IP - Week 13 mod 3

Eltonjohn Oketch

2022-06-05

Kira Plastinina Online Brand Sale.

Defining the Question

The brand's Sales and Marketing team would like to understand the characteristics of customer groups.

Metrics of Success

1. To Perform clustering stating insights drawn from our analysis and visualizations.
2. Provide insights that will help inform the team in formulating the marketing and sales strategies of the brand.
3. Provide comparisons between the approaches learned this week.

Understanding the Context

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.

Recording the experimental design.

The following steps will be followed in conducting this study:

1. Define the question, the metric for success, the context, experimental design taken.
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 Relevance

The dataset for this Independent project can be found here [http://bit.ly/EcommerceCustomersDataset].

> The dataset consists of 10 numerical and 8 categorical attributes. The 'Revenue' attribute can be used as the class label. "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. 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. 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. 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.

Data sourcing

Loading the dataset and libraries.

```
Ecommerce_ds <- read.csv("http://bit.ly/EcommerceCustomersDataset")

head(Ecommerce_ds)
```

```
##   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
```

```
# finding the data summary
summary(Ecommerce_ds)
```

Checking the summary and data type

```
##  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
##  Length:12330       Min.   :1.000     Min.   : 1.000   Min.   :1.000
##  Class :character   1st Qu.:2.000     1st Qu.: 2.000   1st Qu.:1.000
##  Mode  :character   Median :2.000     Median : 2.000   Median :3.000
##                     Mean   :2.124     Mean   : 2.357   Mean   :3.147
##                     3rd Qu.:3.000     3rd Qu.: 2.000   3rd Qu.:4.000
##                     Max.   :8.000     Max.   :13.000   Max.   :9.000
##
##   TrafficType     VisitorType          Weekend          Revenue
##  Min.   : 1.00   Length:12330       Mode :logical   Mode :logical
##  1st Qu.: 2.00   Class :character   FALSE:9462      FALSE:10422
##  Median : 2.00   Mode  :character   TRUE :2868      TRUE :1908
##  Mean   : 4.07
##  3rd Qu.: 4.00
##  Max.   :20.00
##
```

```
# finding the data types of each column
str(Ecommerce_ds)
```

```
## 'data.frame':    12330 obs. of  18 variables:
##  $ Administrative         : int  0 0 0 0 0 0 0 1 0 0 ...
## $ Administrative_Duration: num  0 0 -1 0 0 0 -1 -1 0 0 ...
## $ Informational          : int  0 0 0 0 0 0 0 0 0 0 ...
## $ Informational_Duration : num  0 0 -1 0 0 0 -1 -1 0 0 ...
## $ ProductRelated         : int  1 2 1 2 10 19 1 1 2 3 ...
## $ ProductRelated_Duration: num  0 64 -1 2.67 627.5 ...
## $ BounceRates            : num  0.2 0 0.2 0.05 0.02 ...
## $ ExitRates              : num  0.2 0.1 0.2 0.14 0.05 ...
## $ PageValues             : num  0 0 0 0 0 0 0 0 0 0 ...
## $ SpecialDay             : num  0 0 0 0 0 0 0.4 0 0.8 0.4 ...
## $ Month                  : chr  "Feb" "Feb" "Feb" "Feb" ...
## $ OperatingSystems       : int  1 2 4 3 3 2 2 1 2 2 ...
## $ Browser                : int  1 2 1 2 3 2 4 2 2 4 ...
## $ Region                 : int  1 1 9 2 1 1 3 1 2 1 ...
## $ TrafficType            : int  1 2 3 4 4 3 3 5 3 2 ...
## $ VisitorType            : chr  "Returning_Visitor" "Returning_Visitor" "Returning_Visitor" "Return
## $ Weekend                : logi  FALSE FALSE FALSE FALSE TRUE FALSE ...
## $ Revenue                : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
```

Data Cleaning

Finding the missing data

```
# Lets Identify missing data in your dataset
#
#
colSums(is.na(Ecommerce_ds))
```

```
##          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
```

Dropping the null values

```
# Viewing the null values

colnames(Ecommerce_ds)[apply(Ecommerce_ds, 2, anyNA)]
```

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

```
# Droping the null values
library(dplyr)
```

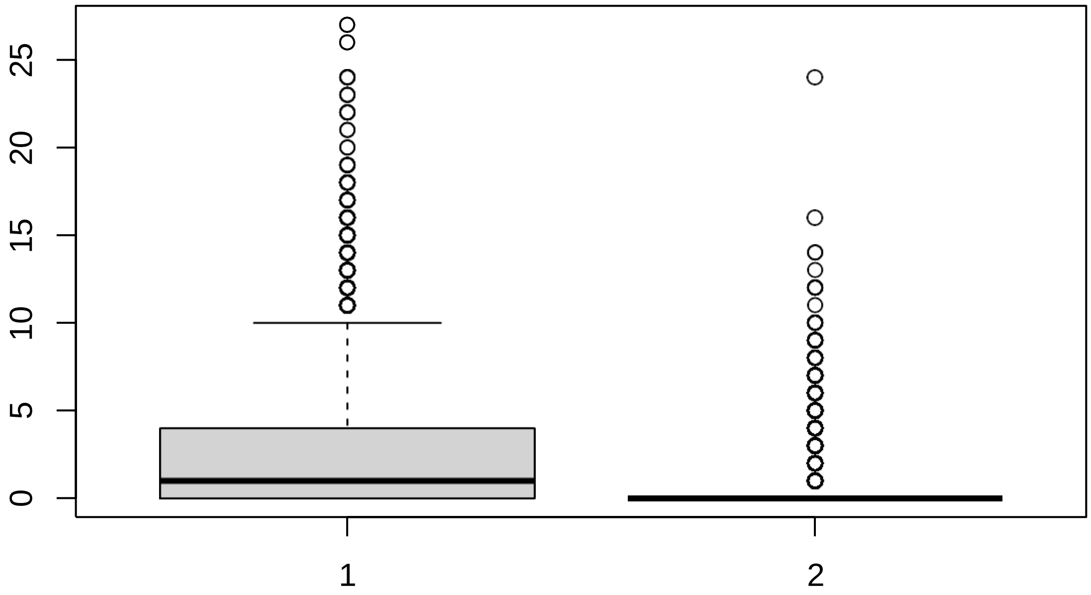
```
##
## Attaching package: 'dplyr'

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

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

```
Ecommerce_ds<- na.omit(Ecommerce_ds)
head(Ecommerce_ds)
```

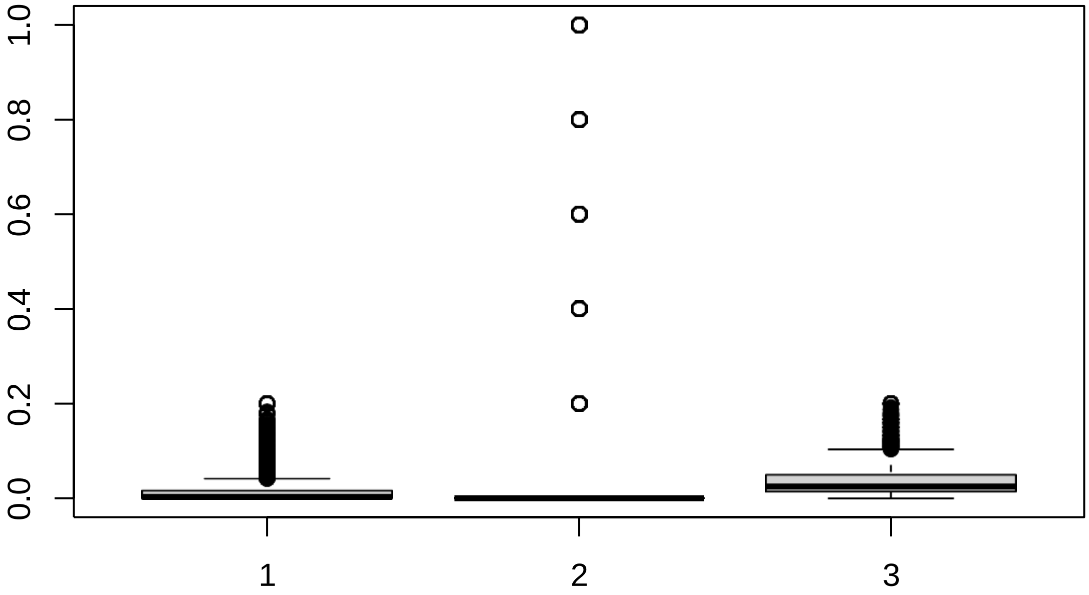
```
##   Administrative Administrative_Duration Informational Informational_Duration
## 1              0                       0             0                      0
## 2              0                       0             0                      0
## 3              0                      -1             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
```

Checking for the outliers

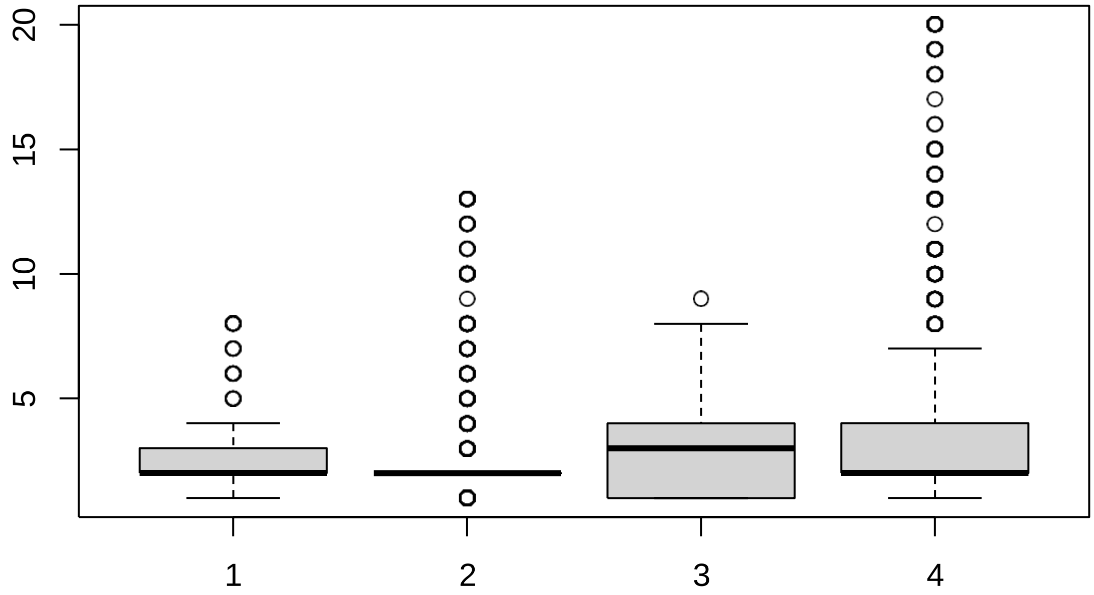
```
# we shall check for the outliers in the dataset using the boxplot
#Checking for outliers in administrative and information columns
boxplot(Ecommerce_ds$Administrative, Ecommerce_ds$Informational)
```



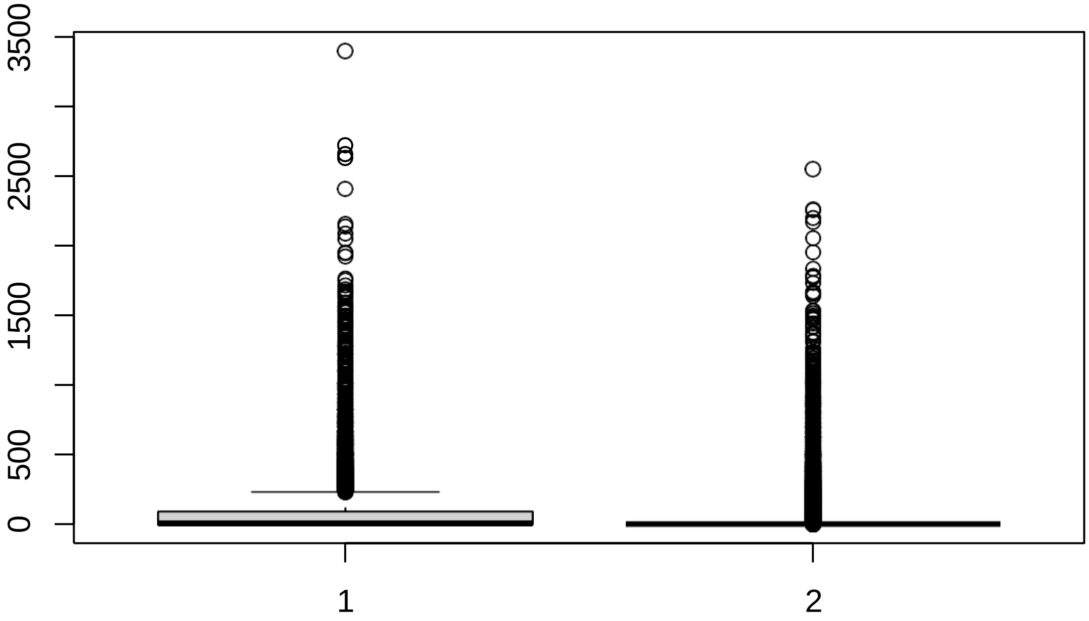
```
# Checking for outliers in bouncerates, special day and exitrates
boxplot(Ecommerce_ds$BounceRates,  Ecommerce_ds$SpecialDay,  Ecommerce_ds$ExitRates)
```



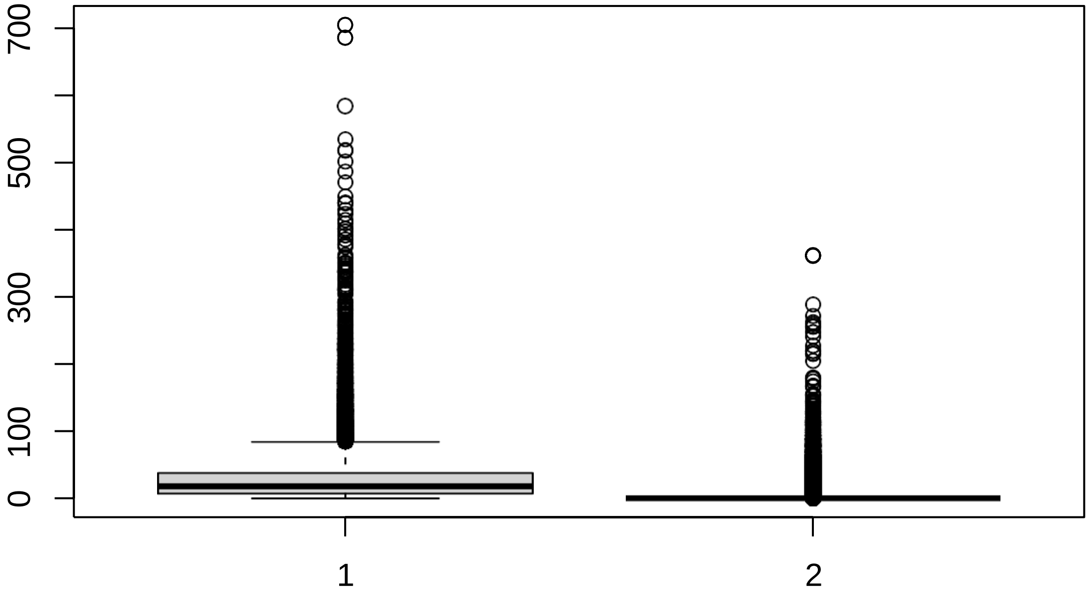
```
# checking for operating system browser, traffic type and region
boxplot(Ecommerce_ds$OperatingSystems, Ecommerce_ds$Browser, Ecommerce_ds$Region, Ecommerce_ds$TrafficT
```



```
# checking for outliers in administrative duration and information duration
boxplot(Ecommerce_ds$Administrative_Duration, Ecommerce_ds$Informational_Duration)
```



```
# checking for outliers in product related and page values
boxplot(Ecommerce_ds$ProductRelated,    Ecommerce_ds$PageValues)
```



Checking for duplicates

```
# checking for duplicated data
duplicated_rows <- Ecommerce_ds[duplicated(Ecommerce_ds),]
# printing the duplicated_rows
head(duplicated_rows)
```

```
##       Administrative  Administrative_Duration  Informational  Informational_Duration
## 159                0                        0              0                       0
## 179                0                        0              0                       0
## 419                0                        0              0                       0
## 457                0                        0              0                       0
## 484                0                        0              0                       0
## 513                0                        0              0                       0
##       ProductRelated  ProductRelated_Duration  BounceRates  ExitRates  PageValues
## 159                1                        0          0.2        0.2           0
## 179                1                        0          0.2        0.2           0
## 419                1                        0          0.2        0.2           0
## 457                1                        0          0.2        0.2           0
## 484                1                        0          0.2        0.2           0
## 513                1                        0          0.2        0.2           0
##      SpecialDay  Month  OperatingSystems  Browser  Region  TrafficType
## 159           0    Feb                 1        1       1            3
## 179           0    Feb                 3        2       3            3
## 419           0    Mar                 1        1       1            1
## 457           0    Mar                 2        2       4            1
## 484          0   Mar                3       2      3           1
## 513          0   Mar                2       2      1           1
##           VisitorType Weekend Revenue
## 159 Returning_Visitor   FALSE   FALSE
## 179 Returning_Visitor   FALSE   FALSE
## 419 Returning_Visitor    TRUE   FALSE
## 457 Returning_Visitor   FALSE   FALSE
## 484 Returning_Visitor   FALSE   FALSE
## 513 Returning_Visitor   FALSE   FALSE
```

```
# since some values are common we drop them
head(drop(duplicated_rows))
```

Dropping duplicated rows

```
##     Administrative Administrative_Duration Informational Informational_Duration
## 159              0                       0             0                      0
## 179              0                       0             0                      0
## 419              0                       0             0                      0
## 457              0                       0             0                      0
## 484              0                       0             0                      0
## 513              0                       0             0                      0
##     ProductRelated ProductRelated_Duration BounceRates ExitRates PageValues
## 159              1                       0         0.2       0.2          0
## 179              1                       0         0.2       0.2          0
## 419              1                       0         0.2       0.2          0
## 457              1                       0         0.2       0.2          0
## 484              1                       0         0.2       0.2          0
## 513              1                       0         0.2       0.2          0
##     SpecialDay Month OperatingSystems Browser Region TrafficType
## 159          0   Feb                1       1      1           3
## 179          0   Feb                3       2      3           3
## 419          0   Mar                1       1      1           1
## 457          0   Mar                2       2      4           1
## 484          0   Mar                3       2      3           1
## 513          0   Mar                2       2      1           1
##           VisitorType Weekend Revenue
## 159 Returning_Visitor   FALSE   FALSE
## 179 Returning_Visitor   FALSE   FALSE
## 419 Returning_Visitor    TRUE   FALSE
## 457 Returning_Visitor   FALSE   FALSE
## 484 Returning_Visitor   FALSE   FALSE
## 513 Returning_Visitor   FALSE   FALSE
```

Exploratory Data Analysis

Univariate Data Analysis

Checking for the mean of the dataset

```
# Checking for mean of admistrative
Ecommerce_ds.Administrative.mean <- mean(Ecommerce_ds$Administrative)
# Printing out the admistrative mean
# ---
Ecommerce_ds.Administrative.mean
```

```
## [1] 2.317798
```

```
#----
# Checking for mean of information
Ecommerce_ds.Informational.mean <- mean(Ecommerce_ds$Informational)
# Printing out
# ---
Ecommerce_ds.Informational.mean
```

```
## [1] 0.5039786
```

```
#----
# Checking for the mean of Bounce Rate
# ---
Ecommerce_ds.BounceRates.mean <- mean(Ecommerce_ds$BounceRates)
# Printing out
# ---
Ecommerce_ds.BounceRates.mean
```

```
## [1] 0.02215246
```

```
# Checking for mean of Special day
Ecommerce_ds.SpecialDay.mean <- mean(Ecommerce_ds$SpecialDay)
# Printing out
# ---
Ecommerce_ds.SpecialDay.mean
```

```
## [1] 0.06149724
```

```
#
# Checking for mean of information
Ecommerce_ds.Informational.mean <- mean(Ecommerce_ds$Informational)
# Printing out
# ---
Ecommerce_ds.Informational.mean
```

```
## [1] 0.5039786
```

```
#----
# Checking for the mean of Exit Rates
# ---
Ecommerce_ds.ExitRates.mean <- mean(Ecommerce_ds$ExitRates)
# Printing out
# ---
Ecommerce_ds.ExitRates.mean
```

```
## [1] 0.04300254
```

```
# Checking for mean of Operating System
Ecommerce_ds.OperatingSystem.mean <- mean(Ecommerce_ds$OperatingSystem)
# Printing out
# ---
Ecommerce_ds.OperatingSystem.mean
```

```
## [1] 2.124147
```

```
# Checking for mean of Browser
Ecommerce_ds.Browser.mean <- mean(Ecommerce_ds$Browser)
# Printing out
# ---
Ecommerce_ds.Browser.mean
```

```
## [1] 2.357584
```

```
# Checking for mean of Region
Ecommerce_ds.Region.mean <- mean(Ecommerce_ds$Region)
# Printing out
# ---
Ecommerce_ds.Region.mean
```

```
## [1] 3.148019
```

```
# Checking for mean of Operating System
Ecommerce_ds.TrafficType.mean <- mean(Ecommerce_ds$TrafficType)
# Printing out
# ---
Ecommerce_ds.TrafficType.mean
```

```
## [1] 4.070477
```

```
# Checking for mean of Operating System
Ecommerce_ds.Administrative_Duration.mean <- mean(Ecommerce_ds$Administrative_Duration)
# Printing out
# ---
Ecommerce_ds.Administrative_Duration.mean
```

```
## [1] 80.90618
```

```
# Checking for mean of Operating System
Ecommerce_ds.Informational_Duration.mean <- mean(Ecommerce_ds$Informational_Duration)
# Printing out
# ---
Ecommerce_ds.Informational_Duration.mean
```

```
## [1] 34.50639
```

```
# Checking for mean of Operating System
Ecommerce_ds.ProductRelated.mean <- mean(Ecommerce_ds$ProductRelated)
# Printing out
# ---
Ecommerce_ds.ProductRelated.mean
```

```
## [1] 31.76388
```

```
# Checking for mean of Operating System
Ecommerce_ds.PageValues.mean <- mean(Ecommerce_ds$PageValues)
# Printing out
# ---
Ecommerce_ds.PageValues.mean
```

```
## [1] 5.895952
```

Checking for the median of the dataset

```
# Checking for median of admistrative
Ecommerce_ds.Administrative.median <- median(Ecommerce_ds$Administrative)
# Printing out
# ---
Ecommerce_ds.Administrative.median
```

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

```
#----
# Checking for median of information
Ecommerce_ds.Informational.median <- median(Ecommerce_ds$Informational)
# Printing out
# ---
Ecommerce_ds.Informational.median
```

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

```
#----
# Checking for the median of Bounce Rate
# ---
Ecommerce_ds.BounceRates.median  <-  median(Ecommerce_ds$BounceRates)
# Printing out
# ---
Ecommerce_ds.BounceRates.median
```

```
## [1] 0.003119412
```

```
# Checking for median of Special day
Ecommerce_ds.SpecialDay.median <- median(Ecommerce_ds$SpecialDay)
# Printing out
# ---
Ecommerce_ds.SpecialDay.median
```

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

```
#
# Checking for median of information
Ecommerce_ds.Informational.median <- median(Ecommerce_ds$Informational)
# Printing out
# ---
Ecommerce_ds.Informational.median
```

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

```
#----
# Checking for the mean of Exit Rates
# ---
Ecommerce_ds.ExitRates.median <- median(Ecommerce_ds$ExitRates)
# Printing out
# ---
Ecommerce_ds.ExitRates.median
```

```
## [1] 0.02512449
```

```
# Checking for median of Operating System
Ecommerce_ds.OperatingSystem.median <- median(Ecommerce_ds$OperatingSystem)
# Printing out
# ---
Ecommerce_ds.OperatingSystem.median
```

```
## [1] 2
```

```
#  Checking  for  median  of  Browser
Ecommerce_ds.Browser.median <- median(Ecommerce_ds$Browser)
# Printing out
# ---
Ecommerce_ds.Browser.median
```

```
## [1] 2
```

```
#  Checking  for  median  of  Region
Ecommerce_ds.Region.median <- median(Ecommerce_ds$Region)
# Printing out
# ---
Ecommerce_ds.Region.median
```

```
## [1] 3
```

```
# Checking for median of Operating System
Ecommerce_ds.TrafficType.median  <-  median(Ecommerce_ds$TrafficType)
# Printing out
# ---
Ecommerce_ds.TrafficType.median
```

```
## [1] 2
```

```
# Checking for median of Operating System
Ecommerce_ds.Administrative_Duration.median <- median(Ecommerce_ds$Administrative_Duration)
# Printing out
# ---
Ecommerce_ds.Administrative_Duration.median
```

```
## [1] 8
```

```
# Checking for median of Operating System
Ecommerce_ds.Informational_Duration.median <- median(Ecommerce_ds$Informational_Duration)
# Printing out
# ---
Ecommerce_ds.Informational_Duration.median
```

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

```
# Checking for median of Operating System
Ecommerce_ds.ProductRelated.median <- median(Ecommerce_ds$ProductRelated)
# Printing out
# ---
Ecommerce_ds.ProductRelated.median
```

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

```
# Checking for median of Operating System
Ecommerce_ds.PageValues.median <- median(Ecommerce_ds$PageValues)
# Printing out
# ---
Ecommerce_ds.PageValues.median
```

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

Checking for the minimum of the dataset

```
# Checking for minimum of admistrative
Ecommerce_ds.Administrative.min <- min(Ecommerce_ds$Administrative)
# Printing out
# ---
Ecommerce_ds.Administrative.min
```

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

```
#----
# Checking for minimum of information
Ecommerce_ds.Informational.min <- min(Ecommerce_ds$Informational)
# Printing out
# ---
Ecommerce_ds.Informational.min
```

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

```
#----
# Checking for the minimum of Bounce Rate
# ---
Ecommerce_ds.BounceRates.min <- min(Ecommerce_ds$BounceRates)
# Printing out
# ---
Ecommerce_ds.BounceRates.min
```

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

```
# Checking for minimum of Special day
Ecommerce_ds.SpecialDay.min <- min(Ecommerce_ds$SpecialDay)
# Printing out
# ---
Ecommerce_ds.SpecialDay.min
```

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

```
#
# Checking for minimum of information
Ecommerce_ds.Informational.min <- min(Ecommerce_ds$Informational)
# Printing out
# ---
Ecommerce_ds.Informational.min
```

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

```
#----
# Checking for the minimum of Exit Rates
# ---
Ecommerce_ds.ExitRates.min <- min(Ecommerce_ds$ExitRates)
# Printing out
# ---
Ecommerce_ds.ExitRates.min
```

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

```
# Checking for minimum of Operating System
Ecommerce_ds.OperatingSystem.min <- min(Ecommerce_ds$OperatingSystem)
# Printing out
# ---
Ecommerce_ds.OperatingSystem.min
```

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

```
# Checking for minimum of Browser
Ecommerce_ds.Browser.min <- min(Ecommerce_ds$Browser)
# Printing out
# ---
Ecommerce_ds.Browser.min
```

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

```
# Checking for minimum of Region
Ecommerce_ds.Region.min <- min(Ecommerce_ds$Region)
# Printing out
# ---
Ecommerce_ds.Region.min
```

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

```
# Checking for minimum of Operating System
Ecommerce_ds.TrafficType.min <- min(Ecommerce_ds$TrafficType)
# Printing out
# ---
Ecommerce_ds.TrafficType.min
```

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

```
# Checking for minimum of Operating System
Ecommerce_ds.Administrative_Duration.min <- min(Ecommerce_ds$Administrative_Duration)
# Printing out
# ---
Ecommerce_ds.Administrative_Duration.min
```

```
## [1] -1
```

```
# Checking for minimum of Operating System
Ecommerce_ds.Informational_Duration.min <- min(Ecommerce_ds$Informational_Duration)
# Printing out
# ---
Ecommerce_ds.Informational_Duration.min
```

```
## [1] -1
```

```
# Checking for median of Operating System
Ecommerce_ds.ProductRelated.min <- min(Ecommerce_ds$ProductRelated)
# Printing out
# ---
Ecommerce_ds.ProductRelated.min
```

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

```
# Checking for median of Operating System
Ecommerce_ds.PageValues.min <- min(Ecommerce_ds$PageValues)
# Printing out
# ---
Ecommerce_ds.PageValues.min
```

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

Checking for the maximum of the dataset

```
# Checking for maximum of admistrative
Ecommerce_ds.Administrative.max <- max(Ecommerce_ds$Administrative)
# Printing out
# ---
Ecommerce_ds.Administrative.max
```

```
## [1] 27
```

```
#----
# Checking for maximum of information
Ecommerce_ds.Informational.max <- max(Ecommerce_ds$Informational)
# Printing out
# ---
Ecommerce_ds.Informational.max
```

```
## [1] 24
```

```
#----
# Checking for the maximum of Bounce Rate
# ---
Ecommerce_ds.BounceRates.max <- max(Ecommerce_ds$BounceRates)
# Printing out
# ---
Ecommerce_ds.BounceRates.max
```

```
## [1] 0.2
```

```
# Checking for minimum of Special day
Ecommerce_ds.SpecialDay.max <- max(Ecommerce_ds$SpecialDay)
# Printing out
# ---
Ecommerce_ds.SpecialDay.max
```

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

```
#
# Checking for minimum of information
Ecommerce_ds.Informational.max <- max(Ecommerce_ds$Informational)
# Printing out
# ---
Ecommerce_ds.Informational.max
```

```
## [1] 24
```

```
#----
# Checking for the minimum of Exit Rates
# ---
Ecommerce_ds.ExitRates.max <- max(Ecommerce_ds$ExitRates)
# Printing out
# ---
Ecommerce_ds.ExitRates.max
```

```
## [1] 0.2
```

```
# Checking for minimum of Operating System
Ecommerce_ds.OperatingSystem.max <- max(Ecommerce_ds$OperatingSystem)
# Printing out
# ---
Ecommerce_ds.OperatingSystem.max
```

```
## [1] 8
```

```
# Checking for minimum of Browser
Ecommerce_ds.Browser.max <- max(Ecommerce_ds$Browser)
# Printing out
# ---
Ecommerce_ds.Browser.max
```

```
## [1] 13
```

```
# Checking for minimum of Region
Ecommerce_ds.Region.max <- max(Ecommerce_ds$Region)
# Printing out
# ---
Ecommerce_ds.Region.max
```

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

```
# Checking for minimum of Operating System
Ecommerce_ds.TrafficType.max <- max(Ecommerce_ds$TrafficType)
# Printing out
# ---
Ecommerce_ds.TrafficType.max
```

```
## [1] 20
```

```
# Checking for minimum of Operating System
Ecommerce_ds.Administrative_Duration.max <- max(Ecommerce_ds$Administrative_Duration)
# Printing out
# ---
Ecommerce_ds.Administrative_Duration.max
```

```
## [1] 3398.75
```

```
# Checking for minimum of Operating System
Ecommerce_ds.Informational_Duration.max <- max(Ecommerce_ds$Informational_Duration)
# Printing out
# ---
Ecommerce_ds.Informational_Duration.max
```

```
## [1] 2549.375
```

```
# Checking for minimu of Operating System
Ecommerce_ds.ProductRelated.max <- max(Ecommerce_ds$ProductRelated)
# Printing out
# ---
Ecommerce_ds.ProductRelated.max
```

```
## [1] 705
```

```
# Checking for minimum of Operating System
Ecommerce_ds.PageValues.max <- max(Ecommerce_ds$PageValues)
# Printing out
# ---
Ecommerce_ds.PageValues.max
```

```
## [1] 361.7637
```

Checking for the quantile of the dataset

```
# Checking for quantile of admistrative
Ecommerce_ds.Administrative.quantile  <-  quantile(Ecommerce_ds$Administrative)
# Printing out
# ---
Ecommerce_ds.Administrative.quantile
```

```
##   0%  25%  50%  75% 100% 
##    0    0    1    4   27
```

```
#----
# Checking for quantile of information
Ecommerce_ds.Informational.quantile <- quantile(Ecommerce_ds$Informational)
# Printing out
# ---
Ecommerce_ds.Informational.quantile
```

```
##   0%  25%  50%  75% 100% 
##    0    0    0    0   24
```

```
#----
# Checking for the quantile of Bounce Rate
# ---
Ecommerce_ds.BounceRates.quantile <- quantile(Ecommerce_ds$BounceRates)
# Printing out
# ---
Ecommerce_ds.BounceRates.quantile
```

```
##          0%         25%         50%         75%        100% 
## 0.000000000 0.000000000 0.003119412 0.016683674 0.200000000
```

```
# Checking for quantile of Special day
Ecommerce_ds.SpecialDay.quantile <- quantile(Ecommerce_ds$SpecialDay)
# Printing out
# ---
Ecommerce_ds.SpecialDay.quantile
```

```
##   0%  25%  50%  75% 100% 
##    0    0    0    0    1
```

```
#
# Checking for quantile of information
Ecommerce_ds.Informational.quantile <- quantile(Ecommerce_ds$Informational)
# Printing out
# ---
Ecommerce_ds.Informational.quantile
```

```
##   0%  25%  50%  75% 100% 
##    0    0    0    0   24
```

```
#----
# Checking for the quantile of Exit Rates
# ---
Ecommerce_ds.ExitRates.quantile <- quantile(Ecommerce_ds$ExitRates)
# Printing out
# ---
Ecommerce_ds.ExitRates.quantile
```

```
##         0%        25%        50%        75%       100% 
## 0.00000000 0.01428571 0.02512449 0.05000000 0.20000000
```

```
# Checking for minimum of Operating System
Ecommerce_ds.OperatingSystem.quantile <- quantile(Ecommerce_ds$OperatingSystem)
# Printing out
# ---
Ecommerce_ds.OperatingSystem.quantile
```

```
##   0%  25%  50%  75% 100% 
##    1    2    2    3    8
```

```
# Checking for minimum of Browser
Ecommerce_ds.Browser.quantile <- quantile(Ecommerce_ds$Browser)
# Printing out
# ---
Ecommerce_ds.Browser.quantile
```

```
##   0%  25%  50%  75% 100% 
##    1    2    2    2   13
```

```r
# Checking for minimum of Region
Ecommerce_ds.Region.quantile <- quantile(Ecommerce_ds$Region)
# Printing out
# ---
Ecommerce_ds.Region.quantile
```

```
##   0%  25%  50%  75% 100%
##    1    1    3    4    9
```

```r
# Checking for minimum of Operating System
Ecommerce_ds.TrafficType.quantile <- quantile(Ecommerce_ds$TrafficType)
# Printing out
# ---
Ecommerce_ds.TrafficType.quantile
```

```
##   0%  25%  50%  75% 100%
##    1    2    2    4   20
```

```r
# Checking for minimum of Operating System
Ecommerce_ds.Administrative_Duration.quantile <- quantile(Ecommerce_ds$Administrative_Duration)
# Printing out
# ---
Ecommerce_ds.Administrative_Duration.quantile
```

```
##      0%     25%     50%     75%    100%
##   -1.00    0.00    8.00   93.50 3398.75
```

```r
# Checking for minimum of Operating System
Ecommerce_ds.Informational_Duration.quantile <- quantile(Ecommerce_ds$Informational_Duration)
# Printing out
# ---
Ecommerce_ds.Informational_Duration.quantile
```

```
##       0%      25%      50%      75%     100%
##   -1.000    0.000    0.000    0.000 2549.375
```

```r
# Checking for median of Operating System
Ecommerce_ds.ProductRelated.quantile <- quantile(Ecommerce_ds$ProductRelated)
# Printing out
# ---
Ecommerce_ds.ProductRelated.quantile
```

```
##   0%  25%  50%  75% 100%
##    0    7   18   38  705
```

```r
# Checking for median of Operating System
Ecommerce_ds.PageValues.quantile <- quantile(Ecommerce_ds$PageValues)
# Printing out
# ---
Ecommerce_ds.PageValues.quantile
```

```
##        0%       25%       50%       75%      100%
##    0.0000    0.0000    0.0000    0.0000  361.7637
```

Checking for the standard deviation of the dataset

```
# Checking for sd of admistrative
Ecommerce_ds.Administrative.sd <- sd(Ecommerce_ds$Administrative)
# Printing out
# ---
Ecommerce_ds.Administrative.sd
```

```
## [1] 3.322754
```

```
#----
# Checking for sd of information
Ecommerce_ds.Informational.sd <- sd(Ecommerce_ds$Informational)
# Printing out
# ---
Ecommerce_ds.Informational.sd
```

```
## [1] 1.270701
```

```
#----
# Checking for the sd of Bounce Rate
# ---
Ecommerce_ds.BounceRates.sd <- sd(Ecommerce_ds$BounceRates)
# Printing out
# ---
Ecommerce_ds.BounceRates.sd
```

```
## [1] 0.04842713
```

```
# Checking for sd of Special day
Ecommerce_ds.SpecialDay.sd <- sd(Ecommerce_ds$SpecialDay)
# Printing out
# ---
Ecommerce_ds.SpecialDay.sd
```

```
## [1] 0.1990195
```

```
#
# Checking for sd of information
Ecommerce_ds.Informational.sd <- sd(Ecommerce_ds$Informational)
# Printing out
# ---
Ecommerce_ds.Informational.sd
```

```
## [1] 1.270701
```

```
#----
# Checking for the sd of Exit Rates
# ---
Ecommerce_ds.ExitRates.sd <- sd(Ecommerce_ds$ExitRates)
# Printing out
# ---
Ecommerce_ds.ExitRates.sd
```

```
## [1] 0.0485273
```

```
# Checking for sd of Operating System
Ecommerce_ds.OperatingSystem.sd <- sd(Ecommerce_ds$OperatingSystem)
# Printing out
# ---
Ecommerce_ds.OperatingSystem.sd
```

```
## [1] 0.9115659
```

```
# Checking for sd of Browser
Ecommerce_ds.Browser.sd <- sd(Ecommerce_ds$Browser)
# Printing out
# ---
Ecommerce_ds.Browser.sd
```

```
## [1] 1.718028
```

```
# Checking for sd of Region
Ecommerce_ds.Region.sd <- sd(Ecommerce_ds$Region)
# Printing out
# ---
Ecommerce_ds.Region.sd
```

```
## [1] 2.402211
```

```
# Checking for sd of Operating System
Ecommerce_ds.TrafficType.sd <- sd(Ecommerce_ds$TrafficType)
# Printing out
# ---
Ecommerce_ds.TrafficType.sd
```

```
## [1] 4.024598
```

```
# Checking for sd of Operating System
Ecommerce_ds.Administrative_Duration.sd <- sd(Ecommerce_ds$Administrative_Duration)
# Printing out
# ---
Ecommerce_ds.Administrative_Duration.sd
```

```
## [1] 176.8604
```

```
# Checking for sd of Operating System
Ecommerce_ds.Informational_Duration.sd <- sd(Ecommerce_ds$Informational_Duration)
# Printing out
# ---
Ecommerce_ds.Informational_Duration.sd
```

```
## [1] 140.8255
```

```
# Checking for sd of Operating System
Ecommerce_ds.ProductRelated.sd <- sd(Ecommerce_ds$ProductRelated)
# Printing out
# ---
Ecommerce_ds.ProductRelated.sd
```

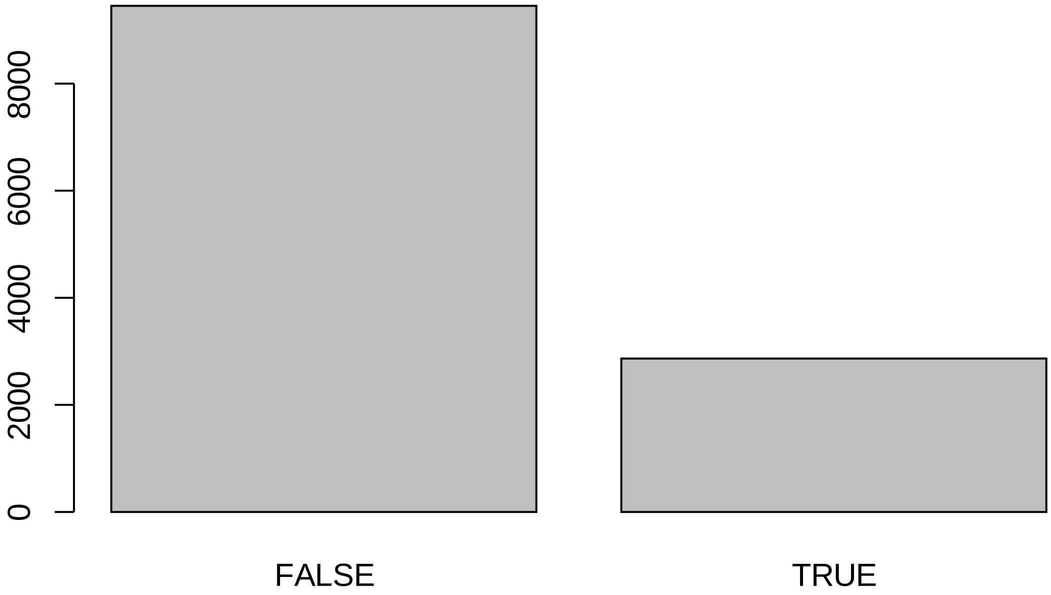
```
## [1] 44.49034
```

```
# Checking for sd of Operating System
Ecommerce_ds.PageValues.sd <- sd(Ecommerce_ds$PageValues)
# Printing out
# ---
Ecommerce_ds.PageValues.sd
```

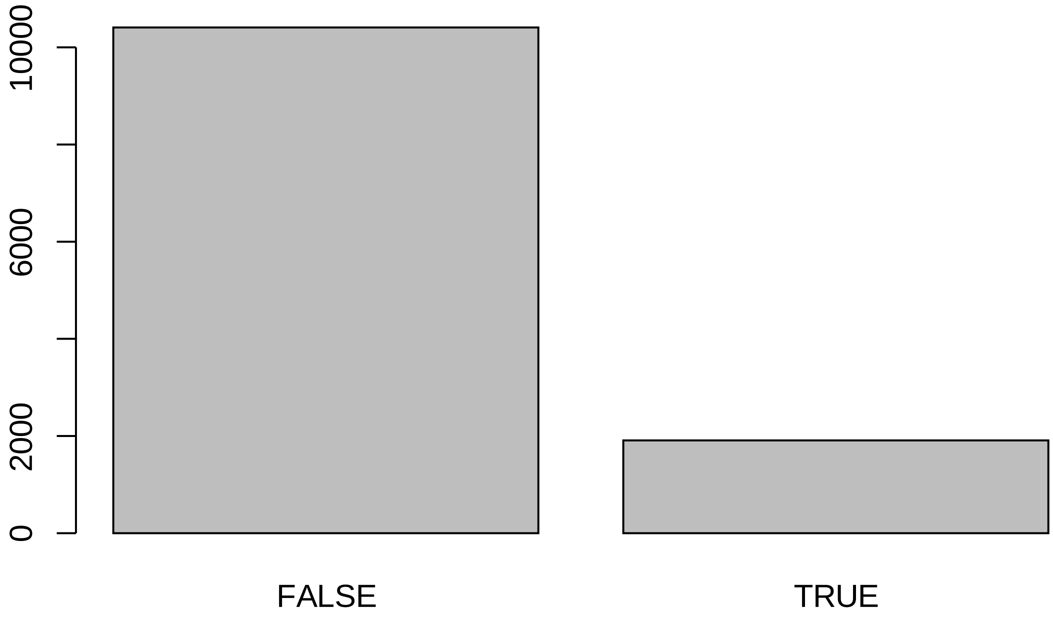
```
## [1] 18.57793
```

Plotting Bar graph

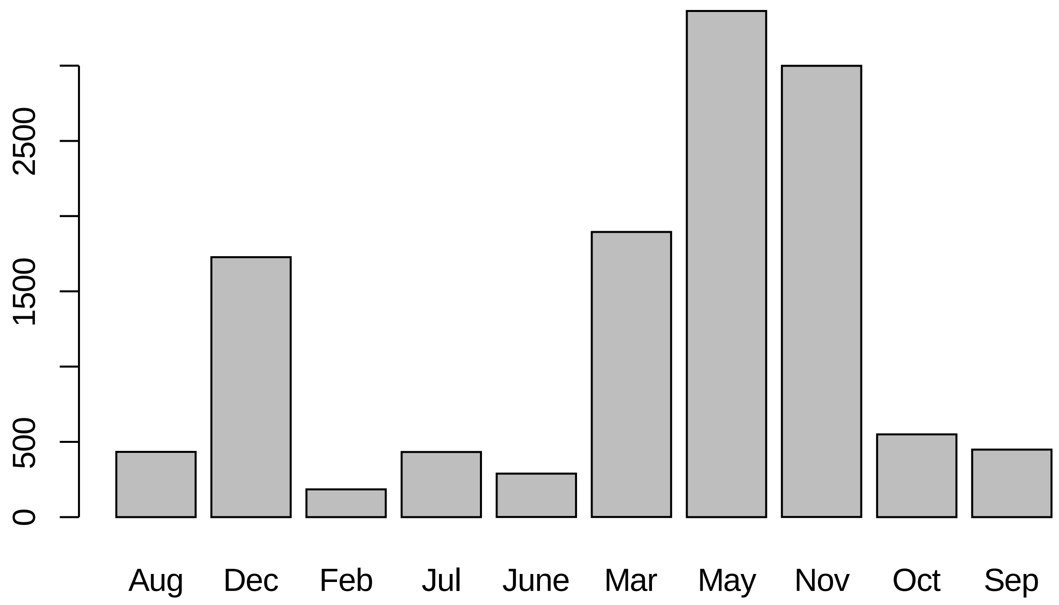
```
# plotting bar graph on weekend
Ecommerce <- Ecommerce_ds$Weekend
# ---
# Applying table
Ecommerce_frequency <- table(Ecommerce)
# Then applying the barplot function to produce its bar graph
# ---
#
barplot(Ecommerce_frequency)
```



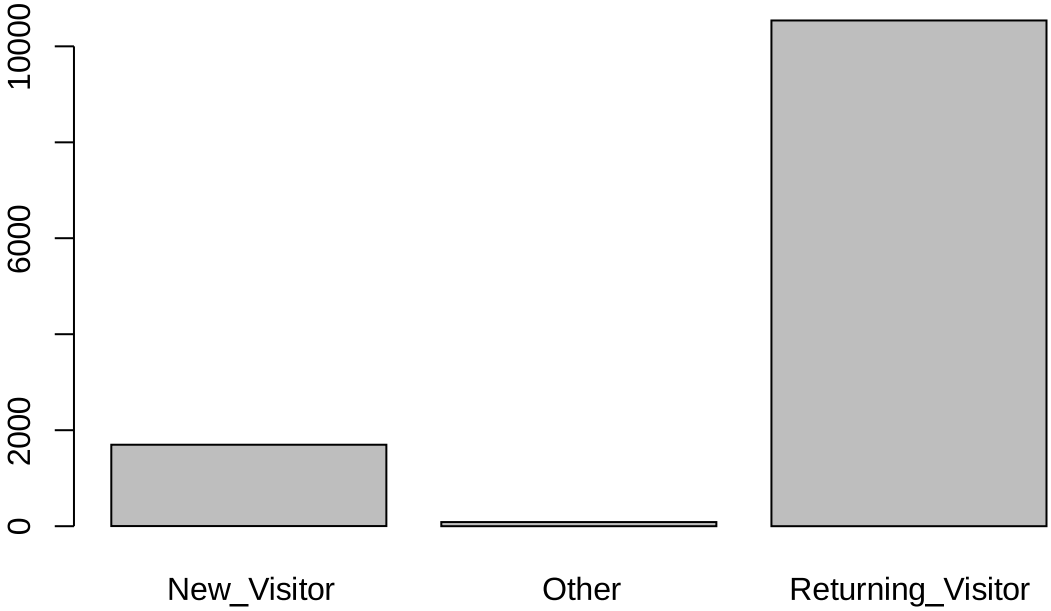
```
#----
# ploting bar on Revenue
Ecommerce <- Ecommerce_ds$Revenue
# Applying table
Ecommerce_frequency <- table(Ecommerce)
# Then applying the barplot function to produce its bar graph
# ---
#
barplot(Ecommerce_frequency)
```



```
# ploting bar on Months
Ecommerce <- Ecommerce_ds$Month
# Applying table
Ecommerce_frequency <- table(Ecommerce)
# Then applying the barplot function to produce its bar graph
# ---
#
barplot(Ecommerce_frequency)
```

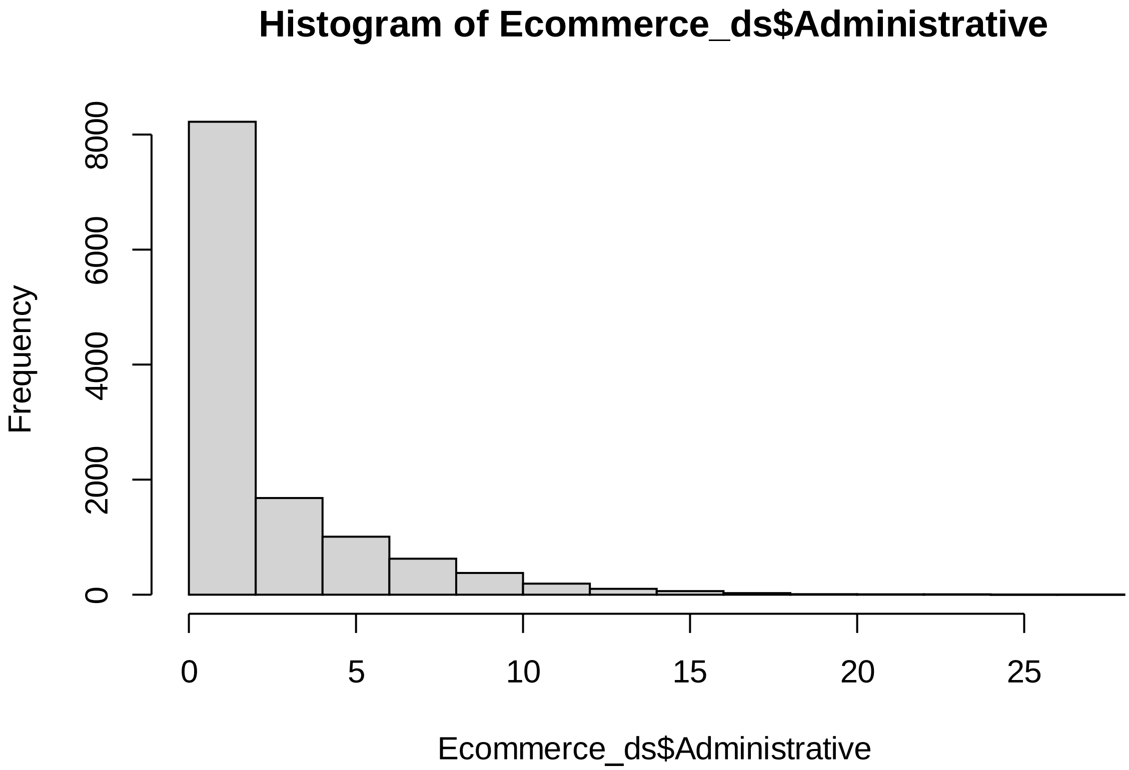


```
# ploting bar on Revenue
Ecommerce <- Ecommerce_ds$VisitorType
# Applying table
Ecommerce_frequency <- table(Ecommerce)
# Then applying the barplot function to produce its bar graph
# ---
#
barplot(Ecommerce_frequency)
```

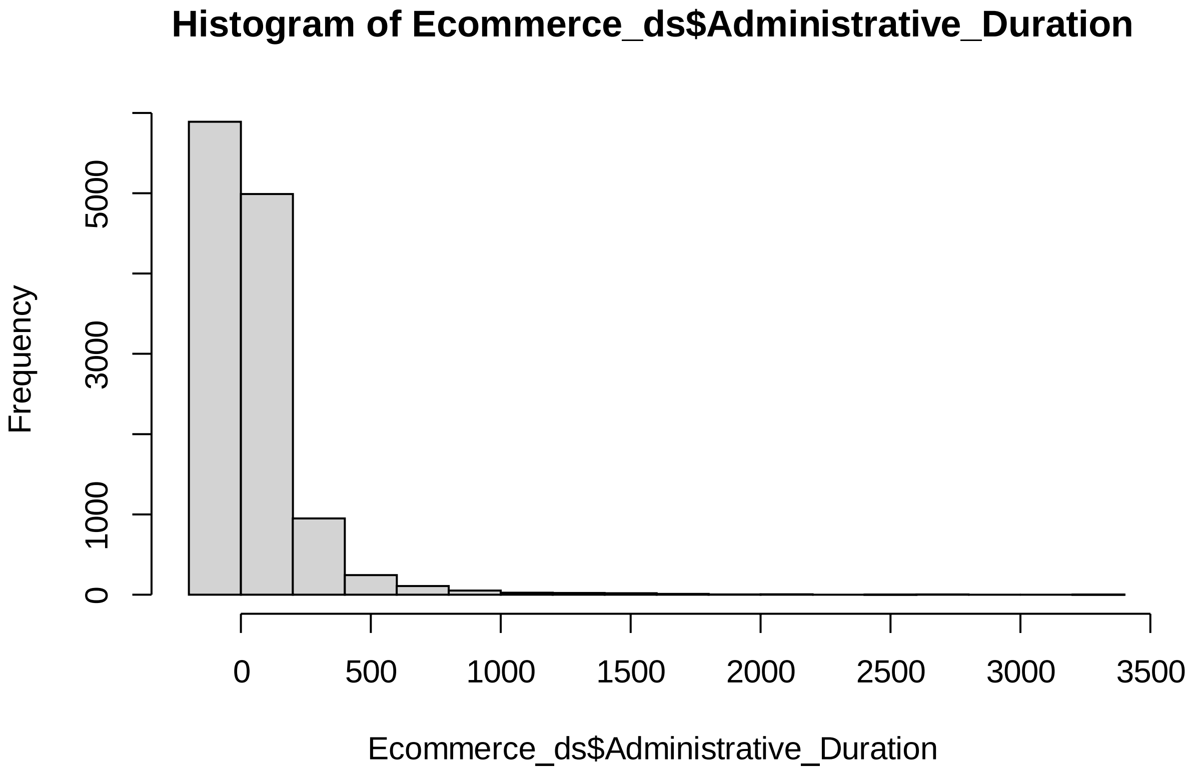


Plotting histogram

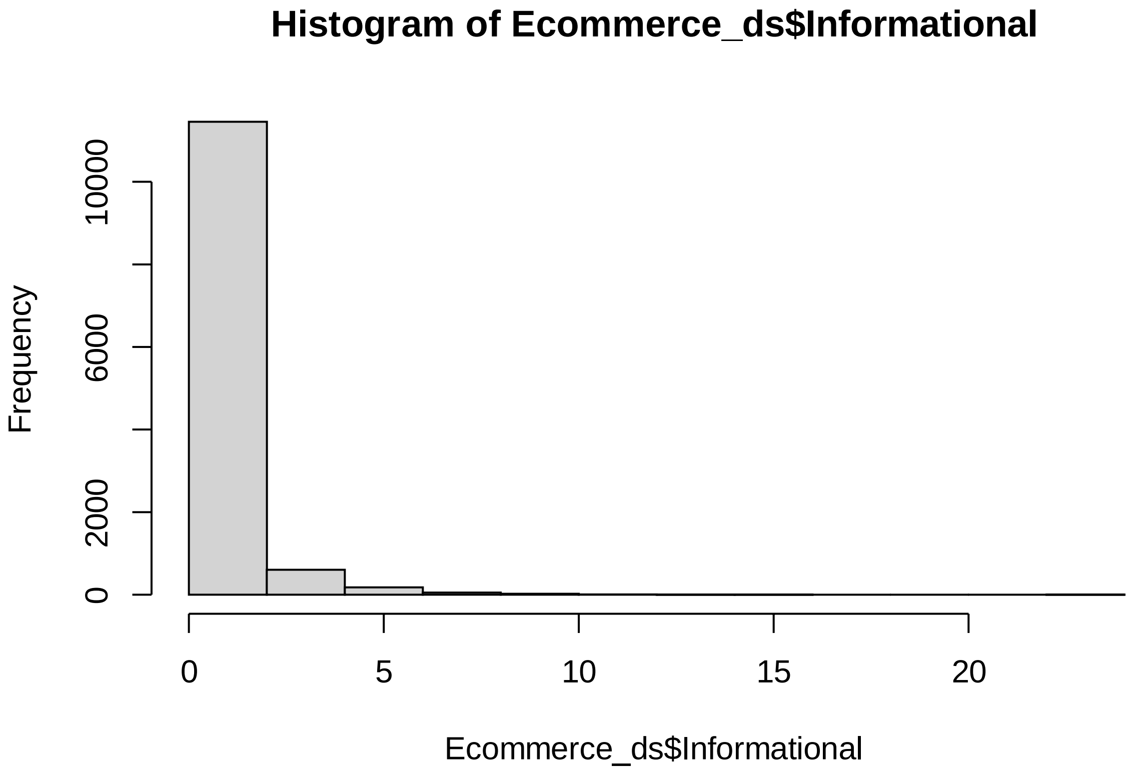
```
# histogram for Administrative
#
hist(Ecommerce_ds$Administrative)
```



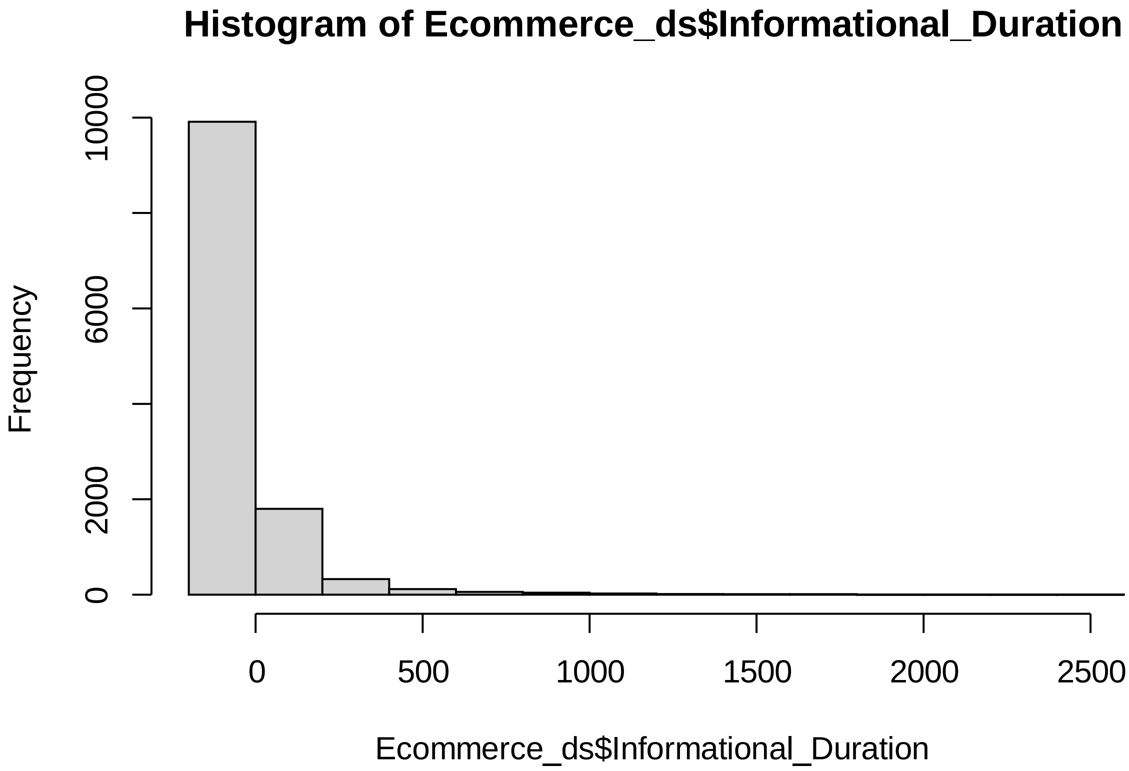
```
# histogram for administrative duration

hist(Ecommerce_ds$Administrative_Duration)
```



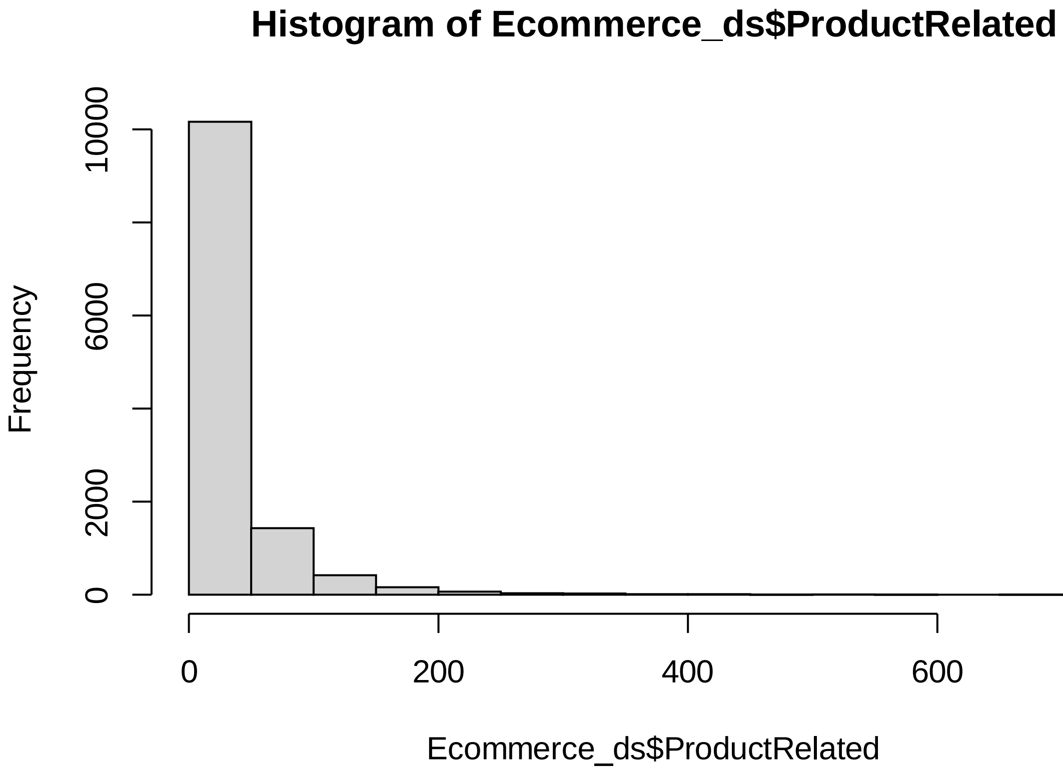
```
# histogram for informational

hist(Ecommerce_ds$Informational)
```



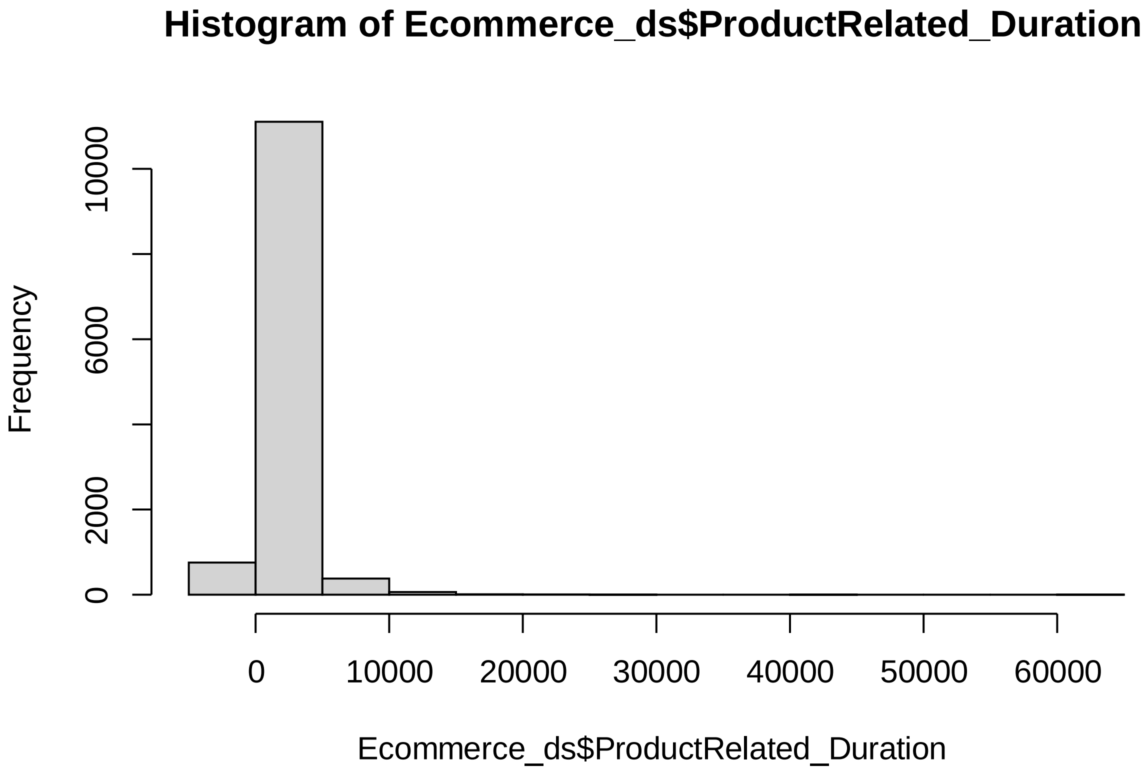
```
# histogram for informatinal duration

hist(Ecommerce_ds$Informational_Duration)
```



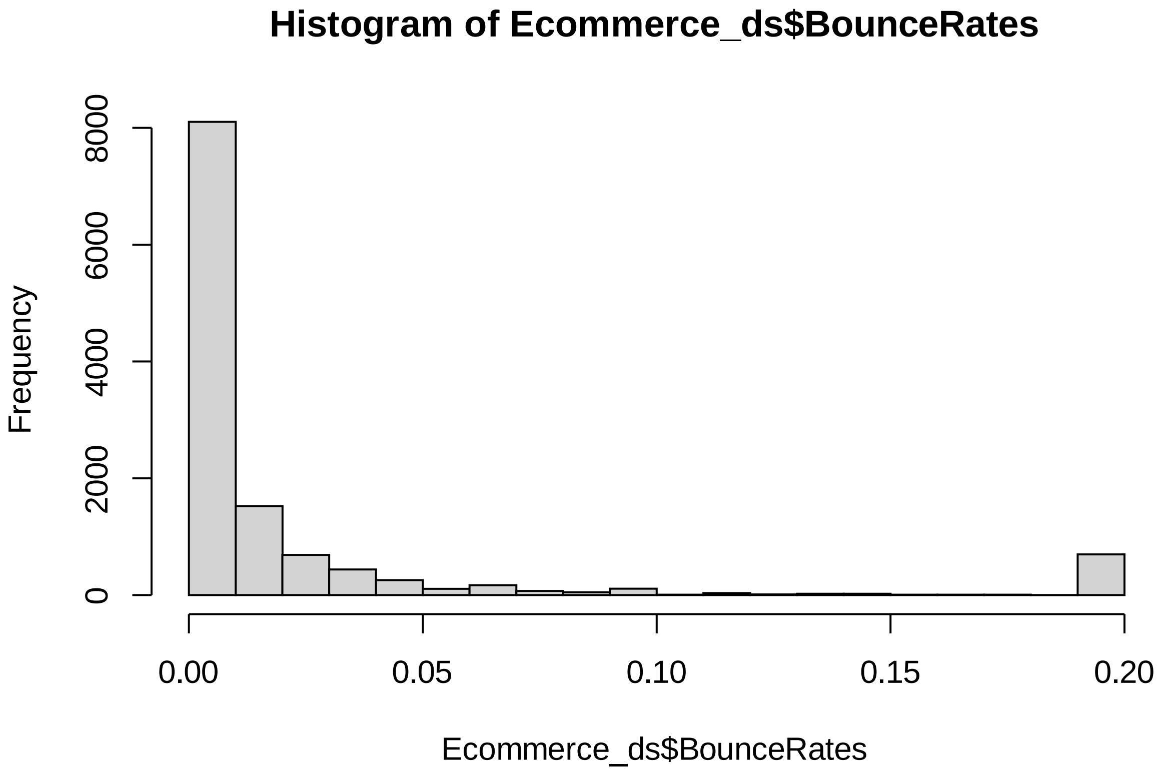
```
# histogram for product related

hist(Ecommerce_ds$ProductRelated)
```



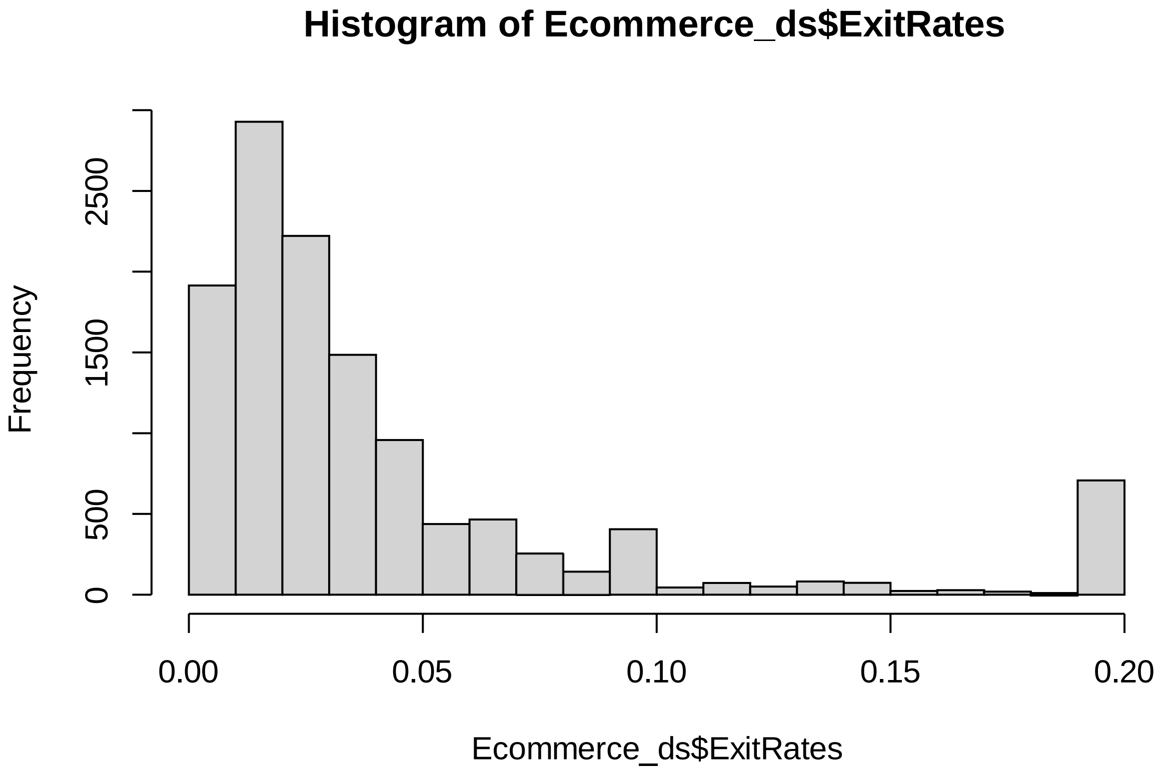
```
# histogram for product related duration
hist(Ecommerce_ds$ProductRelated_Duration)
```



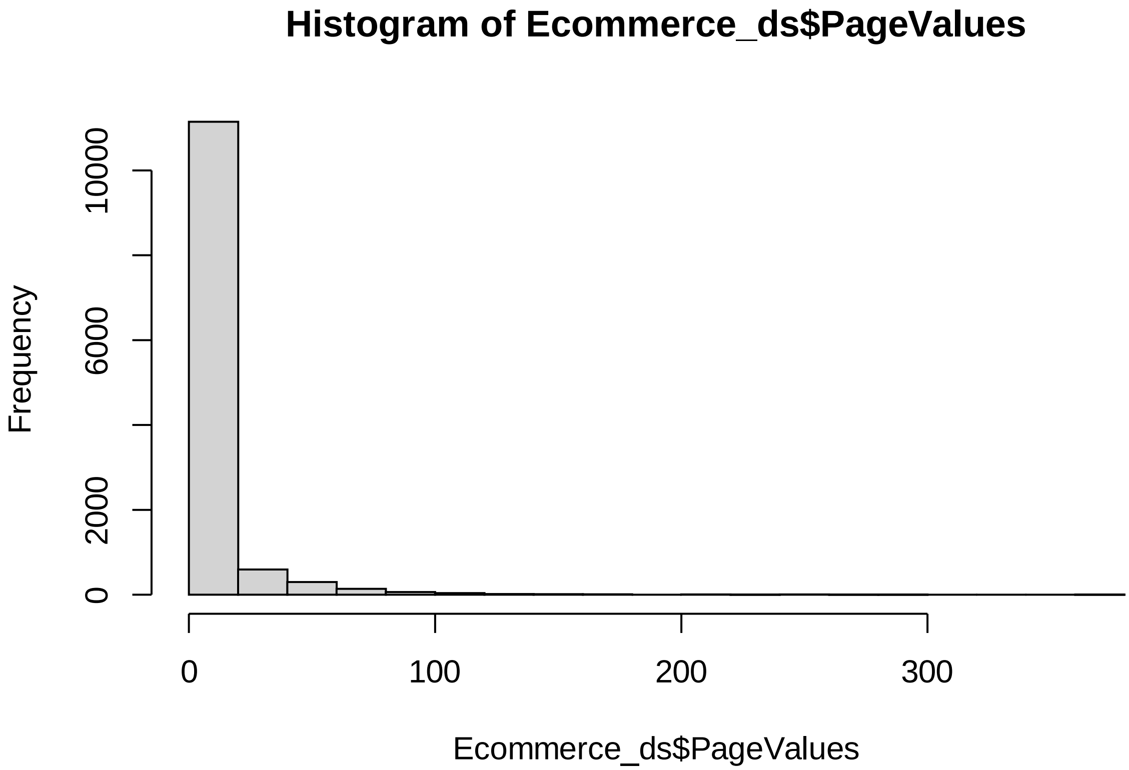
```
# histogram for bounce rates

hist(Ecommerce_ds$BounceRates)
```



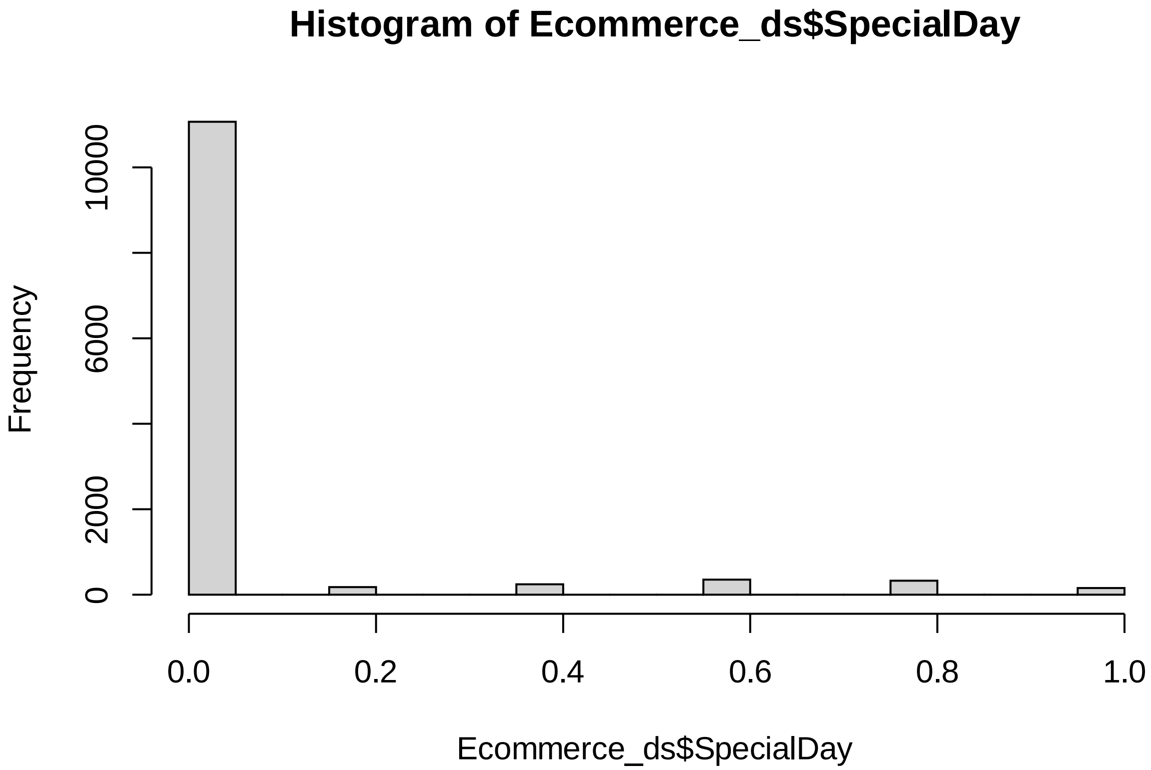
```
# histogram for exit rates

hist(Ecommerce_ds$ExitRates)
```



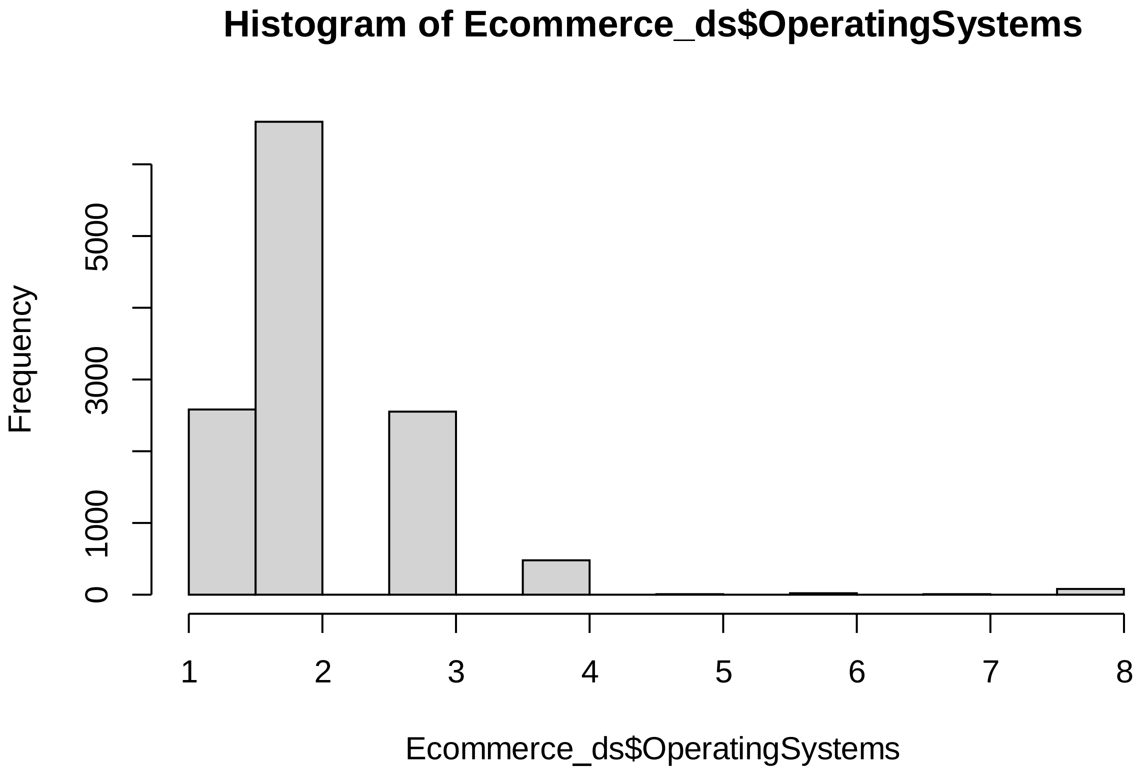
```
# histogram for page values

hist(Ecommerce_ds$PageValues)
```



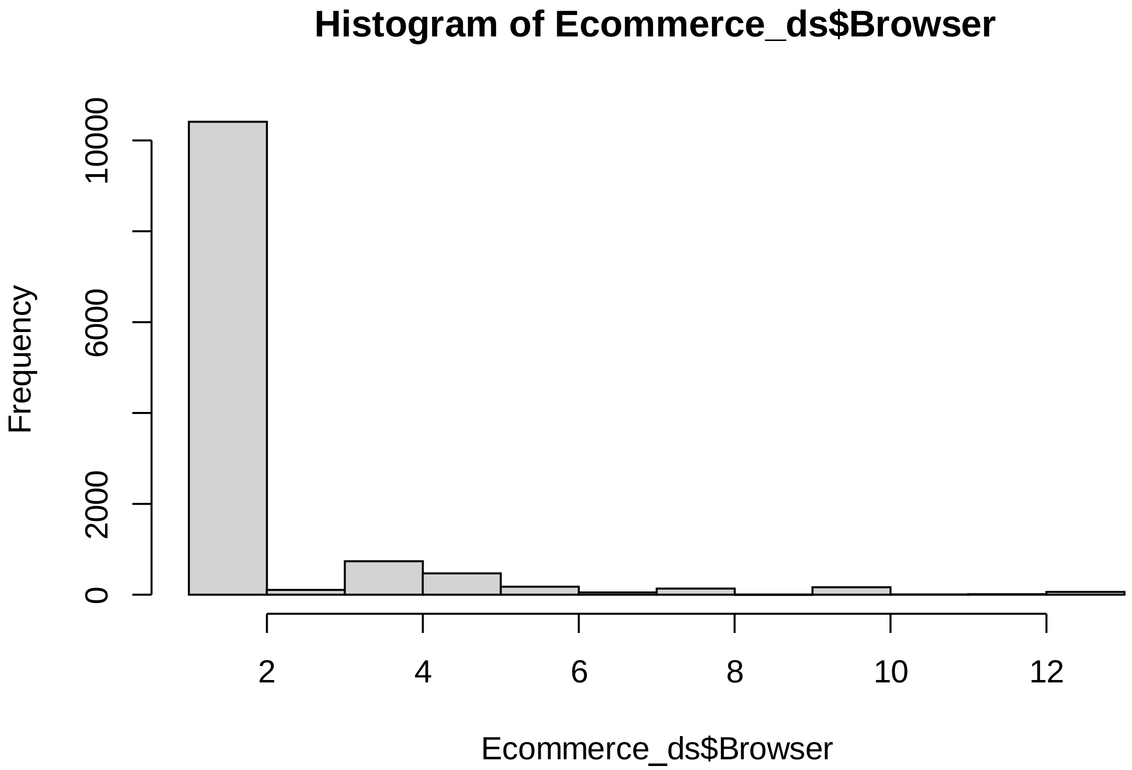
```
# histogram for special day

hist(Ecommerce_ds$SpecialDay)
```



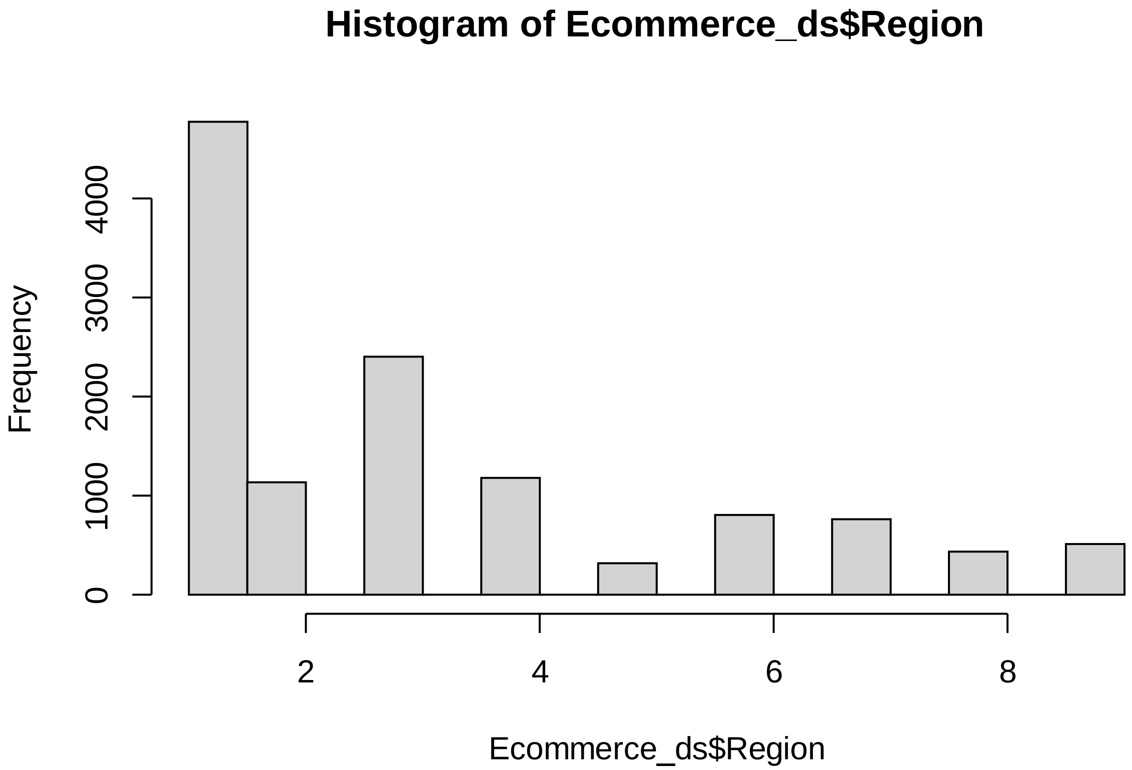
```
# histogram for operating system

hist(Ecommerce_ds$OperatingSystems)
```



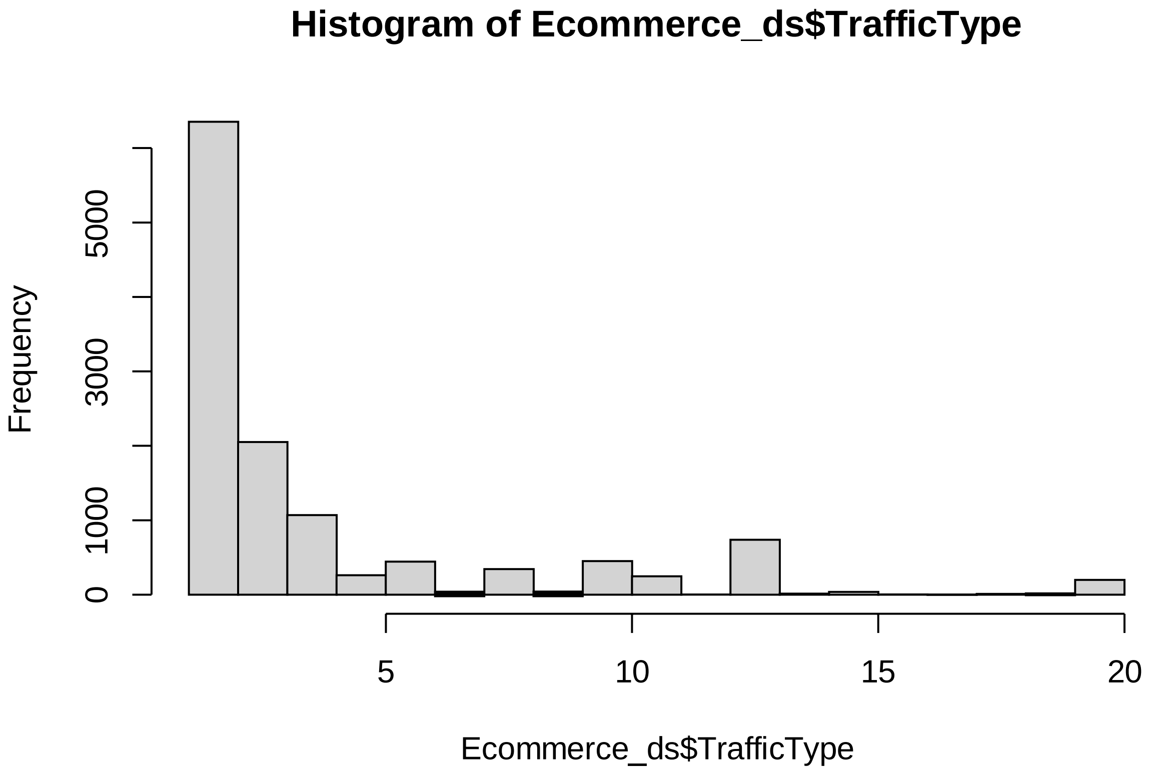
```
# histogram for browser

hist(Ecommerce_ds$Browser)
```



```
# histogram for region

hist(Ecommerce_ds$Region)
```



```
# histogram for traffic type

hist(Ecommerce_ds$TrafficType)
```



## Bivariate Analysis

Finding the correlation of the dataset

```
# assigning the admin to administrative column
admin <- Ecommerce_ds$Administrative
# assigning the admind to variable administrative duration
admind <- Ecommerce_ds$Administrative_Duration
# finding the correlation
cor(admin, admind)
```

```
## [1] 0.6014662
```

```
# assigning the informational column to variable info
info <- Ecommerce_ds$Informational
# assigning the informational duration column to variable infod
infod <- Ecommerce_ds$Informational_Duration
# finding the correlation
cor(info, infod)
```

```
## [1] 0.6189651
```

```
# assigning the product related column to variable prodr
prodr <- Ecommerce_ds$ProductRelated
# assigning the product related duration column to variable prodrd
prodrd <- Ecommerce_ds$ProductRelated_Duration
```

```
# finding the correlation
cor(prodr, prodrd)
```

```
## [1] 0.8608682
```

```
# assigning the browser column to variable brow
brow <- Ecommerce_ds$Browser
# assigning the region column to variable reg
reg <- Ecommerce_ds$Region
# finding the correlation
cor(brow, reg)
```

```
## [1] 0.09729745
```

```
# assigning the bounce rates column to variable brates
brates <- Ecommerce_ds$BounceRates
# assigning the exit rates column to variable Erates
Erates <- Ecommerce_ds$ExitRates
# finding the correlation
cor(brates, Erates)
```

```
## [1] 0.9134364
```

```
# assigning the region column to variable reg
reg <- Ecommerce_ds$Region
# assigning the Traffic type column to variable trafr
trafr <- Ecommerce_ds$TrafficType
# finding the correlation
cor(reg, trafr)
```

```
## [1] 0.04726601
```

Finding the skewness of the dataset

```
# Checking for skewness
library(e1071)
skewness(Ecommerce_ds$Administrative)
```

```
## [1] 1.958399
```

```
# skewness for administrative duration

skewness(Ecommerce_ds$Administrative_Duration)
```

```
## [1] 5.611594
```

```
# skewness for informational

skewness(Ecommerce_ds$Informational)
```

```
## [1] 4.03384
```

```r
# skewness for informatinal duration
skewness(Ecommerce_ds$Informational_Duration)
```

```
## [1] 7.572937
```

```r
# skewness for product related
skewness(Ecommerce_ds$ProductRelated)
```

```
## [1] 4.339165
```

```r
# skewness for product related duration
skewness(Ecommerce_ds$ProductRelated_Duration)
```

```
## [1] 7.259923
```

```r
# skewness for bounce rates
skewness(Ecommerce_ds$BounceRates)
```

```
## [1] 2.951747
```

```r
# skewness for exit rates
skewness(Ecommerce_ds$ExitRates)
```

```
## [1] 2.152229
```

```r
# skewness for page values
skewness(Ecommerce_ds$PageValues)
```

```
## [1] 6.377836
```

```r
# skewness for special day
skewness(Ecommerce_ds$SpecialDay)
```

```
## [1] 3.299505
```

```r
# skewness for operating system
skewness(Ecommerce_ds$OperatingSystems)
```

```
## [1] 2.066268
```

```
# skewness for browser
skewness(Ecommerce_ds$Browser)
```

```
## [1] 3.240196
```

```
# skewness for region
skewness(Ecommerce_ds$Region)
```

```
## [1] 0.9830298
```

```
# skewness for traffic type
skewness(Ecommerce_ds$TrafficType)
```

```
## [1] 1.962697
```

```
# kurtosis for administrative
kurtosis(Ecommerce_ds$Administrative)
```

```
## [1] 4.690786
```

```
# kurtosis for administrative duration
kurtosis(Ecommerce_ds$Administrative_Duration)
```

```
## [1] 50.47826
```

```
# kurtosis for informational
kurtosis(Ecommerce_ds$Informational)
```

```
## [1] 26.89329
```

```
# kurtosis for informatinal duration
kurtosis(Ecommerce_ds$Informational_Duration)
```

```
## [1] 76.18376
```

```
# kurtosis for product related
kurtosis(Ecommerce_ds$ProductRelated)
```

```
## [1] 31.1734
```

```
# kurtosis for product related duration
kurtosis(Ecommerce_ds$ProductRelated_Duration)
```

```
## [1] 137.0289
```

```
# kurtosis for bounce rates
kurtosis(Ecommerce_ds$BounceRates)
```

```
## [1] 7.748958
```

```
# kurtosis for exit rates
kurtosis(Ecommerce_ds$ExitRates)
```

```
## [1] 4.03674
```

```
# kurtosis for page values
kurtosis(Ecommerce_ds$PageValues)
```

```
## [1] 65.52603
```

```
# kurtosis for special day
kurtosis(Ecommerce_ds$SpecialDay)
```

```
## [1] 9.890555
```

```
# kurtosis for operating system
kurtosis(Ecommerce_ds$OperatingSystems)
```

```
## [1] 10.44894
```

```
# kurtosis for browser
kurtosis(Ecommerce_ds$Browser)
```

```
## [1] 12.72503
```

```
# kurtosis for region
kurtosis(Ecommerce_ds$Region)
```

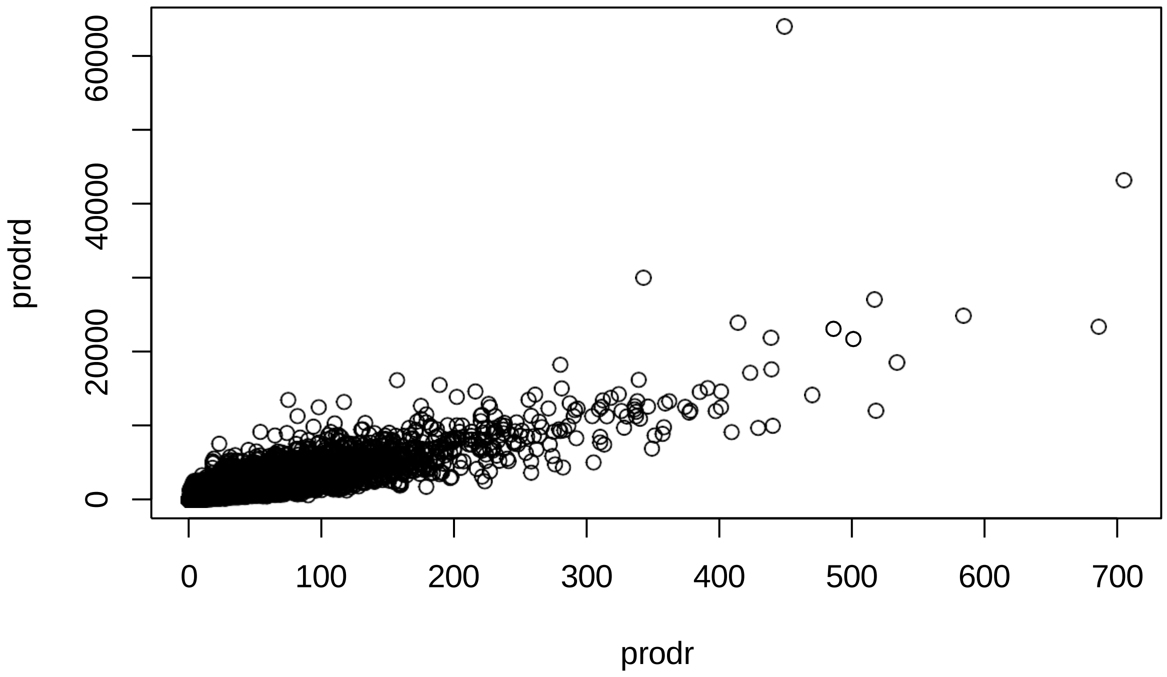
```
## [1] -0.1508587
```

```
# kurtosis for traffic type

kurtosis(Ecommerce_ds$TrafficType)
```

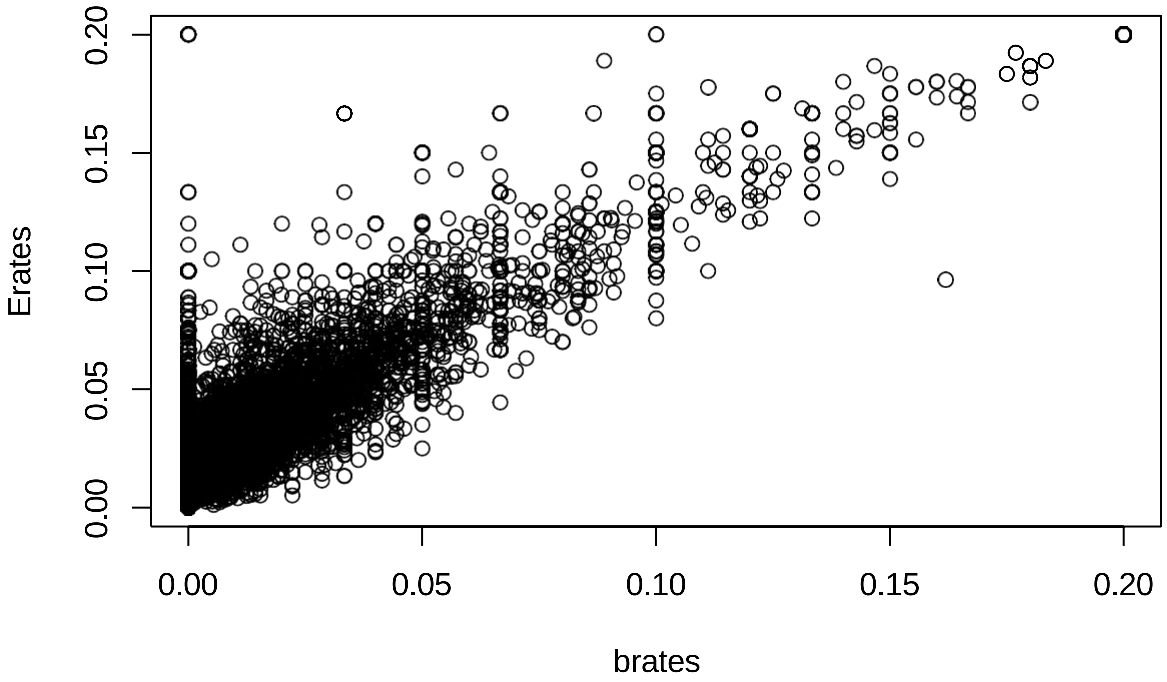
```
## [1] 3.479468
```

Scatter plot

```
# assigning the product related column to variable prodr
prodr <- Ecommerce_ds$ProductRelated
# assigning the product related duration column to variable prodrd
prodrd <- Ecommerce_ds$ProductRelated_Duration
# finding the correlation
plot(prodr, prodrd)
```



```
# assigning the bounce rates column to variable brates
brates <- Ecommerce_ds$BounceRates
# assigning the exit rates column to variable Erates
Erates <- Ecommerce_ds$ExitRates
# finding the correlation
plot(brates, Erates)
```



Implementing the solution

K-Means Clustering

Label Encoding

```
# label encoding weekend column data
Ecommerce_ds$Weekend<-as.integer(as.factor(Ecommerce_ds$Weekend))

# Label encoding continuous data for month

Ecommerce_ds$Month<-as.integer(as.factor(Ecommerce_ds$Month))
# Label encoding traffic data
Ecommerce_ds$VisitorType<-as.integer(as.factor(Ecommerce_ds$VisitorType))

summary(Ecommerce_ds)
```

```
##   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
## 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
##     BounceRates          ExitRates          PageValues          SpecialDay
##   Min.    :0.000000     Min.    :0.00000     Min.    :   0.000     Min.    :0.0000
##   1st  Qu.:0.000000     1st  Qu.:0.01429     1st  Qu.:   0.000     1st  Qu.:0.0000
##   Median  :0.003119     Median  :0.02512     Median  :   0.000     Median  :0.0000
##   Mean    :0.022152     Mean    :0.04300    Mean     :   5.896     Mean    :0.0615
##   3rd  Qu.:0.016684     3rd  Qu.:0.05000    3rd  Qu.:   0.000     3rd  Qu.:0.0000
##   Max.    :0.200000     Max.    :0.20000     Max.    :361.764     Max.    :1.0000
##         Month           OperatingSystems      Browser               Region
##   Min.    :  1.000    Min.    :1.000      Min.    :  1.000     Min.    :1.000
##   1st  Qu.:  6.000    1st  Qu.:2.000      1st  Qu.:  2.000     1st  Qu.:1.000
##   Median  :  7.000    Median  :2.000      Median  :  2.000     Median  :3.000
##   Mean    :  6.164    Mean    :2.124      Mean    :  2.358     Mean    :3.148
##   3rd  Qu.:  8.000    3rd  Qu.:3.000      3rd  Qu.:  2.000     3rd  Qu.:4.000
##   Max.    :10.000     Max.    :8.000      Max.    :13.000      Max.    :9.000
##     TrafficType         VisitorType           Weekend          Revenue
##   Min.    :  1.00     Min.    :1.000     Min.    :1.000     Mode  :logical
##   1st  Qu.:  2.00     1st  Qu.:3.000     1st  Qu.:1.000     FALSE:10408
##   Median  :  2.00     Median  :3.000     Median  :1.000     TRUE  :1908
##   Mean    :  4.07     Mean    :2.718     Mean    :1.233
##   3rd  Qu.:  4.00     3rd  Qu.:3.000     3rd  Qu.:1.000
##   Max.    :20.00      Max.    :3.000     Max.    :2.000
```

Preprocessing our dataset

```
# Pre processing the dataset
# Since clustering is a type of Unsupervised Learning,
# we would not require Class Label(output) during execution of our algorithm.
# We would then normalize the attributes between 0 and 1 using our own function.
# ---
#
Ecommerce        <-        Ecommerce_ds[,        c(1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17)]
Ecommerce_ds.class<- Ecommerce_ds[, "Revenue"]
head(Ecommerce_ds.class)
```

```
## [1] FALSE FALSE FALSE FALSE FALSE FALSE
```

Normalizing our dataset

```
# Normalizing the dataset so that no particular attribute
# has more impact on clustering algorithm than others.
# ---
#
normalize <- function(x){
 return ((x-min(x)) / (max(x)-min(x)))
}
Ecommerce$Administrative<-  normalize(Ecommerce$Administrative)
```

```
Ecommerce$Administrative_Duration<- normalize(Ecommerce$Administrative_Duration)
Ecommerce$Informational<- normalize(Ecommerce$Informational)
Ecommerce$Informational_Duration<- normalize(Ecommerce$Informational_Duration)
Ecommerce$ProductRelated<- normalize(Ecommerce$ProductRelated)
Ecommerce$ProductRelated_Duration<- normalize(Ecommerce$ProductRelated_Duration)
Ecommerce$BounceRates<- normalize(Ecommerce$BounceRates)
Ecommerce$ExitRates<- normalize(Ecommerce$ExitRates)
Ecommerce$PageValues<- normalize(Ecommerce$ExitRates)
Ecommerce$PageValues<- normalize(Ecommerce$PageValues)
Ecommerce$SpecialDay<- normalize(Ecommerce$SpecialDay)
Ecommerce$SpecialDay<- normalize(Ecommerce$SpecialDay)
Ecommerce$Month<- normalize(Ecommerce$Month)
Ecommerce$OperatingSystems<- normalize(Ecommerce$OperatingSystems)
Ecommerce$Browser<- normalize(Ecommerce$Browser)
Ecommerce$Region<- normalize(Ecommerce$Region)
Ecommerce$TrafficType<- normalize(Ecommerce$TrafficType)
Ecommerce$VisitorType<- normalize(Ecommerce$VisitorType)
head(Ecommerce)
```

```
##   Administrative Administrative_Duration Informational Informational_Duration
## 1              0            0.0002941393             0           0.0003920992
## 2              0            0.0002941393             0           0.0003920992
## 3              0            0.0000000000             0           0.0000000000
## 4              0            0.0002941393             0           0.0003920992
## 5              0            0.0002941393             0           0.0003920992
## 6              0            0.0002941393             0           0.0003920992
##   ProductRelated ProductRelated_Duration BounceRates ExitRates PageValues
## 1    0.001418440            1.563122e-05  1.00000000  1.000000   1.000000
## 2    0.002836879            1.016029e-03  0.00000000  0.500000   0.500000
## 3    0.001418440            0.000000e+00  1.00000000  1.000000   1.000000
## 4    0.002836879            5.731448e-05  0.25000000  0.700000   0.700000
## 5    0.014184397            9.824223e-03  0.10000000  0.250000   0.250000
## 6    0.026950355            2.426226e-03  0.07894737  0.122807   0.122807
##   SpecialDay     Month OperatingSystems    Browser Region TrafficType
## 1          0 0.2222222        0.0000000 0.00000000  0.000  0.00000000
## 2          0 0.2222222        0.1428571 0.08333333  0.000  0.05263158
## 3          0 0.2222222        0.4285714 0.00000000  1.000  0.10526316
## 4          0 0.2222222        0.2857143 0.08333333  0.125  0.15789474
## 5          0 0.2222222        0.2857143 0.16666667  0.000  0.15789474
## 6          0 0.2222222        0.1428571 0.08333333  0.000  0.10526316
##   VisitorType Weekend
## 1           1       1
## 2           1       1
## 3           1       1
## 4           1       1
## 5           1       2
## 6           1       1
```

Applying the kmeans clastering

```
# Applying the K-means clustering algorithm with no. of centroids (k)=3
# ---
#
```

```
result<-  kmeans(Ecommerce,3)
# Previewing the no. of records in each cluster
#
result$size
```

```
## [1] 8593 2722 1001
```

```
# Getting the value of cluster center datapoint value(3 centers for k=5)
# ---
#
result$centers
```

```
##    Administrative Administrative_Duration Informational Informational_Duration
## 1     0.092301658            0.0259545470  0.0218831219           0.0144657731
## 2     0.096470460            0.0268327658  0.0256857703           0.0171722255
## 3     0.001517002            0.0006480369  0.0006660007           0.0004160717
##   ProductRelated ProductRelated_Duration BounceRates ExitRates PageValues
## 1    0.048517637            0.0203490561  0.04539471 0.1552114  0.1552114
## 2    0.049230593            0.0200883047  0.04501661 0.1460399  0.1460399
## 3    0.003977583            0.0009058467  0.85068599 0.9159283  0.9159283
##   SpecialDay     Month OperatingSystems   Browser    Region TrafficType
## 1 0.05897824 0.5702574        0.1595318 0.1164029 0.2697399   0.1566208
## 2 0.05481264 0.5922116        0.1614884 0.1037228 0.2676800   0.1597703
## 3 0.10129870 0.5540016        0.1672613 0.1106394 0.2601149   0.2093696
##   VisitorType  Weekend
## 1   0.8575585 1.000000
## 2   0.8253123 2.000000
## 3   0.9630370 1.142857
```

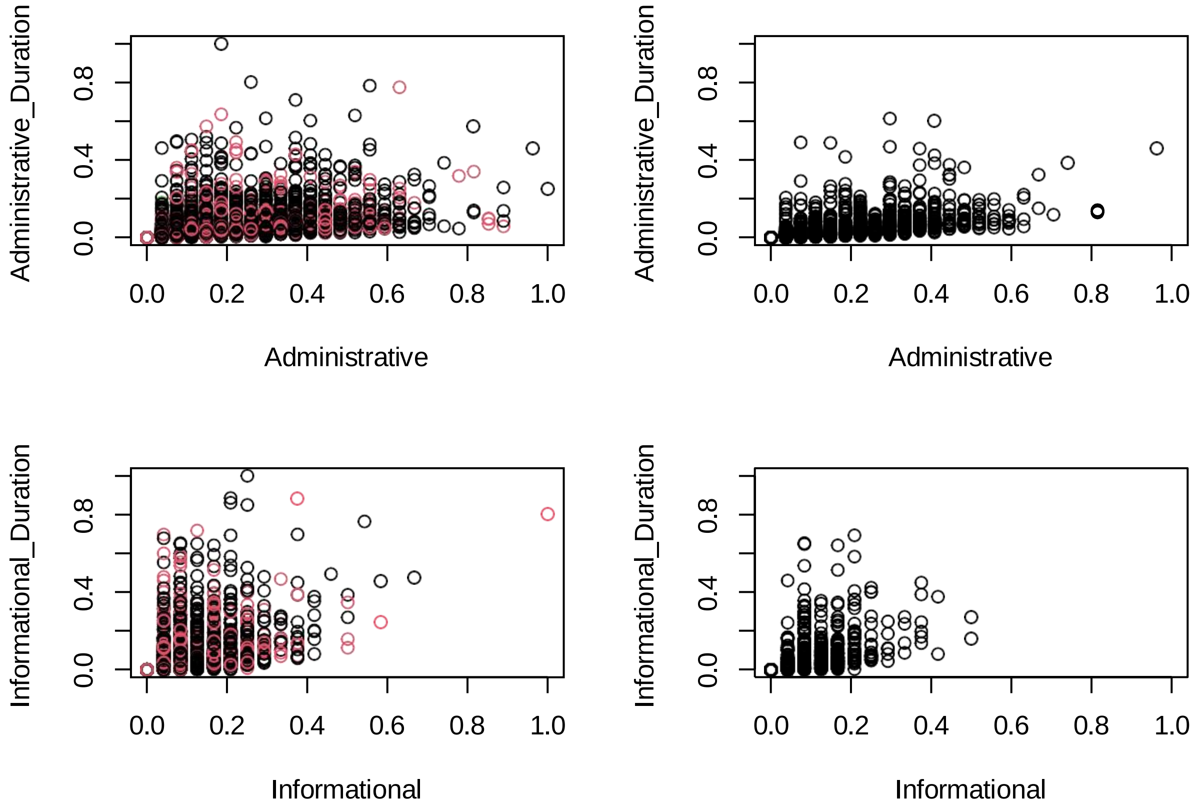
```
# Getting the cluster vector that shows the cluster where each record falls
# ---
#
head(result$cluster)
```

```
## 1 2 3 4 5 6
## 3 1 3 3 2 1
```

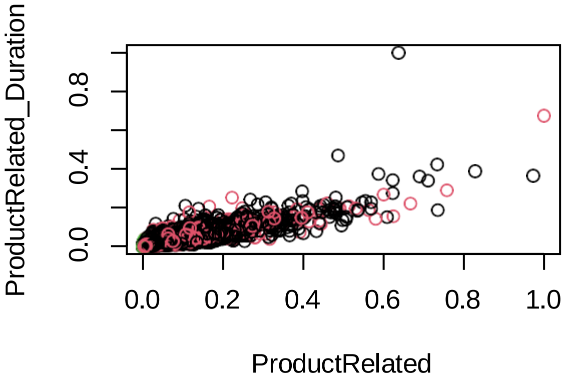
```
# The graph shows that we have got 5 clearly distinguishable clusters for Ozone and Solar.R data points
# Let's see how clustering has performed on Wind and Temp attributes.
```

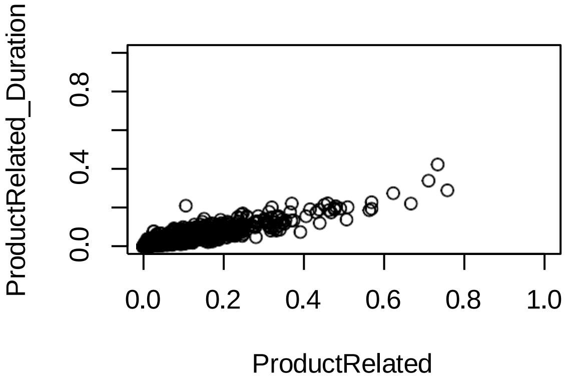
```
# Verifying the results of clustering
# ---
#
par(mfrow = c(2,2), mar = c(5,4,2,2))
# Plotting to see how Administrative and Administrative_Duration points have been distributed in cluste
plot(Ecommerce[c(1,2)], col = result$cluster)
# Plotting to see how Administrative and Administrative_Duration data points have been distributed
# originally as per "class" attribute in dataset
# ---
#
plot(Ecommerce[c(1,2)], col = Ecommerce_ds.class)
```

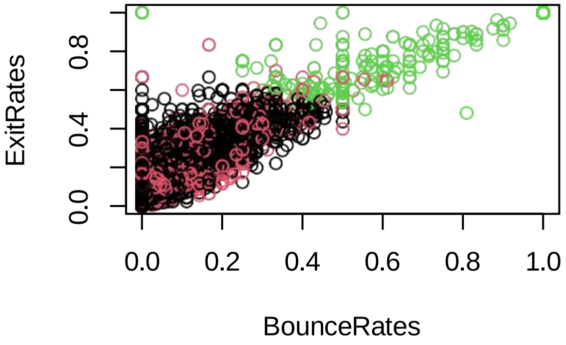
```
# Plotting to see how Informational and Informational_Duration data points have been distributed in clu
# ---
#
plot(Ecommerce[c(3,4)], col = result$cluster)
plot(Ecommerce[c(3,4)], col = Ecommerce_ds.class)
```

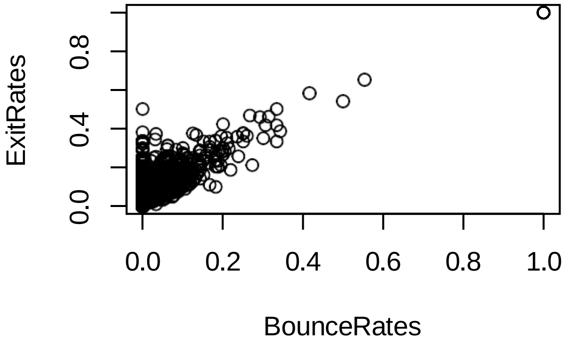


```
# Plotting to see how ProductRelated and ProductRelated_Duration data points have been distributed in c
# ---
#
plot(Ecommerce[c(5,6)], col = result$cluster)
plot(Ecommerce[c(5,6)], col = Ecommerce_ds.class)
# Plotting to see how BounceRates and ExitRates data points have been distributed in clusters
# ---
#
plot(Ecommerce[c(7,8)], col = result$cluster)
plot(Ecommerce[c(7,8)], col = Ecommerce_ds.class)
```

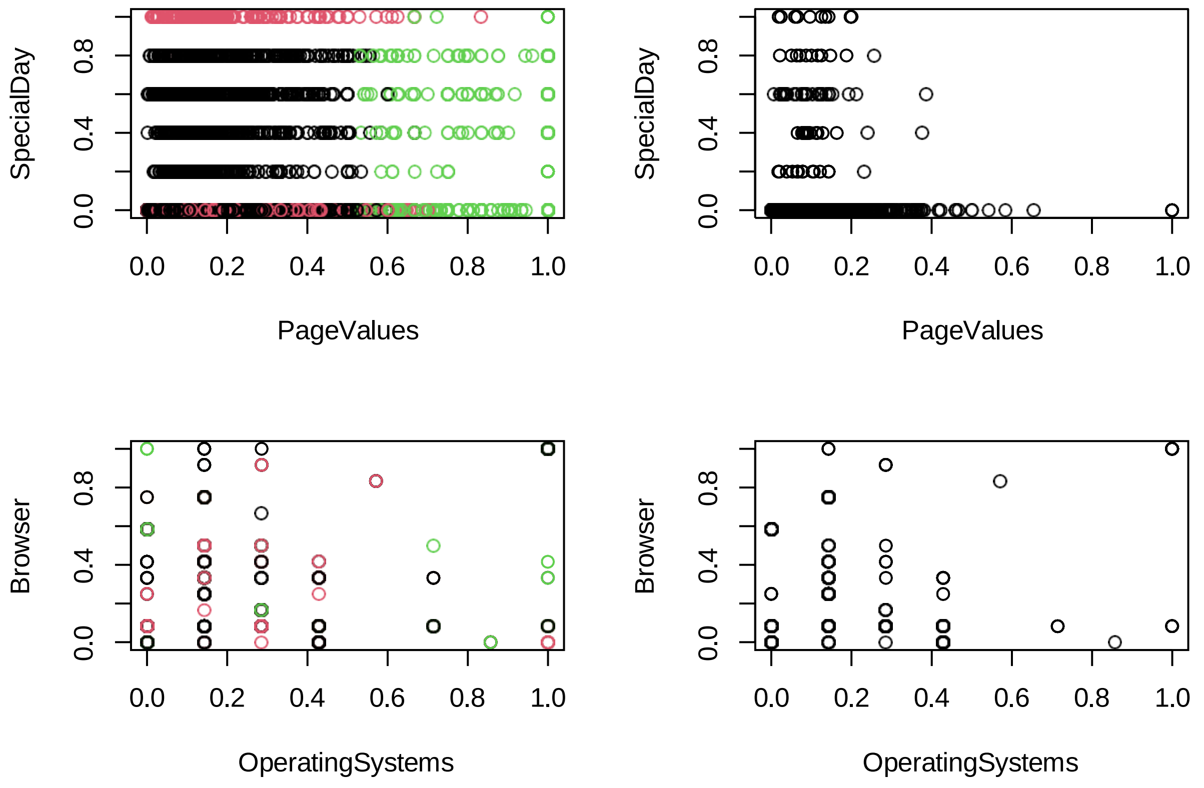




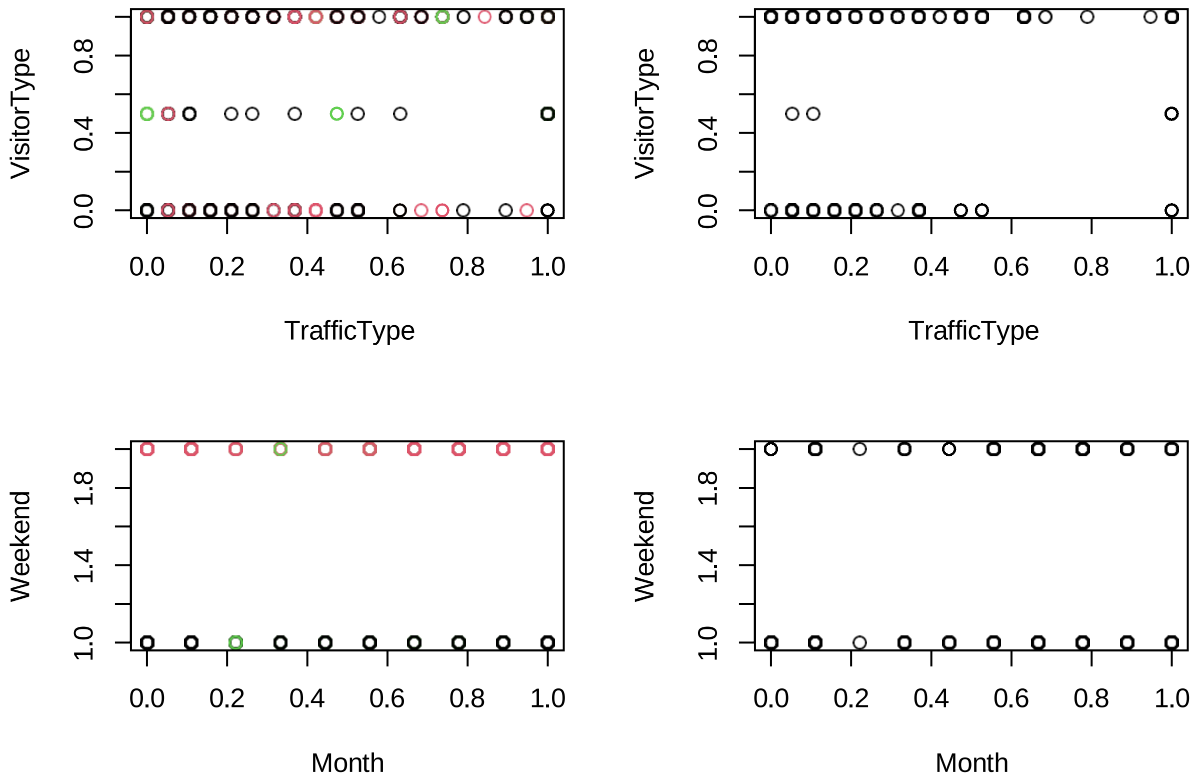




```
# Plotting to see how PageValues and SpecialDay data points have been distributed in clusters
# ---
#
plot(Ecommerce[c(9,10)], col = result$cluster)
plot(Ecommerce[c(9,10)], col = Ecommerce_ds.class)
# Plotting to see how OperatingSystems and Browser data points have been distributed in clusters
# ---
#
plot(Ecommerce[c(12,13)], col = result$cluster)
plot(Ecommerce[c(12,13)], col = Ecommerce_ds.class)
```



```
# Plotting to see how TrafficType and VisitorType data points have been distributed in clusters
# ---
#
plot(Ecommerce[c(15,16)], col = result$cluster)
plot(Ecommerce[c(15,16)], col = Ecommerce_ds.class)
# Plotting to see how Month and Weekend data points have been distributed in clusters
# ---
#
plot(Ecommerce[c(11,17)], col = result$cluster)
plot(Ecommerce[c(11,17)], col = Ecommerce_ds.class)
```



```
# Result of table shows that Cluster 1 corresponds to Virginica,
# Cluster 2 corresponds to Versicolor and Cluster 3 to Setosa.
# ---
#
table(result$cluster,  Ecommerce_ds.class)
```

```
##       Ecommerce_ds.class
##        FALSE TRUE
##     1   7189 1404
##     2   2223  499
##     3    996    5
```

Haerachical Clastering

Scaling the dataset

```
# As we don't want the hierarchical clustering result to depend to an arbitrary variable unit,
# we start by scaling the data using the R function scale() as follows
# ---
#
Ecommerce_ds <- scale(Ecommerce_ds)
head(Ecommerce_ds)
```

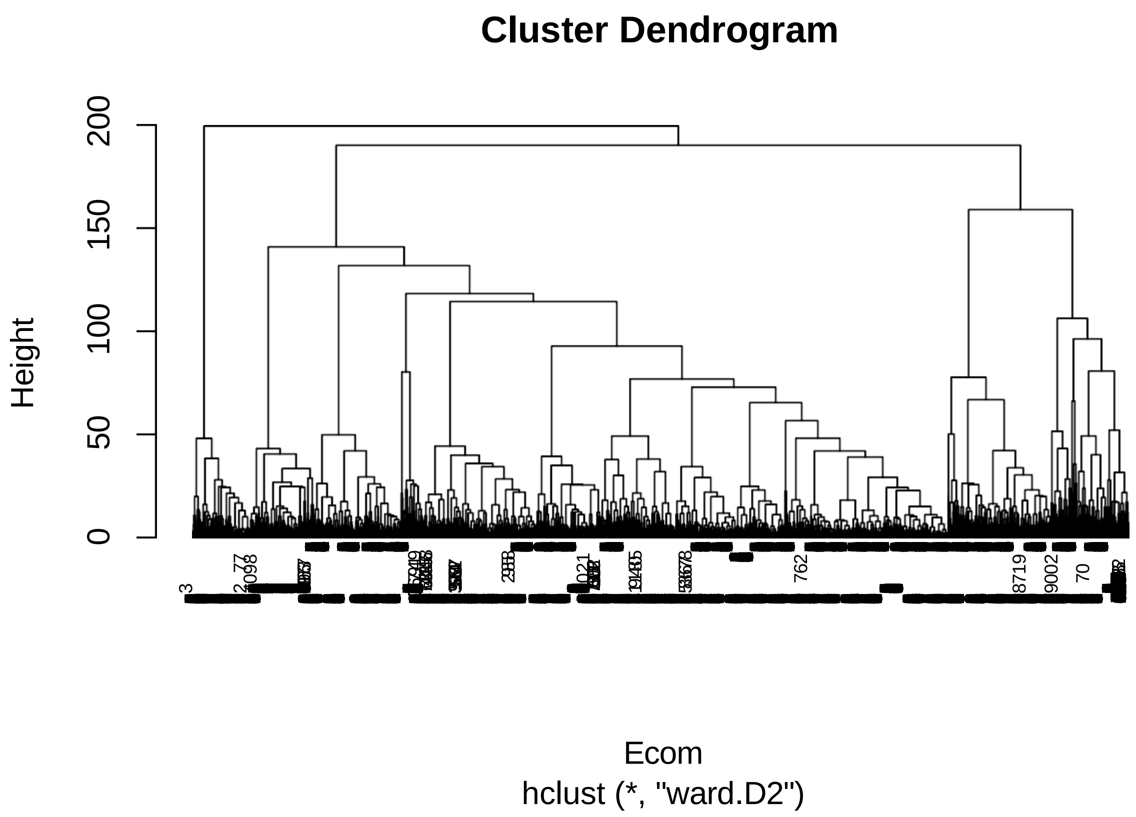
```
##     Administrative Administrative_Duration Informational Informational_Duration
## 1      -0.6975533               -0.4574578    -0.3966145              -0.2450294
## 2      -0.6975533               -0.4574578    -0.3966145              -0.2450294
## 3      -0.6975533               -0.4631119    -0.3966145              -0.2521304
## 4      -0.6975533               -0.4574578    -0.3966145              -0.2450294
## 5      -0.6975533               -0.4574578    -0.3966145              -0.2450294
## 6      -0.6975533               -0.4574578    -0.3966145              -0.2450294
##   ProductRelated ProductRelated_Duration BounceRates  ExitRates PageValues
## 1     -0.6914734              -0.6247671  3.67247746  3.2352400 -0.3173633
## 2     -0.6689966              -0.5913358 -0.45743910  1.1745443 -0.3173633
## 3     -0.6914734              -0.6252895  3.67247746  3.2352400 -0.3173633
## 4     -0.6689966              -0.6233742  0.57504004  1.9988226 -0.3173633
## 5     -0.4891823              -0.2969835 -0.04444744  0.1441964 -0.3173633
## 6     -0.2868911              -0.5442099 -0.13139305 -0.3800157 -0.3173633
##   SpecialDay     Month OperatingSystems    Browser     Region TrafficType
## 1  -0.309001 -1.334201       -1.2332048 -0.7901988 -0.8941841 -0.76292777
## 2  -0.309001 -1.334201       -0.1361914 -0.2081361 -0.8941841 -0.51445574
## 3  -0.309001 -1.334201        2.0578354 -0.7901988  2.4360812 -0.26598370
## 4  -0.309001 -1.334201        0.9608220 -0.2081361 -0.4779009 -0.01751167
## 5  -0.309001 -1.334201        0.9608220  0.3739266 -0.8941841 -0.01751167
## 6  -0.309001 -1.334201       -0.1361914 -0.2081361 -0.8941841 -0.26598370
##   VisitorType    Weekend    Revenue
## 1   0.4080401 -0.5505615 -0.4281421
## 2   0.4080401 -0.5505615 -0.4281421
## 3   0.4080401 -0.5505615 -0.4281421
## 4   0.4080401 -0.5505615 -0.4281421
## 5   0.4080401  1.8161802 -0.4281421
## 6   0.4080401 -0.5505615 -0.4281421
```

Performing Hierachical Clustering

```r
# We now use the R function hclust() for hierarchical clustering
# ---
#
# First we use the dist() function to compute the Euclidean distance between observations,
# d will be the first argument in the hclust() function dissimilarity matrix
# ---
#
Ecom <- dist(Ecommerce_ds, method = "euclidean")
# We then hierarchical clustering using the Ward’s method
# ---
#
res.hc <- hclust(Ecom, method = "ward.D2" )
```

Plot the dendrogram

```r
# Lastly, we plot the obtained dendrogram
# ---
#
plot(res.hc, cex = 0.6, hang = -1)
```



### DBSCAN Clustering

```
# Removing the class label
# ---
#
library("dbscan")
Ecomm<-Ecommerce_ds[,c(1:17)]
head(Ecomm)
```

```
##   Administrative Administrative_Duration Informational Informational_Duration
## 1     -0.6975533              -0.4574578    -0.3966145             -0.2450294
## 2     -0.6975533              -0.4574578    -0.3966145             -0.2450294
## 3     -0.6975533              -0.4631119    -0.3966145             -0.2521304
## 4     -0.6975533              -0.4574578    -0.3966145             -0.2450294
## 5     -0.6975533              -0.4574578    -0.3966145             -0.2450294
## 6     -0.6975533              -0.4574578    -0.3966145             -0.2450294
##   ProductRelated ProductRelated_Duration BounceRates  ExitRates PageValues
## 1     -0.6914734              -0.6247671  3.67247746  3.2352400 -0.3173633
## 2     -0.6689966              -0.5913358 -0.45743910  1.1745443 -0.3173633
## 3     -0.6914734              -0.6252895  3.67247746  3.2352400 -0.3173633
## 4     -0.6689966              -0.6233742  0.57504004  1.9988226 -0.3173633
## 5     -0.4891823              -0.2969835 -0.04444744  0.1441964 -0.3173633
## 6     -0.2868911              -0.5442099 -0.13139305 -0.3800157 -0.3173633
##   SpecialDay     Month OperatingSystems    Browser     Region TrafficType
## 1  -0.309001 -1.334201       -1.2332048 -0.7901988 -0.8941841 -0.76292777
## 2  -0.309001 -1.334201       -0.1361914 -0.2081361 -0.8941841 -0.51445574
## 3  -0.309001 -1.334201        2.0578354 -0.7901988  2.4360812 -0.26598370
## 4  -0.309001 -1.334201          0.9608220 -0.2081361 -0.4779009 -0.01751167
## 5  -0.309001 -1.334201          0.9608220  0.3739266 -0.8941841 -0.01751167
## 6  -0.309001 -1.334201         -0.1361914 -0.2081361 -0.8941841 -0.26598370
##   VisitorType    Weekend
## 1   0.4080401 -0.5505615
## 2   0.4080401 -0.5505615
## 3   0.4080401 -0.5505615
## 4   0.4080401 -0.5505615
## 5   0.4080401  1.8161802
## 6   0.4080401 -0.5505615
```

Applying DBSCAN Algorithm

```
# Applying our DBSCAN algorithm
# ---
# We want minimum 17 points with in a distance of eps(0.4)
#
db<-dbscan(Ecomm, eps=2, MinPts = 17)
```

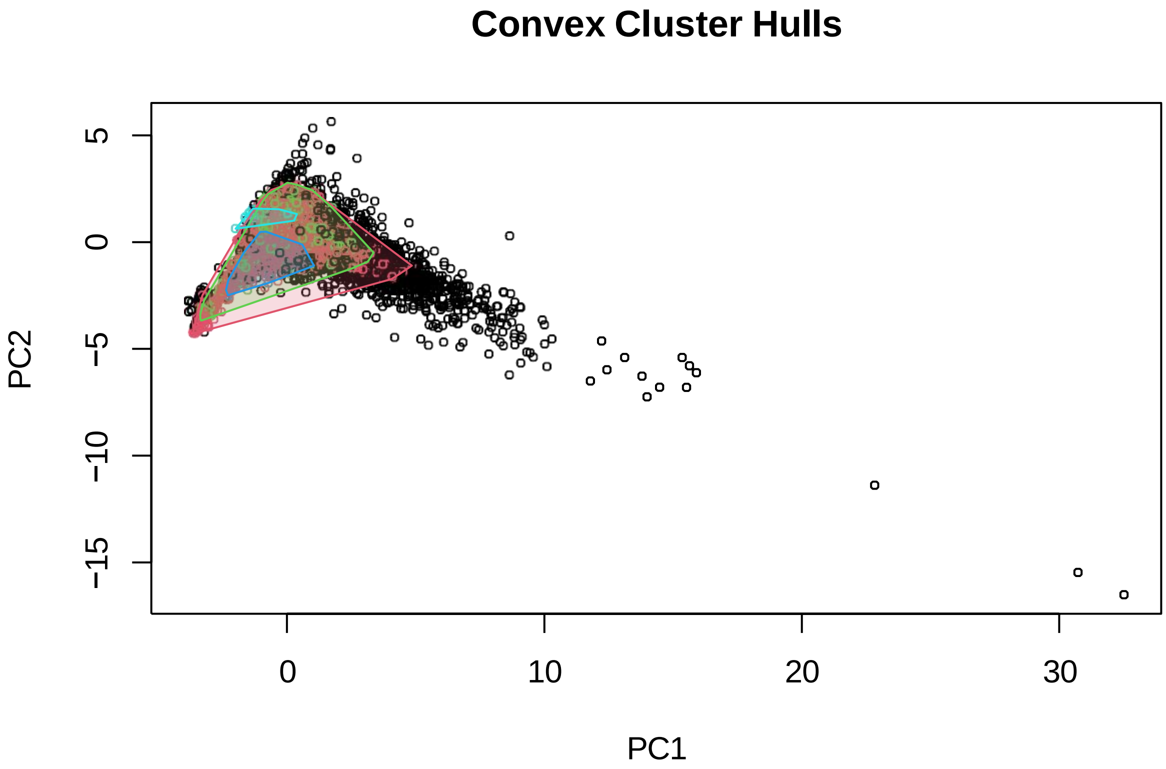
```
## Warning in dbscan(Ecomm, eps = 2, MinPts = 17): converting argument MinPts (fpc)
## to minPts (dbscan)!
```

```
# Printing out the clustering results
# ---
#
print(db)
```

```
## DBSCAN clustering for 12316 objects.
## Parameters: eps = 2, minPts = 17
## The clustering contains 4 cluster(s) and 1934 noise points.
##
##    0    1    2    3    4
## 1934 8127 2135   94   26
##
## Available fields: cluster, eps, minPts
```

Plotting our clasters

```
# We also plot our clusters as shown
# ---
# The dataset and cluster method of dbscan is used to plot the clusters.
#
hullplot(Ecomm,db$cluster)
```



Challenging our Solution

Haerachical Clastering

Scaling the dataset

```
# As we don't want the hierarchical clustering result to depend to an arbitrary variable unit,
# we start by scaling the data using the R function scale() as follows
# ---
#
Ecommerce_ds <- scale(Ecommerce_ds)
head(Ecommerce_ds)
```

```
##    Administrative  Administrative_Duration  Informational  Informational_Duration
## 1      -0.6975533               -0.4574578     -0.3966145              -0.2450294
## 2      -0.6975533               -0.4574578     -0.3966145              -0.2450294
## 3      -0.6975533               -0.4631119     -0.3966145              -0.2521304
## 4      -0.6975533               -0.4574578     -0.3966145              -0.2450294
## 5      -0.6975533               -0.4574578     -0.3966145              -0.2450294
## 6      -0.6975533               -0.4574578     -0.3966145              -0.2450294
##    ProductRelated  ProductRelated_Duration  BounceRates  ExitRates  PageValues
## 1      -0.6914734               -0.6247671   3.67247746  3.2352400  -0.3173633
## 2      -0.6689966               -0.5913358  -0.45743910  1.1745443  -0.3173633
## 3      -0.6914734               -0.6252895   3.67247746  3.2352400  -0.3173633
## 4        -0.6689966                    -0.6233742   0.57504004  1.9988226 -0.3173633
## 5        -0.4891823                    -0.2969835  -0.04444744  0.1441964 -0.3173633
## 6        -0.2868911                    -0.5442099  -0.13139305 -0.3800157 -0.3173633
##    SpecialDay     Month OperatingSystems    Browser     Region  TrafficType
## 1   -0.309001 -1.334201       -1.2332048 -0.7901988 -0.8941841 -0.76292777
## 2   -0.309001 -1.334201       -0.1361914 -0.2081361 -0.8941841 -0.51445574
## 3   -0.309001 -1.334201        2.0578354 -0.7901988  2.4360812 -0.26598370
## 4   -0.309001 -1.334201        0.9608220 -0.2081361 -0.4779009 -0.01751167
## 5   -0.309001 -1.334201        0.9608220  0.3739266 -0.8941841 -0.01751167
## 6   -0.309001 -1.334201       -0.1361914 -0.2081361 -0.8941841 -0.26598370
##    VisitorType    Weekend    Revenue
## 1    0.4080401 -0.5505615 -0.4281421
## 2    0.4080401 -0.5505615 -0.4281421
## 3    0.4080401 -0.5505615 -0.4281421
## 4    0.4080401 -0.5505615 -0.4281421
## 5    0.4080401  1.8161802 -0.4281421
## 6    0.4080401 -0.5505615 -0.4281421
```

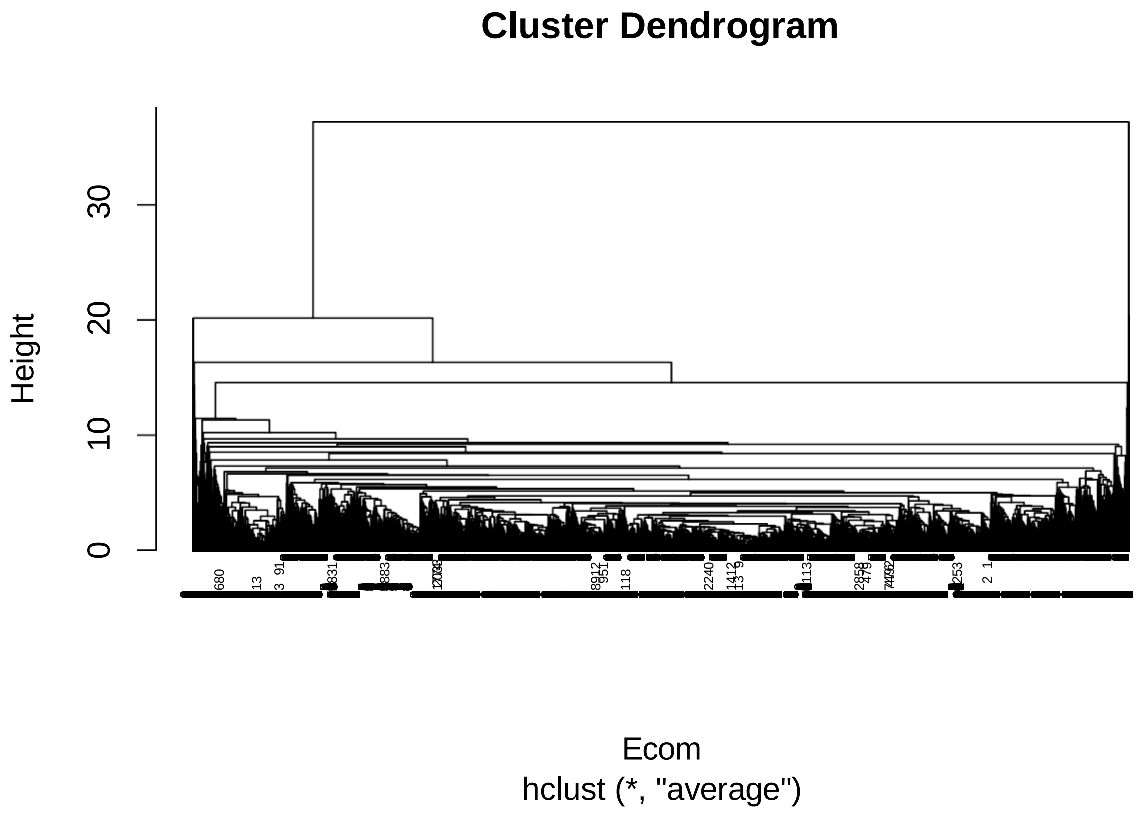
Performing Hierachical Clustering

Performing hierachical clustering using single

```
# We now use the R function hclust() for hierarchical clustering
# ---
#
# First we use the dist() function to compute the Euclidean distance between observations,
# d will be the first argument in the hclust() function dissimilarity matrix
# ---
#
Ecom <- dist(Ecommerce_ds, method = "euclidean")
# We then hierarchical clustering using the Ward’s method
# ---
#
res.hc <- hclust(Ecom, method = "average")
```

Plot the dendrogram

```
# Lastly, we plot the obtained dendrogram
# ---
#
plot(res.hc, cex = 0.4, hang = -1)
```



### DBSCAN Clustering

```
# Removing the class label
# ---
#
library("dbscan")
Ecomm<-Ecommerce_ds[,c(1:17)]
head(Ecomm)
```

```
##     Administrative  Administrative_Duration  Informational  Informational_Duration
## 1       -0.6975533               -0.4574578     -0.3966145              -0.2450294
## 2       -0.6975533               -0.4574578     -0.3966145              -0.2450294
## 3       -0.6975533               -0.4631119     -0.3966145              -0.2521304
## 4       -0.6975533               -0.4574578     -0.3966145              -0.2450294
## 5       -0.6975533               -0.4574578     -0.3966145              -0.2450294
## 6       -0.6975533               -0.4574578     -0.3966145              -0.2450294
##     ProductRelated  ProductRelated_Duration  BounceRates  ExitRates  PageValues
## 1       -0.6914734               -0.6247671   3.67247746  3.2352400  -0.3173633
## 2       -0.6689966               -0.5913358  -0.45743910  1.1745443  -0.3173633
## 3       -0.6914734               -0.6252895   3.67247746  3.2352400  -0.3173633
## 4       -0.6689966               -0.6233742   0.57504004  1.9988226  -0.3173633
## 5       -0.4891823               -0.2969835  -0.04444744  0.1441964  -0.3173633
## 6       -0.2868911               -0.5442099  -0.13139305 -0.3800157  -0.3173633
##   SpecialDay     Month OperatingSystems    Browser     Region TrafficType
## 1  -0.309001 -1.334201       -1.2332048 -0.7901988 -0.8941841 -0.76292777
## 2  -0.309001 -1.334201       -0.1361914 -0.2081361 -0.8941841 -0.51445574
## 3  -0.309001 -1.334201        2.0578354 -0.7901988  2.4360812 -0.26598370
## 4  -0.309001 -1.334201           0.9608220 -0.2081361 -0.4779009 -0.01751167
## 5  -0.309001 -1.334201           0.9608220  0.3739266 -0.8941841 -0.01751167
## 6  -0.309001 -1.334201          -0.1361914 -0.2081361 -0.8941841 -0.26598370
##   VisitorType    Weekend
## 1   0.4080401 -0.5505615
## 2   0.4080401 -0.5505615
## 3   0.4080401 -0.5505615
## 4   0.4080401 -0.5505615
## 5   0.4080401  1.8161802
## 6   0.4080401 -0.5505615
```

Applying DBSCAN Algorithm using 25 minimum points

```
# Applying our DBSCAN algorithm
# ---
# We want minimum 10 points with in a distance of eps(0.4)
#
db<-dbscan(Ecomm, eps=3, MinPts = 10)
```

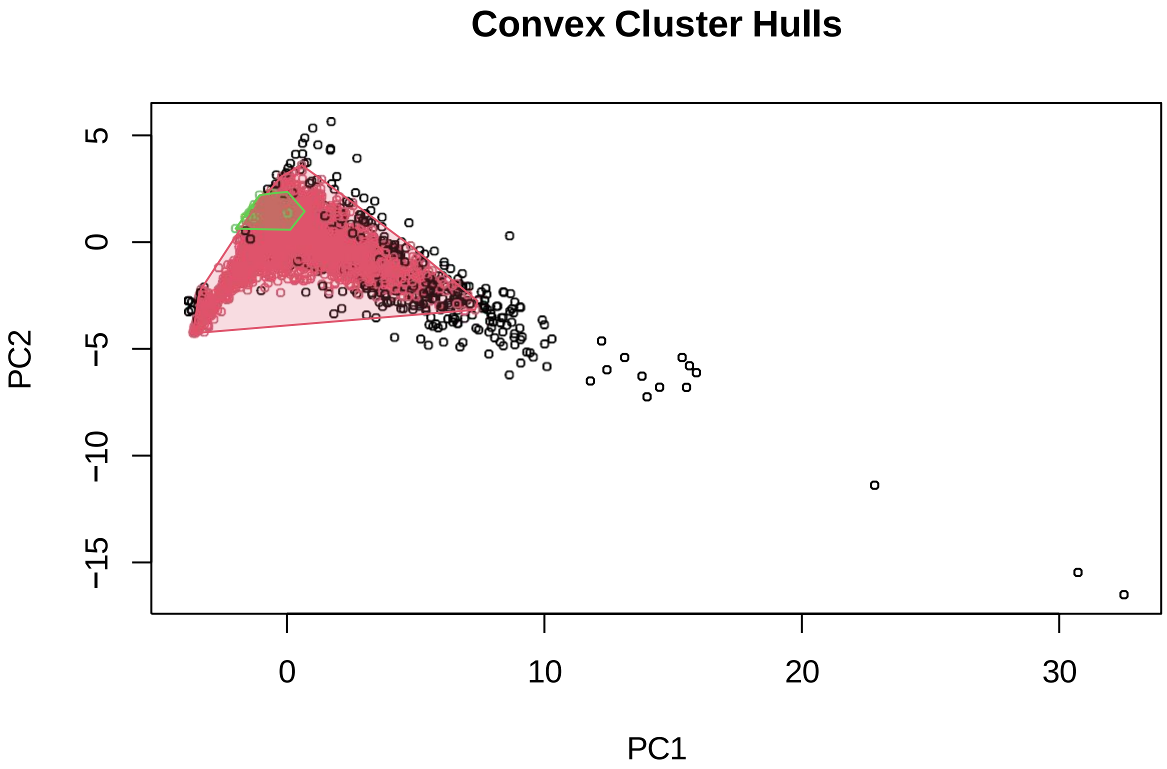
```
## Warning in dbscan(Ecomm, eps = 3, MinPts = 10): converting argument MinPts (fpc)
## to minPts (dbscan)!
```

```
# Printing out the clustering results
# ---
#
print(db)
```

```
## DBSCAN clustering for 12316 objects.
## Parameters: eps = 3, minPts = 10
## The clustering contains 2 cluster(s) and 455 noise points.
##
##     0     1     2
##   455 11825    36
##
## Available fields: cluster, eps, minPts
```

Plotting our clasters

```
# We also plot our clusters as shown
# ---
# The dataset and cluster method of dbscan is used to plot the clusters.
#
hullplot(Ecomm,db$cluster)
```


Convex Cluster Hulls
PC2
5
0
−5
−10
−15
PC1
0
10
20
30