907     0.3389  0.1551  4.3465 0.0832  3e-04  1.1722
3.6405
## 10    0.1977  924.5088     0.3385 -0.0657  4.7673 0.0832  3e-04  1.1345
3.5764
## 11    0.1936  814.5868     0.3480  0.3741  4.7776 0.1069  3e-04  1.2460
3.5393
## 12    0.1894  725.3733     0.3370 -0.2236  5.3863 0.0877  4e-04  1.2316
3.4908
## 13    0.1888  630.4379     0.3480  0.7576  5.2168 0.0751  4e-04  1.2610
3.4148
## 14    0.1827  575.1744     0.3288 -0.0026  6.0538 0.0592  4e-04  1.1530
3.3530
## 15    0.1803  520.5218     0.3386  0.1780  6.2065 0.0683  4e-04  1.1125
3.3080
##      SDbw
## 2  1.0343
## 3  1.0131
## 4  1.0847
## 5  1.0826
## 6  1.0264
## 7  1.1543
## 8  0.9999
## 9  1.1191
## 10 1.0955
## 11 1.1430
## 12 1.1056
## 13 1.0576
## 14 0.8899
## 15 0.8762
##
## $All.CriticalValues
##    CritValue_Duda CritValue_PseudoT2 Fvalue_Beale
## 2          0.9250            23.1970       1.0000
## 3          0.8760            25.4727       1.0000
## 4          0.8950            34.6064       1.0000
## 5          0.9152            19.7479       0.9879
## 6          0.9100            17.1002       1.0000
## 7          0.8441            34.7312       1.0000
## 8          0.9009            12.4301       1.0000
## 9          0.8471            12.8144       1.0000
## 10         0.8441            16.4419       1.0000
```

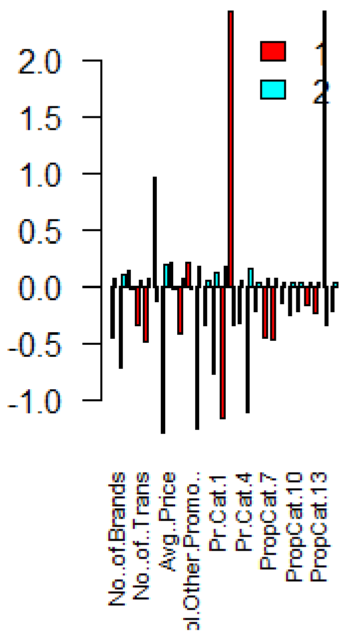
```
## 11         0.8471           19.3118       1.0000
## 12         0.8441           14.9640       1.0000
## 13         0.8911           14.4248       1.0000
## 14         0.8811           21.9928       1.0000
## 15         0.8898           10.0343       0.0000
##
## $Best.nc
##                      KL      CH Hartigan     CCC    Scott     Marriot
TrCovW
## Number_clusters 13.0000  3.0000  13.0000 15.0000    3.000 3.00000e+00
3.0
## Value_Index      4.6229 68.4916  26.0602 49.9215 1174.589 3.11235e+57
107452.5
##                   TraceW Friedman   Rubin  Cindex      DB Silhouette
Duda
## Number_clusters   3.0000     3.00 13.0000 15.0000 13.0000    13.0000
2.0000
## Value_Index     499.5785  9186.22 -0.0802  0.1973  1.7147     0.1577
1.1921
##                 PseudoT2   Beale Ratkowsky    Ball PtBiserial Frey McClain
## Number_clusters   2.0000  2.0000    4.0000    3.00     13.000    1  2.0000
## Value_Index     -46.0855 -3.0475    0.2165 2904.81      0.348   NA  0.8921
##                    Dunn Hubert SDindex Dindex    SDbw
## Number_clusters 11.0000      0 15.0000      0 15.0000
## Value_Index      0.1069      0  1.1125      0  0.8762
##
## $Best.partition
##   [1] 2 3 1 1 2 2 2 1 2 3 2 2 2 2 2 2 2 2 2 3 1 1 1 1 1 3 2 2 2 1 1 2 1 1
1 2 2
##  [38] 2 1 1 2 1 1 1 2 2 2 2 1 3 1 2 1 2 1 2 1 2 1 2 2 1 1 2 2 1 2 3 2 2 2
2 1 2
##  [75] 2 3 2 1 2 2 2 1 1 3 2 2 2 2 3 1 3 2 1 3 3 2 1 2 1 2 2 2 2 2 2 2 3 2
3 2 1
## [112] 2 2 3 3 3 3 1 3 2 2 3 2 3 2 1 2 2 2 3 3 3 3 2 1 3 2 2 2 2 2 1 2 1 3
1 1 2
## [149] 2 3 2 2 1 1 2 3 3 1 2 1 1 1 2 2 2 2 2 2 2 2 2 3 2 1 2 2 2 1 1 2 2 2
3 3 3
## [186] 2 2 2 2 3 2 3 2 2 2 2 2 2 3 2 2 2 2 2 1 2 2 2 2 3 2 2 3 2 2 3 2 3 1
3 2 2
## [223] 1 3 2 2 2 3 3 2 1 2 1 2 1 1 1 1 1 3 2 3 2 2 1 2 1 3 3 2 2 3 2 3 2 2
2 1 2
## [260] 2 3 2 3 2 3 2 2 2 2 3 2 2 2 2 3 3 3 3 2 2 2 2 2 2 2 2 2 2 2 3 3 3 2
2 2 2
## [297] 1 2 2 3 3 2 2 2 3 2 3 2 2 3 3 3 3 3 2 3 2 2 3 3 2 3 2 2 2 1 2 1 2 2
2 2 2
## [334] 3 2 3 3 3 2 2 2 2 2 3 2 3 3 3 2 2 2 2 3 2 3 2 3 2 3 3 3 2 2 3 3 3 3
3 3 2
## [371] 2 3 2 2 1 3 2 3 2 2 3 3 3 3 2 3 2 3 2 3 3 3 3 2 3 3 3 3 3 2 2 3 2 3
3 3 3
## [408] 2 3 2 2 3 3 2 3 3 2 3 2 2 2 2 3 2 2 3 3 3 3 2 2 2 2 3 3 2 3 3 2 3 2
```

```
3 2 3
## [445] 3 2 2 2 2 2 3 2 2 2 2 2 3 3 3 3 3 3 3 3 3 2 2 2 3 3 3 3 3 3 2 2 2 3
3 3 3
## [482] 3 3 3 3 2 3 2 3 3 3 3 2 2 3 3 3 3 2 2 1 3 2 2 3 3 2 2 2 3 3 2 2 3 3
2 3 3
## [519] 3 2 2 3 2 2 2 2 2 2 2 2 3 1 2 2 3 2 1 2 2 2 3 3 2 3 3 2 2 2 2 3 2 3
3 2 2
## [556] 2 2 3 3 3 3 2 3 2 3 3 2 3 2 3 2 2 3 3 3 1 3 3 3 3 3 3 2 3 2 3 2 2 3
3 2 3
## [593] 3 2 2 2 3 2 2 2

# According to the majority rule, the best number of clusters is  2

# Traditional Approches
fviz_nbclust(BScScale, kmeans, method = "wss")+labs(subtitle = "Elbow
method")
```



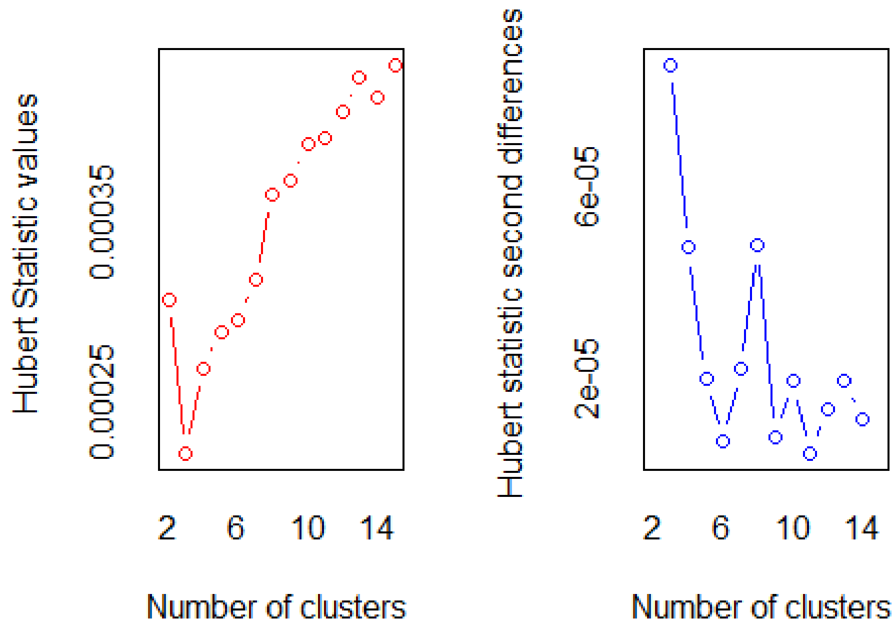

```
#2 is the best cluster from 3 approches
BScK <- kmeans(BScScale,centers =2,nstart = 25)

# Allow us to represent the cluster solution into 2 dimensions
#clusplot(BScScale,BScK$cluster,lines = 0,labels = 2,shade = TRUE,color =
TRUE,main = #'2D-Representation of the Cluster solution - Purchase Behavior')
BScK$size

## [1]  75 525
```

```r
#dev.off()
barplot(BScK$centers,beside = TRUE,col = rainbow(2),cex.names = .7,las=2,
        args.legend = list(x ='topright', bty='n'),legend.text =
(1:nrow(BScK$centers)))
```

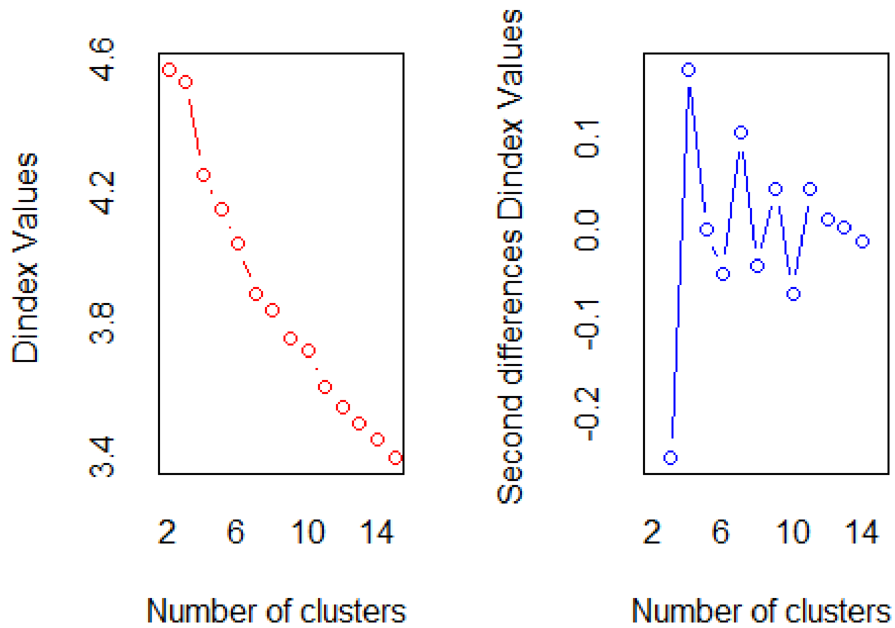


2.Select what you think is the best segmentation and comment on the characteristics (demographic,brand loyalty, and basis for purchase) of these clusters.

```r
BS2 <-
select(BathSoapDF,"No..of.Brands","Brand.Runs","Total.Volume","No..of..Trans"
,"Value",

"Trans...Brand.Runs","Avg..Price","Brand_1_Other_2_Loyal","LoyalBrand_1_8",

"Pur.Vol.No.Promo....","Pur.Vol.Promo.6..","Pur.Vol.Other.Promo..",
              "SEC","FEH","MT","SEX","EDU","HS","CS",

"Br..Cd..57..144","Br..Cd..55","Br..Cd..272","Br..Cd..286","Br..Cd..24","Br..
Cd..481","Br..Cd..352","Br..Cd..5","Others.999")

BS2Scale <- scale(BS2)
NbClust(data = BS2Scale,distance = "euclidean",min.nc = 2, max.nc = 15,
method = "kmeans")
```




```
## *** : The Hubert index is a graphical method of determining the number of clusters.
##                 In the plot of Hubert index, we seek a significant knee that corresponds to a
##                 significant increase of the value of the measure i.e the significant peak in Hubert
##                 index second differences plot.
##
```

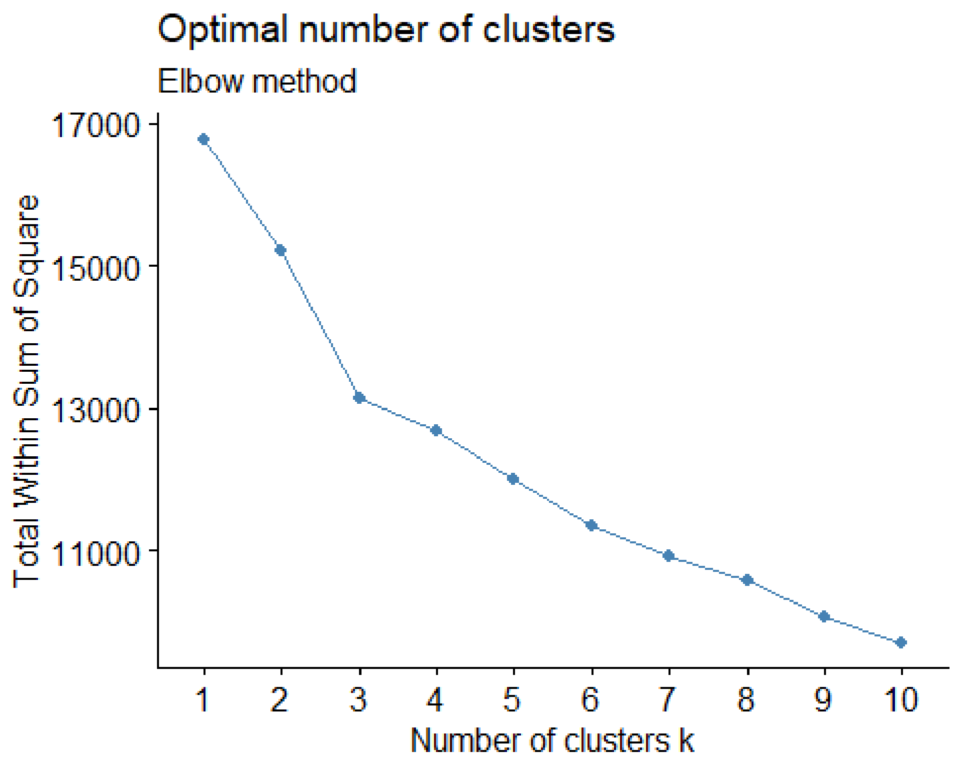



```
## *** : The D index is a graphical method of determining the number of
clusters.
##                 In the plot of D index, we seek a significant knee (the
significant peak in Dindex
##                 second differences plot) that corresponds to a significant
increase of the value of
##                 the measure.
##
## *******************************************************************
## * Among all indices:
## * 12 proposed 2 as the best number of clusters
## * 3 proposed 3 as the best number of clusters
## * 6 proposed 4 as the best number of clusters
## * 2 proposed 14 as the best number of clusters
## * 1 proposed 15 as the best number of clusters
##
##                    ***** Conclusion *****
##
## * According to the majority rule, the best number of clusters is  2
##
##
## *******************************************************************

## $All.index
##         KL      CH Hartigan      CCC     Scott      Marriot   TrCovW
TraceW
## 2  8.6033 94.5557  10.3474  -0.0797 1710.322 3.254901e+64 527192.1
```

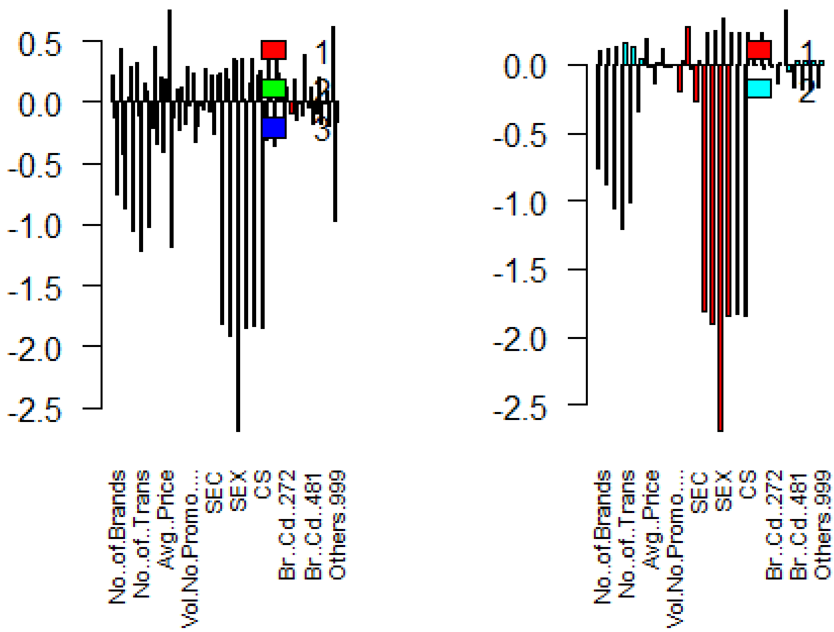
```
14482.092
## 3  0.1290 53.1805  73.3944 -11.8458 1193.457 1.733167e+65 463018.2
14235.766
## 4  2.8871 64.1695  31.3261  -1.0599 3227.442 1.038652e+64 374464.4
12677.243
## 5  0.9614 58.3901  31.1924   1.1679 3798.437 6.265998e+63 336587.4
12044.193
## 6  0.7827 55.3063  37.9054   4.5446 3996.488 6.486319e+63 296120.3
11444.238
## 7  1.4639 55.2540  27.8050  10.7128 4693.863 2.761311e+63 252402.7
10757.745
## 8  1.2745 53.4631  22.8271  15.2680 5345.306 1.217779e+63 229464.2
10275.920
## 9  1.4389 51.3505  17.0826  19.0180 5782.696 7.434995e+62 206667.0
9894.399
## 10 0.4201 48.7797  35.5892  21.5359 6087.338 5.524441e+62 194346.4
9616.440
## 11 1.2368 50.0239  30.2731  29.8710 6904.696 1.711788e+62 171513.8
9069.369
## 12 2.9906 50.4799  12.3819  37.1260 7803.696 4.553121e+61 160812.4
8626.015
## 13 0.6190 48.1973  18.0071  38.9926 7837.558 5.050374e+61 149296.6
8448.117
## 14 0.7664 47.1593  22.9456  42.8494 8304.662 2.689004e+61 142737.0
8196.672
## 15 1.2697 47.0640  18.8966  47.8963 8718.855 1.547801e+61 130234.9
7887.814
##     Friedman  Rubin Cindex     DB Silhouette   Duda  Pseudot2    Beale
## 2    75.0977 1.1581 0.2882 1.6414     0.2846 4.0650 -186.9919 -14.6335
## 3  2894.6922 1.1782 0.2883 2.5936     0.1318 0.3679  139.1619  33.6687
## 4  3023.1976 1.3230 0.2630 2.1369     0.1480 3.0018 -157.3817 -12.9425
## 5  4440.5617 1.3925 0.2506 2.1056     0.1520 1.1539  -27.8724  -2.6121
## 6  4233.4960 1.4655 0.2455 2.1617     0.1232 1.6558  -81.1907  -7.7320
## 7  6112.3002 1.5591 0.2322 2.0536     0.1329 1.9009  -77.7259  -9.1938
## 8  6274.1752 1.6322 0.2344 1.8976     0.1356 2.0148  -75.0460  -9.8077
## 9  6467.2204 1.6951 0.2344 1.9832     0.1226 1.2372  -23.1947  -3.7167
## 10 7013.7933 1.7441 0.2259 1.9518     0.1280 2.3700  -89.5983 -11.2498
## 11 7046.9270 1.8493 0.2232 1.8780     0.1346 1.0083   -1.1186  -0.1603
## 12 7783.1336 1.9444 0.2176 1.6880     0.1498 1.4562  -57.6428  -6.1150
## 13 7628.3366 1.9853 0.2135 1.8033     0.1367 2.8508  -98.0318 -12.6385
## 14 7694.9077 2.0462 0.2028 1.6621     0.1390 1.9074  -43.7659  -9.2367
## 15 8388.9503 2.1263 0.2394 1.7300     0.1436 1.6890  -62.0037  -7.9433
##    Ratkowsky      Ball Ptbiserial       Frey McClain   Dunn Hubert SDindex
## 2     0.1806 7241.0460     0.4314     3.9808  0.1806 0.1892  3e-04  1.1217
## 3     0.2007 4745.2553     0.3240    -0.1123  0.9800 0.0814  2e-04  1.2743
## 4     0.2166 3169.3106     0.3618     0.0678  1.1166 0.0811  3e-04  1.2389
## 5     0.2120 2408.8387     0.3990     2.4089  1.3686 0.1031  3e-04  1.2374
## 6     0.2081 1907.3730     0.3314     0.1185  2.3560 0.0768  3e-04  1.2355
## 7     0.2099 1536.8207     0.3490    -0.2087  2.9050 0.0701  3e-04  1.2496
## 8     0.2097 1284.4900     0.3890 -1686.9334  2.8242 0.0808  4e-04  1.6002
```

```
## 9     0.2057 1099.3777     0.3280    -0.3404  4.0435 0.0697  4e-04  1.6198
## 10    0.1990  961.6440     0.3709     1.5523  3.5981 0.0808  4e-04  1.6149
## 11    0.1983  824.4881     0.3319    -0.4483  4.7360 0.0808  4e-04  1.4814
## 12    0.1970  718.8346     0.3670     0.8697  4.1615 0.0809  4e-04  1.4074
## 13    0.1914  649.8551     0.3333    -0.5799  5.3809 0.0808  4e-04  1.4411
## 14    0.1863  585.4765     0.3283     0.0025  5.5097 0.0767  4e-04  1.2227
## 15    0.1841  525.8543     0.3384     0.0306  5.6903 0.0933  4e-04  1.2637
##    Dindex   SDbw
## 2  4.5726 1.2344
## 3  4.5383 1.2431
## 4  4.2560 1.0650
## 5  4.1494 1.1187
## 6  4.0438 1.0312
## 7  3.8913 1.0484
## 8  3.8450 1.6232
## 9  3.7595 1.4997
## 10 3.7201 1.5548
## 11 3.6121 1.3164
## 12 3.5505 1.2323
## 13 3.5009 1.1890
## 14 3.4554 0.8662
## 15 3.3985 0.9567
##
## $All.CriticalValues
##    CritValue_Duda CritValue_PseudoT2 Fvalue_Beale
## 2          0.8751            35.4081            1
## 3          0.9160             7.4317            0
## 4          0.8751            33.6948            1
## 5          0.9130            19.9148            1
## 6          0.8973            23.4704            1
## 7          0.8735            23.7448            1
## 8          0.8869            18.9991            1
## 9          0.8719            17.7745            1
## 10         0.8848            20.1891            1
## 11         0.8899            16.8308            1
## 12         0.8965            21.2332            1
## 13         0.8859            19.4572            1
## 14         0.8765            12.9589            1
## 15         0.8869            19.3816            1
##
## $Best.nc
##                      KL      CH Hartigan     CCC    Scott      Marriot
TrCovW
## Number_clusters 2.0000  2.0000    3.000 15.0000    4.000 4.000000e+00
4.00
## Value_Index     8.6033 94.5557   63.047 47.8963 2033.986 1.588096e+65
88553.76
##                   TraceW Friedman   Rubin  Cindex     DB Silhouette  Duda
## Number_clusters   4.0000    3.000  4.0000 14.0000 2.0000     2.0000 2.000
## Value_Index     925.4743 2819.595 -0.0753  0.2028 1.6414     0.2846 4.065
```

```
##                    PseudoT2     Beale Ratkowsky      Ball PtBiserial    Frey
McClain
## Number_clusters     2.0000    2.0000    4.0000    3.000      2.0000 2.0000
2.0000
## Value_Index     -186.9919 -14.6335    0.2166 2495.791      0.4314 3.9808
0.1806
##                     Dunn Hubert SDindex Dindex    SDbw
## Number_clusters 2.0000      0  2.0000      0 14.0000
## Value_Index     0.1892      0  1.1217      0  0.8662
##
## $Best.partition
##   [1] 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 2 2 1 1 2 2 2 2 1 2 2 1 2
2 2 2
##  [38] 2 2 1 2 2 2 1 2 2 1 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
2 2 2
##  [75] 2 2 2 2 2 2 2 2 2 2 1 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
2 2 2
## [112] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2
2 2 2
## [149] 2 2 1 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
2 2 2
## [186] 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 1 2 2 2
2 2 2
## [223] 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 1 2 1 2 2 2 2 2 2 2 2 2 2 1 2 1 2 2 2
2 2 2
## [260] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 1 2 2 2 2 2 2 2
2 2 2
## [297] 2 2 2 2 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2
2 2 2
## [334] 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
2 2 1
## [371] 2 2 2 2 2 2 2 2 2 2 1 1 2 1 2 2 2 2 2 2 2 2 1 2 2 2 1 1 2 1 2 2 2
1 1 1
## [408] 2 2 2 2 1 2 2 2 2 2 1 2 2 2 1 1 2 2 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 1
2 2 2
## [445] 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 1 2 2 2 2 2 2
2 2 1
## [482] 2 2 2 2 2 1 2 2 2 2 2 1 1 2 2 2 2 2 2 2 1 2 2 2 2 2 1 1 2 2 1 2 2 2 2
2 2 2
## [519] 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
1 2 1
## [556] 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2
2 2 2
## [593] 2 2 2 2 2 2 2 2

# According to the majority rule, the best number of clusters is  3

# Traditional Approches
fviz_nbclust(BS2Scale, kmeans, method = "wss")+labs(subtitle = "Elbow
method")
```




```
#Using 3 cluster size
BS2K <- kmeans(BS2Scale,centers =3,nstart = 25)

# Allow us to represent the cluster solution into 2 dimensions
BS2K$size

## [1] 331 201  68

#dev.off()
barplot(BS2K$centers,beside = TRUE,col = rainbow(3),cex.names = .7,las=2,
        args.legend = list(x ='topright', bty='n'),legend.text =
(1:nrow(BS2K$centers)))

#USing 2 cluster size
BS2K <- kmeans(BS2Scale,centers =2,nstart = 25)

# Allow us to represent the cluster solution into 2 dimensions
BS2K$size

## [1]  68 532

#dev.off()
barplot(BS2K$centers,beside = TRUE,col = rainbow(2),cex.names = .7,las=2,
        args.legend = list(x ='topright', bty='n'),legend.text =
(1:nrow(BS2K$centers)))
```



Question 3 - Model that classifies the data into these segments.

```
BS2$Cluster <- BS2K$cluster
BS2K$size

## [1]  68 532

Mail <- BS2[BS2$Cluster==1,]

head(Mail)

##    No..of.Brands Brand.Runs Total.Volume No..of..Trans Value
Trans...Brand.Runs
## 4              2          4         1500             4 114.0
1.00
## 20             2          4          675             4 123.5
1.00
## 21             2          4         6400             8 368.0
2.00
## 24             3          3         1600             4 111.0
1.33
## 25             3          5         2825             5 300.0
1.00
## 30             2          2         6050             8 381.5
4.00
##    Avg..Price Brand_1_Other_2_Loyal LoyalBrand_1_8 Pur.Vol.No.Promo....
```

```
## 4        7.60                     1             2                1.00
## 20      18.30                     2             5                0.33
## 21       5.75                     1             2                0.58
## 24       6.94                     1             2                1.00
## 25      10.62                     1             2                1.00
## 30       6.31                     1             2                1.00
##    Pur.Vol.Promo.6.. Pur.Vol.Other.Promo.. SEC FEH MT SEX EDU HS CS
## 4               0.00                  0.00   4   0  0   0   0  0  0
## 20              0.67                  0.00   3   0  0   0   0  0  0
## 21              0.00                  0.42   4   0  0   0   0  0  0
## 24              0.00                  0.00   3   0  0   0   0  0  0
## 25              0.00                  0.00   4   0  0   0   0  0  0
## 30              0.00                  0.00   4   0  0   0   0  0  0
##    Br..Cd..57..144 Br..Cd..55 Br..Cd..272 Br..Cd..286 Br..Cd..24
Br..Cd..481
## 4             0.40       0.60        0.00           0       0.00
0
## 20            0.00       0.00        0.00           0       0.11
0
## 21            0.00       0.94        0.00           0       0.00
0
## 24            0.19       0.75        0.00           0       0.00
0
## 25            0.00       0.53        0.32           0       0.00
0
## 30            0.13       0.87        0.00           0       0.00
0
##    Br..Cd..352 Br..Cd..5 Others.999 Cluster
## 4            0         0      0.000       1
## 20           0         0      0.889       1
## 21           0         0      0.063       1
## 24           0         0      0.063       1
## 25           0         0      0.150       1
## 30           0         0      0.000       1

# Excluding these people as they are loyal so we need to exclude the 68 out
of the total 600 entries.
# So therefore on targeting direct-mail promotions for this market segment
would succeed
```