Marketing and advertisement modelling in R

Vivian Njau

3/5/2020

Problem Statement

Kira Plastinina is a Russian brand that is sold through a defunct chain of retail stores in Russia, Ukraine, Kazakhstan, Belarus, China, Philippines, and Armenia.

The brand's Sales and Marketing team would like to understand their customer's behavior from data that they have collected over the past year.

More specifically, they would like to learn the characteristics of customer groups.

Perform clustering stating insights drawn from your analysis and visualizations.

Upon implementation, provide comparisons between K-Means clustering vs Hierarchical clustering highlighting the strengths and limitations of each approach in the context of your analysis.

Your findings should help inform the team in formulating the marketing and sales strategies of the brand.

Markdown Sections.

1.Problem Definition

2.Data Sourcing

3.Check the Data

4.Perform Data Cleaning

5.Perform Exploratory Data Analysis (Univariate, Bivariate & Multivariate)

6.Implement the Solution

7.Challenge the Solution

8.Follow up Questions

Data

The dataset consists of 10 numerical and 8 categorical attributes.

The 'Revenue' attribute can be used as the class label.

Types of Pages: Administrative, Informational

Time spent on pages: Admin Duration and Info Duration

“Administrative”, “Administrative Duration”, “Informational”, “Informational Duration”, “Product Related” and “Product Related Duration” represents the number of different types of pages visited by the visitor in that session and total time spent in each of these page categories.

The values of these features are derived from the URL information of the pages visited by the user and updated in real-time when a user takes an action, e.g. moving from one page to another.

Metrics: Bounce rate, Exit rate and Page Value

The “Bounce Rate”, “Exit Rate” and “Page Value” features represent the metrics measured by “Google Analytics” for each page in the e-commerce site.

The value of the “Bounce Rate” feature for a web page refers to the percentage of visitors who enter the site from that page and then leave (“bounce”) without triggering any other requests to the analytics server during that session.

The value of the “Exit Rate” feature for a specific web page is calculated as for all pageviews to the page, the percentage that was the last in the session.

The “Page Value” feature represents the average value for a web page that a user visited before completing an e-commerce transaction.

Type of days: Speical or Ordinary

The “Special Day” feature indicates the closeness of the site visiting time to a specific special day (e.g. Mother’s Day, Valentine’s Day) in which the sessions are more likely to be finalized with the transaction.

The value of this attribute is determined by considering the dynamics of e-commerce such as the duration between the order date and delivery date. For example, for Valentina’s day, this value takes a nonzero value between February 2 and February 12, zero before and after this date unless it is close to another special day, and its maximum value of 1 on February 8.

Type of visit, Operating system, Browser and region(location)

The dataset also includes the operating system, browser, region, traffic type, visitor type as returning or new visitor, a Boolean value indicating whether the date of the visit is weekend, and month of the year.

Installing packages.

```r
install.packages("devtools")
library(devtools)
```

```
install_github("vqv/ggbiplot")
install.packages("rtools")
install.packages("DataExplorer")
install.packages("Hmisc")
install.packages("pastecs")
install.packages("psych")
install.packages("corrplot")
install.packages("factoextra")
install.packages("caret")
```

Loading the libraries

```
#specify the path where the file is located
library("data.table")
library(tidyverse)
library(magrittr)
library(warn = -1)

library("ggbiplot")
library(ggplot2)
library(lattice)
library(corrplot)

library(DataExplorer)
library(Hmisc)
library(pastecs)
library(psych)
library(factoextra)
library(caret)
```

Loading the data

```
#specify the path where the file is located
library("data.table")
```

obtaining the path to the working directroy

```
getwd()

## [1] "C:/Users/hp/Documents"
```

Loading the datasets

```
library("readr")
df <- read.csv("online_shoppers_intention.csv")
head(df)

##   Administrative Administrative_Duration Informational
Informational_Duration
## 1              0                       0             0
0
## 2              0                       0             0
0
```

```
## 3              0                      -1             0
-1
## 4              0                       0             0
0
## 5              0                       0             0
0
## 6              0                       0             0
0
##   ProductRelated ProductRelated_Duration BounceRates ExitRates PageValues
## 1              1                0.000000  0.20000000 0.2000000          0
## 2              2               64.000000  0.00000000 0.1000000          0
## 3              1               -1.000000  0.20000000 0.2000000          0
## 4              2                2.666667  0.05000000 0.1400000          0
## 5             10              627.500000  0.02000000 0.0500000          0
## 6             19              154.216667  0.01578947 0.0245614          0
##   SpecialDay Month OperatingSystems Browser Region TrafficType
## 1          0   Feb                1       1      1           1
## 2          0   Feb                2       2      1           2
## 3          0   Feb                4       1      9           3
## 4          0   Feb                3       2      2           4
## 5          0   Feb                3       3      1           4
## 6          0   Feb                2       2      1           3
##         VisitorType Weekend Revenue
## 1 Returning_Visitor   FALSE   FALSE
## 2 Returning_Visitor   FALSE   FALSE
## 3 Returning_Visitor   FALSE   FALSE
## 4 Returning_Visitor   FALSE   FALSE
## 5 Returning_Visitor    TRUE   FALSE
## 6 Returning_Visitor   FALSE   FALSE
```

Previewing the top of the dataset

```
market_df <- data.frame(df)
head(market_df)
```

```
##   Administrative Administrative_Duration Informational
Informational_Duration
## 1              0                       0             0
0
## 2              0                       0             0
0
## 3              0                      -1             0
-1
## 4              0                       0             0
0
## 5              0                       0             0
0
## 6              0                       0             0
0
##   ProductRelated ProductRelated_Duration BounceRates ExitRates PageValues
## 1              1                0.000000  0.20000000 0.2000000          0
## 2              2               64.000000  0.00000000 0.1000000          0
## 3              1               -1.000000  0.20000000 0.2000000          0
## 4              2                2.666667  0.05000000 0.1400000          0
## 5             10              627.500000  0.02000000 0.0500000          0
## 6             19              154.216667  0.01578947 0.0245614          0
##   SpecialDay Month OperatingSystems Browser Region TrafficType
## 1          0   Feb                1       1      1           1
## 2          0   Feb                2       2      1           2
## 3          0   Feb                4       1      9           3
## 4          0   Feb                3       2      2           4
## 5          0   Feb                3       3      1           4
## 6          0   Feb                2       2      1           3
##         VisitorType Weekend Revenue
## 1 Returning_Visitor   FALSE   FALSE
## 2 Returning_Visitor   FALSE   FALSE
## 3 Returning_Visitor   FALSE   FALSE
## 4 Returning_Visitor   FALSE   FALSE
## 5 Returning_Visitor    TRUE   FALSE
## 6 Returning_Visitor   FALSE   FALSE
```

Previewing the summary of the dataset

```
summary(market_df)
```

```
##  Administrative   Administrative_Duration Informational
##  Min.   : 0.000   Min.   :  -1.00         Min.   : 0.000
##  1st Qu.: 0.000   1st Qu.:   0.00         1st Qu.: 0.000
##  Median : 1.000   Median :   8.00         Median : 0.000
##  Mean   : 2.318   Mean   :  80.91         Mean   : 0.504
##  3rd Qu.: 4.000   3rd Qu.:  93.50         3rd Qu.: 0.000
##  Max.   :27.000   Max.   :3398.75         Max.   :24.000
##  NA's   :14       NA's   :14              NA's   :14
##  Informational_Duration ProductRelated   ProductRelated_Duration
##  Min.   :  -1.00        Min.   :  0.00   Min.   :   -1.0
##  1st Qu.:   0.00        1st Qu.:  7.00   1st Qu.:  185.0
##  Median :   0.00        Median : 18.00   Median :  599.8
##  Mean   :  34.51        Mean   : 31.76   Mean   : 1196.0
##  3rd Qu.:   0.00        3rd Qu.: 38.00   3rd Qu.: 1466.5
##  Max.   :2549.38        Max.   :705.00   Max.   :63973.5
##  NA's   :14             NA's   :14       NA's   :14
##   BounceRates        ExitRates          PageValues        SpecialDay
##  Min.   :0.000000   Min.   :0.00000   Min.   :  0.000   Min.   :0.00000
##  1st Qu.:0.000000   1st Qu.:0.01429   1st Qu.:  0.000   1st Qu.:0.00000
##  Median :0.003119   Median :0.02512   Median :  0.000   Median :0.00000
##  Mean   :0.022152   Mean   :0.04300   Mean   :  5.889   Mean   :0.06143
##  3rd Qu.:0.016684   3rd Qu.:0.05000   3rd Qu.:  0.000   3rd Qu.:0.00000
##  Max.   :0.200000   Max.   :0.20000   Max.   :361.764   Max.   :1.00000
##  NA's   :14         NA's   :14
##      Month      OperatingSystems    Browser           Region
##  May    :3364   Min.   :1.000    Min.   : 1.000   Min.   :1.000
##  Nov    :2998   1st Qu.:2.000    1st Qu.: 2.000   1st Qu.:1.000
##  Mar    :1907   Median :2.000    Median : 2.000   Median :3.000
##  Dec    :1727   Mean   :2.124    Mean   : 2.357   Mean   :3.147
##  Oct    : 549   3rd Qu.:3.000    3rd Qu.: 2.000   3rd Qu.:4.000
##  Sep    : 448   Max.   :8.000    Max.   :13.000   Max.   :9.000
##  (Other):1337
##   TrafficType               VisitorType     Weekend         Revenue
##  Min.   : 1.00   New_Visitor      : 1694   Mode :logical   Mode :logical
##  1st Qu.: 2.00   Other            :   85   FALSE:9462      FALSE:10422
##  Median : 2.00   Returning_Visitor:10551   TRUE :2868      TRUE :1908
##  Mean   : 4.07
##  3rd Qu.: 4.00
##  Max.   :20.00
##
```

Properties of the dataset

Length

```
length(market_df)

## [1] 18

#The dataframe has 18 columns.
```

Dimensions

```
dim(market_df)

## [1] 12330    18

#The dataframe has 12330 row entries and 18 columns
```

Column Names

```
colnames(market_df)

##  [1] "Administrative"          "Administrative_Duration"
##  [3] "Informational"           "Informational_Duration"
##  [5] "ProductRelated"          "ProductRelated_Duration"
##  [7] "BounceRates"             "ExitRates"
##  [9] "PageValues"              "SpecialDay"
## [11] "Month"                   "OperatingSystems"
## [13] "Browser"                 "Region"
## [15] "TrafficType"             "VisitorType"
## [17] "Weekend"                 "Revenue"

#The Eighteen column names are:
```

Column data types

```
sapply(market_df, class)

##          Administrative Administrative_Duration           Informational
##               "integer"               "numeric"               "integer"
##  Informational_Duration          ProductRelated ProductRelated_Duration
##               "numeric"               "integer"               "numeric"
##             BounceRates               ExitRates              PageValues
##               "numeric"               "numeric"               "numeric"
##              SpecialDay                   Month        OperatingSystems
##               "numeric"                "factor"               "integer"
##                 Browser                  Region             TrafficType
##               "integer"               "integer"               "integer"
##             VisitorType                 Weekend                 Revenue
##                "factor"               "logical"               "logical"
```

Data Cleaning

Missing Values

```
sum(is.na(market_df))

## [1] 112

#There are 112 missing values in the data.
```

Missing values per column

```
#Checking the sum of missing values per column
colSums(is.na(market_df))

##          Administrative Administrative_Duration           Informational
##                      14                      14                      14
##  Informational_Duration          ProductRelated ProductRelated_Duration
##                      14                      14                      14
##             BounceRates               ExitRates              PageValues
##                      14                      14                       0
##              SpecialDay                   Month        OperatingSystems
##                       0                       0                       0
##                 Browser                  Region             TrafficType
##                       0                       0                       0
##             VisitorType                 Weekend                 Revenue
##                       0                       0                       0

#there are no misssing values in the data
```

The list of columns with null values

```
# Return the column names containing missing observations
list_na <- colnames(market_df)[ apply(market_df, 2, anyNA) ]
list_na

## [1] "Administrative"          "Administrative_Duration"
## [3] "Informational"           "Informational_Duration" 
## [5] "ProductRelated"          "ProductRelated_Duration"
## [7] "BounceRates"             "ExitRates"
```

Duplicates

```
duplicated_rows <- market_df[duplicated(market_df),]
dim(duplicated_rows)
```

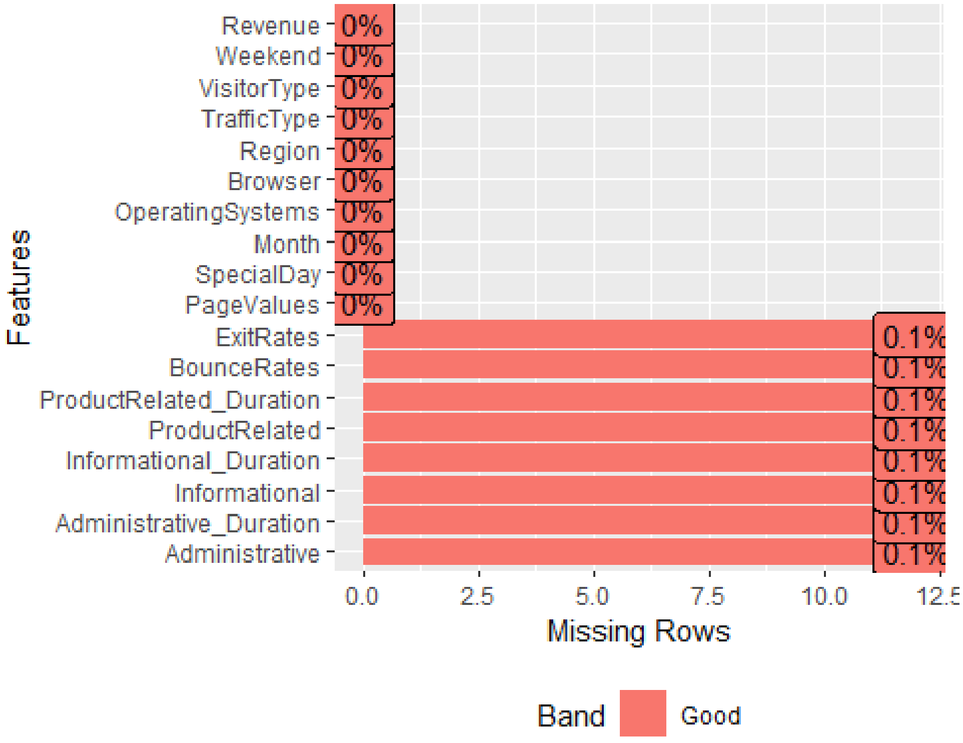
```
## [1] 119  18
```

Removing duplicates

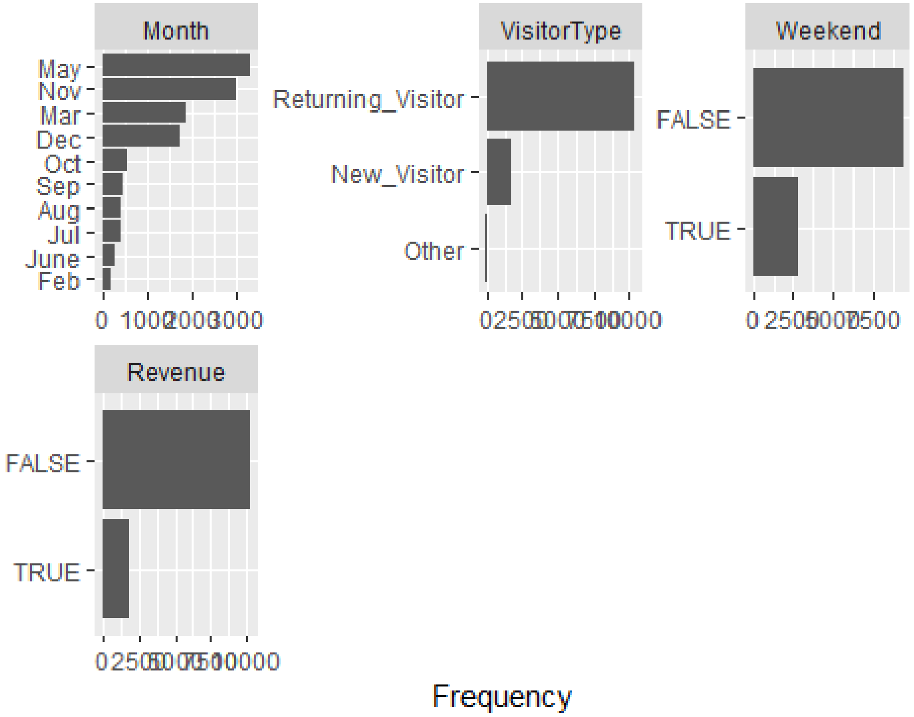
```
new_market_df <- market_df[-which(duplicated(market_df)), ]
dim(new_market_df)

## [1] 12211    18

#119 rows deleted
```

Exploring the data with Data Explorer

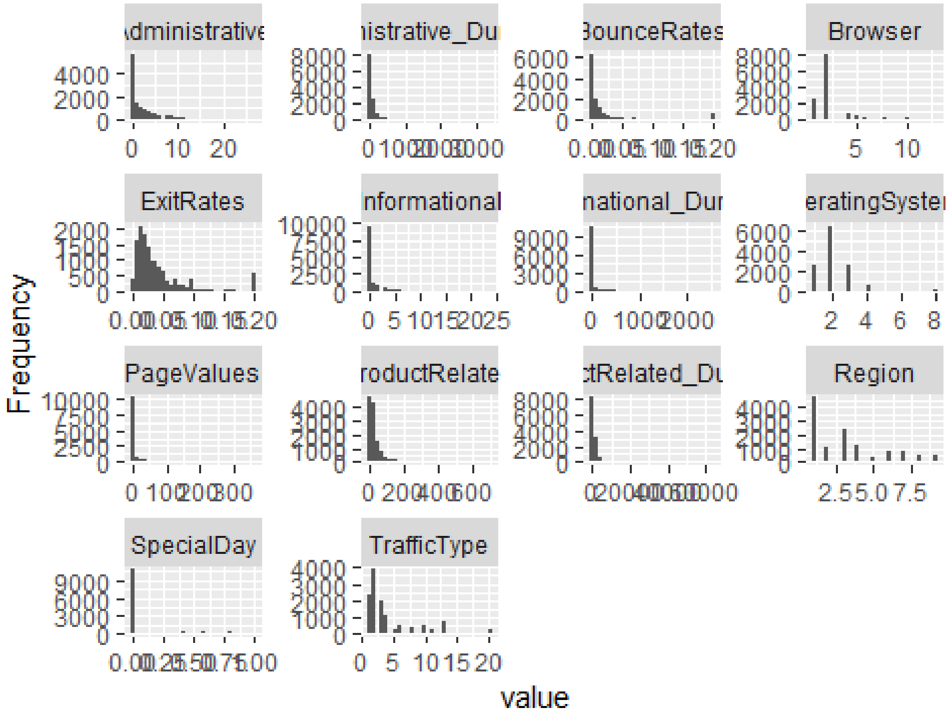
```
library(DataExplorer)

plot_missing(new_market_df) ## Are there missing values, and what is the
missing data profile?
```



```
plot_bar(new_market_df) ## How does the categorical frequency for each
discrete variable look like?
```



```
plot_histogram(new_market_df) ## What is the distribution of each continuous variable?
```



```
plot_str(new_market_df)
```

Data Types

```
sapply(new_market_df, class)

##          Administrative Administrative_Duration           Informational
##               "integer"               "numeric"               "integer"
##  Informational_Duration          ProductRelated ProductRelated_Duration
##               "numeric"               "integer"               "numeric"
##             BounceRates               ExitRates              PageValues
##               "numeric"               "numeric"               "numeric"
##              SpecialDay                   Month        OperatingSystems
##               "numeric"                "factor"               "integer"
##                 Browser                  Region             TrafficType
##               "integer"               "integer"               "integer"
##             VisitorType                 Weekend                 Revenue
##                "factor"               "logical"               "logical"
```

Perform Exploratory Data Analysis (Univariate, Bivariate & Multivariate)

Univariate Analysis

Administrative

```
unique(new_market_df$Administrative)

##  [1]  0  1  2  4 12  3 10  6  5  9  8 16 13 11  7 18 14 17 19 15 NA 24 22
21 20
## [26] 23 27 26

factor(unique(new_market_df$Administrative))

##  [1] 0    1    2    4    12   3    10   6    5    9    8    16   13   11
7
## [16] 18   14   17   19   15   <NA> 24   22   21   20   23   27   26
## 27 Levels: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23
... 27

#There are 27 levels [27 unique elements in the Administrative column]
```

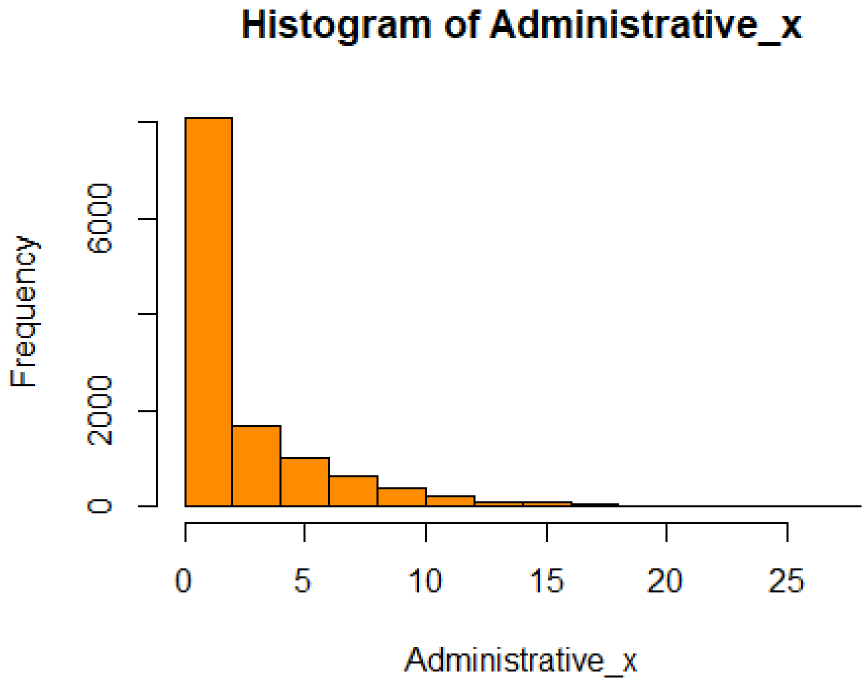
There are 14 missing values in this column thus we shall use the mean/mode to impute.

Before performing any analysis on the column we have to drop the missing values.

```
length(new_market_df$Administrative)

## [1] 12211

12211

## [1] 12211
```

```
dim(new_market_df)

## [1] 12211    18

sum(is.na(new_market_df))

## [1] 96

#there are 96 missing values in the new_market_df dataframe
markert_df2 <- new_market_df[-which(is.na(new_market_df)), ]
sum(is.na(markert_df2))

## [1] 0

dim(markert_df2)

## [1] 12199    18

colSums(is.na(markert_df2))

##          Administrative Administrative_Duration           Informational 
##                       0                       0                       0 
##  Informational_Duration          ProductRelated ProductRelated_Duration 
##                       0                       0                       0 
##             BounceRates               ExitRates              PageValues 
##                       0                       0                       0 
##              SpecialDay                   Month        OperatingSystems 
##                       0                       0                       0 
##                 Browser                  Region             TrafficType 
##                       0                       0                       0 
##             VisitorType                 Weekend                 Revenue 
##                       0                       0                       0

summary(markert_df2$Administrative)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##    0.00    0.00    1.00    2.34    4.00   27.00

adm <- markert_df2$Administrative
# median
median(markert_df2$Administrative)

## [1] 1

# mode
Administrative_x <- markert_df2$Administrative
#sort(Daily.Internet.Usage_x)
names(table(Administrative_x))[table(Administrative_x)==max(table(Administrat
ive_x))]

## [1] "0"
```

```
#each of the values printed below appear thrice in the dataset

#distribution
hist(Administrative_x, col=c("darkorange"))
```



The adm distribution is right skewed.

The highest value in the administrative column is 27

The lowest value in the column is zero and it has the highest frequency.

The mean is 2.34

Administrative_Duration

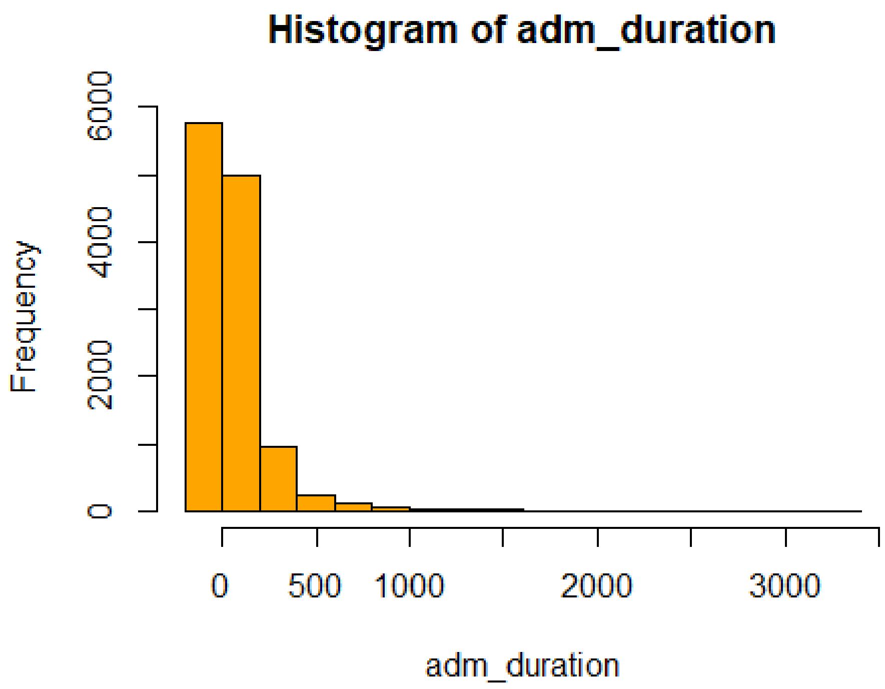
```
length(unique(markert_df2$Administrative_Duration))

## [1] 3336

#there are 3336 unique elements in admin duration column

summary(markert_df2$Administrative_Duration)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   -1.00    0.00    9.00   81.68   94.75 3398.75

adm_duration <- markert_df2$Administrative_Duration
# median
median(adm_duration)
```

```
## [1] 9
```

```r
# mode

#sort(adm_duration)
names(table(adm_duration))[table(adm_duration)==max(table(adm_duration ))]
```

```
## [1] "0"
```

```r
#distribution
hist(adm_duration, col=c("orange"))
```



The adm_duration distribution is right skewed.

The highest value in the administrative column is 3398.75

The lowest value in the column is 0 and it has the highest frequency.

The mean is 81.68

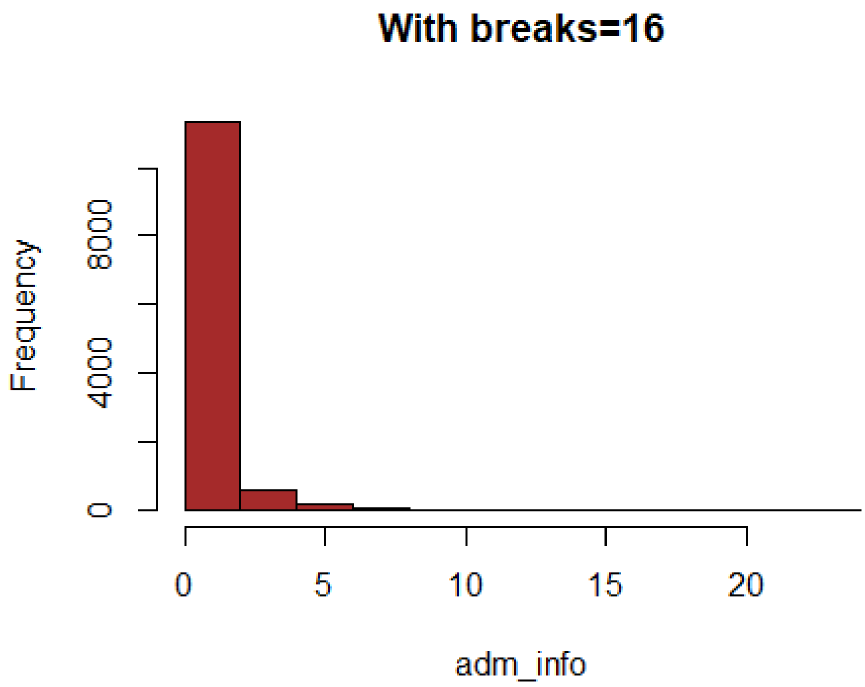
The median is 9

Informational

```r
length(unique(markert_df2$Informational))
```

```
## [1] 17
```

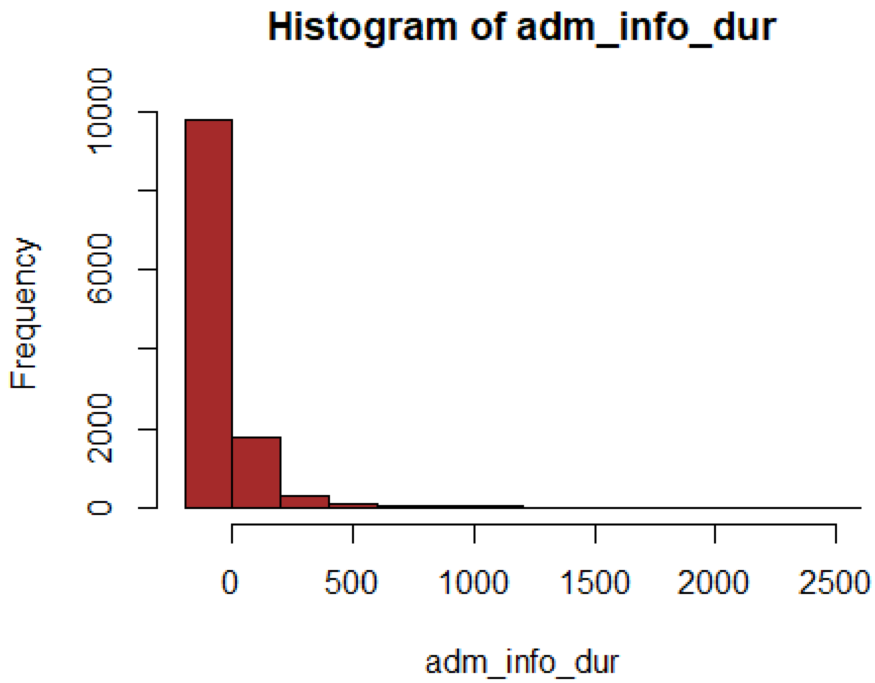
```
#there are 17 unique elements in Informational column

summary(markert_df2$Informational)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##  0.0000  0.0000  0.0000  0.5088  0.0000 24.0000

adm_info <- markert_df2$Informational
# median
median(adm_info)

## [1] 0

# mode

#sort(adm_duration)
names(table(adm_info))[table(adm_info)==max(table(adm_info ))]

## [1] "0"

#The modal value in the information dataset is 0

#distribution
hist(adm_info,breaks = 16 , main="With breaks=16", col=c("brown"))
```



Informational_Duration

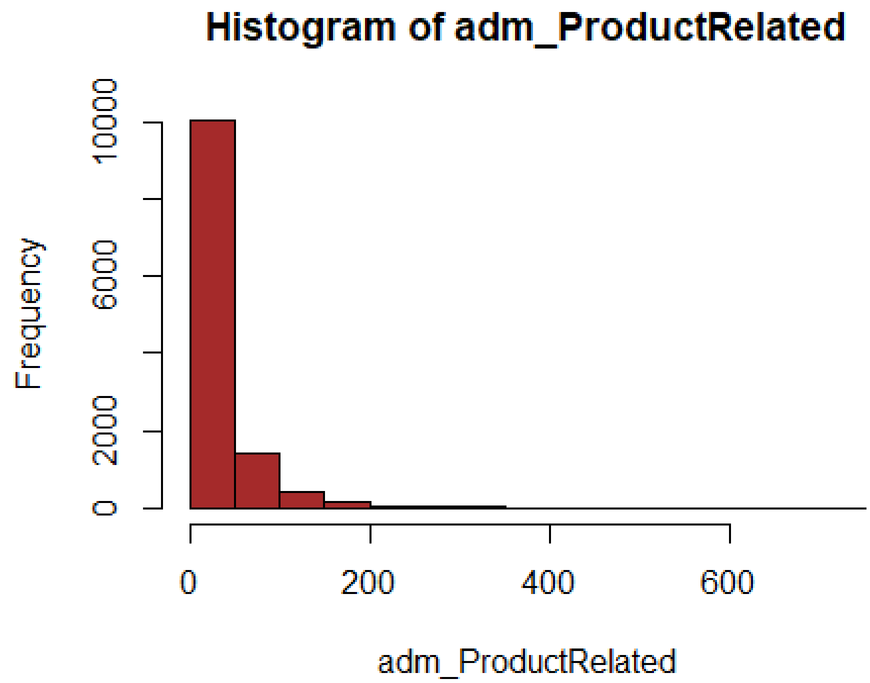
```
length(unique(markert_df2$Informational_Duration))
```

```
## [1] 1259

#there are 1259 unique elements in Informational duration column

summary(markert_df2$Informational_Duration)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   -1.00    0.00    0.00   34.84    0.00 2549.38

adm_info_dur <- markert_df2$Informational_Duration
# median
median(adm_info)

## [1] 0

# mode

#sort(adm_info_dur)
names(table(adm_info_dur))[table(adm_info_dur)==max(table(adm_info_dur ))]

## [1] "0"

#The modal value in the information dataset is 0

#distribution
hist(adm_info_dur,col=c("brown"))
```



ProductRelated

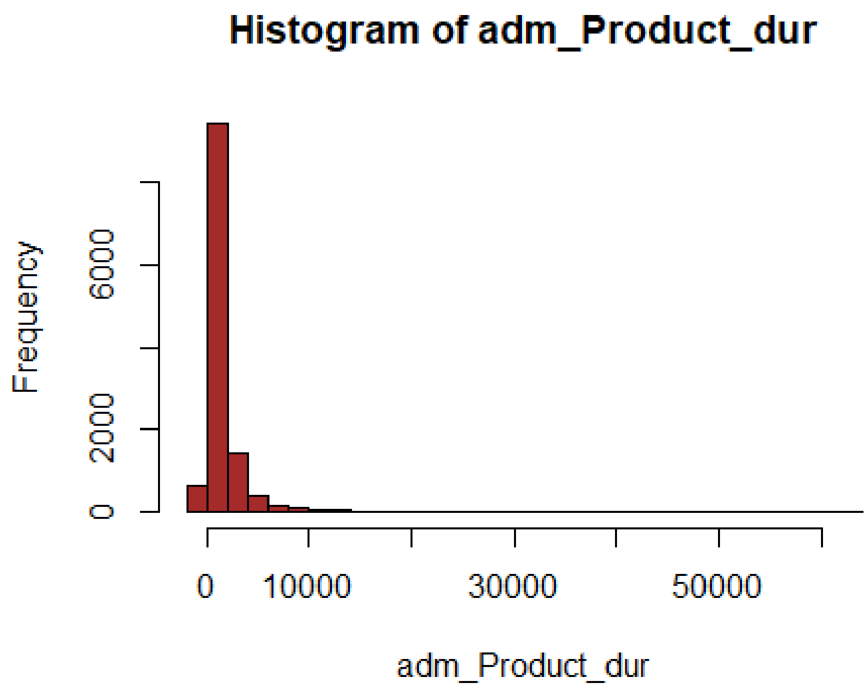
```
length(unique(markert_df2$ProductRelated))

## [1] 311

#there are 311 unique elements in ProductRelated column

summary(markert_df2$ProductRelated)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##    0.00    8.00   18.00   32.06   38.00  705.00

adm_ProductRelated <- markert_df2$ProductRelated
# median
median(adm_ProductRelated)

## [1] 18

# mode

#sort(adm_info_dur)
names(table(adm_ProductRelated))[table(adm_ProductRelated)==max(table(adm_Pro
ductRelated ))]

## [1] "1"

#The modal value in the information dataset is 0

#distribution
hist(adm_ProductRelated,col=c("brown"))
```



ProductRelated_Duration

```
length(unique(markert_df2$ProductRelated_Duration))

## [1] 9552

#there are 9552 unique elements in ProductRelated durationcolumn

summary(markert_df2$ProductRelated_Duration)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    -1.0   193.6   609.5  1207.5  1477.6 63973.5

adm_Product_dur <- markert_df2$ProductRelated_Duration
# median
median(adm_Product_dur)

## [1] 609.5417

# mode

#sort(adm_info_dur)
names(table(adm_Product_dur))[table(adm_Product_dur)==max(table(adm_Product_d
ur ))]

## [1] "0"

#The modal value in the information dataset is 0
```

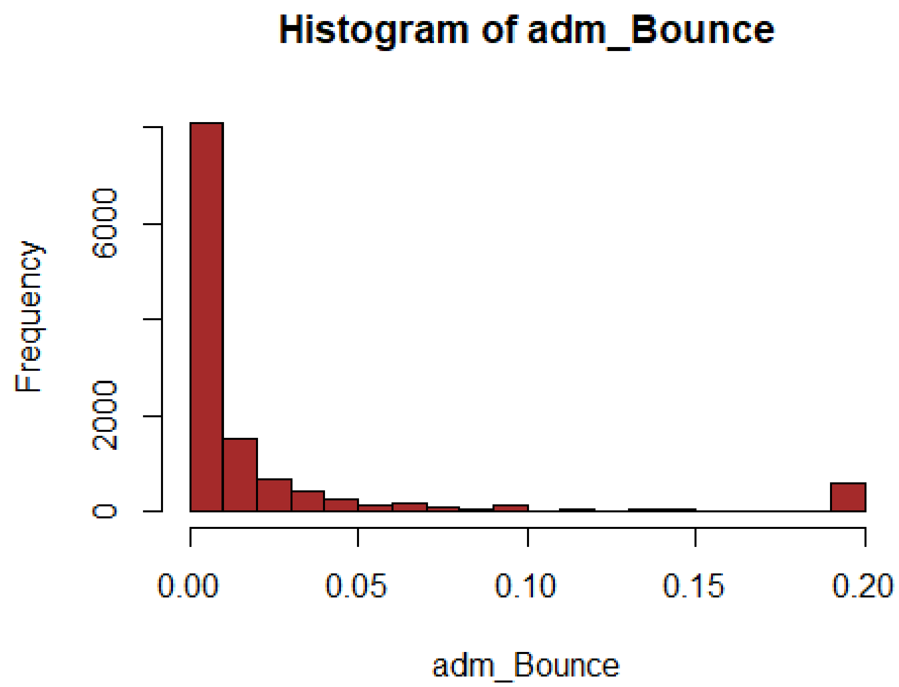
```
#distribution
hist(adm_Product_dur,breaks=30,col=c("brown"))
```



BounceRates

```
length(unique(markert_df2$BounceRates))

## [1] 1872

#there are 1872 unique elements in Bounce rate column

summary(markert_df2$BounceRates)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.00000 0.00000 0.00293 0.02045 0.01667 0.20000

adm_Bounce <- markert_df2$BounceRates
# median
median(adm_Bounce)

## [1] 0.002930403

# mode

#sort(adm_info_dur)
names(table(adm_Bounce))[table(adm_Bounce)==max(table(adm_Bounce ))]

## [1] "0"
```

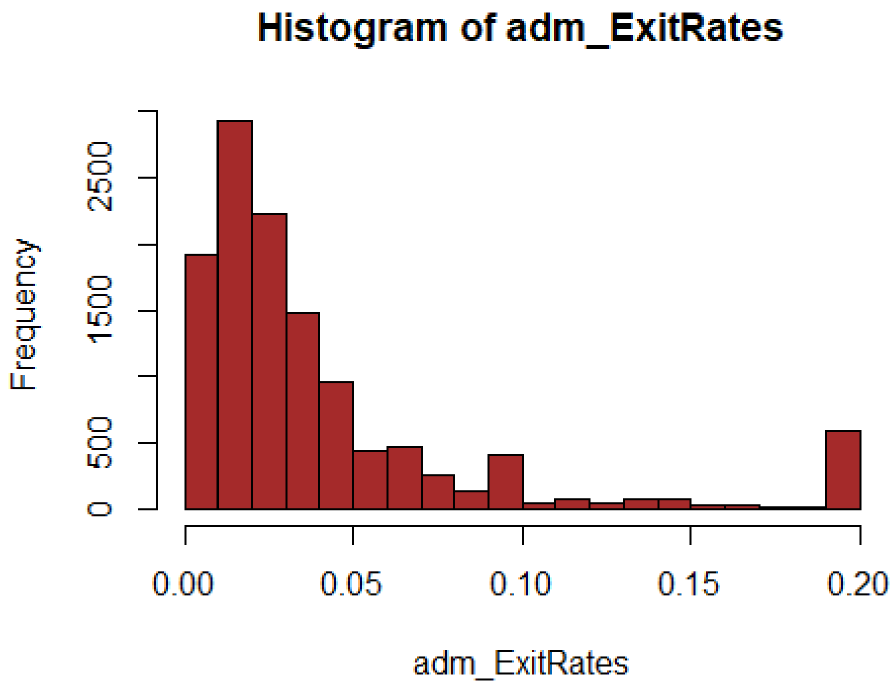
```
#The modal value in the information dataset is 0

#distribution
hist(adm_Bounce,col=c("brown"))
```



ExitRates

```
length(unique(markert_df2$ExitRates))

## [1] 4777

#there are 4777 unique elements in Exit rates column

summary(markert_df2$ExitRates)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
## 0.00000 0.01422 0.02500 0.04150 0.04848 0.20000

adm_ExitRates <- markert_df2$ExitRates
# median
median(adm_ExitRates)

## [1] 0.025

# mode

#sort(adm_info_dur)
names(table(adm_ExitRates))[table(adm_ExitRates)==max(table(adm_ExitRates ))]
```

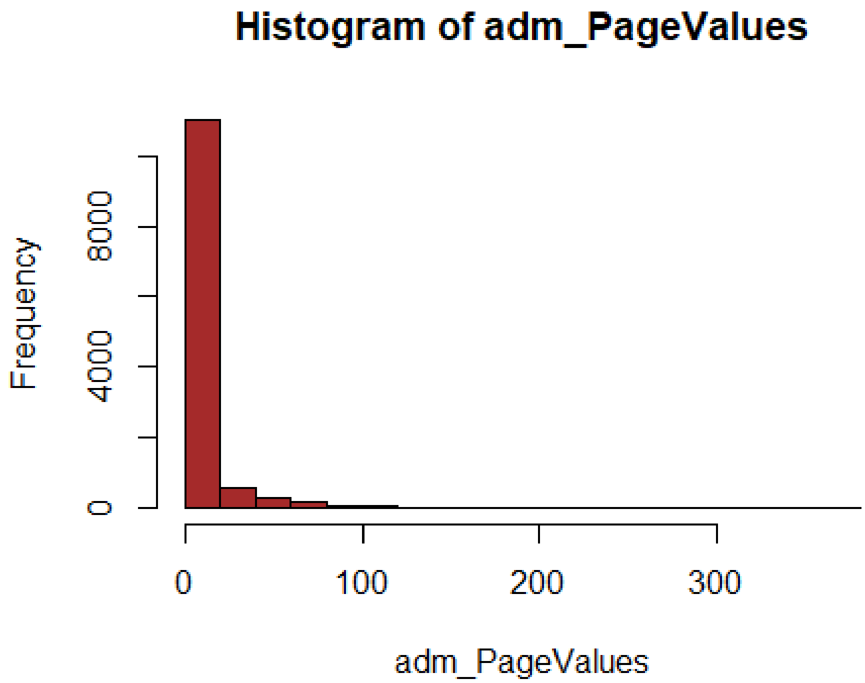
```
## [1] "0.2"

#The modal value in the information dataset is 0

#distribution
hist(adm_ExitRates,col=c("brown"))
```



Page Values

```
length(unique(markert_df2$PageValues))

## [1] 2704

#there are 2704 unique elements in Page Values column

summary(markert_df2$PageValues)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   0.000   0.000   0.000   5.952   0.000 361.764

adm_PageValues <- markert_df2$PageValues
# median
median(adm_PageValues)

## [1] 0

# mode

#sort(adm_info_dur)
```

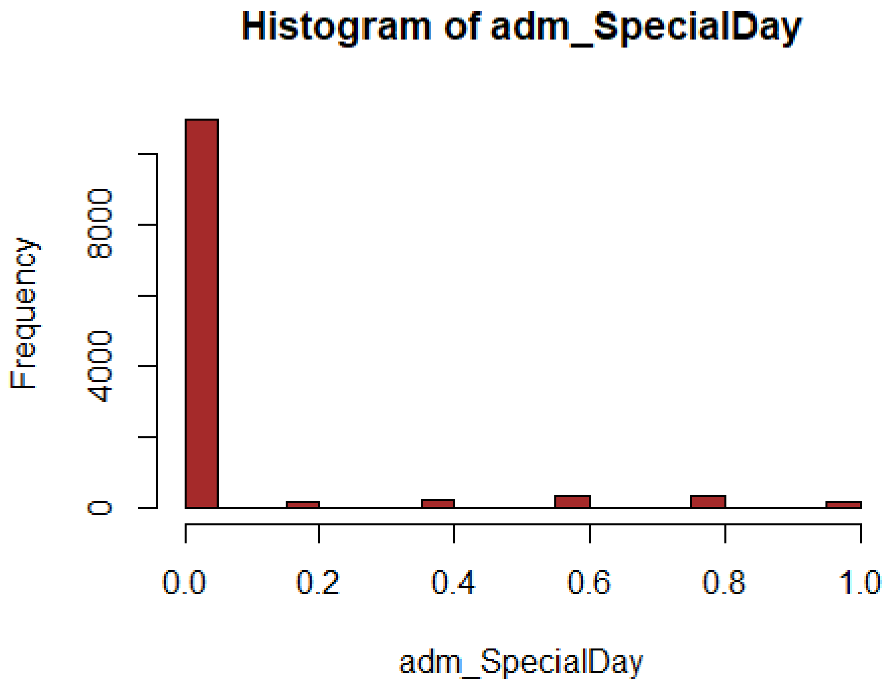
```
names(table(adm_PageValues))[table(adm_PageValues)==max(table(adm_PageValues
))]

## [1] "0"

#The modal value in the information dataset is 0

#distribution
hist(adm_PageValues,col=c("brown"))
```



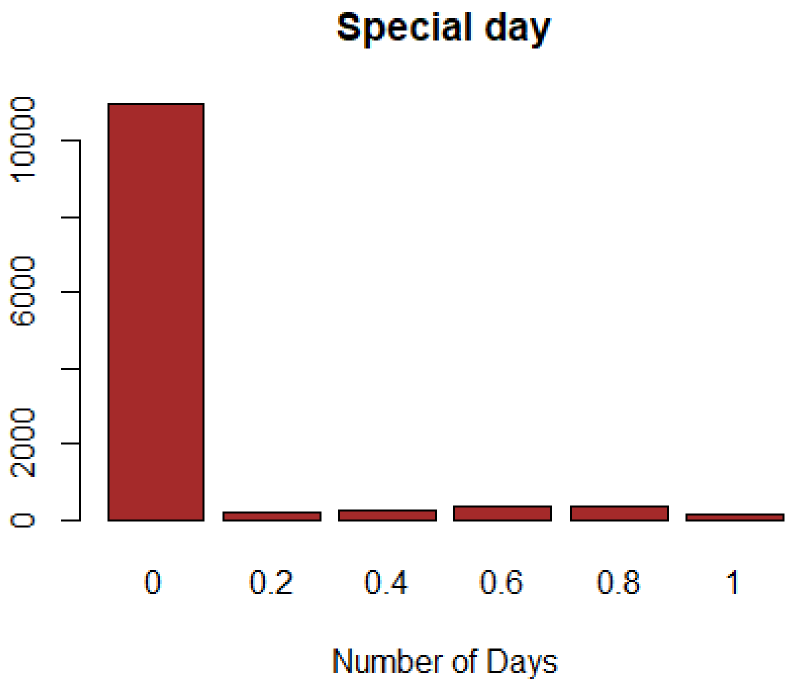
SpecialDay

```
length(unique(markert_df2$SpecialDay))

## [1] 6

#there are 6 unique elements in ProductRelated column

summary(markert_df2$SpecialDay)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.00000 0.00000 0.00000 0.06197 0.00000 1.00000

adm_SpecialDay <- markert_df2$SpecialDay
# median
median(adm_SpecialDay)

## [1] 0
```

```
# mode

#sort(adm_info_dur)
names(table(adm_SpecialDay))[table(adm_SpecialDay)==max(table(adm_SpecialDay
))]

## [1] "0"

#The modal value in the information dataset is 0

#distribution
hist(adm_SpecialDay,col=c("brown"))
```



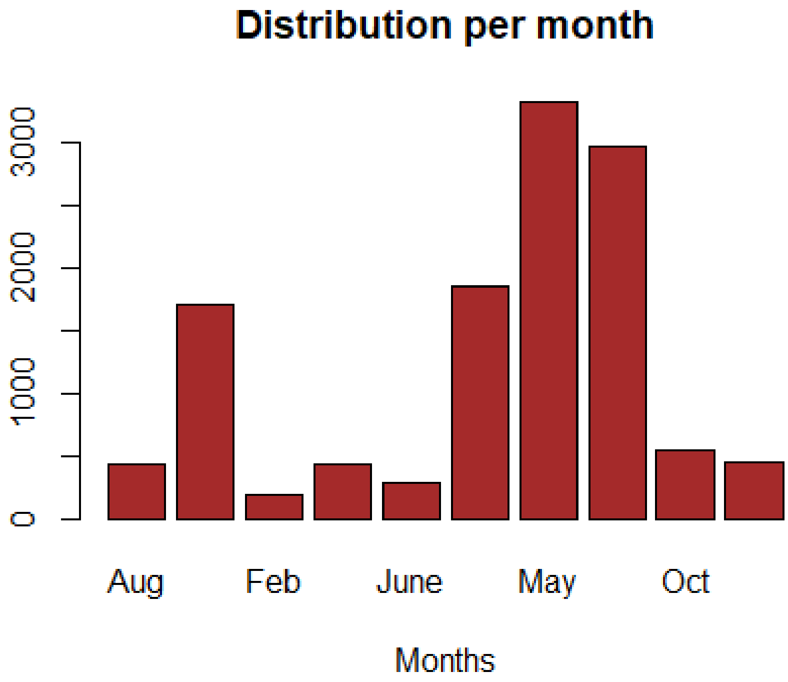
```
# Simple Bar Plot
counts <- table(adm_SpecialDay)
barplot(counts, main="Special day",col=c("brown"),
   xlab="Number of Days")
```



Month

```
length(unique(markert_df2$Month))

## [1] 10

#there are 10 unique elements in Month column

summary(markert_df2$Month)

##   Aug  Dec  Feb  Jul June  Mar  May  Nov  Oct  Sep 
##   433 1706  182  432  285 1853 3328 2983  549  448

adm_Month <- markert_df2$Month

# mode

#sort(adm_info_dur)
names(table(adm_Month))[table(adm_Month)==max(table(adm_Month ))]

## [1] "May"

#The modal value in the information dataset is 0

#distribution

# Simple Bar Plot
counts <- table(adm_Month)
```

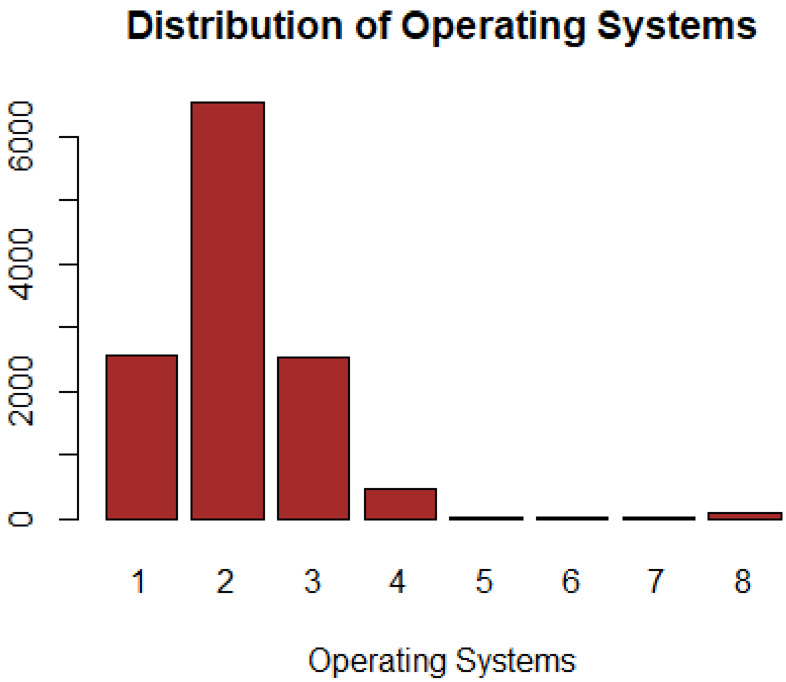
```
barplot(counts, main="Distribution per month",col=c("brown"),
   xlab="Months")
```



OperatingSystems

```
length(unique(markert_df2$OperatingSystems))

## [1] 8

#there are 8 unique elements in Operating Systems column

summary(markert_df2$OperatingSystems)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   1.000   2.000   2.000   2.124   3.000   8.000

adm_OperatingSystems <- markert_df2$OperatingSystems
# median
median(adm_OperatingSystems)

## [1] 2

# mode

#sort(adm_info_dur)
names(table(adm_OperatingSystems))[table(adm_OperatingSystems)==max(table(adm
_OperatingSystems ))]

## [1] "2"
```

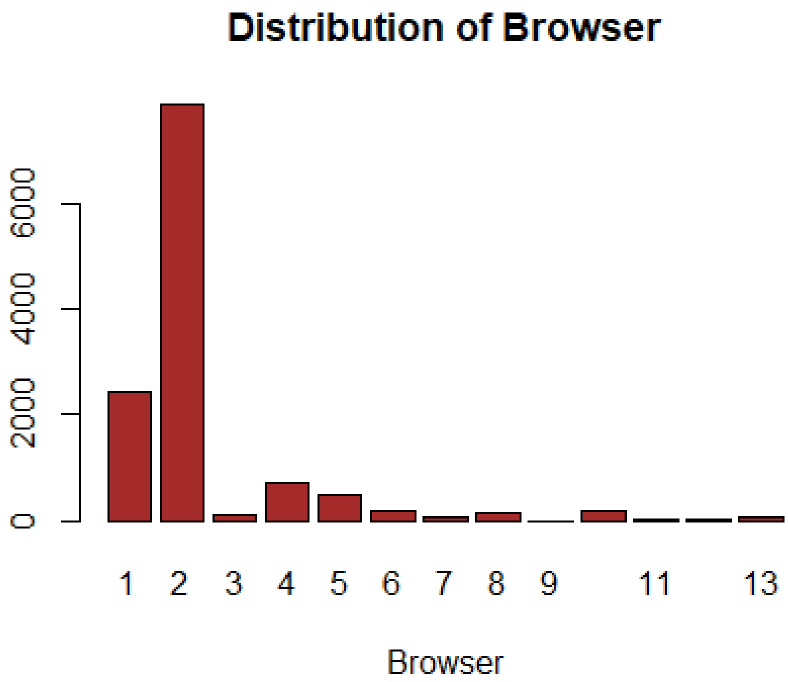
```
#The modal value in the information dataset is 0

#distribution
counts <- table(adm_OperatingSystems)
barplot(counts, main="Distribution of Operating Systems",col=c("brown"),
   xlab="Operating Systems")
```



Browser

```
length(unique(markert_df2$Browser))

## [1] 13

#there are 13 unique elements in Browser column

summary(markert_df2$Browser)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   1.000   2.000   2.000   2.358   2.000  13.000

adm_Browser <- markert_df2$Browser
# median
median(adm_Browser)

## [1] 2

# mode
```

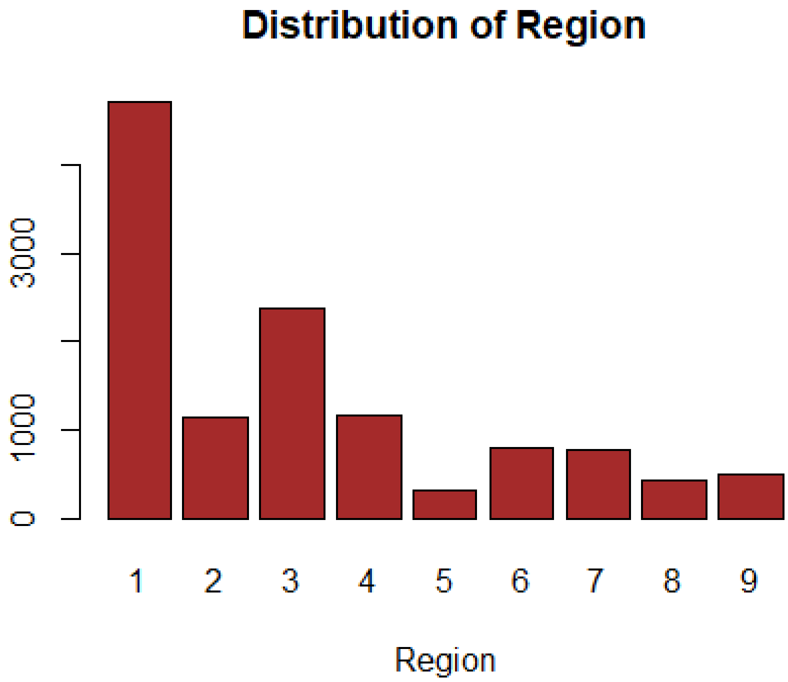
```
#sort(adm_info_dur)
names(table(adm_Browser))[table(adm_Browser)==max(table(adm_Browser ))]

## [1] "2"

#The modal value in the information dataset is 0

#distribution
counts <- table(adm_Browser)
barplot(counts, main="Distribution of Browser",col=c("brown"),
   xlab="Browser")
```



Region

```
length(unique(markert_df2$Region))

## [1] 9

#there are 9 unique elements in Region column

summary(markert_df2$Region)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   1.000   1.000   3.000   3.153   4.000   9.000

adm_Region <- markert_df2$Region
# median
median(adm_Region)
```

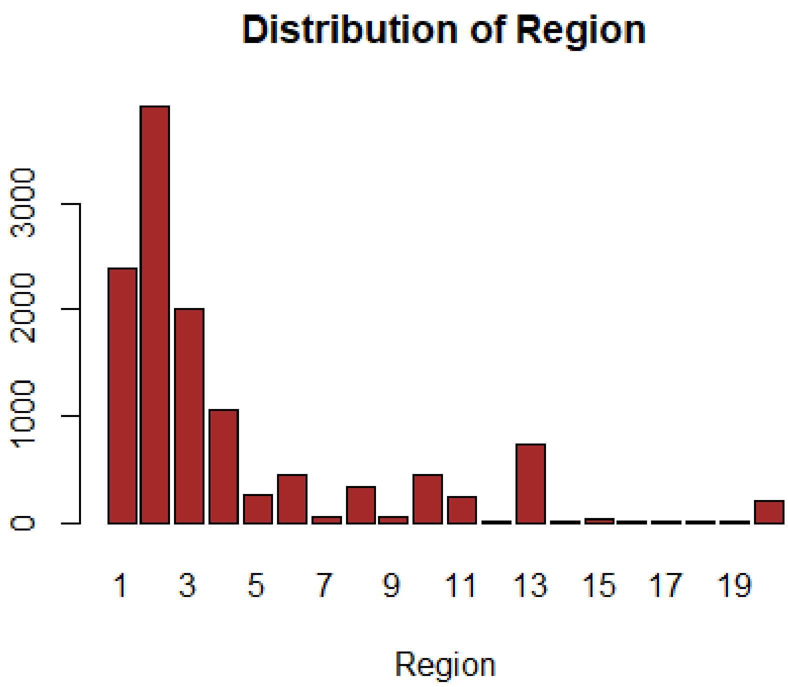
```
## [1] 3

# mode

#sort(adm_Region)
names(table(adm_Region))[table(adm_Region)==max(table(adm_Region ))]

## [1] "1"

#The modal value in the information dataset is 0

#distribution
counts <- table(adm_Region)
barplot(counts, main="Distribution of Region",col=c("brown"),
   xlab="Region")
```



TrafficType

```
length(unique(markert_df2$TrafficType))

## [1] 20

#there are 311 unique elements in ProductRelated column

summary(markert_df2$TrafficType)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   1.000   2.000   2.000   4.075   4.000  20.000
```

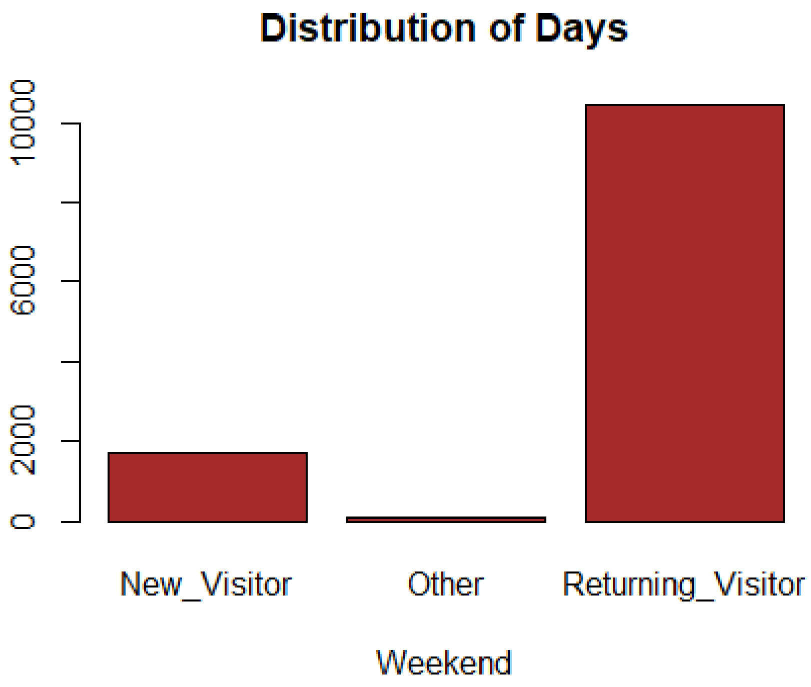
```
adm_TrafficType <- markert_df2$TrafficType
# median
median(adm_TrafficType)

## [1] 2

# mode

#sort(adm_info_dur)
names(table(adm_TrafficType))[table(adm_TrafficType)==max(table(adm_TrafficTy
pe ))]

## [1] "2"

#The modal value in the information dataset is 0

#distribution
counts <- table(adm_TrafficType)
barplot(counts, main="Distribution of Region",col=c("brown"),
   xlab="Region")
```



VisitorType

```
length(unique(markert_df2$VisitorType))

## [1] 3
```

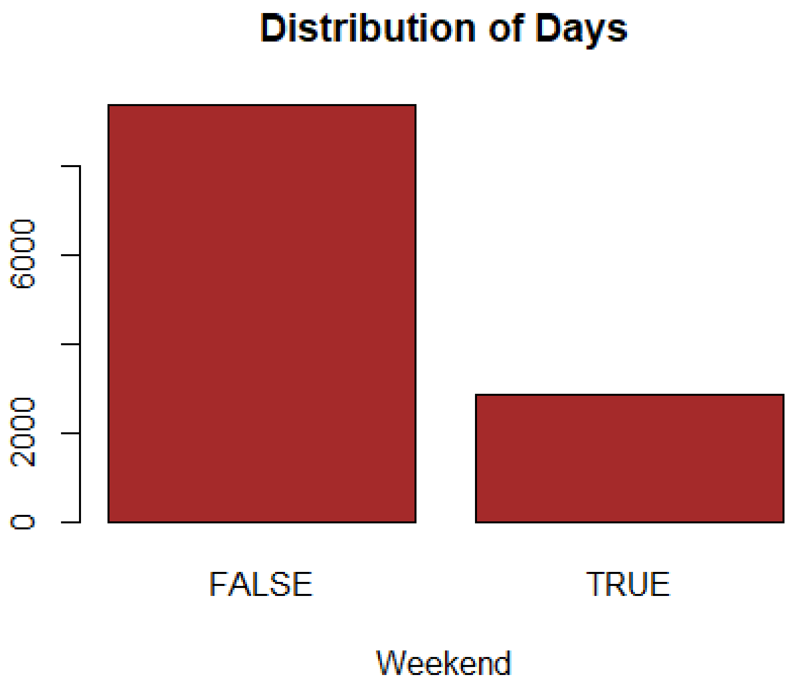
```
#there are 3 unique elements in VisitorType column

summary(markert_df2$VisitorType)

##       New_Visitor             Other Returning_Visitor
##              1693                81             10425

adm_VisitorType <- markert_df2$VisitorType
# median

# mode

#sort(adm_info_dur)
names(table(adm_VisitorType))[table(adm_VisitorType)==max(table(adm_VisitorTy
pe ))]

## [1] "Returning_Visitor"

#The modal value in the information dataset is 0

#distribution
counts <- table(adm_VisitorType)
barplot(counts, main="Distribution of Days",col=c("brown"),
   xlab="Weekend")
```



Weekend

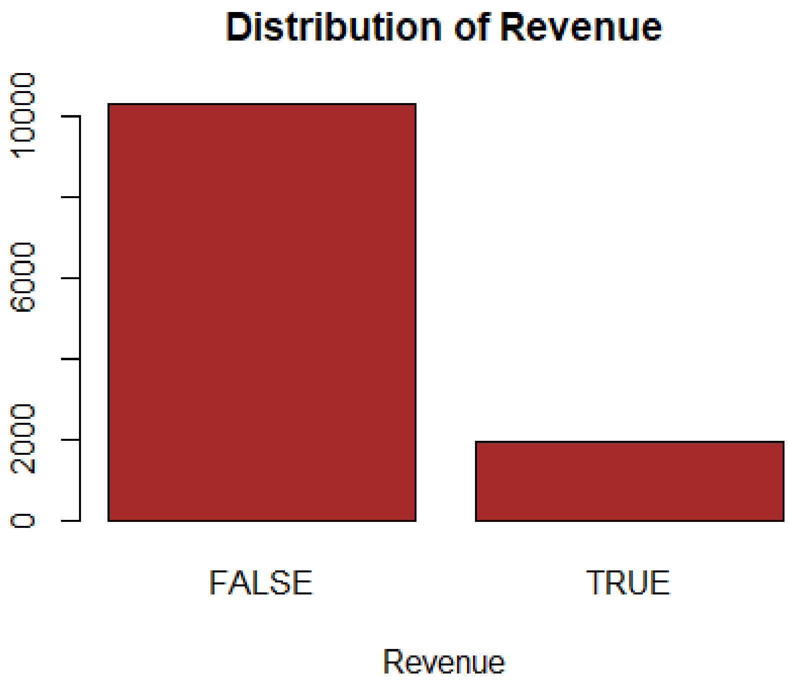
```
length(unique(markert_df2$Weekend))
```

```
## [1] 2

#there are 2 unique elements in Weekend column

summary(markert_df2$Weekend)

##    Mode   FALSE    TRUE 
## logical    9343    2856

adm_Weekend <- markert_df2$Weekend
# median
median(adm_Weekend)

## [1] FALSE

# mode

#sort(adm_Weekend)
names(table(adm_Weekend))[table(adm_Weekend)==max(table(adm_Weekend ))]

## [1] "FALSE"

#The modal value in the information dataset is 0

#distribution
counts <- table(adm_Weekend)
barplot(counts, main="Distribution of Days",col=c("brown"),
   xlab="Weekend")
```



Revenue

```
length(unique(markert_df2$Revenue))

## [1] 2

#there are 2 unique elements in Revenue column

summary(markert_df2$Revenue)

##    Mode   FALSE    TRUE
## logical   10291    1908

adm_Revenue <- markert_df2$Revenue
# median
median(adm_Revenue)

## [1] FALSE

# mode

#sort(adm_info_dur)
names(table(adm_Revenue))[table(adm_Revenue)==max(table(adm_Revenue ))]

## [1] "FALSE"

#The modal value in the information dataset is 0

#distribution
counts <- table(adm_Revenue)
barplot(counts, main="Distribution of Revenue",col=c("brown"),
   xlab="Revenue")
```



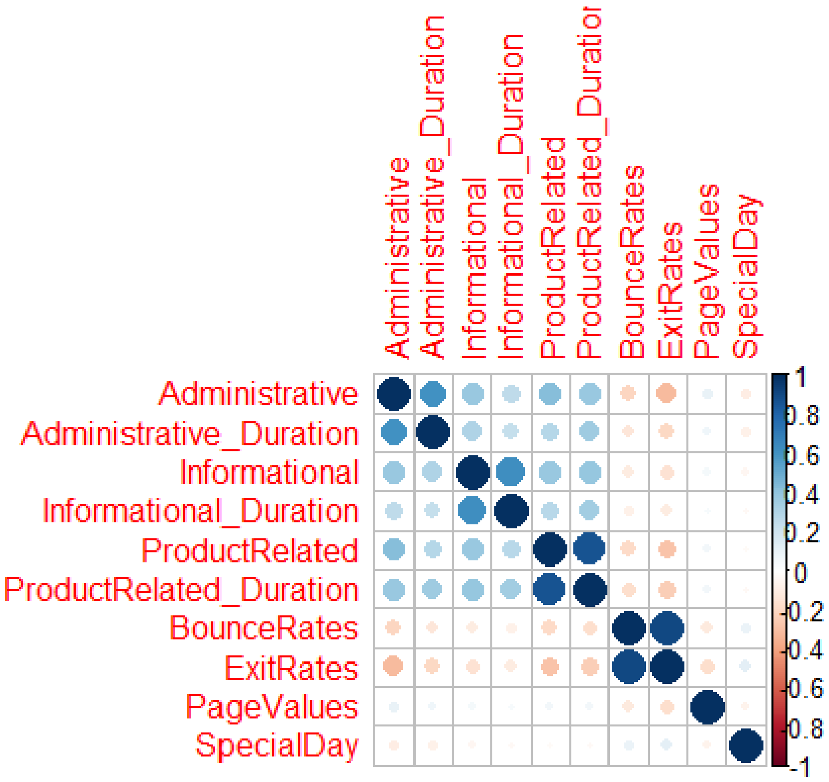
Bivariate Analysis

```
# calculate correlations
correlations <- cor(markert_df2[,1:10])
correlations

##                         Administrative Administrative_Duration
Informational
## Administrative              1.00000000              0.60040965
0.37528761
## Administrative_Duration     0.60040965              1.00000000
0.30143630
## Informational               0.37528761              0.30143630
1.00000000
## Informational_Duration      0.25478602              0.23718986
0.61867795
## ProductRelated              0.42819151              0.28678391
0.37260472
## ProductRelated_Duration     0.37102722              0.35351379
0.38608372
## BounceRates                -0.21366664             -0.13733340   -
0.10950530
## ExitRates                  -0.31127413             -0.20202445   -
0.15956681
## PageValues                  0.09692097              0.06616837
0.04739015
## SpecialDay                 -0.09707210             -0.07473689   -
```

```
0.04937677
##                           Informational_Duration ProductRelated
## Administrative                        0.25478602     0.42819151
## Administrative_Duration               0.23718986     0.28678391
## Informational                         0.61867795     0.37260472
## Informational_Duration                1.00000000     0.27906195
## ProductRelated                        0.27906195     1.00000000
## ProductRelated_Duration               0.34658069     0.86030819
## BounceRates                          -0.07015947    -0.19351577
## ExitRates                            -0.10293268    -0.28616321
## PageValues                            0.03006416     0.05411549
## SpecialDay                           -0.03129304    -0.02593062
##                           ProductRelated_Duration BounceRates  ExitRates
## Administrative                         0.37102722 -0.21366664 -0.3112741
## Administrative_Duration                0.35351379 -0.13733340 -0.2020245
## Informational                          0.38608372 -0.10950530 -0.1595668
## Informational_Duration                 0.34658069 -0.07015947 -0.1029327
## ProductRelated                         0.86030819 -0.19351577 -0.2861632
## ProductRelated_Duration                1.00000000 -0.17437550 -0.2453340
## BounceRates                           -0.17437550  1.00000000  0.9033582
## ExitRates                             -0.24533401  0.90335819  1.0000000
## PageValues                             0.05084062 -0.11599198 -0.1735715
## SpecialDay                            -0.03821065  0.08783999  0.1167838
##                            PageValues  SpecialDay
## Administrative             0.09692097 -0.09707210
## Administrative_Duration    0.06616837 -0.07473689
## Informational              0.04739015 -0.04937677
## Informational_Duration     0.03006416 -0.03129304
## ProductRelated             0.05411549 -0.02593062
## ProductRelated_Duration    0.05084062 -0.03821065
## BounceRates               -0.11599198  0.08783999
## ExitRates                 -0.17357154  0.11678376
## PageValues                 1.00000000 -0.06453271
## SpecialDay                -0.06453271  1.00000000
```

Correlation Plot

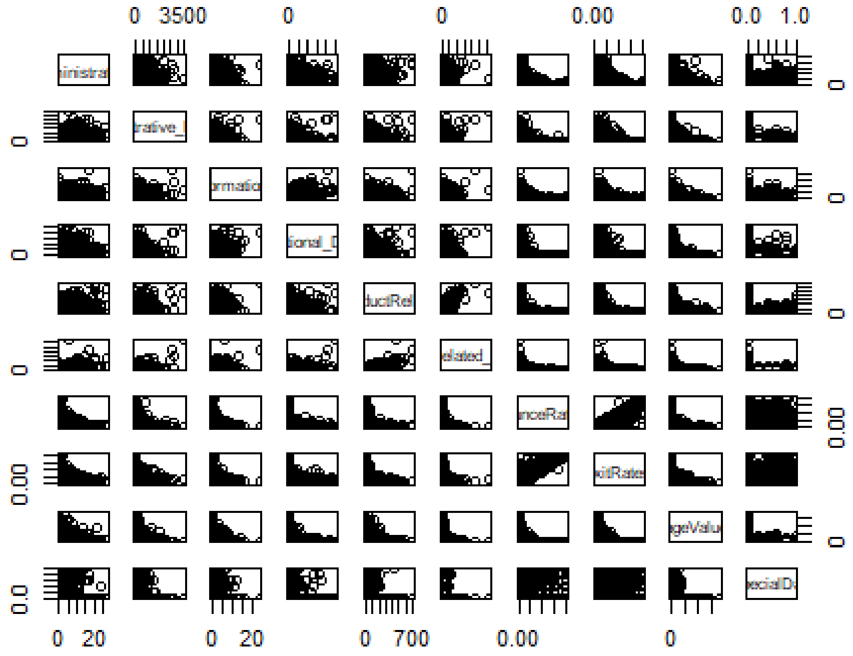
```
# create correlation plot
library(corrplot)

## corrplot 0.84 loaded

corrplot(correlations, method="circle")
```



From the plot above, we can see that most of the variables have low Positive and Negative correlation

Pair Plots

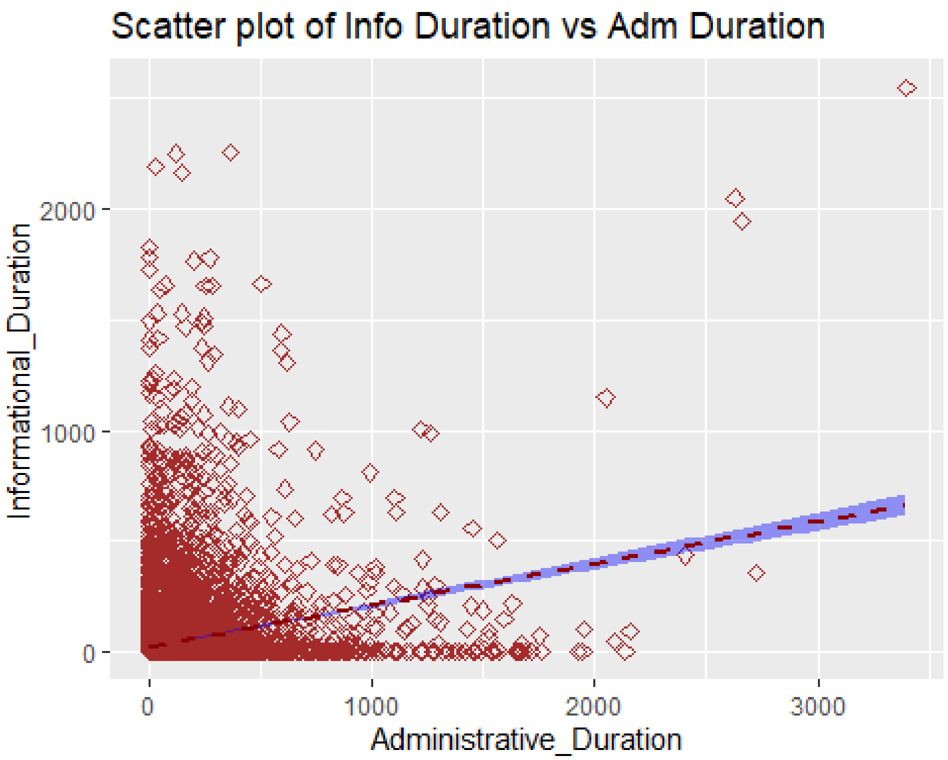
```r
pairs(markert_df2[,1:10])
```



Sites Visited Duration

Scatter plot of Administrative_Duration vs Informational_Duration

```
library(ggplot2)
ggplot(markert_df2, aes(x = Administrative_Duration, y =
Informational_Duration)) +
        geom_point(size = 2, color= "brown", shape = 23)+
        geom_smooth(method=lm,  linetype="dashed",color="darkred",
fill="blue")+
        labs(title = "Scatter plot of Info Duration vs Adm Duration")

## `geom_smooth()` using formula 'y ~ x'
```

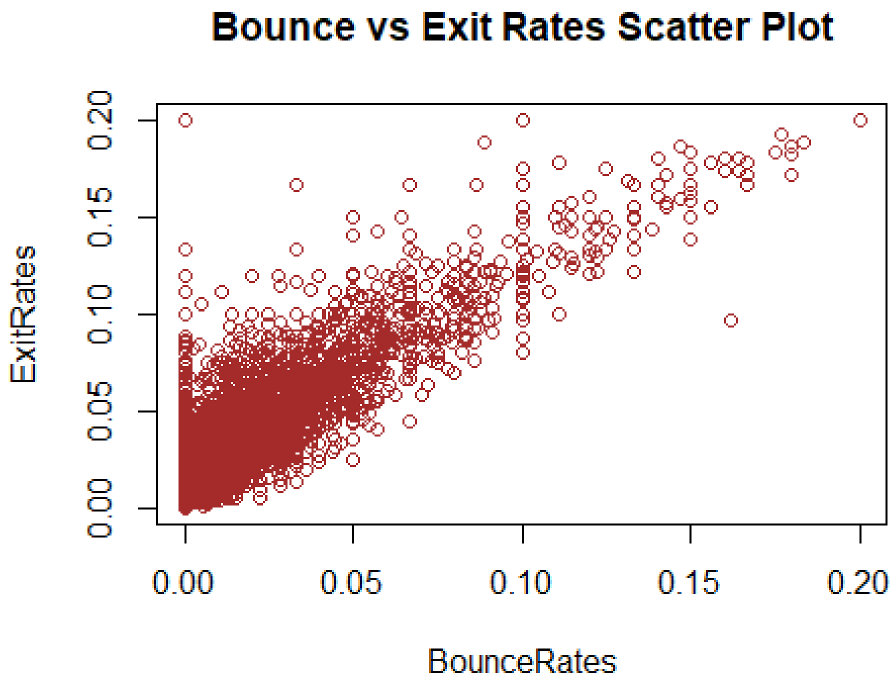


There is a positive non-linear correlation between the time spent on the Administrative site and the Informational site

Metrics

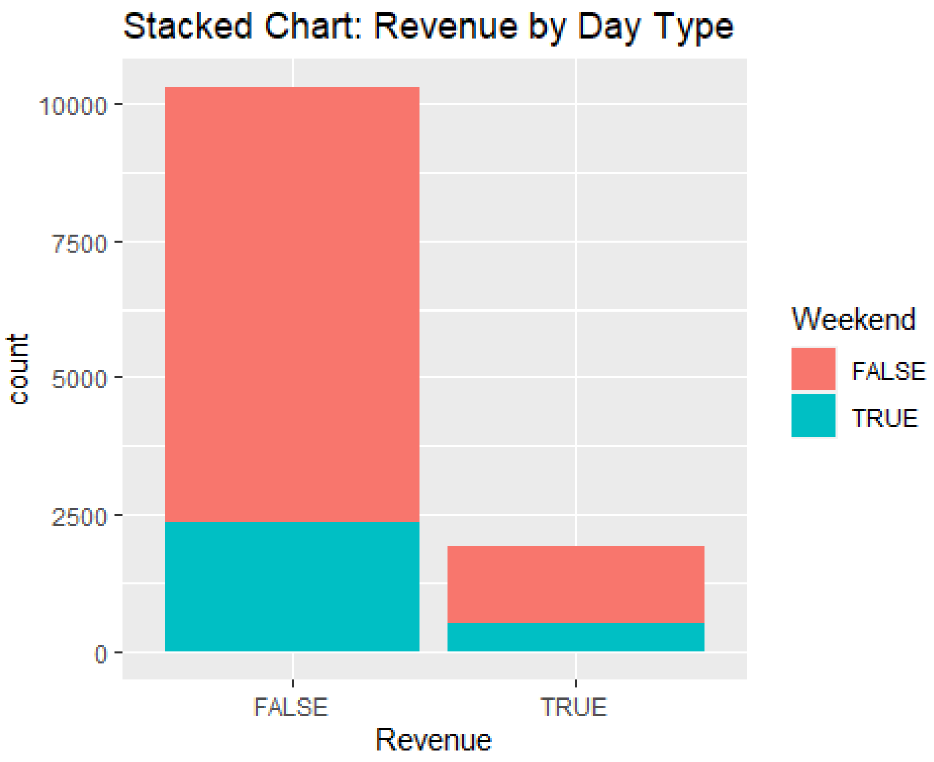
Scatter plot Bounce vs Exit Rates Scatter Plot

```
plot(ExitRates ~ BounceRates, dat = markert_df2,
      col = "brown",
      main = "Bounce vs Exit Rates Scatter Plot")
```



Stacked bar chart: Revenue vs Day Type

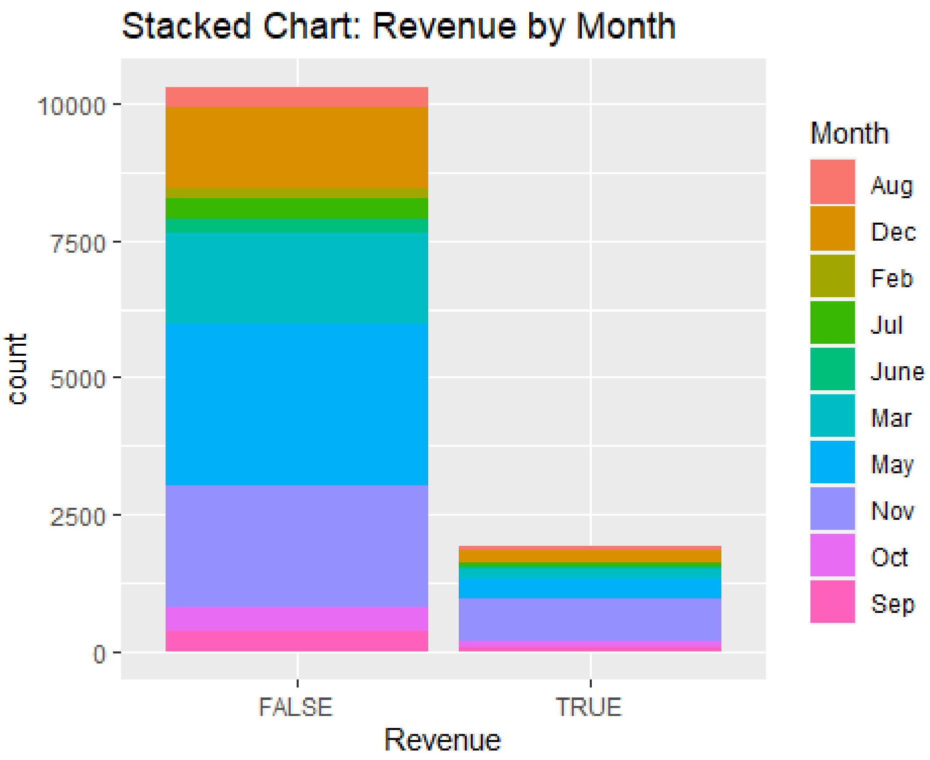
```r
library(magrittr)
markert_df2 %>%
    ggplot(aes(Revenue)) +
    geom_bar(aes(fill = Weekend))+
    labs(title = "Stacked Chart: Revenue by Day Type")
```



From the stacked chart, we can see that most of the revenue is generated during the week and not over the weekend

Revenue vs Month

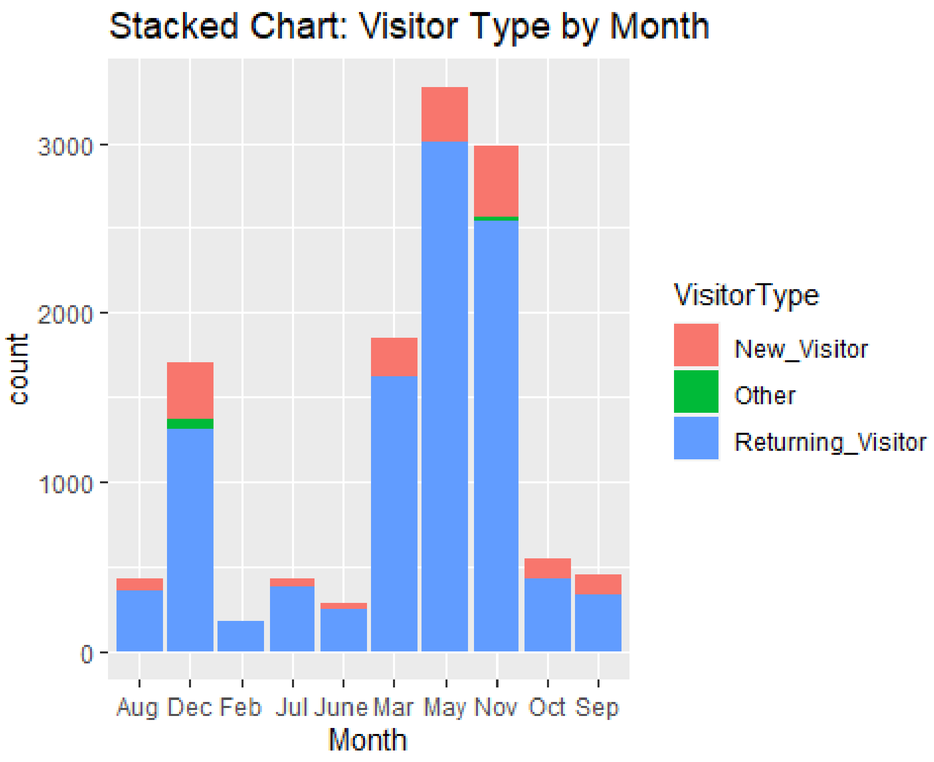
```
# Stacked bar chart: Revenue vs Month
markert_df2 %>%
    ggplot(aes(Revenue)) +
    geom_bar(aes(fill = Month))+
    labs(title = "Stacked Chart: Revenue by Month")
```



Type of visitor

Stacked bar chart: Visitor Type vs Month

```
markert_df2 %>%
    ggplot(aes(Month)) +
    geom_bar(aes(fill = VisitorType))+
    labs(title = "Stacked Chart: Visitor Type by Month")
```



Multivariate Analysis

```
# A glimpse of the data
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:data.table':
##
##     between, first, last

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

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

glimpse(markert_df2)

## Observations: 12,199
## Variables: 18
## $ Administrative          <int> 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0,
0...
## $ Administrative_Duration <dbl> 0, 0, -1, 0, 0, 0, -1, -1, 0, 0, 0, 0, 0,
0...
```

```
## $ Informational           <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0...
## $ Informational_Duration  <dbl> 0, 0, -1, 0, 0, 0, -1, -1, 0, 0, 0, 0, 0,
0...
## $ ProductRelated          <int> 1, 2, 1, 2, 10, 19, 1, 1, 2, 3, 3, 16, 7,
6...
## $ ProductRelated_Duration <dbl> 0.000000, 64.000000, -1.000000, 2.666667,
6...
## $ BounceRates             <dbl> 0.200000000, 0.000000000, 0.200000000,
0.05...
## $ ExitRates               <dbl> 0.200000000, 0.100000000, 0.200000000,
0.14...
## $ PageValues              <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0...
## $ SpecialDay              <dbl> 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.4, 0.0,
0.8...
## $ Month                   <fct> Feb, Feb, Feb, Feb, Feb, Feb, Feb, Feb,
Feb...
## $ OperatingSystems        <int> 1, 2, 4, 3, 3, 2, 2, 1, 2, 2, 1, 1, 1, 2,
3...
## $ Browser                 <int> 1, 2, 1, 2, 3, 2, 4, 2, 2, 4, 1, 1, 1, 5,
2...
## $ Region                  <int> 1, 1, 9, 2, 1, 1, 3, 1, 2, 1, 3, 4, 1, 1,
3...
## $ TrafficType             <int> 1, 2, 3, 4, 4, 3, 3, 5, 3, 2, 3, 3, 3, 3,
3...
## $ VisitorType             <fct> Returning_Visitor, Returning_Visitor,
Retur...
## $ Weekend                 <lgl> FALSE, FALSE, FALSE, FALSE, TRUE, FALSE,
FA...
## $ Revenue                 <lgl> FALSE, FALSE, FALSE, FALSE, FALSE, FALSE,
F...
```

dummify the data

```
# One hot encoding of the factor variables.

library(caret)

## Loading required package: lattice

dmy <- dummyVars(" ~ .", data = markert_df2)
dummy_df <- data.frame(predict(dmy, newdata = markert_df2))
#print(dummy_df)
glimpse(dummy_df)

## Observations: 12,199
## Variables: 31
## $ Administrative                <dbl> 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0,
0...
## $ Administrative_Duration       <dbl> 0, 0, -1, 0, 0, 0, -1, -1, 0, 0, 0,
0...
## $ Informational                 <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0...
## $ Informational_Duration        <dbl> 0, 0, -1, 0, 0, 0, -1, -1, 0, 0, 0,
0...
## $ ProductRelated                <dbl> 1, 2, 1, 2, 10, 19, 1, 1, 2, 3, 3,
16...
## $ ProductRelated_Duration       <dbl> 0.000000, 64.000000, -1.000000,
2.666...
## $ BounceRates                   <dbl> 0.200000000, 0.000000000,
0.200000000...
## $ ExitRates                     <dbl> 0.200000000, 0.100000000,
0.200000000...
## $ PageValues                    <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0...
## $ SpecialDay                    <dbl> 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.4,
0....
## $ Month.Aug                     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0...
## $ Month.Dec                     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0...
## $ Month.Feb                     <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1...
## $ Month.Jul                     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0...
## $ Month.June                    <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0...
## $ Month.Mar                     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0...
## $ Month.May                     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0...
## $ Month.Nov                     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0...
## $ Month.Oct                     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0...
## $ Month.Sep                     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0...
## $ OperatingSystems              <dbl> 1, 2, 4, 3, 3, 2, 2, 1, 2, 2, 1, 1,
1...
## $ Browser                       <dbl> 1, 2, 1, 2, 3, 2, 4, 2, 2, 4, 1, 1,
1...
## $ Region                        <dbl> 1, 1, 9, 2, 1, 1, 3, 1, 2, 1, 3, 4,
1...
## $ TrafficType                   <dbl> 1, 2, 3, 4, 4, 3, 3, 5, 3, 2, 3, 3,
3...
## $ VisitorType.New_Visitor       <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0...
## $ VisitorType.Other             <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0...
## $ VisitorType.Returning_Visitor <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1...
## $ WeekendFALSE                  <dbl> 1, 1, 1, 1, 0, 1, 1, 0, 1, 1, 1, 1,
1...
## $ WeekendTRUE                   <dbl> 0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0,
0...
## $ RevenueFALSE                  <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,
1...
## $ RevenueTRUE                   <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0...
```

Checking the resultant datatype

```
sapply(dummy_df, class)
```

```
##                Administrative       Administrative_Duration
##                     "numeric"                     "numeric"
##                 Informational        Informational_Duration
##                     "numeric"                     "numeric"
##                ProductRelated       ProductRelated_Duration
##                     "numeric"                     "numeric"
##                   BounceRates                     ExitRates
##                     "numeric"                     "numeric"
##                    PageValues                    SpecialDay
##                     "numeric"                     "numeric"
##                     Month.Aug                     Month.Dec
##                     "numeric"                     "numeric"
##                     Month.Feb                     Month.Jul
##                     "numeric"                     "numeric"
##                    Month.June                     Month.Mar
##                     "numeric"                     "numeric"
##                     Month.May                     Month.Nov
##                     "numeric"                     "numeric"
##                     Month.Oct                     Month.Sep
##                     "numeric"                     "numeric"
##              OperatingSystems                       Browser
##                     "numeric"                     "numeric"
##                        Region                   TrafficType
##                     "numeric"                     "numeric"
##       VisitorType.New_Visitor             VisitorType.Other
##                     "numeric"                     "numeric"
## VisitorType.Returning_Visitor                  WeekendFALSE
##                     "numeric"                     "numeric"
##                   WeekendTRUE                  RevenueFALSE
##                     "numeric"                     "numeric"
##                   RevenueTRUE
##                     "numeric"
```

Seperating the dependent and independent variables

```
#removing the revenue column from the data
#we select all the column indexes before 30
```

```
dummy_df2 <- dummy_df[, -c(30:31)]
dim(dummy_df2)

## [1] 12199    29

#29 columns in dummy_df2

dummy_df.class<- markert_df2[, "Revenue"]
```

SCALING VS NORMALIZATION

Scaling

In this step the data is transformed to fit withing the range between 0 and 1

```
dummy_df2_scaled <- scale(dummy_df2)
summary(dummy_df2_scaled)

##  Administrative    Administrative_Duration Informational   
##  Min.   :-0.7025   Min.   :-0.46574        Min.   :-0.3988  
##  1st Qu.:-0.7025   1st Qu.:-0.46011        1st Qu.:-0.3988  
##  Median :-0.4023   Median :-0.40941        Median :-0.3988  
##  Mean   : 0.0000   Mean   : 0.00000        Mean   : 0.0000  
##  3rd Qu.: 0.4984   3rd Qu.: 0.07361        3rd Qu.:-0.3988  
##  Max.   : 7.4035   Max.   :18.68474        Max.   :18.4127  
##  Informational_Duration ProductRelated    ProductRelated_Duration
##  Min.   :-0.2533        Min.   :-0.7188   Min.   :-0.6295        
##  1st Qu.:-0.2463        1st Qu.:-0.5394   1st Qu.:-0.5281        
##  Median :-0.2463        Median :-0.3152   Median :-0.3115        
##  Mean   : 0.0000        Mean   : 0.0000   Mean   : 0.0000        
##  3rd Qu.:-0.2463        3rd Qu.: 0.1332   3rd Qu.: 0.1407        
##  Max.   :17.7758        Max.   :15.0881   Max.   :32.6919        
##   BounceRates         ExitRates         PageValues       SpecialDay    
##  Min.   :-0.45034   Min.   :-0.8973   Min.   :-0.319   Min.   :-0.3103  
##  1st Qu.:-0.45034   1st Qu.:-0.5897   1st Qu.:-0.319   1st Qu.:-0.3103  
##  Median :-0.38580   Median :-0.3567   Median :-0.319   Median :-0.3103  
##  Mean   : 0.00000   Mean   : 0.0000   Mean   : 0.000   Mean   : 0.0000  
##  3rd Qu.:-0.08326   3rd Qu.: 0.1511   3rd Qu.:-0.319   3rd Qu.:-0.3103  
##  Max.   : 3.95470   Max.   : 3.4273   Max.   :19.070   Max.   : 4.6969  
##    Month.Aug         Month.Dec         Month.Feb         Month.Jul     
##  Min.   :-0.1918   Min.   :-0.4032   Min.   :-0.1231   Min.   :-0.1916  
##  1st Qu.:-0.1918   1st Qu.:-0.4032   1st Qu.:-0.1231   1st Qu.:-0.1916  
##  Median :-0.1918   Median :-0.4032   Median :-0.1231   Median :-0.1916  
##  Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000  
##  3rd Qu.:-0.1918   3rd Qu.:-0.4032   3rd Qu.:-0.1231   3rd Qu.:-0.1916  
##  Max.   : 5.2126   Max.   : 2.4799   Max.   : 8.1254   Max.   : 5.2188  
##    Month.June        Month.Mar         Month.May         Month.Nov     
##  Min.   :-0.1547   Min.   :-0.4232   Min.   :-0.6125   Min.   :-0.5689  
##  1st Qu.:-0.1547   1st Qu.:-0.4232   1st Qu.:-0.6125   1st Qu.:-0.5689  
##  Median :-0.1547   Median :-0.4232   Median :-0.6125   Median :-0.5689  
##  Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000  
##  3rd Qu.:-0.1547   3rd Qu.:-0.4232   3rd Qu.: 1.6326   3rd Qu.:-0.5689  
##  Max.   : 6.4653   Max.   : 2.3628   Max.   : 1.6326   Max.   : 1.7576  
##    Month.Oct         Month.Sep       OperatingSystems     Browser       
##  Min.   :-0.2171   Min.   :-0.1952   Min.   :-1.2397   Min.   :-0.7940  
##  1st Qu.:-0.2171   1st Qu.:-0.1952   1st Qu.:-0.1371   1st Qu.:-0.2094  
##  Median :-0.2171   Median :-0.1952   Median :-0.1371   Median :-0.2094  
##  Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000  
##  3rd Qu.:-0.2171   3rd Qu.:-0.1952   3rd Qu.: 0.9654   3rd Qu.:-0.2094  
##  Max.   : 4.6064   Max.   : 5.1213   Max.   : 6.4782   Max.   : 6.2212  
##      Region          TrafficType       VisitorType.New_Visitor
##  Min.   :-0.89629   Min.   :-0.76562   Min.   :-0.4014        
##  1st Qu.:-0.89629   1st Qu.:-0.51661   1st Qu.:-0.4014        
##  Median :-0.06381   Median :-0.51661   Median :-0.4014        
##  Mean   : 0.00000   Mean   : 0.00000   Mean   : 0.0000        
##  3rd Qu.: 0.35244   3rd Qu.:-0.01858   3rd Qu.:-0.4014        
##  Max.   : 2.43366   Max.   : 3.96567   Max.   : 2.4910        
##  VisitorType.Other  VisitorType.Returning_Visitor  WeekendFALSE    
##  Min.   :-0.08175   Min.   :-2.4241               Min.   :-1.8086  
##  1st Qu.:-0.08175   1st Qu.: 0.4125               1st Qu.: 0.5529  
##  Median :-0.08175   Median : 0.4125               Median : 0.5529  
##  Mean   : 0.00000   Mean   : 0.0000               Mean   : 0.0000  
##  3rd Qu.:-0.08175   3rd Qu.: 0.4125               3rd Qu.: 0.5529  
##  Max.   :12.23081   Max.   : 0.4125               Max.   : 0.5529  
##   WeekendTRUE     
##  Min.   :-0.5529  
##  1st Qu.:-0.5529  
##  Median :-0.5529  
##  Mean   : 0.0000  
##  3rd Qu.:-0.5529  
##  Max.   : 1.8086
```

Normalizing

Normalization is a technique often applied to change the values of numeric columns in the dataset to a common scale, without distorting differences in the ranges of values.

```
dummy_df2_norm <- as.data.frame(apply(dummy_df2, 2, function(x) (x -
min(x))/(max(x)-min(x))))
summary(dummy_df2_norm)
```

```
##  Administrative    Administrative_Duration Informational   
##  Min.   :0.00000   Min.   :0.0000000       Min.   :0.0000  
##  1st Qu.:0.00000   1st Qu.:0.0002941       1st Qu.:0.0000  
##  Median :0.03704   Median :0.0029414       Median :0.0000  
##  Mean   :0.08667   Mean   :0.0243201       Mean   :0.0212  
##  3rd Qu.:0.14815   3rd Qu.:0.0281638       3rd Qu.:0.0000  
##  Max.   :1.00000   Max.   :1.0000000       Max.   :1.0000  
##  Informational_Duration ProductRelated    ProductRelated_Duration
##  Min.   :0.0000000      Min.   :0.00000   Min.   :0.000000       
##  1st Qu.:0.0003921      1st Qu.:0.01135   1st Qu.:0.003042       
##  Median :0.0003921      Median :0.02553   Median :0.009543
##  Mean   :0.0140518      Mean   :0.04547   Mean   :0.018891
##  3rd Qu.:0.0003921      3rd Qu.:0.05390   3rd Qu.:0.023112
##  Max.   :1.0000000      Max.   :1.00000   Max.   :1.000000
##   BounceRates        ExitRates         PageValues        SpecialDay
##  Min.   :0.00000   Min.   :0.00000   Min.   :0.00000   Min.   :0.00000
##  1st Qu.:0.00000   1st Qu.:0.07111   1st Qu.:0.00000   1st Qu.:0.00000
##  Median :0.01465   Median :0.12500   Median :0.00000   Median :0.00000
##  Mean   :0.10223   Mean   :0.20748   Mean   :0.01645   Mean   :0.06197
##  3rd Qu.:0.08333   3rd Qu.:0.24242   3rd Qu.:0.00000   3rd Qu.:0.00000
##  Max.   :1.00000   Max.   :1.00000   Max.   :1.00000   Max.   :1.00000
##    Month.Aug         Month.Dec        Month.Feb         Month.Jul
##  Min.   :0.00000   Min.   :0.0000   Min.   :0.00000   Min.   :0.00000
##  1st Qu.:0.00000   1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:0.00000
##  Median :0.00000   Median :0.0000   Median :0.00000   Median :0.00000
##  Mean   :0.03549   Mean   :0.1398   Mean   :0.01492   Mean   :0.03541
##  3rd Qu.:0.00000   3rd Qu.:0.0000   3rd Qu.:0.00000   3rd Qu.:0.00000
##  Max.   :1.00000   Max.   :1.0000   Max.   :1.00000   Max.   :1.00000
##    Month.June        Month.Mar        Month.May        Month.Nov
##  Min.   :0.00000   Min.   :0.0000   Min.   :0.0000   Min.   :0.0000
##  1st Qu.:0.00000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000
##  Median :0.00000   Median :0.0000   Median :0.0000   Median :0.0000
##  Mean   :0.02336   Mean   :0.1519   Mean   :0.2728   Mean   :0.2445
##  3rd Qu.:0.00000   3rd Qu.:0.0000   3rd Qu.:1.0000   3rd Qu.:0.0000
##  Max.   :1.00000   Max.   :1.0000   Max.   :1.0000   Max.   :1.0000
##    Month.Oct       Month.Sep       OperatingSystems    Browser
##  Min.   :0.000   Min.   :0.00000   Min.   :0.0000   Min.   :0.00000
##  1st Qu.:0.000   1st Qu.:0.00000   1st Qu.:0.1429   1st Qu.:0.08333
##  Median :0.000   Median :0.00000   Median :0.1429   Median :0.08333
##  Mean   :0.045   Mean   :0.03672   Mean   :0.1606   Mean   :0.11318
##  3rd Qu.:0.000   3rd Qu.:0.00000   3rd Qu.:0.2857   3rd Qu.:0.08333
##  Max.   :1.000   Max.   :1.00000   Max.   :1.0000   Max.   :1.00000
##      Region        TrafficType      VisitorType.New_Visitor
VisitorType.Other
##  Min.   :0.0000   Min.   :0.00000   Min.   :0.0000          Min.
 :0.00000
##  1st Qu.:0.0000   1st Qu.:0.05263   1st Qu.:0.0000          1st
Qu.:0.00000
##  Median :0.2500   Median :0.05263   Median :0.0000          Median
 :0.00000
##  Mean   :0.2692   Mean   :0.16182   Mean   :0.1388          Mean
 :0.00664
##  3rd Qu.:0.3750   3rd Qu.:0.15789   3rd Qu.:0.0000          3rd
Qu.:0.00000
##  Max.   :1.0000   Max.   :1.00000   Max.   :1.0000          Max.
 :1.00000
##  VisitorType.Returning_Visitor  WeekendFALSE     WeekendTRUE
##  Min.   :0.0000                Min.   :0.0000   Min.   :0.0000
##  1st Qu.:1.0000                1st Qu.:1.0000   1st Qu.:0.0000
##  Median :1.0000                Median :1.0000   Median :0.0000
##  Mean   :0.8546                Mean   :0.7659   Mean   :0.2341
##  3rd Qu.:1.0000                 3rd Qu.:1.0000   3rd Qu.:0.0000  
##  Max.   :1.0000                 Max.   :1.0000   Max.   :1.0000  
```

visualizing the distance matrix Euclidean Distances

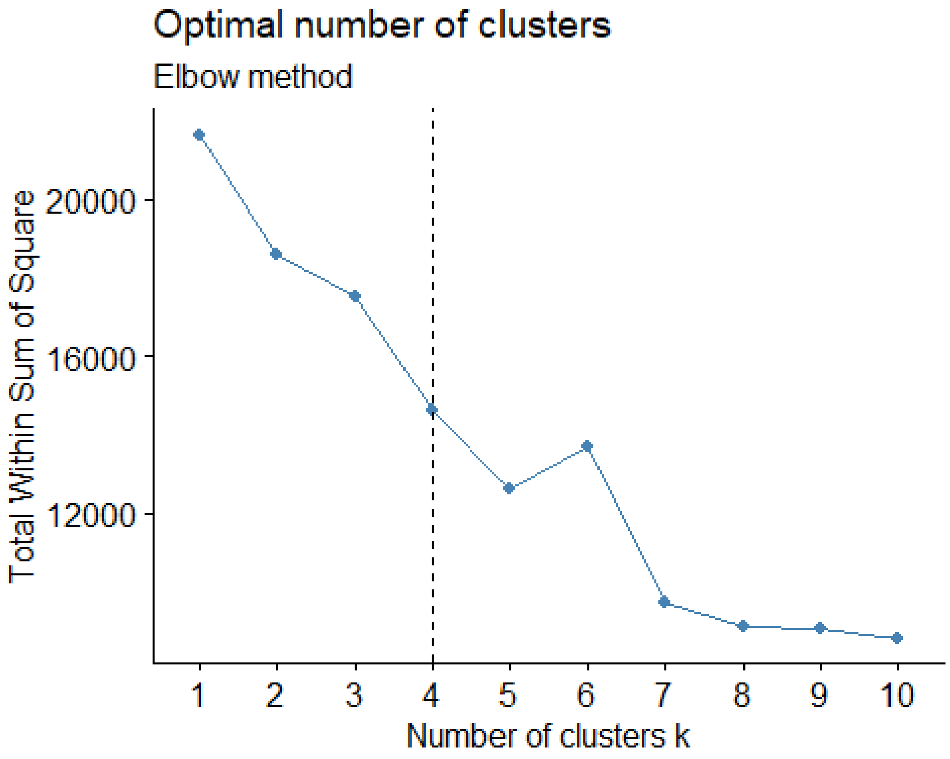
```
#distance <- get_dist(dummy_df2_norm)
#fviz_dist(distance, gradient = list(low = "#00AFBB", mid = "white", high =
"#FC4E07"))
```

The normalized dataset has a smaller range for the values which are between 0 and 1 unlike the standardized dataset which has values ranging from -5 to 19

Finding the Optimal number of clusters

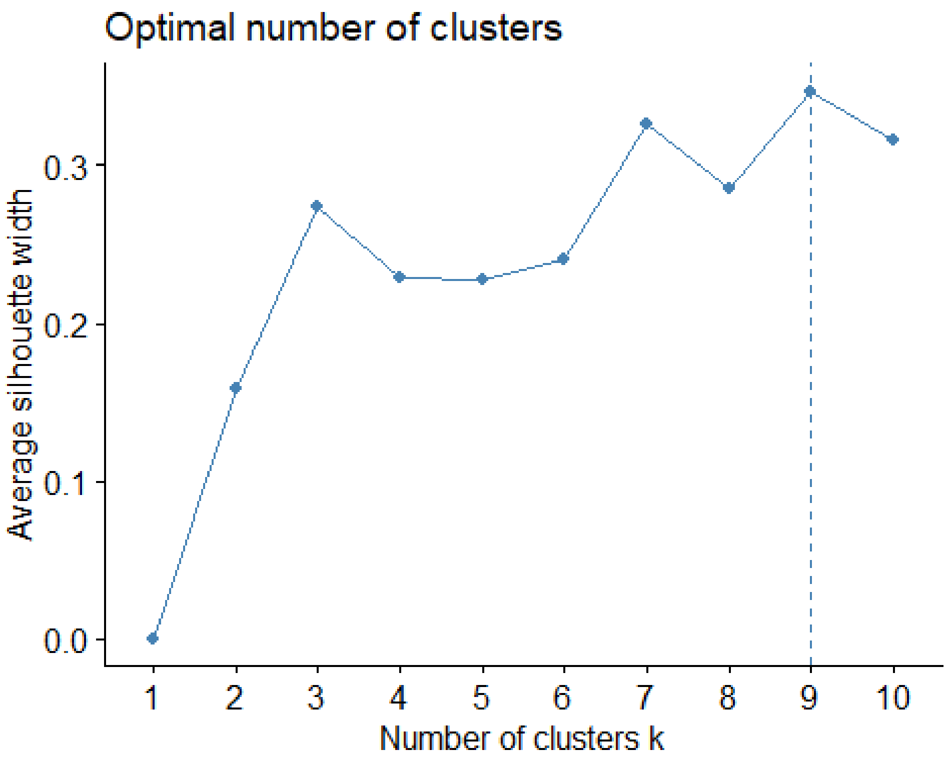
Method 1: Elbow method

```
# Searching for the optimal number of clusters
# # Elbow method

# Searching for the optimal number of clusters
# # Elbow method
library(factoextra)

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

fviz_nbclust(dummy_df2_norm, kmeans, method = "wss") +
    geom_vline(xintercept = 4, linetype = 2)+
  labs(subtitle = "Elbow method")
```



Method 2: Silhouette

```
library(cluster)
fviz_nbclust(dummy_df2_norm, kmeans, method = "silhouette")
```



Implement the Solution

K-MEANS CLUSTERING

```
outputk <- kmeans(dummy_df2_norm, 4)
```

####Results

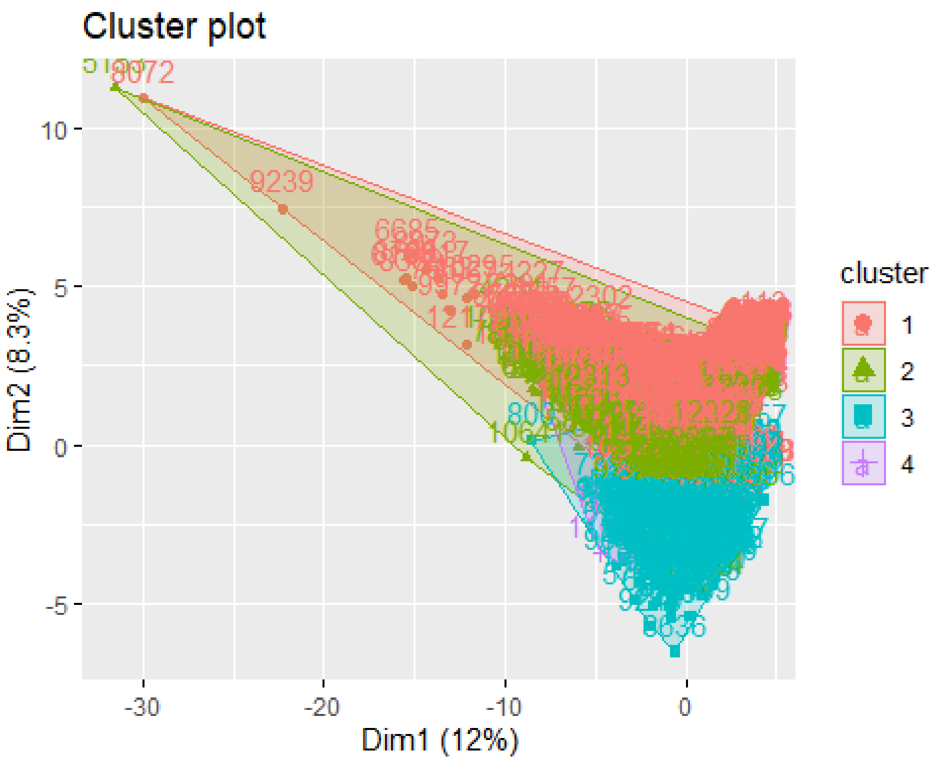
```
# Previewing the number of records in each cluster

outputk$size

## [1] 8065 1993 1666  475
```

The cluster center datapoints Per attribute

```
outputk$centers

##   Administrative Administrative_Duration Informational
Informational_Duration
## 1     0.08374090              0.02343605    0.02124406
0.014217850
## 2     0.09557525              0.02648357    0.02816106
0.019026956
## 3     0.09274821              0.02688726    0.01333033
0.007376798
## 4     0.07766082              0.02124794    0.01885965
0.013769189
##   ProductRelated ProductRelated_Duration BounceRates ExitRates PageValues
## 1     0.04808009              0.02018870  0.11939703 0.2312374 0.01344207
## 2     0.05525225              0.02242966  0.10576042 0.2052659 0.01642964
## 3     0.02555746              0.01001773  0.02602540 0.1067013 0.03168349
## 4     0.03002613              0.01311793  0.06331099 0.1669631 0.01428299
##   SpecialDay  Month.Aug Month.Dec    Month.Feb  Month.Jul Month.June
Month.Mar
## 1 0.07079975 0.03583385 0.1279603 0.0189708617 0.03558586 0.02641042
0.15337880
## 2 0.07566483 0.03612644 0.1455093 0.0140491721 0.04565981 0.02057200
0.00000000
## 3 0.02052821 0.04321729 0.2304922 0.0006002401 0.03241297 0.01860744
0.08463385
## 4 0.00000000 0.00000000 0.0000000 0.0000000000 0.00000000 0.00000000
1.00000000
##   Month.May Month.Nov  Month.Oct  Month.Sep OperatingSystems    Browser
## 1 0.2951023 0.2339740 0.03980161 0.03298202        0.1583739 0.11228560
## 2 0.3156046 0.3331661 0.05218264 0.03712995        0.1620672 0.10394715
## 3 0.1914766 0.2593037 0.07442977 0.06482593        0.1692677 0.13325330
## 4 0.0000000 0.0000000 0.00000000 0.00000000        0.1624060 0.09666667
##      Region TrafficType VisitorType.New_Visitor VisitorType.Other
## 1 0.2630657   0.1593565               0.0000000       0.001239926
## 2 0.2696312   0.1694087               0.0000000       0.003512293
## 3 0.3085984   0.1729007               0.9615846       0.038415366
## 4 0.2323684   0.1329640               0.1915789       0.000000000
##   VisitorType.Returning_Visitor WeekendFALSE WeekendTRUE
## 1                     0.9987601    1.0000000   0.0000000
## 2                     0.9964877    0.0000000   1.0000000
## 3                     0.0000000    0.7671068   0.2328932
## 4                     0.8084211    0.0000000   1.0000000
```

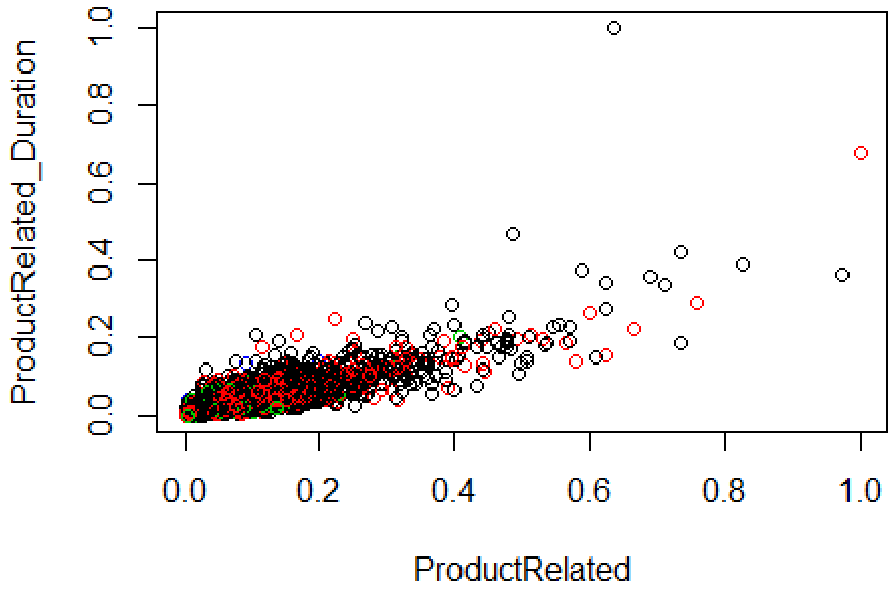
Visualising the clusters of the whole dataset

```
options(repr.plot.width = 11, repr.plot.height = 6)
fviz_cluster(outputk, dummy_df2_norm)
```



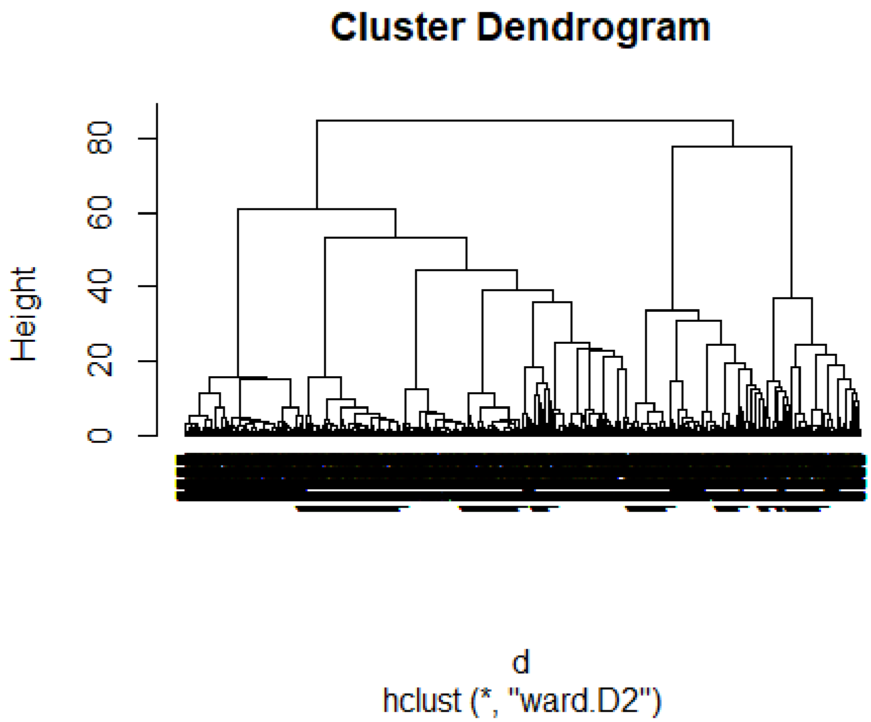
Visualizing variable datatypes on a scatter plot

```r
# Plotting two variables to see how their data points
# have been distributed in the cluster
# Product Related, vs Product Related Duration

plot(dummy_df2_norm[, 5:6], col = outputk$cluster)
```



HIERACHICAL CLUSTERING

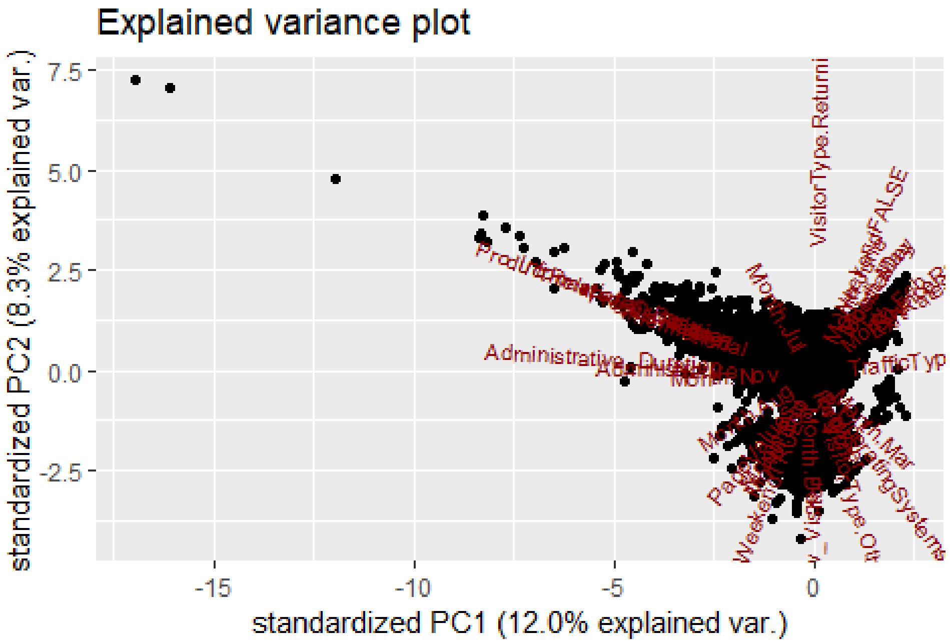
```
d <- dist(dummy_df2_norm, method = "euclidean")

# We then apply hierarchical clustering using the Ward's method

res.hc <- hclust(d, method = "ward.D2")

# Lastly we plot the obtained dendrogram
#--

plot(res.hc, cex = 0.6, hang = -1)
```



Challenging the Solution

PCA

```
# Reducing the dimensionality of the dataset
library(ggbiplot)

## Loading required package: plyr

## ------------------------------------------------------------------------------
----

## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first,
then dplyr:
## library(plyr); library(dplyr)

## ------------------------------------------------------------------------------
----

##
## Attaching package: 'plyr'

## The following objects are masked from 'package:dplyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
## Loading required package: scales

##
## Attaching package: 'scales'

## The following object is masked from 'package:readr':
##
##     col_factor

## Loading required package: grid

pca_residual = prcomp(dummy_df2_norm, scale = T, center = T)

# Visualising the pca results
options(repr.plot.width = 6, repr.plot.height = 6)
ggbiplot(pca_residual) +
  labs(title = 'Explained variance plot')
```



Dummify the variables

```
# Applying PCA
# We pass df_norm to the prcomp().
# We also set two arguments, center and scale,
# to be TRUE then preview our object with summary
dummy_PCA <- prcomp(dummy_df2_norm,
              center = TRUE,
              scale = FALSE)
summary(dummy_PCA)
```

```
## Importance of components:
##                           PC1    PC2    PC3    PC4     PC5     PC6     PC7
## Standard deviation     0.6027 0.5249 0.4890 0.4369 0.37908 0.31341 0.30033
## Proportion of Variance 0.2047 0.1553 0.1348 0.1076 0.08101 0.05537 0.05085
## Cumulative Proportion  0.2047 0.3600 0.4948 0.6024 0.68343 0.73880 0.78965
##                            PC8     PC9    PC10    PC11    PC12    PC13
PC14
## Standard deviation     0.25907 0.21400 0.20283 0.19014 0.18821 0.17371
0.15733
## Proportion of Variance 0.03784 0.02582 0.02319 0.02038 0.01997 0.01701
0.01395
## Cumulative Proportion  0.82748 0.85330 0.87649 0.89687 0.91684 0.93385
0.94781
##                           PC15   PC16    PC17    PC18    PC19    PC20
PC21
## Standard deviation     0.15027 0.1298 0.12147 0.11865 0.08500 0.06923
0.06523
## Proportion of Variance 0.01273 0.0095 0.00832 0.00794 0.00407 0.00270
0.00240
## Cumulative Proportion  0.96054 0.9700 0.97835 0.98629 0.99036 0.99307
0.99546
##                           PC22    PC23    PC24    PC25    PC26      PC27
## Standard deviation     0.05217 0.04953 0.04018 0.03288 0.01328 3.259e-15
## Proportion of Variance 0.00153 0.00138 0.00091 0.00061 0.00010 0.000e+00
## Cumulative Proportion  0.99700 0.99838 0.99929 0.99990 1.00000 1.000e+00
##                             PC28      PC29
## Standard deviation     2.477e-15 1.496e-15
## Proportion of Variance 0.000e+00 0.000e+00
## Cumulative Proportion  1.000e+00 1.000e+00
```

The Principal Components and how well they explain the variance

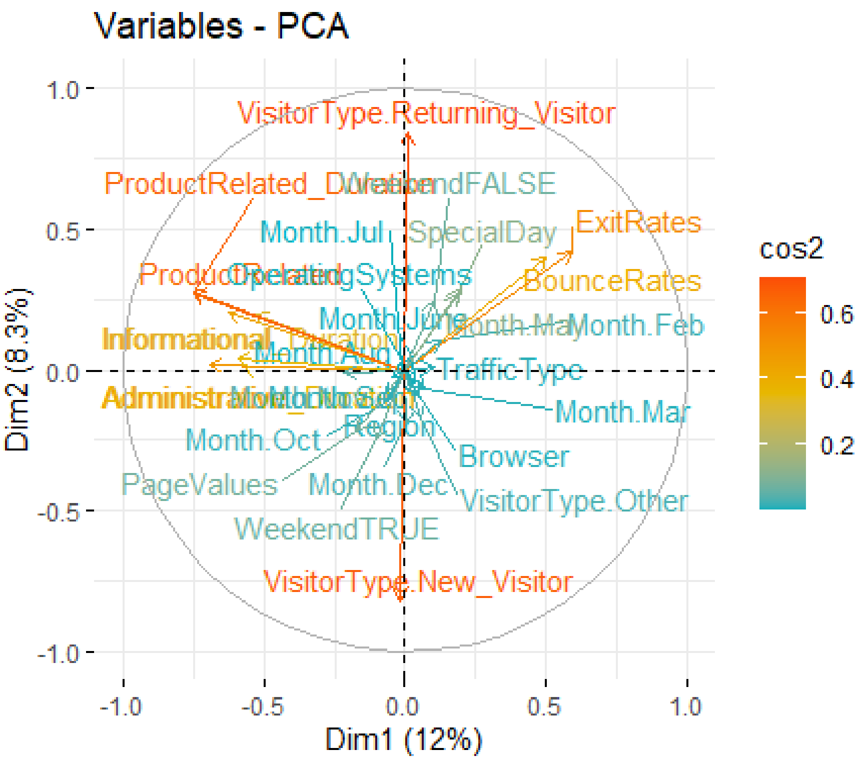
```
var <- get_pca_var(pca_residual)
head(var$contrib, 9)
```

```
##                              Dim.1       Dim.2       Dim.3      Dim.4
## Administrative          13.9170391 0.009209892 0.201793431 0.03599038
## Administrative_Duration 10.1448702 0.057506724 0.207334484 0.13541694
## Informational           11.3501623 1.714523189 0.007380966 0.54436384
## Informational_Duration   8.1540775 1.639658273 0.003786998 0.64146982
## ProductRelated          16.3067695 2.964280882 0.034580216 0.43822803
## ProductRelated_Duration 16.3010236 3.243816954 0.065364681 0.62995372
## BounceRates              7.2582381 6.643907579 0.314692329 3.24508463
## ExitRates               10.1887277 7.308693311 0.310908281 2.65886831
## PageValues               0.8516718 2.009493493 0.394286524 0.24831605
##                                Dim.5        Dim.6      Dim.7     Dim.8
## Administrative          3.018123e-01  1.231027603 1.55064165 5.0567064
## Administrative_Duration 2.573771e-01  2.661759501 2.61659964 5.9767375
## Informational           5.940552e-03  5.956602161 3.92665547 0.8565455
## Informational_Duration  9.002913e-04  6.997529354 5.07142477 1.6233809
## ProductRelated          9.281220e-06  0.001500608 3.01235280 0.5699698
## ProductRelated_Duration 6.429665e-03  0.111362950 1.80705683 0.8119072
## BounceRates             3.296224e+00 24.382637202 0.07777457 0.6713438
## ExitRates               3.223972e+00 19.089300629 0.17881645 0.3115298
## PageValues              4.334139e-01  0.330098643 0.66098067 0.1399238
##                               Dim.9       Dim.10       Dim.11       Dim.12
## Administrative          2.714410102 4.176111e-01 0.4616170786 8.705881e-05
## Administrative_Duration 1.606851104 8.255884e-01 0.7547538629 2.867152e-03
## Informational           4.912421038 5.048420e-02 0.1699183718 6.493103e-02
## Informational_Duration  7.481595699 7.051776e-02 0.4472806645 1.688859e-01
## ProductRelated          0.307059385 1.315947e+00 0.0005384326 1.275772e-01
## ProductRelated_Duration 0.001460471 1.049039e+00 0.0091644974 5.243858e-02
## BounceRates             0.140953500 4.521648e-04 0.1613438972 4.393957e-03
## ExitRates               0.025267109 2.537698e-05 0.1191672626 4.227516e-03
## PageValues              0.161682142 1.981605e+00 0.1546673263 1.769753e-01
##                               Dim.13       Dim.14     Dim.15       Dim.16
## Administrative          0.0004845723 0.3853068766  4.4862317 1.540434e-01
## Administrative_Duration 0.0935894307 0.7805167023  9.1515707 2.158903e-01
## Informational           0.0274253210 0.0024245897 10.7381115 1.144402e+00
## Informational_Duration  0.4144941830 0.0009861781 16.5809998 1.486534e+00
## ProductRelated          0.1236644509 1.3908628317  4.4984200 3.874419e-01
## ProductRelated_Duration 0.0867417784 1.1116032033  4.5688199 2.234820e-01
## BounceRates             0.0011218000 0.5998422257  0.8245177 3.167380e-02
## ExitRates               0.0005334053 0.3758479996  0.5495094 1.088715e-02
## PageValues              0.0235648551 2.0350586838 25.6185252 3.535846e-05
##                              Dim.17     Dim.18      Dim.19      Dim.20
## Administrative           3.74410883  5.8803694 2.105429454 0.086622166
## Administrative_Duration  7.13471105 10.5782802 3.929044941 0.003163934
## Informational            2.07937375  0.8425020 0.386749774 0.053483231
## Informational_Duration   4.63693067  0.2065091 0.257553698 0.016936848
## ProductRelated           0.03307171 12.6974074 5.624192770 0.159303834
## ProductRelated_Duration  0.01430887 12.2550326 5.856253606 0.388440607
## BounceRates              2.25609259  2.8877526 0.043402060 0.021664954
## ExitRates                0.98039593  1.8674661 0.007470498 0.001047996
## PageValues              53.91929560  5.3502577 4.385737477 0.302133689
##                              Dim.21      Dim.22       Dim.23       Dim.24
## Administrative          0.002944602 0.331935086 33.079748750 2.107146e+01
## Administrative_Duration 0.021728532 0.295050630 30.935299946 9.505671e+00
## Informational           0.005587917 0.004845165 15.946677284 3.912280e+01
## Informational_Duration  0.049967039 0.043288309 13.860928395 2.954175e+01
## ProductRelated          0.030669514 0.368782543  1.194145893 9.543002e-02
## ProductRelated_Duration 0.012285631 0.933599205  3.716721852 2.302616e-01
## BounceRates             0.088815019 0.053443100  0.526509416 1.052747e-01
## ExitRates               0.004065392 0.001425502  0.005463358 9.561249e-04
## PageValues              0.072684036 0.560163580  0.052328725 4.402316e-03
##                               Dim.25       Dim.26       Dim.27
Dim.28
## Administrative           2.690571551 8.279934e-02 5.291665e-27 4.901231e-
29
## Administrative_Duration  2.054265621 5.355475e-02 2.231317e-27 2.438692e-
29
```

```
## Informational            0.085227952 4.582118e-04 2.599792e-28 2.066272e-
29
## Informational_Duration   0.594049702 8.566945e-03 1.747734e-28 1.578141e-
28
## ProductRelated          45.005242569 3.312552e+00 4.176491e-28 9.890316e-
28
## ProductRelated_Duration 44.904609044 1.608822e+00 2.246101e-29 2.876558e-
29
## BounceRates              2.322038074 4.404081e+01 3.596795e-28 4.012747e-
27
## ExitRates                2.199604319 5.057582e+01 1.950194e-28 2.399542e-
27
## PageValues               0.003593179 1.291046e-01 7.299501e-30 2.732457e-
30
##                              Dim.29
## Administrative          5.602701e-31
## Administrative_Duration 1.992482e-31
## Informational           6.347817e-32
## Informational_Duration  2.550764e-30
## ProductRelated          1.458682e-29
## ProductRelated_Duration 3.048385e-30
## BounceRates             1.063215e-30
## ExitRates               6.884248e-31
## PageValues              6.328509e-31
```

Correlation Cirlce

```
fviz_pca_var(pca_residual, col.var = "cos2",
             gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),
             repel = TRUE # Avoid text overlapping
             )
```



From the Correlation Circle and PCA we can see that the most important components are

Administrative #site

Administrative_Duration #Time spent on the admin site

Informational #site

Product Related #site

Product Related Duration #Time spent on the Product related site

Bounce Rates #metric

Exit Rates #metric

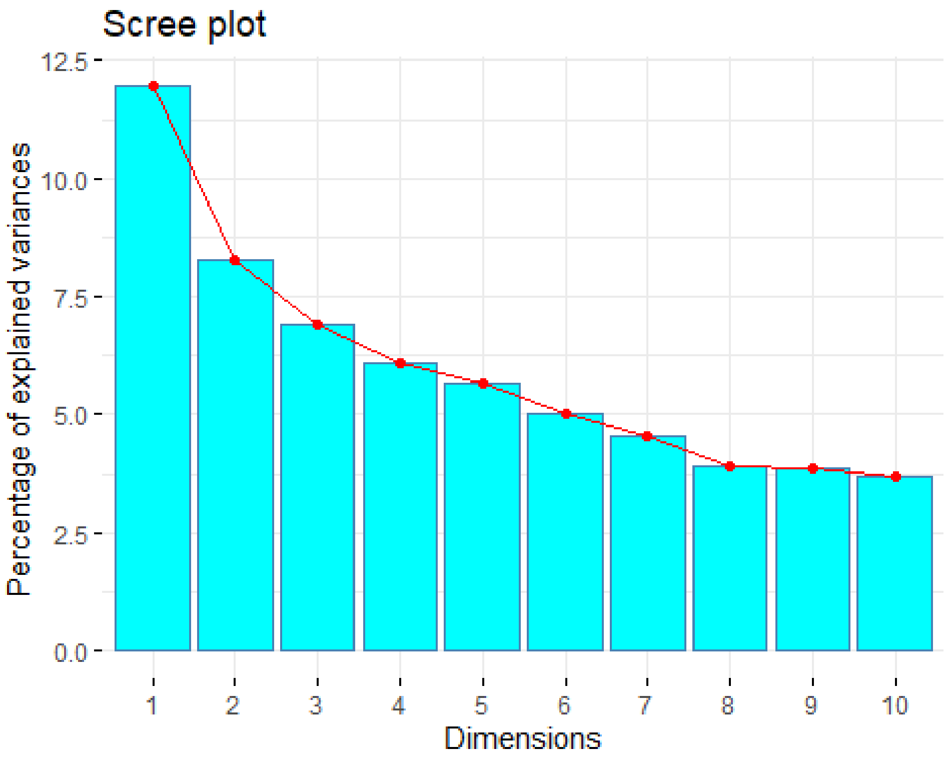
Page Values #metric

SCREE PLOT

A scree plot shows the eigenvalues on the y-axis and the number of factors on the x-axis. It always displays a downward curve.

The point where the slope of the curve is clearly leveling off (the “elbow) indicates the number of factors that should be generated by the analysis.

```
fviz_eig(pca_residual, barfill = 'cyan',linecolor = 'red' )
```



From the plot above, the elbow forms in between the 7th and 8th dimensions. This indicates that the analysis should yield 7 factors.

The first 7 principal components explain about 76% of the variance in the data

Challenging the solution

Using a different number of clusters 9 clusters using the silhouette method

K-MEANS CLUSTERING

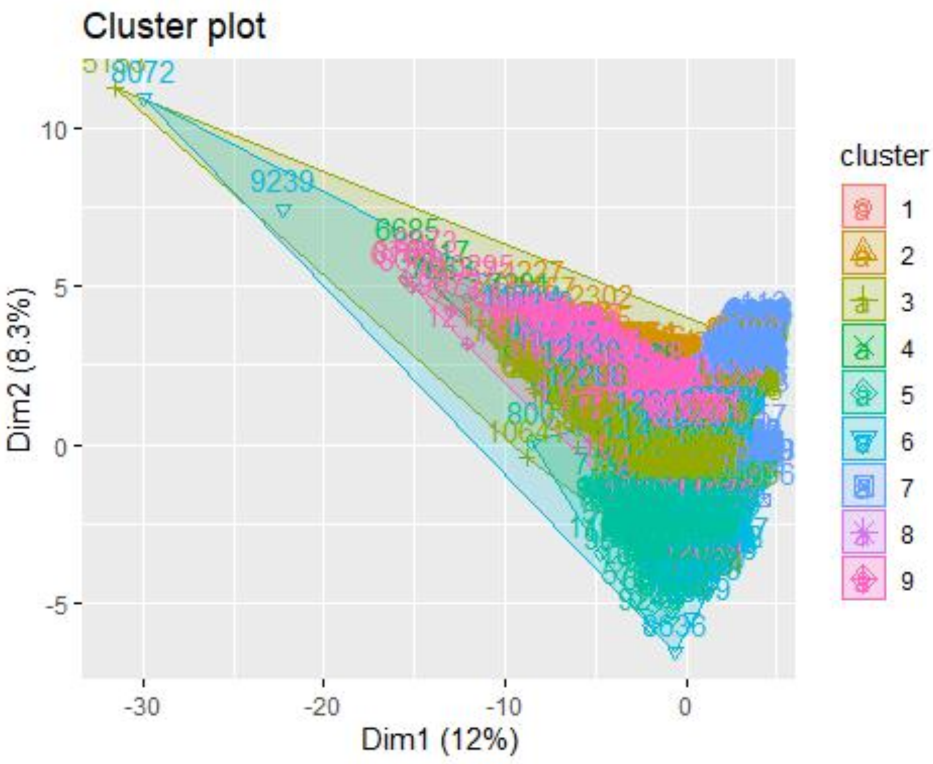
```
outputs <- kmeans(dummy_df2_norm, 9)
```

Results

```
# Previewing the number of records in each cluster

outputs$size

## [1]  273 2221 2377 1167 1676 1025  502 1165 1793
```

Visualising the clusters of the whole dataset

```
options(repr.plot.width = 11, repr.plot.height = 6)
fviz_cluster(outputs, dummy_df2_norm)
```



Summary

Compasiron Between K-MEANS and HIERACHICAL clustering From the Analysis, we can identify that:

1. K-means Cluster Analysis performs much better in identyfing patterns as compared to Hierrachical clustering.
2. Since the dataset is large, visualizing hierrachical clusters is abit cumbersome as compared to K-means clustering.
3. K-means clustering yields better reults using the optimal number of clusters which can be determined by Elbow and Silhouette Methods