Unsupervised_Learning_Part2

Obrein Telly

24/03/2022

Introduction & Overview 1. Defining the Problem

Question/ 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

2. Metric for Success

A model with accuracy of at least 95%

4Experimental Design 1. Problem Definition 2. Data Sourcing 3. Check the Data 4. Perform Data Cleaning 5. Perform Exploratory Data Analysis (Univariate & Bivariate) 6. Implement the Solution 7. Challenge the Solution 8. Follow up Questions

1 (a) Installing the Relevant Packages & Libraries

```
#install.packages(c("tidyverse", "ggplot2", "dplR","DataExplorer","mice","VIM","lubridate", "Hmisc","GG
 #                "moments", "ggcorrplot", "data.tables", "caret", "mlbench", "factoextra", "NbClust",
  #               "cluster", "dbscan", "fpc"))
```

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.0.5
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.8
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1
```

```
## Warning: package 'ggplot2' was built under R version 4.0.5
```

```
## Warning: package 'tibble' was built under R version 4.0.5
```

```
## Warning: package 'tidyr' was built under R version 4.0.5
```

```
## Warning: package 'readr' was built under R version 4.0.5
```

```
## Warning: package 'purrr' was built under R version 4.0.5

## Warning: package 'dplyr' was built under R version 4.0.5

## Warning: package 'stringr' was built under R version 4.0.5

## Warning: package 'forcats' was built under R version 4.0.5

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

```
library(DataExplorer)
```

```
## Warning: package 'DataExplorer' was built under R version 4.0.5
```

```
library(ggplot2)
library(mice)
```

```
## Warning: package 'mice' was built under R version 4.0.5

##
## Attaching package: 'mice'

## The following object is masked from 'package:stats':
##
##     filter

## The following objects are masked from 'package:base':
##
##     cbind, rbind
```

```
library(VIM)
```

```
## Warning: package 'VIM' was built under R version 4.0.5

## Loading required package: colorspace

## Warning: package 'colorspace' was built under R version 4.0.5

## Loading required package: grid

## VIM is ready to use.

## Suggestions and bug-reports can be submitted at: https://github.com/statistikat/VIM/issues

##
## Attaching package: 'VIM'

## The following object is masked from 'package:datasets':
##
##     sleep
```

```
library(lubridate)
```

```
## Warning: package 'lubridate' was built under R version 4.0.5
```

```
##
## Attaching package: 'lubridate'
```

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

```
library(Hmisc)
```

```
## Warning: package 'Hmisc' was built under R version 4.0.5
```

```
## Loading required package: lattice
```

```
## Loading required package: survival
```

```
## Loading required package: Formula
```

```
##
## Attaching package: 'Hmisc'
```

```
## The following objects are masked from 'package:dplyr':
##
##     src, summarize
```

```
## The following objects are masked from 'package:base':
##
##     format.pval, units
```

```
library(GGally)
```

```
## Warning: package 'GGally' was built under R version 4.0.5
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```
library(moments)
library(ggcorrplot)
```

```
## Warning: package 'ggcorrplot' was built under R version 4.0.5
```

```
library(data.table)
```

```
## Warning: package 'data.table' was built under R version 4.0.5
```

```
##
## Attaching package: 'data.table'

## The following objects are masked from 'package:lubridate':
##
##     hour, isoweek, mday, minute, month, quarter, second, wday, week,
##     yday, year

## The following objects are masked from 'package:dplyr':
##
##     between, first, last

## The following object is masked from 'package:purrr':
##
##     transpose
```

```
library(caret)
```

```
## Warning: package 'caret' was built under R version 4.0.5

##
## Attaching package: 'caret'

## The following object is masked from 'package:survival':
##
##     cluster

## The following object is masked from 'package:purrr':
##
##     lift
```

```
library(mlbench)
```

```
## Warning: package 'mlbench' was built under R version 4.0.5
```

```
library(factoextra)
```

```
## Warning: package 'factoextra' was built under R version 4.0.5

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

```
library(NbClust)
library(cluster)
```

```
## Warning: package 'cluster' was built under R version 4.0.5
```

```
library(dbscan)
```

```
## Warning: package 'dbscan' was built under R version 4.0.5
```

```
##
## Attaching package: 'dbscan'
```

```
## The following object is masked from 'package:VIM':
##
##     kNN
```

```
library(fpc)
```

```
## Warning: package 'fpc' was built under R version 4.0.5
```

```
##
## Attaching package: 'fpc'
```

```
## The following object is masked from 'package:dbscan':
##
##     dbscan
```

1 (b) Loading the Dataset

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

1 (c) Previewing the Dataset

```
#Checking the first 6 rows of the data
head(kira)
```

```
##   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
```

```
#Number of rows and columns in the DF using the dim() function

dim(kira)
```

```
## [1] 12330    18
```

Description of Columns

-The dataset consists of 10 numerical and 8 categorical columns/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.

```
#Checking the columns and their datatypes
str(kira)
```

```
## '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" "Returni
##  $ Weekend                : logi  FALSE FALSE FALSE FALSE TRUE FALSE ...
##  $ Revenue                : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
```

```
sapply(kira, 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"             "character"               "integer"
##                 Browser                  Region             TrafficType
##               "integer"               "integer"               "integer"
##             VisitorType                 Weekend                 Revenue
##             "character"               "logical"               "logical"
```

```
#

#Identifying Categorical columns
cat.col <- kira[, 10:18]

#Identifying NUmerical Columns
num.col <- kira[, 1:9]

#Checking unique values in the dataset
unique(kira$OperatingSystems)
```

```
## [1] 1 2 4 3 7 6 8 5
```

```
#Checking for Unique Values in Browser
unique(kira$Browser)
```

```
##  [1]  1  2  3  4  5  6  7 10  8  9 12 13 11
```

```
#Checking unique values in the Month column to ensure in line with expectations
unique(kira$Month)
```

```
##  [1] "Feb"  "Mar"  "May"  "Oct"  "June" "Jul"  "Aug"  "Nov"  "Sep"  "Dec"
```

```
#There are no anomalies in this column

#Unique values in Informational Column
unique(kira$Informational)
```

```
##  [1]  0  1  2  4 16  5  3 14  6 12  7 NA  9 10  8 11 24 13
```

```
#Values as expected

#Checking for unique values in the Administrative Column
unique(kira$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
```

```
#Missing entry in this column

#Checking for unique values in the Region column
unique(kira$Region)
```

```
## [1] 1 9 2 3 4 5 6 7 8
```

```
#Values as expected

#Checking for unique values in the Weekend Column
unique(kira$Weekend)
```

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

```
#Logical values as expected


#Checking for unique values in the visitor type
unique(kira$VisitorType)
```

```
## [1] "Returning_Visitor" "New_Visitor"       "Other"
```

```
#From the description we know that the Visitor Type is either returning or new. There
#is no provision for "Other". Therefore, we will drop all the columns with other.

#Checking for uniqueness in the Revenue attribute
unique(kira$Revenue)
```

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

```
#Values as expected
```

2 (a) Data Cleaning

```
#Renaming the Columns to ensure consistency

kira <- rename(kira, Admin = Administrative, Info = Informational, Info_Duration = Informational_Duratio
               Admin_Duration = Administrative_Duration, Prod_Related =  ProductRelated,
               Prod_Related_Duration =  ProductRelated_Duration,Bounce_Rates = BounceRates, Exit_Rates =
               Page_Values =  PageValues, Special_Day = SpecialDay,  Operating_Systems= OperatingSystems
               Traffic = TrafficType,Visitor = VisitorType)

#Checking the new columns
head(kira)
```

```
##   Admin Admin_Duration Info Info_Duration Prod_Related Prod_Related_Duration
## 1     0              0    0             0            1              0.000000
## 2     0              0    0             0            2             64.000000
## 3     0             -1    0            -1            1             -1.000000
## 4     0              0    0             0            2              2.666667
## 5     0              0    0             0           10            627.500000
## 6     0              0    0             0           19            154.216667
##   Bounce_Rates Exit_Rates Page_Values Special_Day Month Operating_Systems
## 1   0.20000000  0.2000000           0           0   Feb                  1
## 2   0.00000000  0.1000000           0           0   Feb                  2
## 3   0.20000000  0.2000000           0           0   Feb                  4
## 4   0.05000000  0.1400000           0           0   Feb                  3
## 5   0.02000000  0.0500000           0           0   Feb                  3
## 6   0.01578947  0.0245614           0           0   Feb                  2
##   Browser Region Traffic           Visitor Weekend Revenue
## 1       1      1       1 Returning_Visitor   FALSE   FALSE
## 2       2      1       2 Returning_Visitor   FALSE   FALSE
## 3       1      9       3 Returning_Visitor   FALSE   FALSE
## 4       2      2       4 Returning_Visitor   FALSE   FALSE
## 5       3      1       4 Returning_Visitor    TRUE   FALSE
## 6       2      1       3 Returning_Visitor   FALSE   FALSE
```

```
#Checking for duplicates in the dataset
anyDuplicated(kira)
```

```
## [1] 159
```

```
#There are 159 duplicated rows in the dataset

#Dropping the duplicated rows
kira <- kira[!duplicated(kira), ]


#Confirming there are no duplicates in the dataset
any(duplicated(kira))
```

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

```r
#No duplicated rows in the dataset


#Dropping all the columns with other in the Visitor Dataset

sum(kira$Visitor == 'Other')
```

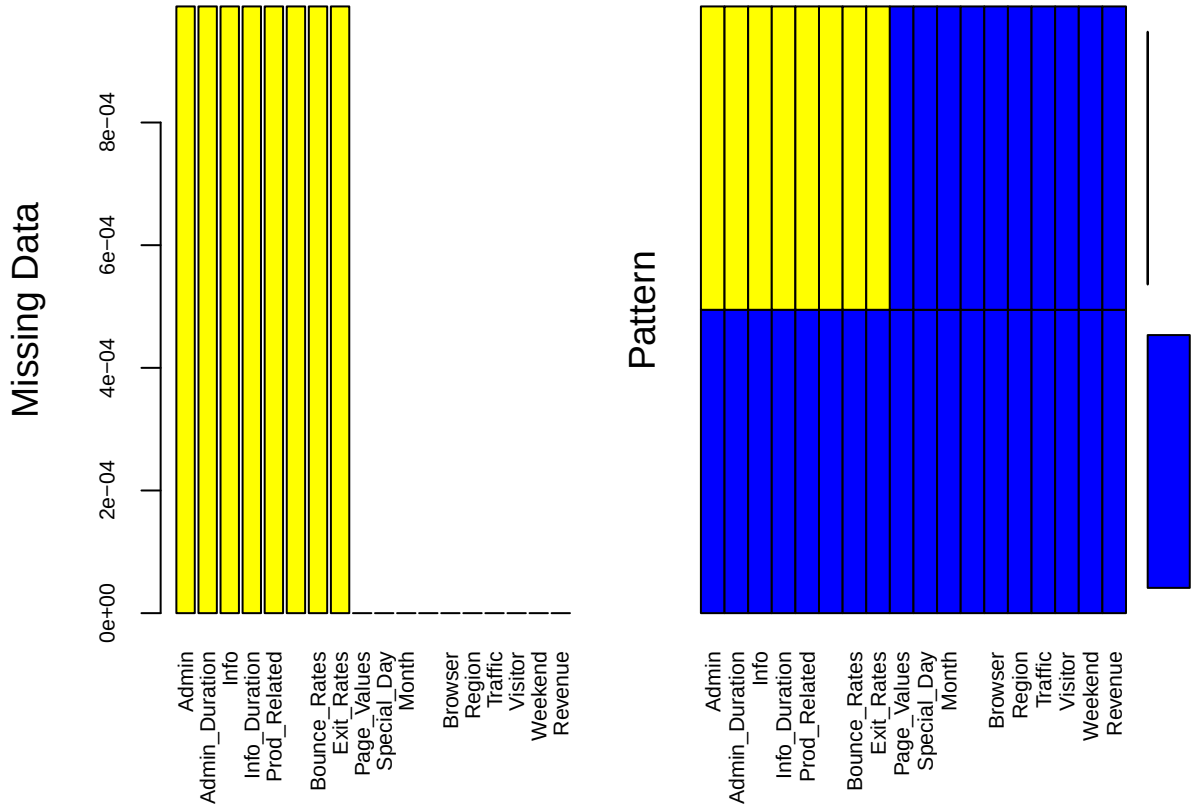
```
## [1] 81
```

```r
#There are 85 Rows with type 'Other'


kira <- kira[!(kira$Visitor == "Other"), ]
# Checking for Missing Values

unique(kira$Visitor)
```

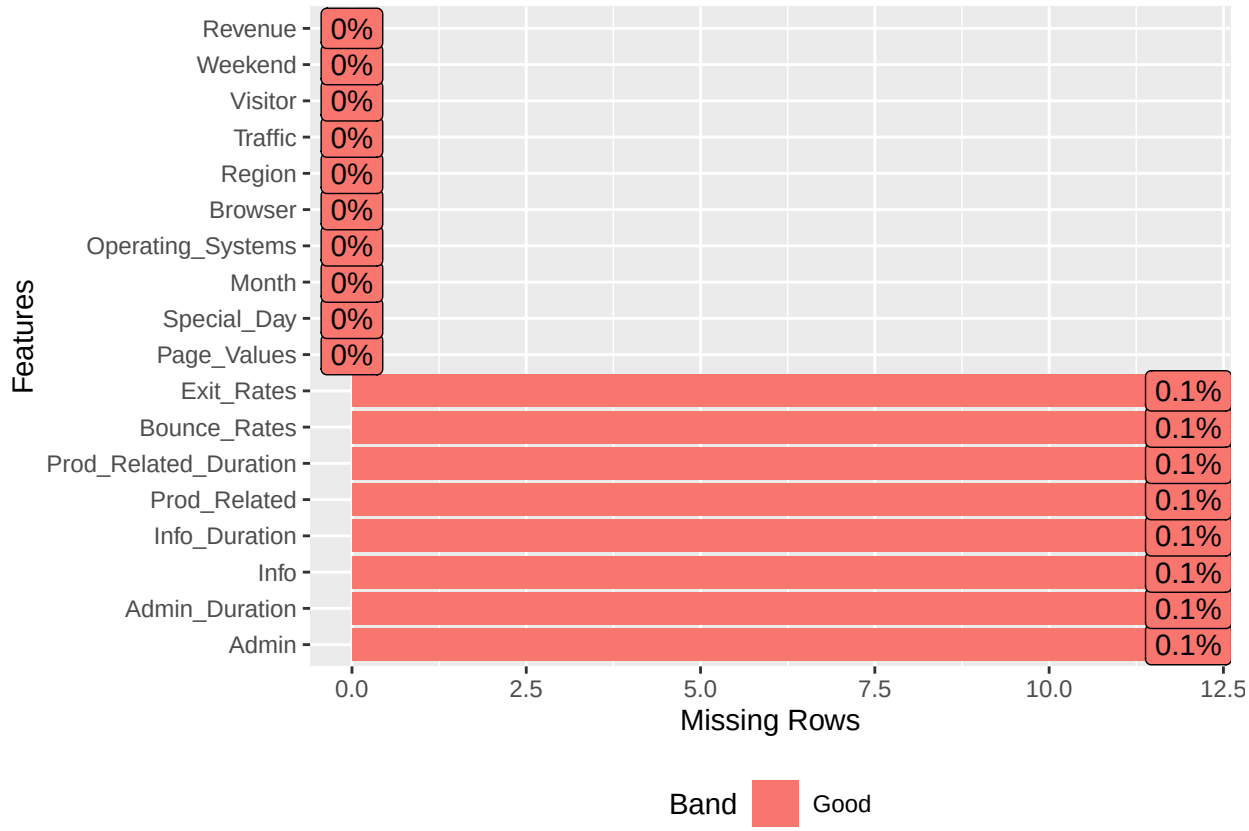
```
## [1] "Returning_Visitor" "New_Visitor"
```

```r
#Visualizing Pattern of missing values

mp <- aggr(kira, col=c('blue', 'yellow'),
           numbers=TRUE, sortvars=TRUE, labels=names(kira), cex.axis =.7, gap=3,
           ylab=c("Missing Data", "Pattern"))
```

```
## Warning in plot.aggr(res, ...): not enough horizontal space to display
## frequencies
```




```
#Plot and Distribution of Missing values
plot_missing(kira)
```




```
#Only ExitRates, BounceRates, ProductRelated_Duration, ProductRelated, information_related,
#informational, administrative_duration & administrative have missing values and about 0.1%
#of the data per column

#We observe the percentage of missing values is not substantial to impact the
#analysis of the dataset should we decide to omit the missing values.

#Dropping the Missing Data using the na.omit() function

kira <- na.omit(kira)

#Confirming there are no missing values

any(is.na(kira))
```

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

```
#There are no missing values

head(kira)
```

```
##   Admin Admin_Duration Info Info_Duration Prod_Related Prod_Related_Duration
## 1     0              0    0             0            1              0.000000
## 2     0              0    0             0            2             64.000000
## 3     0             -1    0            -1            1             -1.000000
```

```
## 4     0              0    0             0            2              2.666667
## 5     0              0    0             0           10            627.500000
## 6     0              0    0             0           19            154.216667
##   Bounce_Rates Exit_Rates Page_Values Special_Day Month Operating_Systems
## 1   0.20000000  0.2000000           0           0   Feb                  1
## 2   0.00000000  0.1000000           0           0   Feb                  2
## 3   0.20000000  0.2000000           0           0   Feb                  4
## 4   0.05000000  0.1400000           0           0   Feb                  3
## 5   0.02000000  0.0500000           0           0   Feb                  3
## 6   0.01578947  0.0245614           0           0   Feb                  2
##   Browser Region Traffic           Visitor Weekend Revenue
## 1       1      1       1 Returning_Visitor   FALSE   FALSE
## 2       2      1       2 Returning_Visitor   FALSE   FALSE
## 3       1      9       3 Returning_Visitor   FALSE   FALSE
## 4       2      2       4 Returning_Visitor   FALSE   FALSE
## 5       3      1       4 Returning_Visitor    TRUE   FALSE
## 6       2      1       3 Returning_Visitor   FALSE   FALSE
```

```
dim(kira)
```

```
## [1] 12118    18
```

```
str(kira)
```

```
## 'data.frame':    12118 obs. of  18 variables:
##  $ Admin                : int  0 0 0 0 0 0 0 1 0 0 ...
##  $ Admin_Duration       : num  0 0 -1 0 0 0 -1 -1 0 0 ...
##  $ Info                 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Info_Duration        : num  0 0 -1 0 0 0 -1 -1 0 0 ...
##  $ Prod_Related         : int  1 2 1 2 10 19 1 1 2 3 ...
##  $ Prod_Related_Duration: num  0 64 -1 2.67 627.5 ...
##  $ Bounce_Rates         : num  0.2 0 0.2 0.05 0.02 ...
##  $ Exit_Rates           : num  0.2 0.1 0.2 0.14 0.05 ...
##  $ Page_Values          : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Special_Day          : num  0 0 0 0 0 0 0.4 0 0.8 0.4 ...
##  $ Month                : chr  "Feb" "Feb" "Feb" "Feb" ...
##  $ Operating_Systems    : 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 ...
##  $ Traffic              : int  1 2 3 4 4 3 3 5 3 2 ...
##  $ Visitor              : chr  "Returning_Visitor" "Returning_Visitor" "Returning_Visitor" "Returnin
##  $ Weekend              : logi  FALSE FALSE FALSE FALSE TRUE FALSE ...
##  $ Revenue              : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
##  - attr(*, "na.action")= 'omit' Named int [1:12] 1050 1116 1117 1118 1119 1443 1444 1445 1446 1996 .
##   ..- attr(*, "names")= chr [1:12] "1066" "1133" "1134" "1135" ...
```

```
#Numerical variables in the Dataset
numeric <- select(kira, 1:9)
str(numeric)
```

```
## 'data.frame':    12118 obs. of  9 variables:
##  $ Admin                : int  0 0 0 0 0 0 0 1 0 0 ...
```

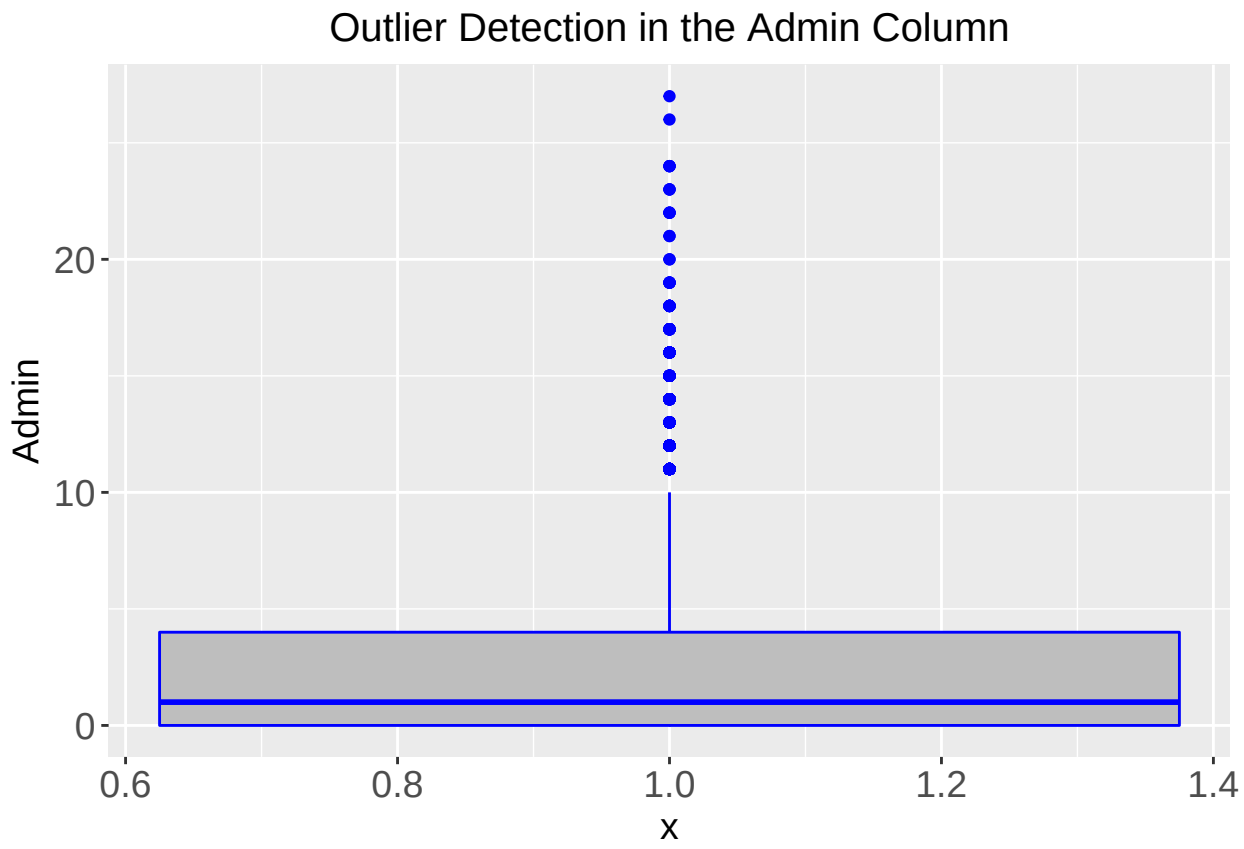
```
##  $ Admin_Duration       : num  0 0 -1 0 0 0 -1 -1 0 0 ...
##  $ Info                 : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Info_Duration        : num  0 0 -1 0 0 0 -1 -1 0 0 ...
##  $ Prod_Related         : int  1 2 1 2 10 19 1 1 2 3 ...
##  $ Prod_Related_Duration: num  0 64 -1 2.67 627.5 ...
##  $ Bounce_Rates         : num  0.2 0 0.2 0.05 0.02 ...
##  $ Exit_Rates           : num  0.2 0.1 0.2 0.14 0.05 ...
##  $ Page_Values          : num  0 0 0 0 0 0 0 0 0 0 ...
##  - attr(*, "na.action")= 'omit' Named int [1:12] 1050 1116 1117 1118 1119 1443 1444 1445 1446 1996 .
##   ..- attr(*, "names")= chr [1:12] "1066" "1133" "1134" "1135" ...
```

```
#Categorical Variables in the Dataset

non.numeric <- select(kira, 10:18)

str(non.numeric)
```

```
## 'data.frame':    12118 obs. of  9 variables:
##  $ Special_Day      : num  0 0 0 0 0 0 0.4 0 0.8 0.4 ...
##  $ Month            : chr  "Feb" "Feb" "Feb" "Feb" ...
##  $ Operating_Systems: 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 ...
##  $ Traffic          : int  1 2 3 4 4 3 3 5 3 2 ...
##  $ Visitor          : chr  "Returning_Visitor" "Returning_Visitor" "Returning_Visitor" "Returning_Vis
##  $ Weekend          : logi  FALSE FALSE FALSE FALSE TRUE FALSE ...
##  $ Revenue          : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
##  - attr(*, "na.action")= 'omit' Named int [1:12] 1050 1116 1117 1118 1119 1443 1444 1445 1446 1996 .
##   ..- attr(*, "names")= chr [1:12] "1066" "1133" "1134" "1135" ...
```

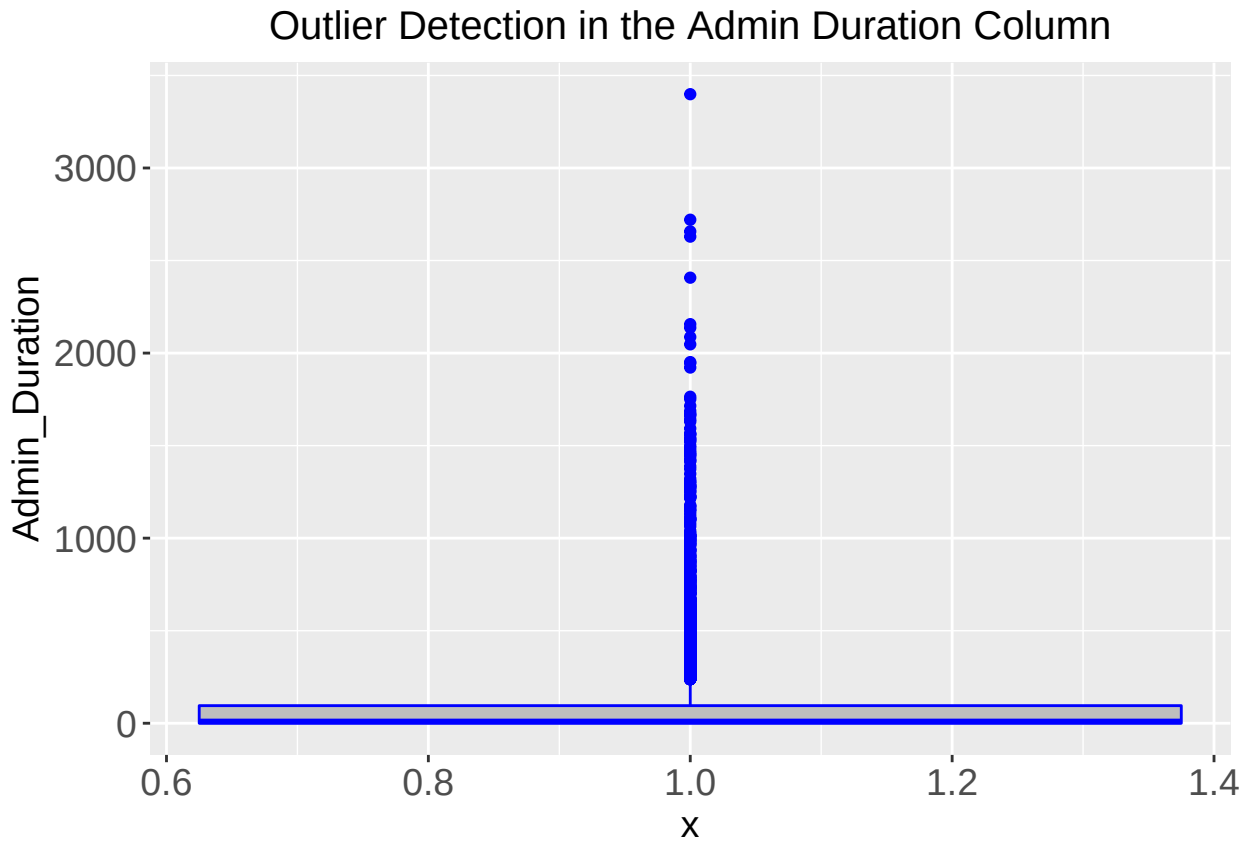
2 (b)Outlier Detection

```
#Checking for Outliers in the Admin Column
kira %>%
  ggplot(aes(x= 1, y=Admin)) +
  geom_boxplot(fill = "grey", color= 'blue') +
  ggtitle("Outlier Detection in the Admin Column") +
  theme(axis.text = element_text(size=14),
        axis.title = element_text(size = 14),
        plot.title = element_text(hjust = 0.5, size = 15))
```



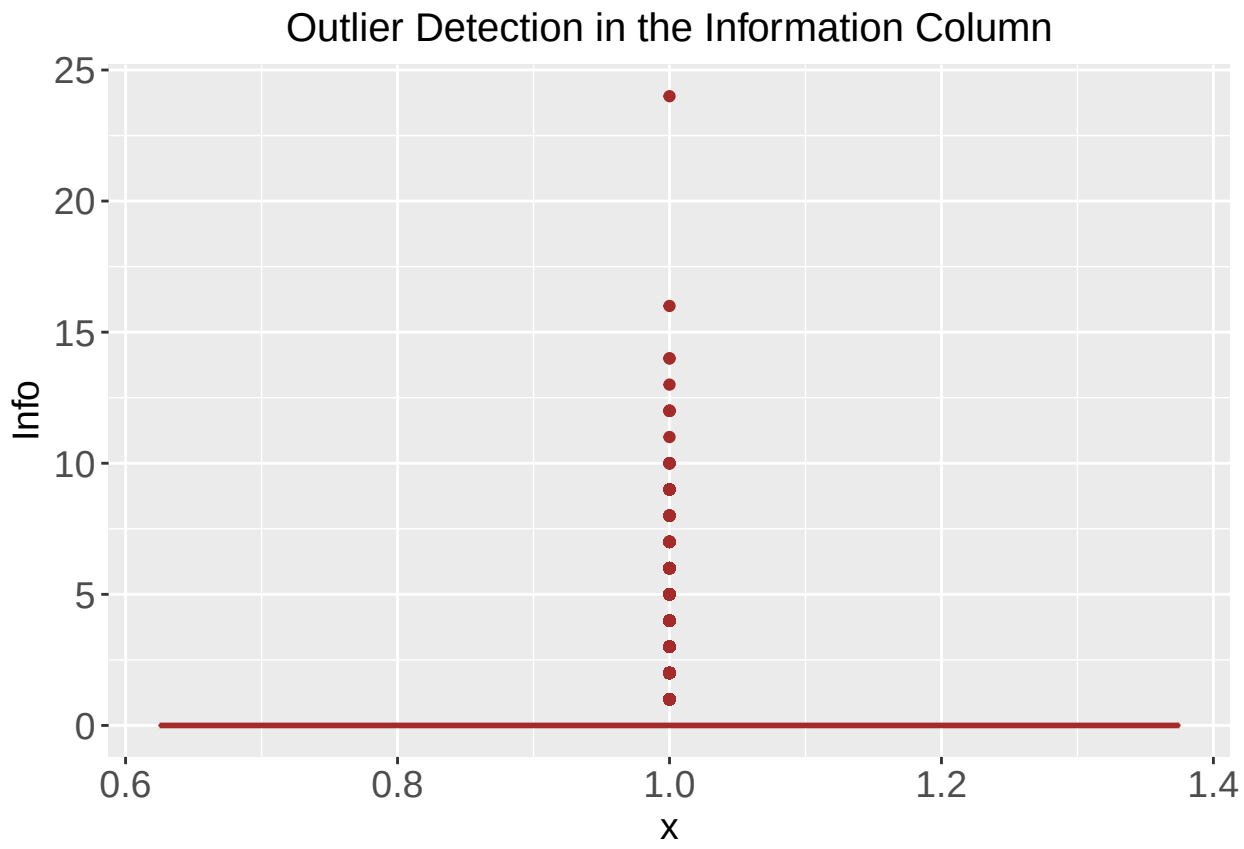

```
#There are outliers in the Admin column
```

```
#Checking for Outliers in the Admin Duration Column
kira %>%
  ggplot(aes(x= 1, y=Admin_Duration)) +
  geom_boxplot(fill = "grey", color= 'blue') +
  ggtitle("Outlier Detection in the Admin Duration Column") +
  theme(axis.text = element_text(size=14),
        axis.title = element_text(size = 14),
        plot.title = element_text(hjust = 0.5, size = 15))
```



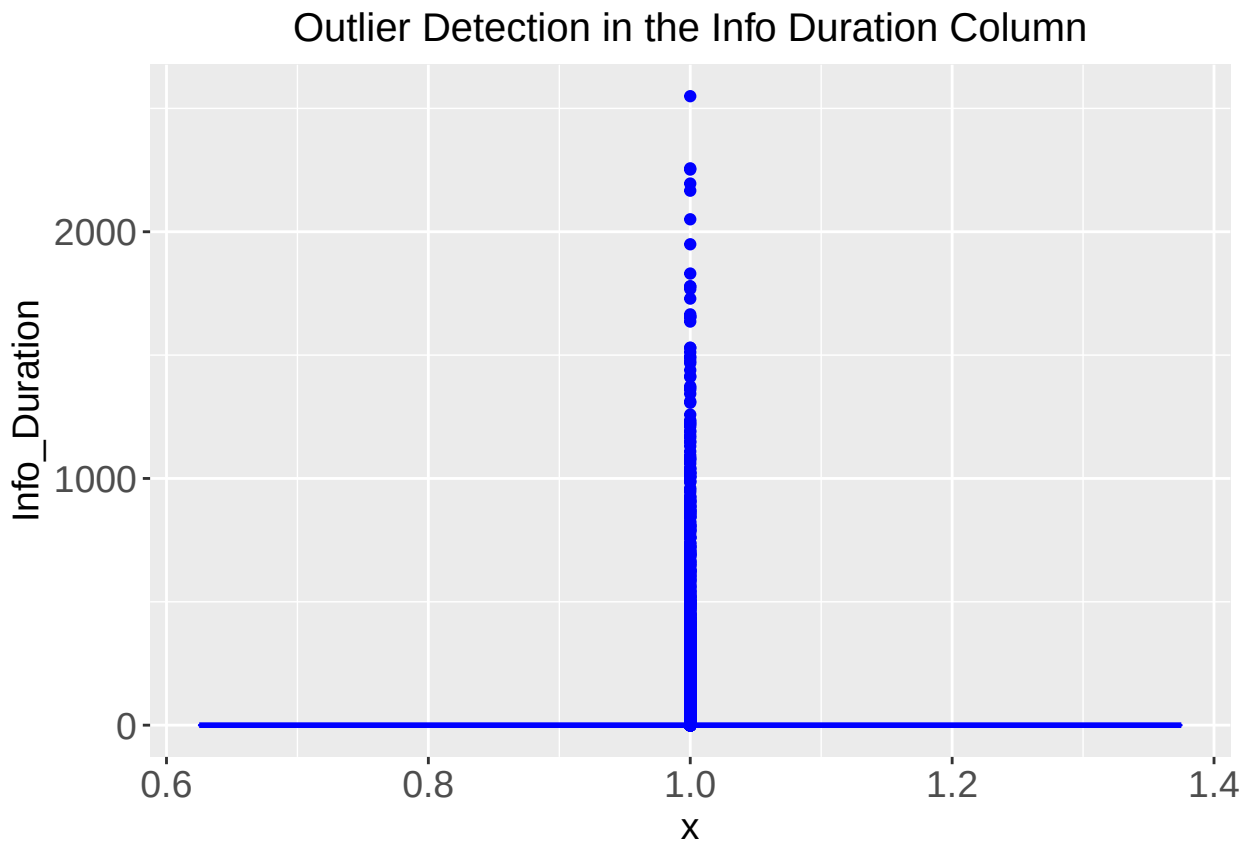

```
#We also have outliers in this column
```

```
#Checking for Outliers in the Info Column
kira %>%
  ggplot(aes(x= 1, y=Info)) +
  geom_boxplot(fill = "grey", color= 'brown') +
  ggtitle("Outlier Detection in the Information Column") +
  theme(axis.text = element_text(size=14),
        axis.title = element_text(size = 14),
        plot.title = element_text(hjust = 0.5, size = 15))
```

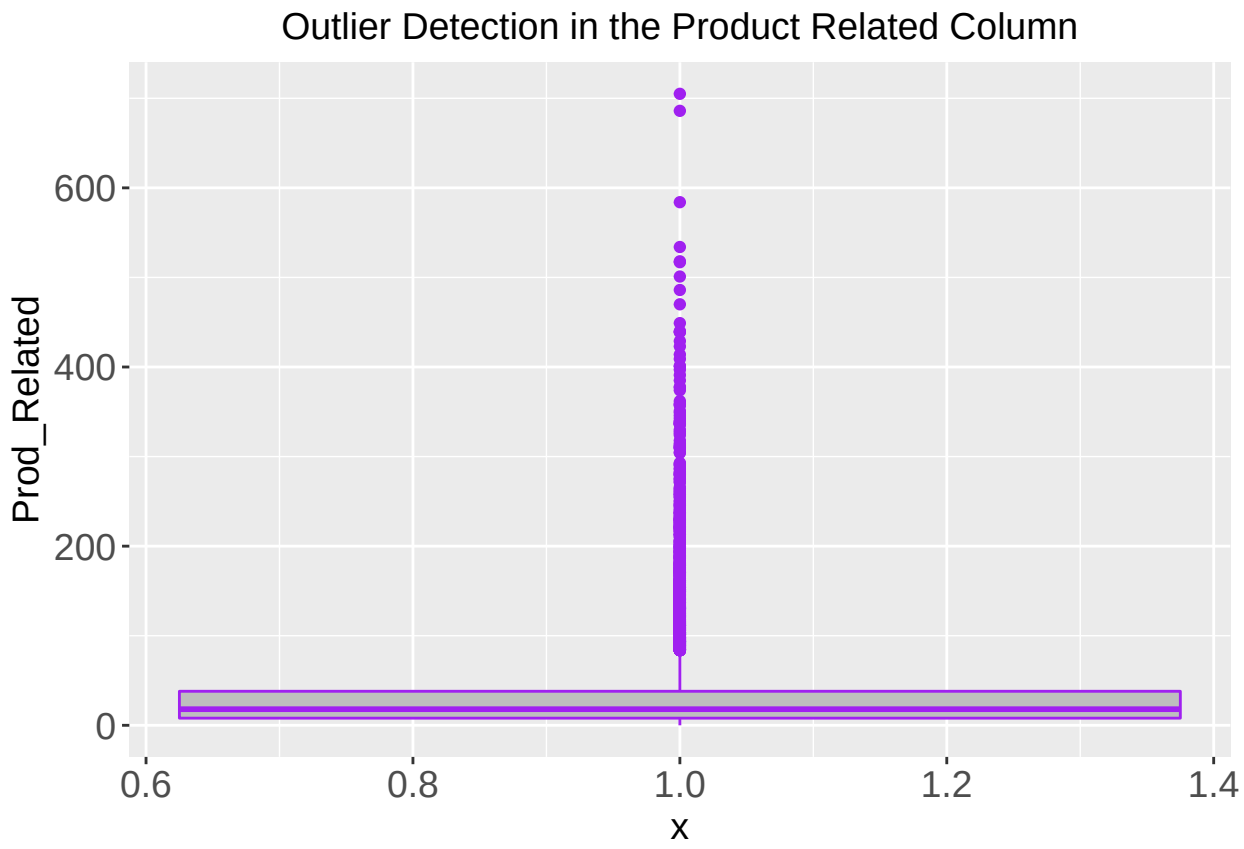



```
# We have Outliers in the Information Column as well
```

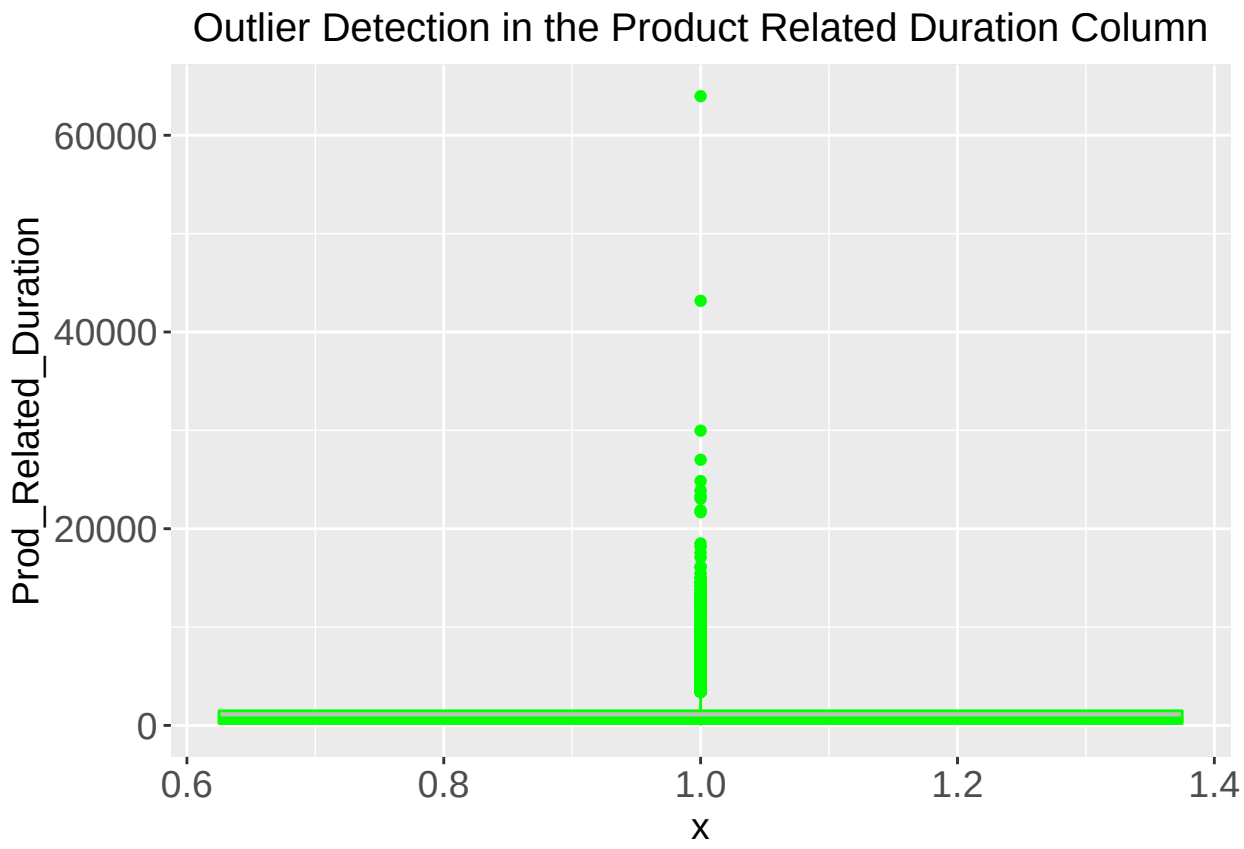
```
#Checking for Outliers in the Information Duration Column
kira %>%
  ggplot(aes(x= 1, y= Info_Duration)) +
  geom_boxplot(fill = "grey", color= 'blue') +
  ggtitle("Outlier Detection in the Info Duration Column") +
  theme(axis.text = element_text(size=14),
        axis.title = element_text(size = 14),
        plot.title = element_text(hjust = 0.5, size = 15))
```



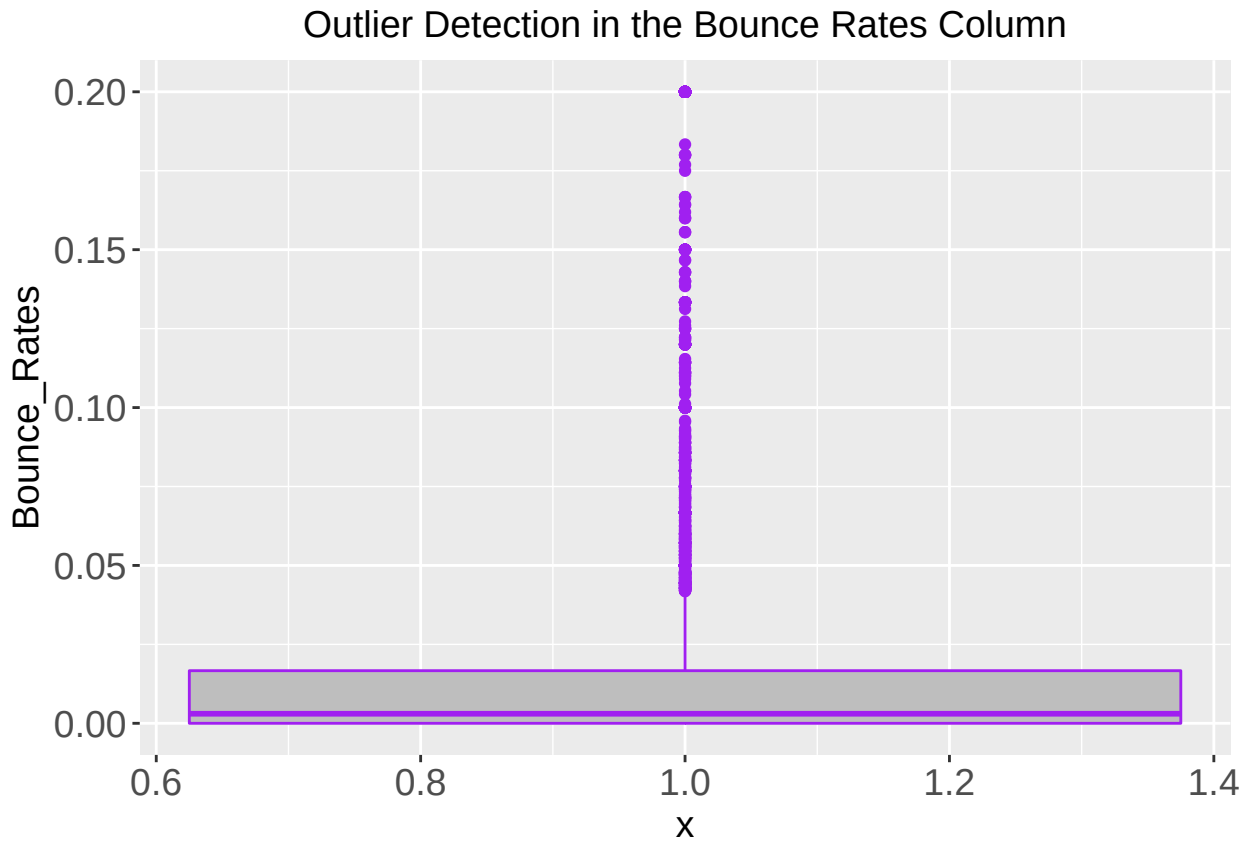

```
#We also have outliers in this column

#Checking for Outliers in the Product Related Column
kira %>%
  ggplot(aes(x= 1, y= Prod_Related)) +
  geom_boxplot(fill = "grey", color= 'purple') +
  ggtitle("Outlier Detection in the Product Related Column") +
  theme(axis.text = element_text(size=14),
        axis.title = element_text(size = 14),
        plot.title = element_text(hjust = 0.5, size = 14))
```



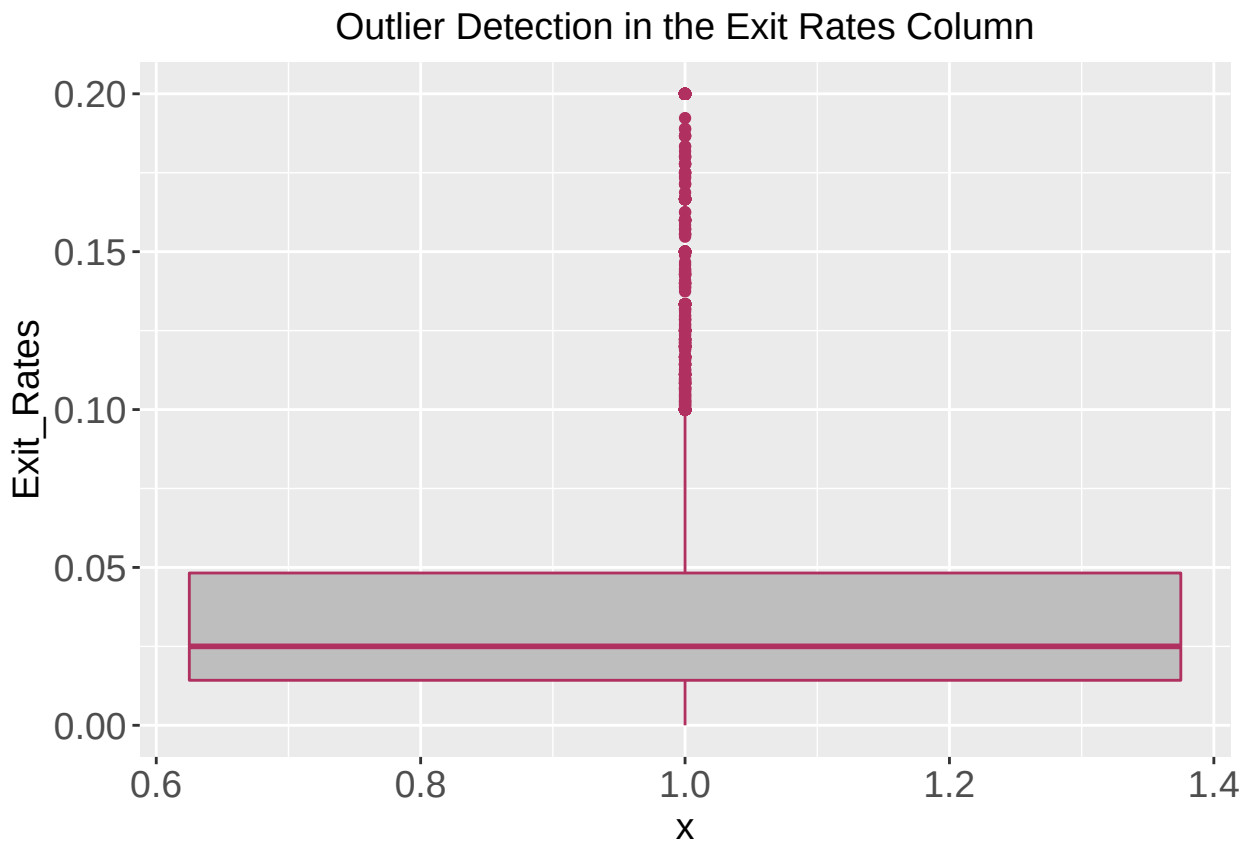

```
#We also have outliers in this column

#Checking for Outliers in the Product Related Duration Column
kira %>%
  ggplot(aes(x= 1, y= Prod_Related_Duration)) +
  geom_boxplot(fill = "grey", color= 'green') +
  ggtitle("Outlier Detection in the Product Related Duration Column") +
  theme(axis.text = element_text(size=14),
        axis.title = element_text(size = 14),
        plot.title = element_text(hjust = 0.5, size = 15))
```



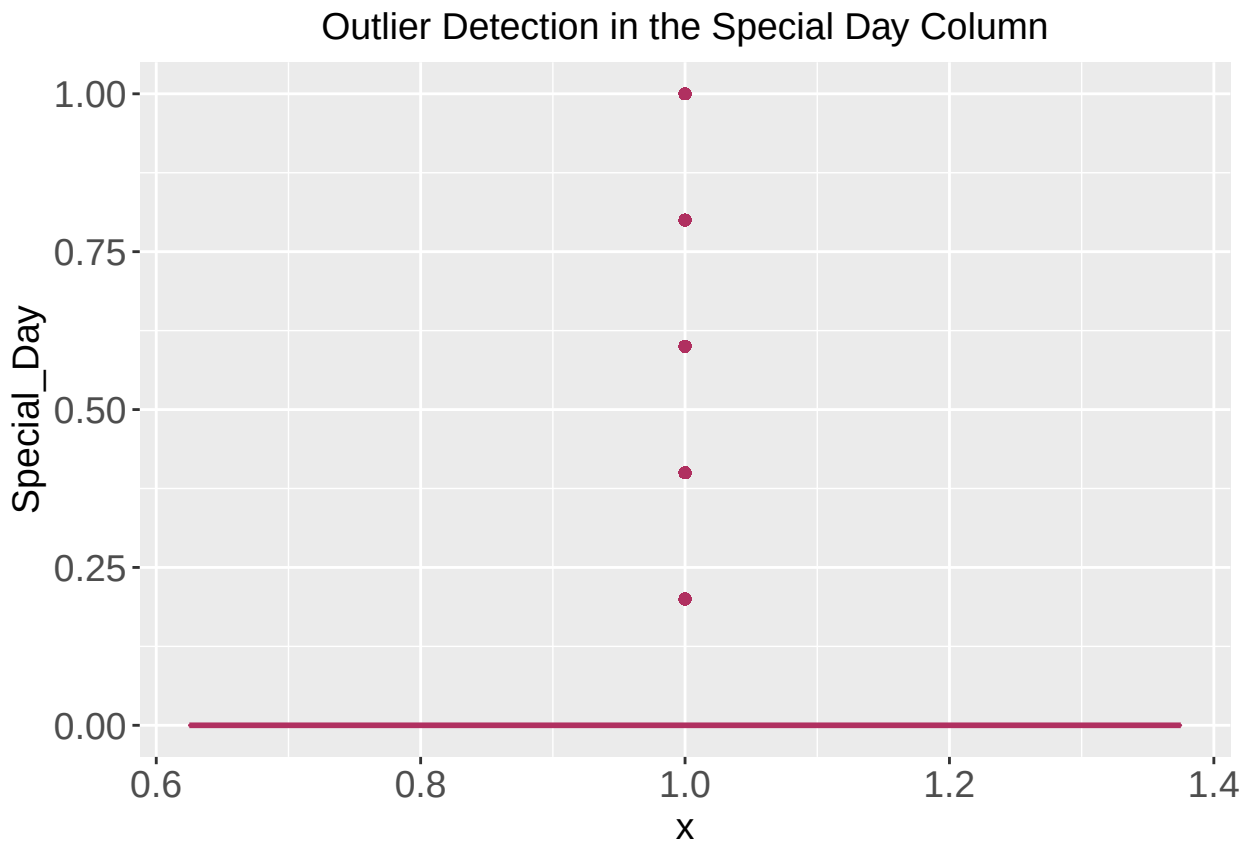

```
#We also have outliers in this column

#Checking for Outliers in the Bounce Rates Column
kira %>%
  ggplot(aes(x= 1, y= Bounce_Rates)) +
  geom_boxplot(fill = "grey", color= 'purple') +
  ggtitle("Outlier Detection in the Bounce Rates Column") +
  theme(axis.text = element_text(size=14),
        axis.title = element_text(size = 14),
        plot.title = element_text(hjust = 0.5, size = 14))
```



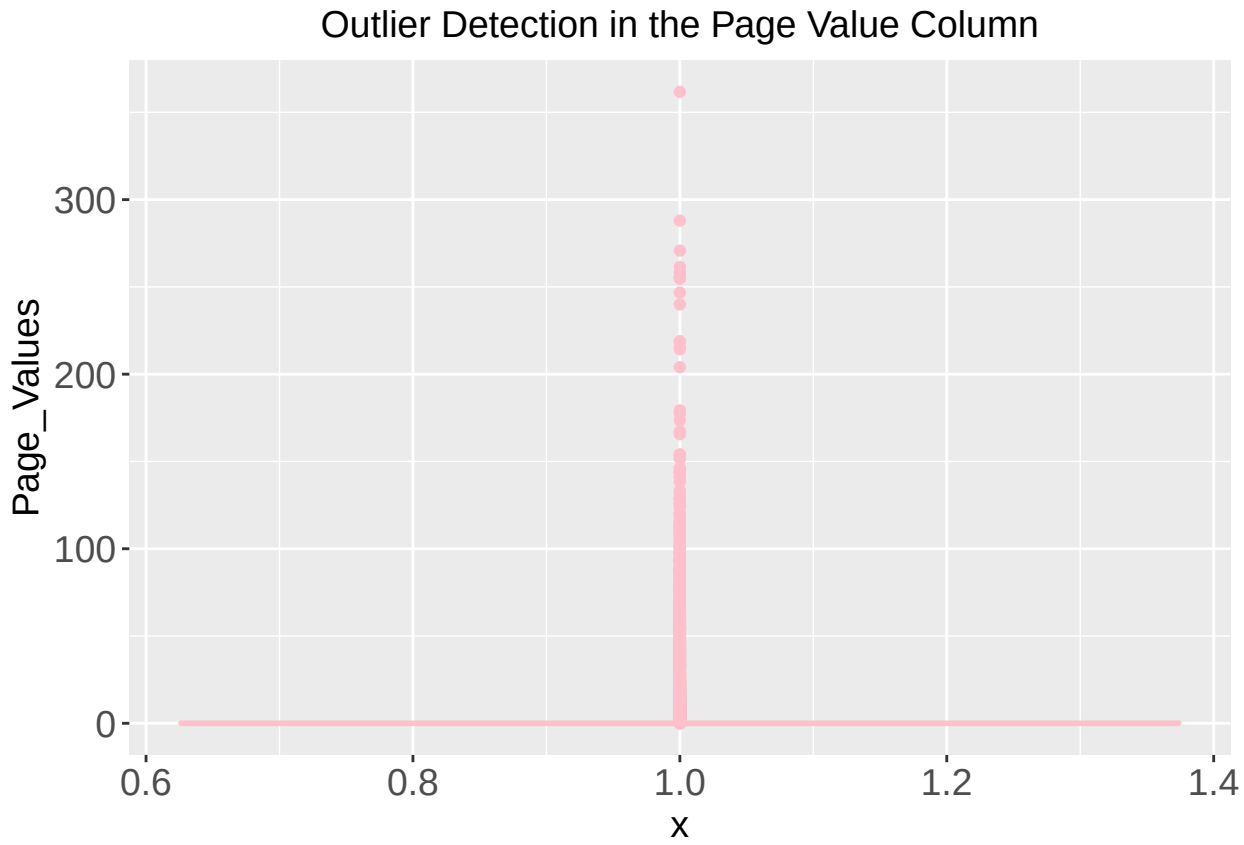

```
#We also have outliers in this column

#Checking for Outliers in the Exit Rates Column
kira %>%
  ggplot(aes(x= 1, y= Exit_Rates)) +
  geom_boxplot(fill = "grey", color= 'maroon') +
  ggtitle("Outlier Detection in the Exit Rates Column") +
  theme(axis.text = element_text(size=14),
        axis.title = element_text(size = 14),
        plot.title = element_text(hjust = 0.5, size = 14))
```




```
#There are outliers in this column

#Checking for Outliers in the Special Day  Column
kira %>%
  ggplot(aes(x= 1, y= Special_Day)) +
  geom_boxplot(fill = "grey", color= 'maroon') +
  ggtitle("Outlier Detection in the Special Day Column") +
  theme(axis.text = element_text(size=14),
        axis.title = element_text(size = 14),
        plot.title = element_text(hjust = 0.5, size = 14))
```




```
#We also have outliers in this column


#Checking for Outliers in the Page Values Column
kira %>%
  ggplot(aes(x= 1, y= Page_Values)) +
  geom_boxplot(fill = "grey", color= 'pink') +
  ggtitle("Outlier Detection in the Page Value Column") +
  theme(axis.text = element_text(size=14),
        axis.title = element_text(size = 14),
        plot.title = element_text(hjust = 0.5, size = 14))
```




```
#Every Numeric Column has outliers. We'll keep them in the dataset for now as they
#represent actual data points.
```

```
#Ascribing the correct Data type to the Some of the Categorical Variables

kira <- transform( kira, Operating_Systems = as.factor(Operating_Systems), Browser =
                    as.factor(Browser), Region = as.factor(Region), Traffic = as.factor(Traffic))
```

3. EXPLORATORY DATA ANALYSIS

a) Univariate Analysis

```
#Measures of Central Tendency and measures of dispersion of the numerical attributes

central.tendecy <- data.frame(
  Mean <- apply(numeric, 2, mean),
  Median <- apply(numeric, 2, median),
  Mode <- apply(numeric, 2, mode),
  Min <- apply(numeric, 2, min),
  Max <- apply(numeric, 2, max),
  Variance <- apply(numeric, 2, var),
  Std <- apply(numeric, 2, sd),
  Skewness <- apply(numeric, 2, skewness),
  Kurtosis <- apply(numeric, 2, kurtosis))
```

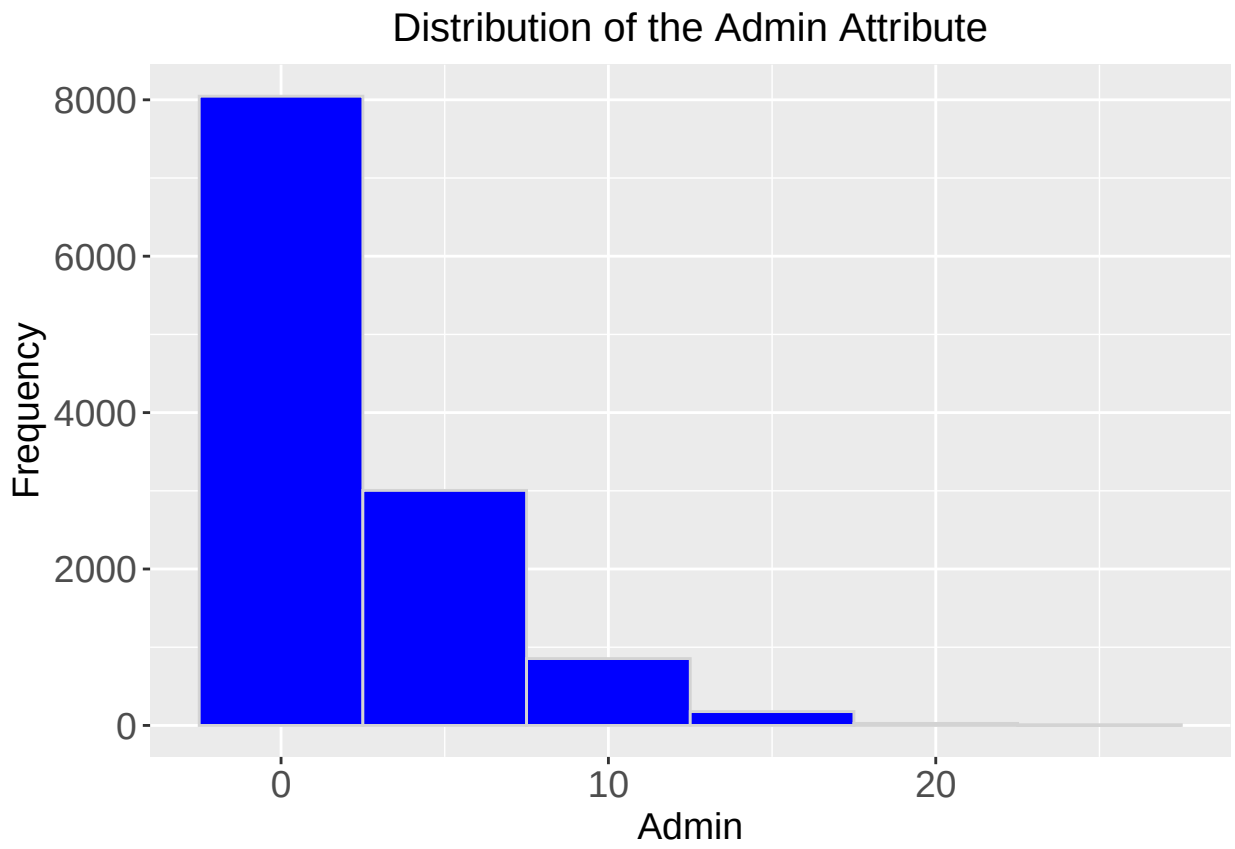
```
print(central.tendecy)
```

```
##                         Mean....apply.numeric..2..mean.
## Admin                                      2.345354e+00
## Admin_Duration                             8.178836e+01
## Info                                       5.109754e-01
## Info_Duration                              3.498823e+01
## Prod_Related                               3.218559e+01
## Prod_Related_Duration                      1.211578e+03
## Bounce_Rates                               2.037901e-02
## Exit_Rates                                 4.139583e-02
## Page_Values                                5.864684e+00
##                         Median....apply.numeric..2..median.
## Admin                                         1.000000e+00
## Admin_Duration                                9.500000e+00
## Info                                          0.000000e+00
## Info_Duration                                 0.000000e+00
## Prod_Related                                  1.800000e+01
## Prod_Related_Duration                         6.146955e+02
## Bounce_Rates                                  3.030303e-03
## Exit_Rates                                    2.500000e-02
## Page_Values                                   0.000000e+00
##                         Mode....apply.numeric..2..mode.
## Admin                                            numeric
## Admin_Duration                                   numeric
## Info                                             numeric
## Info_Duration                                    numeric
## Prod_Related                                     numeric
## Prod_Related_Duration                            numeric
## Bounce_Rates                                     numeric
## Exit_Rates                                       numeric
## Page_Values                                      numeric
##                         Min....apply.numeric..2..min.
## Admin                                               0
## Admin_Duration                                     -1
## Info                                                0
## Info_Duration                                      -1
## Prod_Related                                        0
## Prod_Related_Duration                              -1
## Bounce_Rates                                        0
## Exit_Rates                                          0
## Page_Values                                         0
##                         Max....apply.numeric..2..max.
## Admin                                         27.0000
## Admin_Duration                              3398.7500
## Info                                          24.0000
## Info_Duration                               2549.3750
## Prod_Related                                 705.0000
## Prod_Related_Duration                      63973.5222
## Bounce_Rates                                   0.2000
## Exit_Rates                                     0.2000
## Page_Values                                  361.7637
```

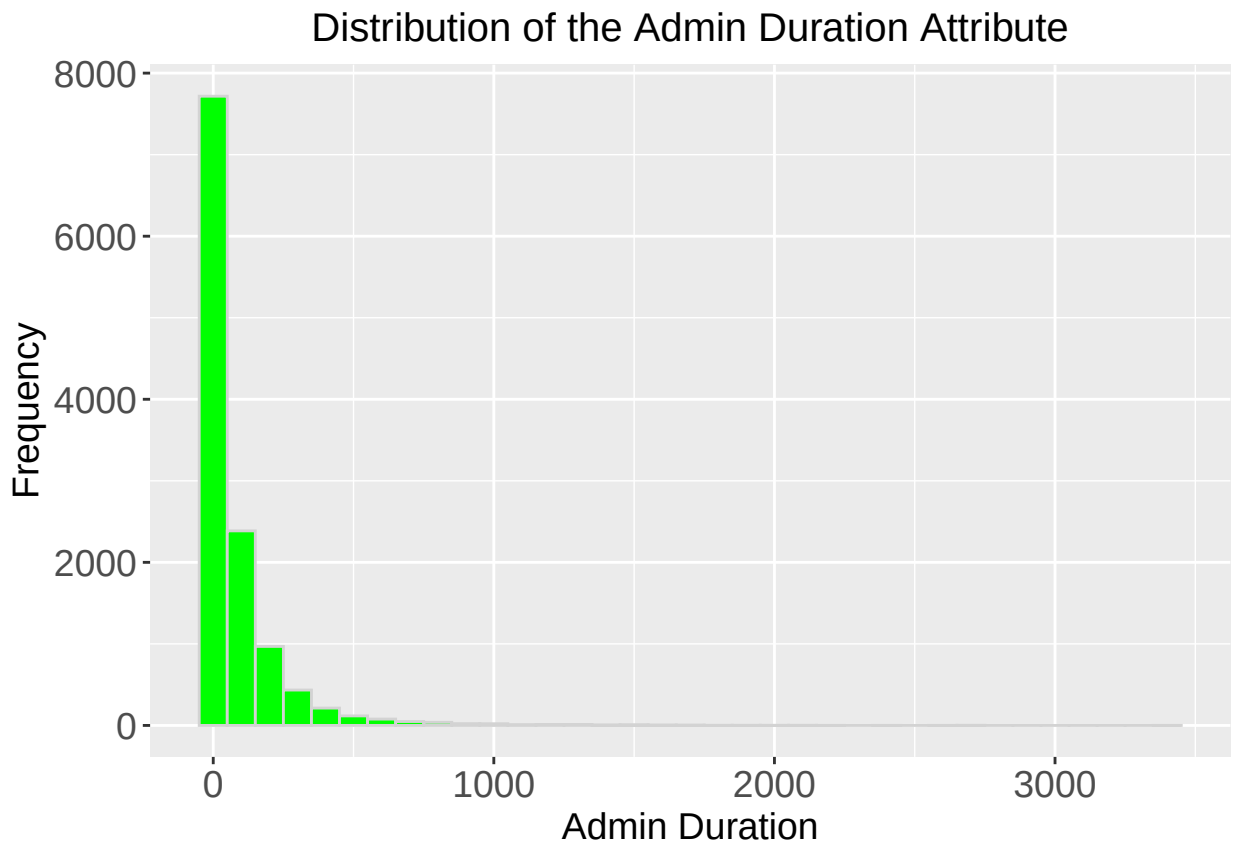
```
##                         Variance....apply.numeric..2..var.
## Admin                                         1.112154e+01
## Admin_Duration                                3.143836e+04
## Info                                          1.634566e+00
## Info_Duration                                 2.011611e+04
## Prod_Related                                  1.995423e+03
## Prod_Related_Duration                         3.697301e+06
## Bounce_Rates                                  2.040854e-03
## Exit_Rates                                    2.122975e-03
## Page_Values                                   3.297918e+02
##                         Std....apply.numeric..2..sd.
## Admin                                    3.334897e+00
## Admin_Duration                           1.773087e+02
## Info                                     1.278502e+00
## Info_Duration                            1.418313e+02
## Prod_Related                             4.467016e+01
## Prod_Related_Duration                    1.922837e+03
## Bounce_Rates                             4.517581e-02
## Exit_Rates                               4.607575e-02
## Page_Values                              1.816017e+01
##                         Skewness....apply.numeric..2..skewness.
## Admin                                                 1.943782
## Admin_Duration                                        5.584833
## Info                                                  4.006558
## Info_Duration                                         7.522768
## Prod_Related                                          4.327166
## Prod_Related_Duration                                 7.253780
## Bounce_Rates                                          3.164762
## Exit_Rates                                            2.239481
## Page_Values                                           6.085829
##                         Kurtosis....apply.numeric..2..kurtosis.
## Admin                                                 7.622079
## Admin_Duration                                       53.176584
## Info                                                 29.554109
## Info_Duration                                        78.124117
## Prod_Related                                         33.987478
## Prod_Related_Duration                               139.582228
## Bounce_Rates                                         12.358046
## Exit_Rates                                            7.668824
## Page_Values                                          61.931671
```

Univariate Analysis for numerical variables in the dataset

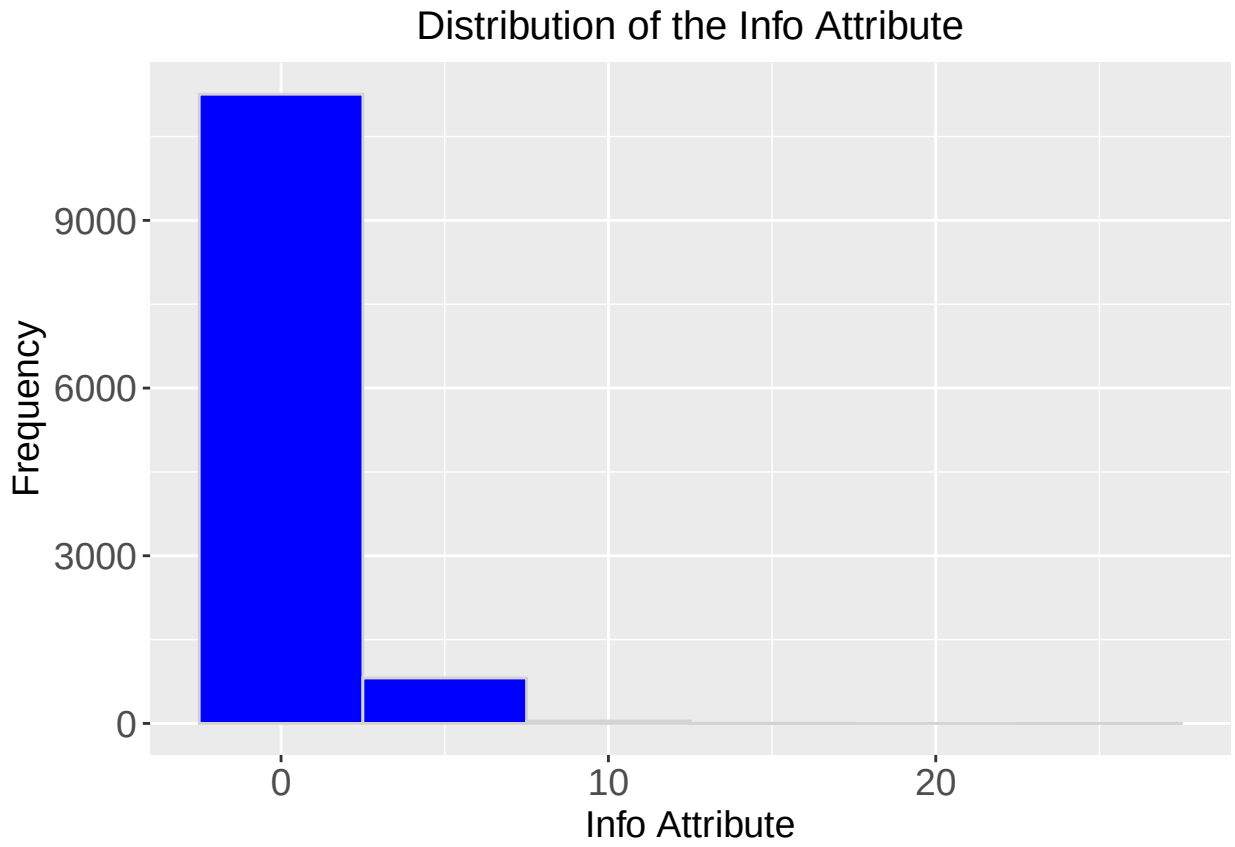
```
# Histogram to visualize the distribution of values in the admin column

options(repr.plot.width = 8, repr.plot.height = 6)
p <- kira %>% ggplot(aes(x = Admin ))

p + geom_histogram(color="lightgray", fill="blue", binwidth = 5) +
  labs(title = "Distribution of the Admin Attribute", x = "Admin", y = "Frequency") +
  theme(axis.title = element_text(size = 14),
        axis.text = element_text(size=14),
        plot.title = element_text(hjust = 0.5, size = 15))
```



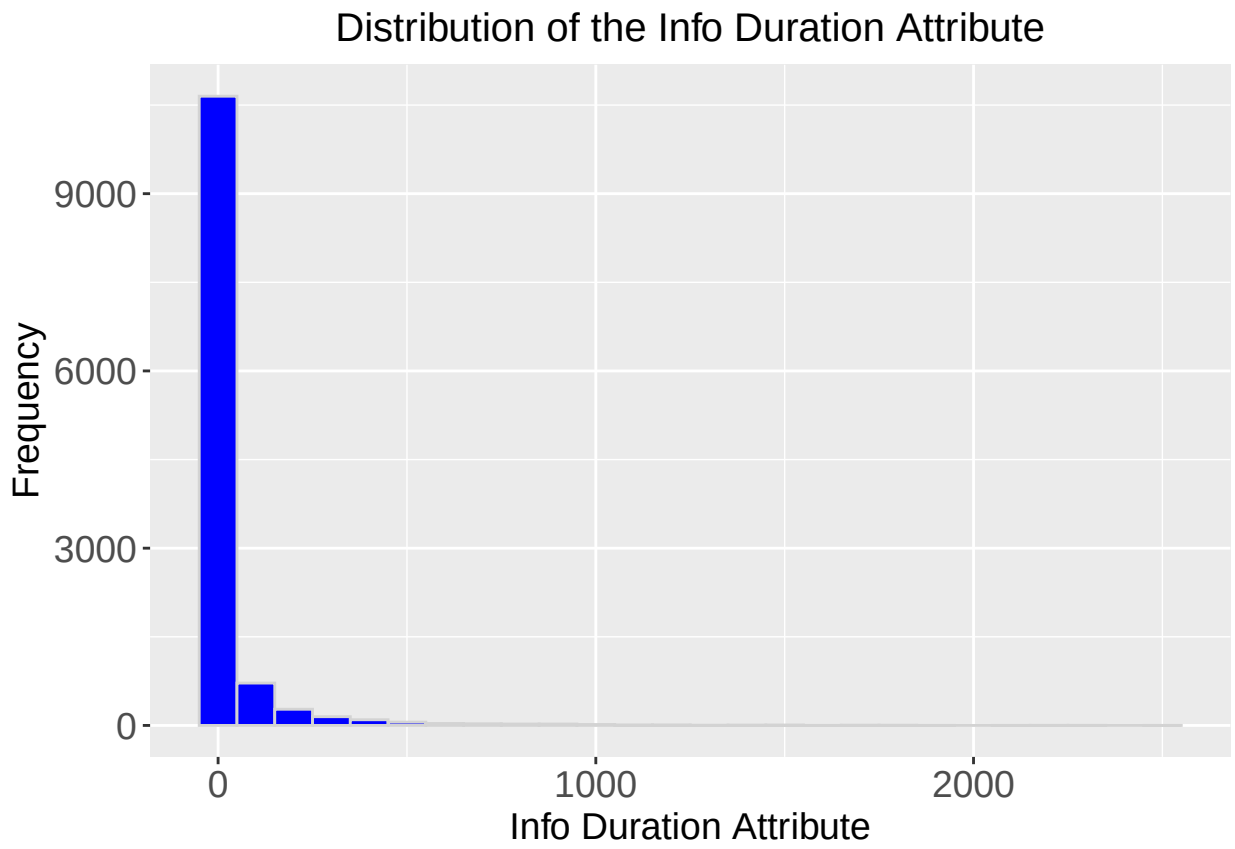

```
# Histogram to visualize the distribution of values in the admin Duration column

options(repr.plot.width = 8, repr.plot.height = 6)
p <- kira %>% ggplot(aes(x = Admin_Duration ))

p + geom_histogram(color="lightgray", fill="green", binwidth = 100) +
  labs(title = "Distribution of the Admin Duration Attribute", x = "Admin Duration", y = "Frequency") +
  theme(axis.title = element_text(size = 14),
        axis.text = element_text(size=14),
        plot.title = element_text(hjust = 0.5, size = 15))
```



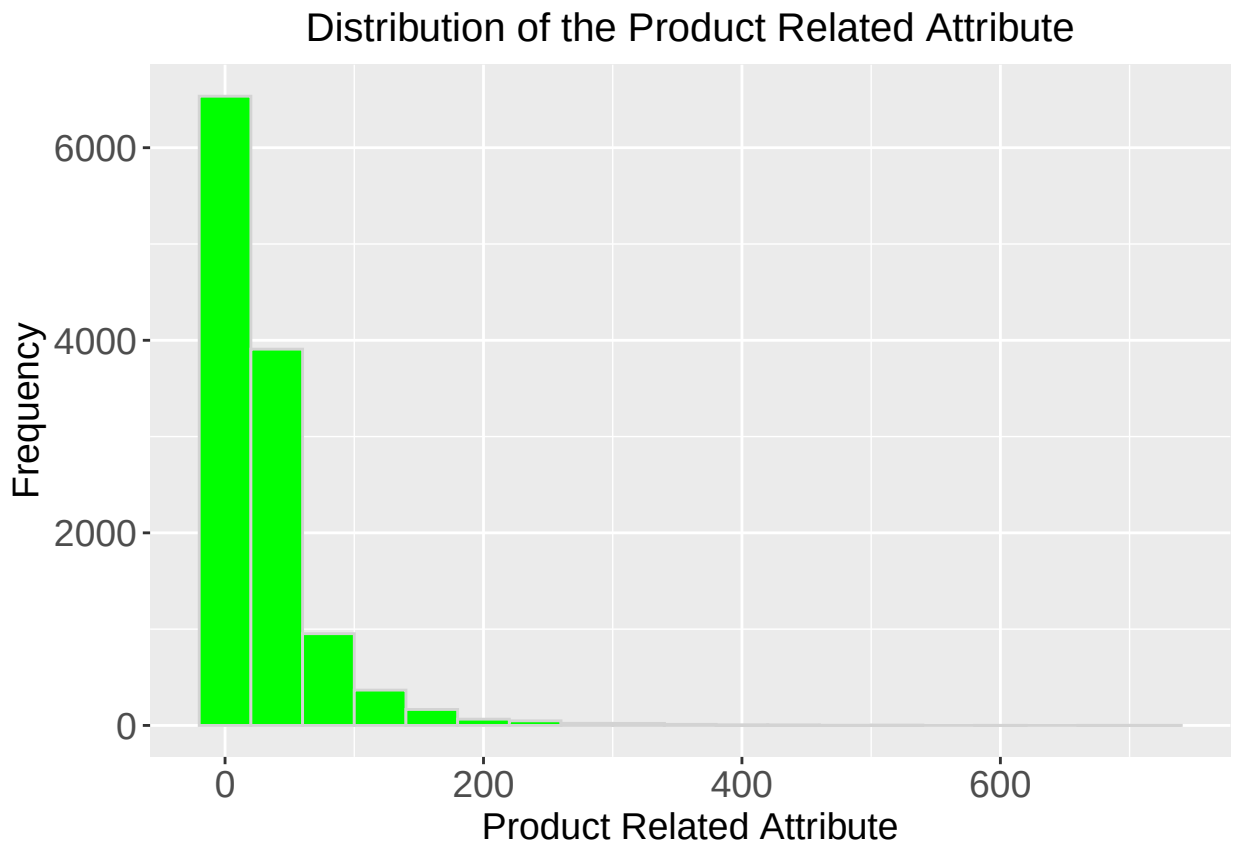

```
# Histogram to visualize the distribution of values in the Info column

options(repr.plot.width = 8, repr.plot.height = 6)
p <- kira %>% ggplot(aes(x = Info ))

p + geom_histogram(color="lightgray", fill="blue", binwidth = 5) +
  labs(title = "Distribution of the Info Attribute", x = "Info Attribute", y = "Frequency") +
  theme(axis.title = element_text(size = 14),
        axis.text = element_text(size=14),
        plot.title = element_text(hjust = 0.5, size = 15))
```



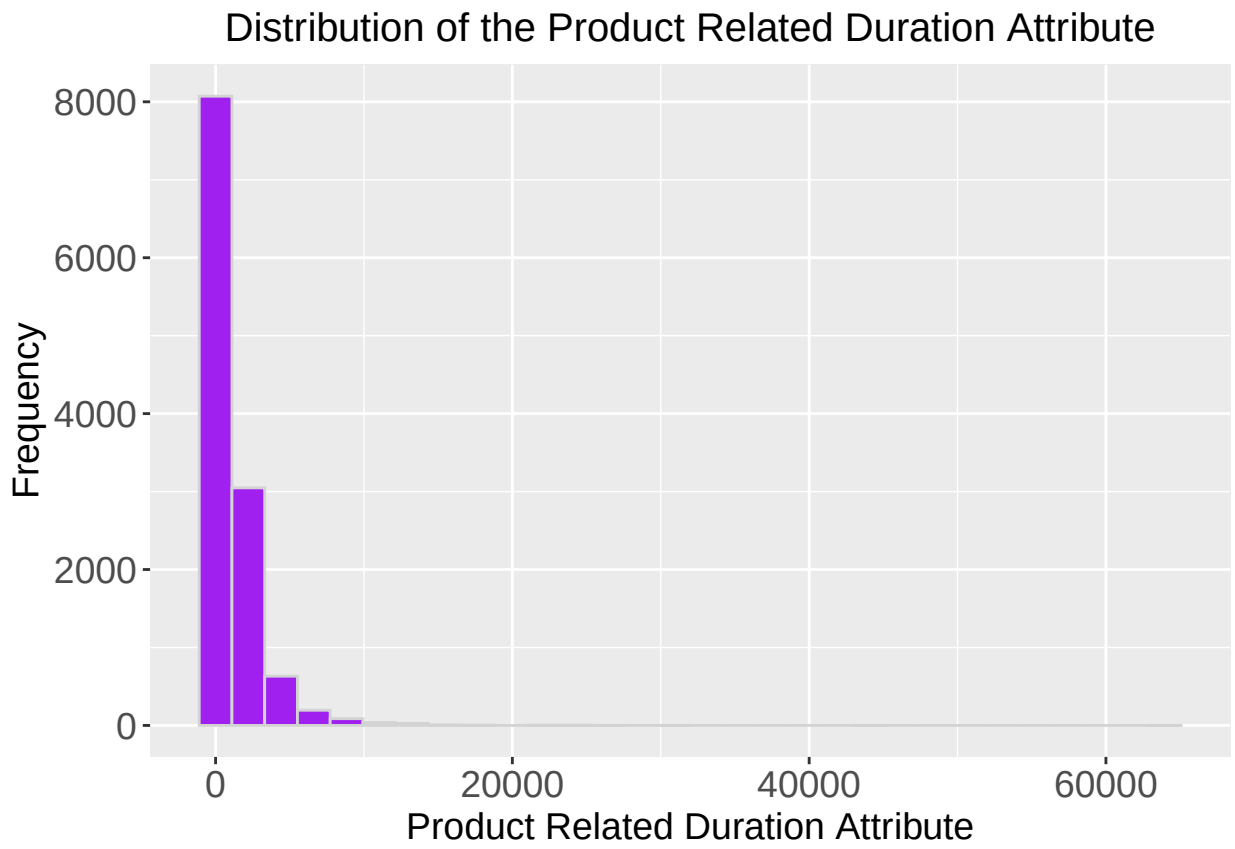

```
# Histogram to visualize the distribution of values in the Info Related column

options(repr.plot.width = 8, repr.plot.height = 6)
p <- kira %>% ggplot(aes(x = Info_Duration ))

p + geom_histogram(color="lightgray", fill="blue", binwidth = 100) +
  labs(title = "Distribution of the Info Duration Attribute", x = "Info Duration Attribute", y = "Frequ
  theme(axis.title = element_text(size = 14),
        axis.text = element_text(size=14),
        plot.title = element_text(hjust = 0.5, size = 15))
```




```
#The data is skewed to the right and extremely leptokurtic

# Histogram to visualize the distribution of values in the Product Related column

options(repr.plot.width = 8, repr.plot.height = 6)
p <- kira %>% ggplot(aes(x = Prod_Related ))

p + geom_histogram(color="lightgray", fill="green", binwidth = 40) +
  labs(title = "Distribution of the Product Related Attribute", x = "Product Related Attribute", y = "Fr
  theme(axis.title = element_text(size = 14),
        axis.text = element_text(size=14),
        plot.title = element_text(hjust = 0.5, size = 15))
```

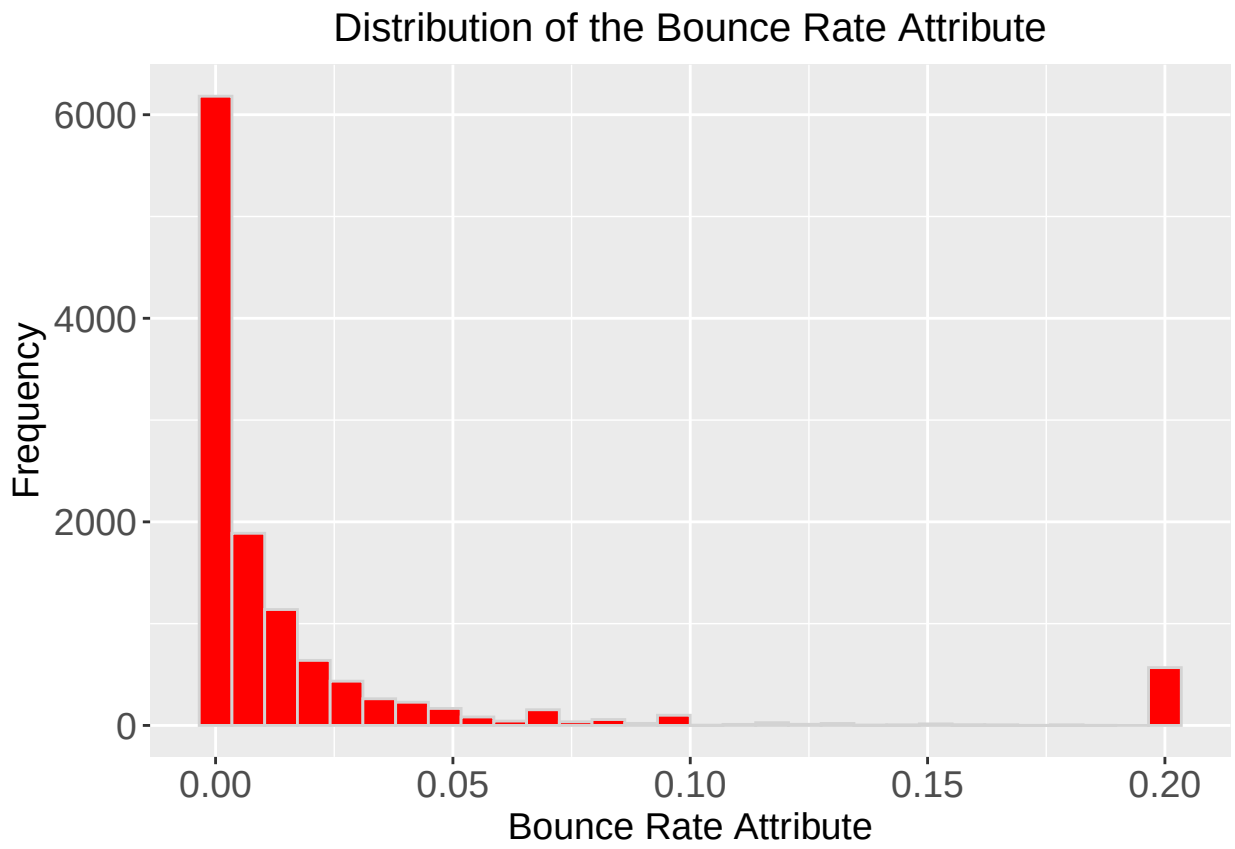



```
# Histogram to visualize the distribution of values in the Product Related Duration column

options(repr.plot.width = 8, repr.plot.height = 6)
p <- kira %>% ggplot(aes(x = Prod_Related_Duration ))

p + geom_histogram(color="lightgray", fill="purple", bins = 30) +
  labs(title = "Distribution of the Product Related Duration Attribute", x = "Product Related Duration A
  theme(axis.title = element_text(size = 14),
        axis.text = element_text(size=14),
        plot.title = element_text(hjust = 0.5, size = 15))
```



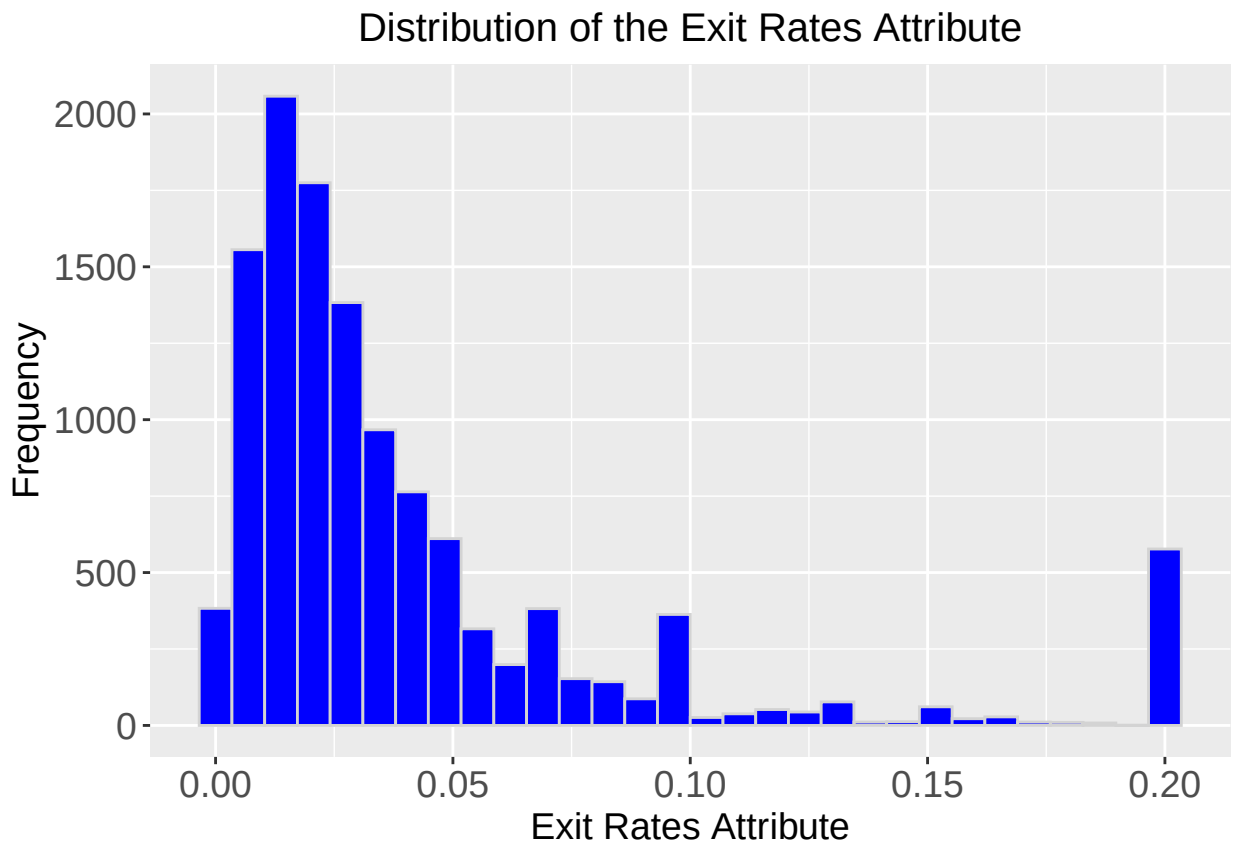

```
# Histogram to visualize the distribution of values in the Bounce Rates column

options(repr.plot.width = 8, repr.plot.height = 6)
p <- kira %>% ggplot(aes(x = Bounce_Rates ))

p + geom_histogram(color="lightgray", fill="red") +
  labs(title = "Distribution of the Bounce Rate Attribute", x = "Bounce Rate Attribute", y = "Frequency
  theme(axis.title = element_text(size = 14),
        axis.text = element_text(size=14),
        plot.title = element_text(hjust = 0.5, size = 15))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



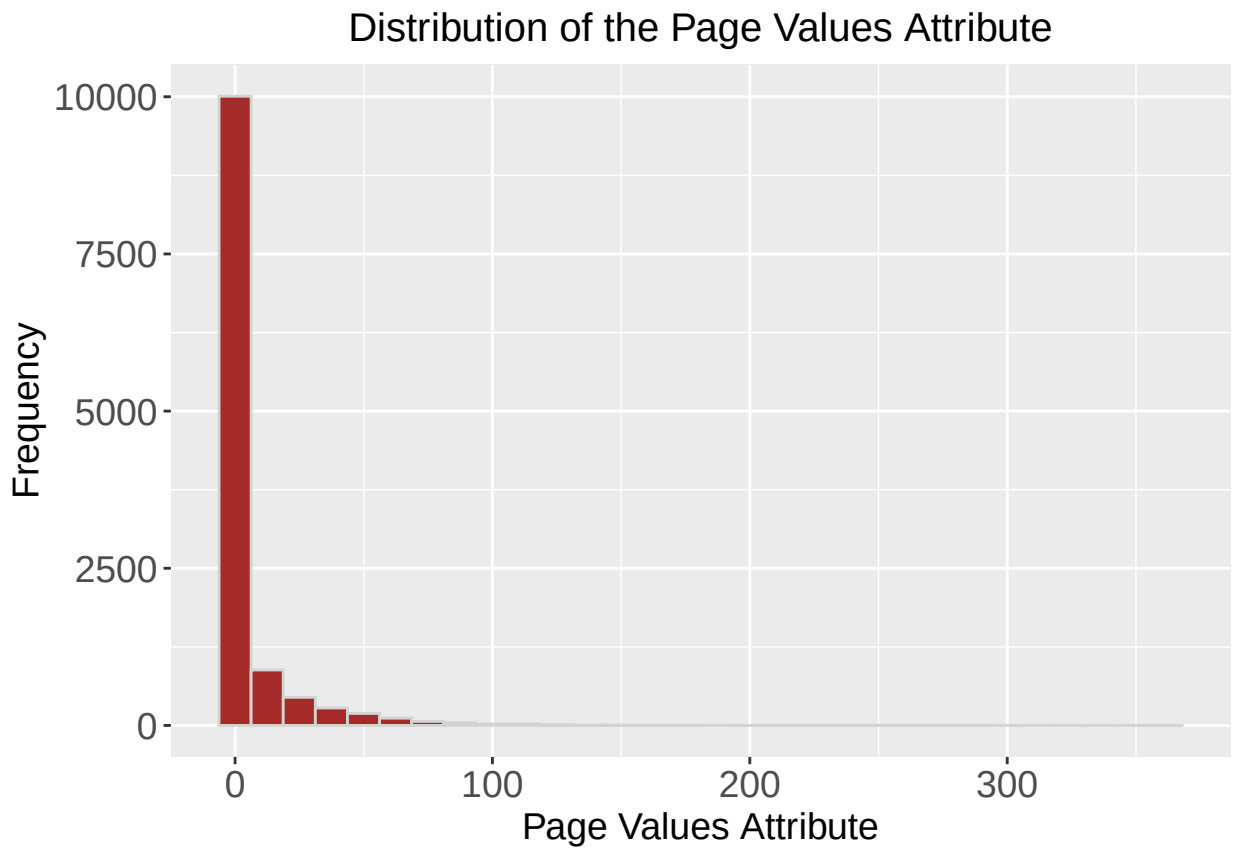

```
# Histogram to visualize the distribution of values in the Exit Rates column

options(repr.plot.width = 8, repr.plot.height = 6)
p <- kira %>% ggplot(aes(x = Exit_Rates ))

p + geom_histogram(color="lightgray", fill="blue") +
  labs(title = "Distribution of the Exit Rates Attribute", x = "Exit Rates Attribute", y = "Frequency")
  theme(axis.title = element_text(size = 14),
        axis.text = element_text(size=14),
        plot.title = element_text(hjust = 0.5, size = 15))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
#Histogram to visualize the distribution of values in the Page Values column

options(repr.plot.width = 8, repr.plot.height = 6)
p <- kira %>% ggplot(aes(x = Page_Values ))

p + geom_histogram(color="lightgray", fill="brown") +
  labs(title = "Distribution of the Page Values Attribute", x = "Page Values Attribute", y = "Frequency
  theme(axis.title = element_text(size = 14),
        axis.text = element_text(size=14),
        plot.title = element_text(hjust = 0.5, size = 15))
```

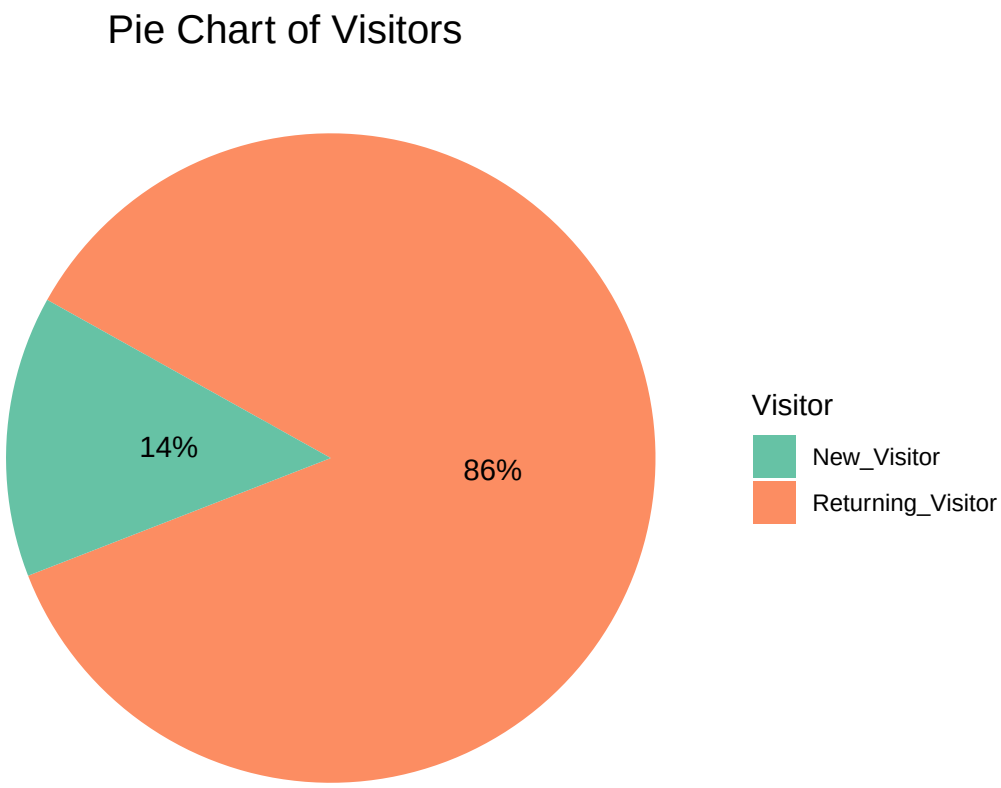
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



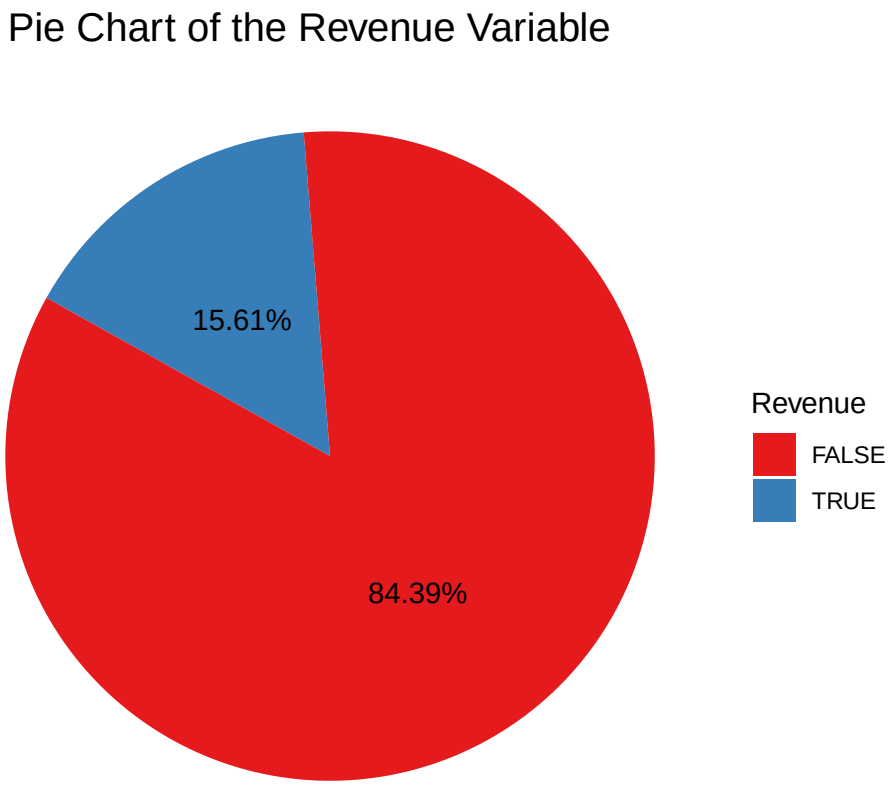

```
#Observation
#A general trend in the analysis above shows that most the numerical columns
#are leptokurtic
#Looking at the histogram analysis and comparing the results with the central tendency
#analysis, the high levels of variance in the attributes is noted.
```

Univariate Analysis for the Categorical Variables

```
#VIsualizing the proportions in the Visitor Column
kira_visitor <- kira %>%
  filter(Visitor != "NA") %>%
  group_by(Visitor) %>%
  count() %>%
  ungroup() %>%
  arrange(desc(Visitor)) %>%
  mutate( percentage = round(n/sum(n), 3)*100, lab.pos = cumsum(percentage)- 0.5 * percentage)
ggplot(kira_visitor, aes(x = "", y= percentage, fill = Visitor)) +
  geom_bar(stat = "identity")+
  coord_polar("y", start = 200) +
  geom_text(aes(y = lab.pos, label = paste(percentage,"%", sep = "")), col = "black") +
  theme_void() + scale_fill_brewer(palette = "Set2") + labs(title= "Pie Chart of Visitors") +
  theme(plot.title = element_text(hjust = 0.4, size = 15))
```

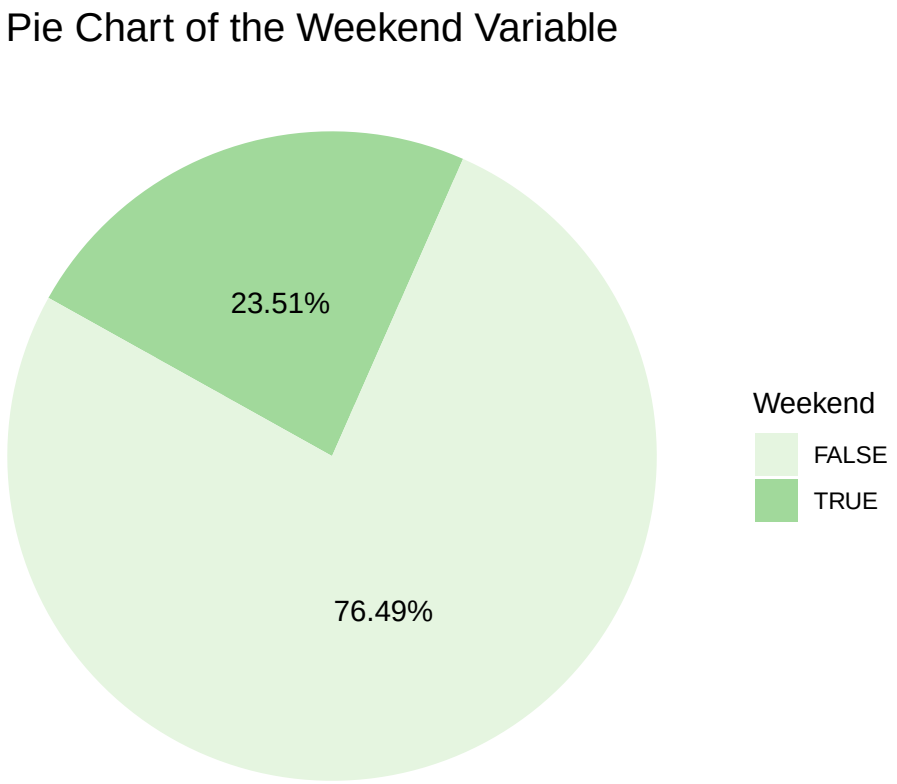



```
#About 90% of the visitors in the Dataset were Returning Visitors.


#Visualizing the label variable

kira_revenue <- kira %>%
  filter(Revenue != "NA") %>%
  group_by(Revenue) %>%
  count() %>%
  ungroup() %>%
  arrange(desc(Revenue)) %>%
  mutate( percentage = round(n/sum(n), 4)*100, lab.pos = cumsum(percentage)- 0.5 * percentage)
ggplot(kira_revenue, aes(x = "", y= percentage, fill = Revenue)) +
  geom_bar(stat = "identity")+
  coord_polar("y", start = 200) +
  geom_text(aes(y = lab.pos, label = paste(percentage,"%", sep = "")), col = "black") +
  theme_void() + scale_fill_brewer(palette = "Set1") + labs(title= "Pie Chart of the Revenue Variable")
  theme(plot.title = element_text(hjust = 0.4, size = 15))
```

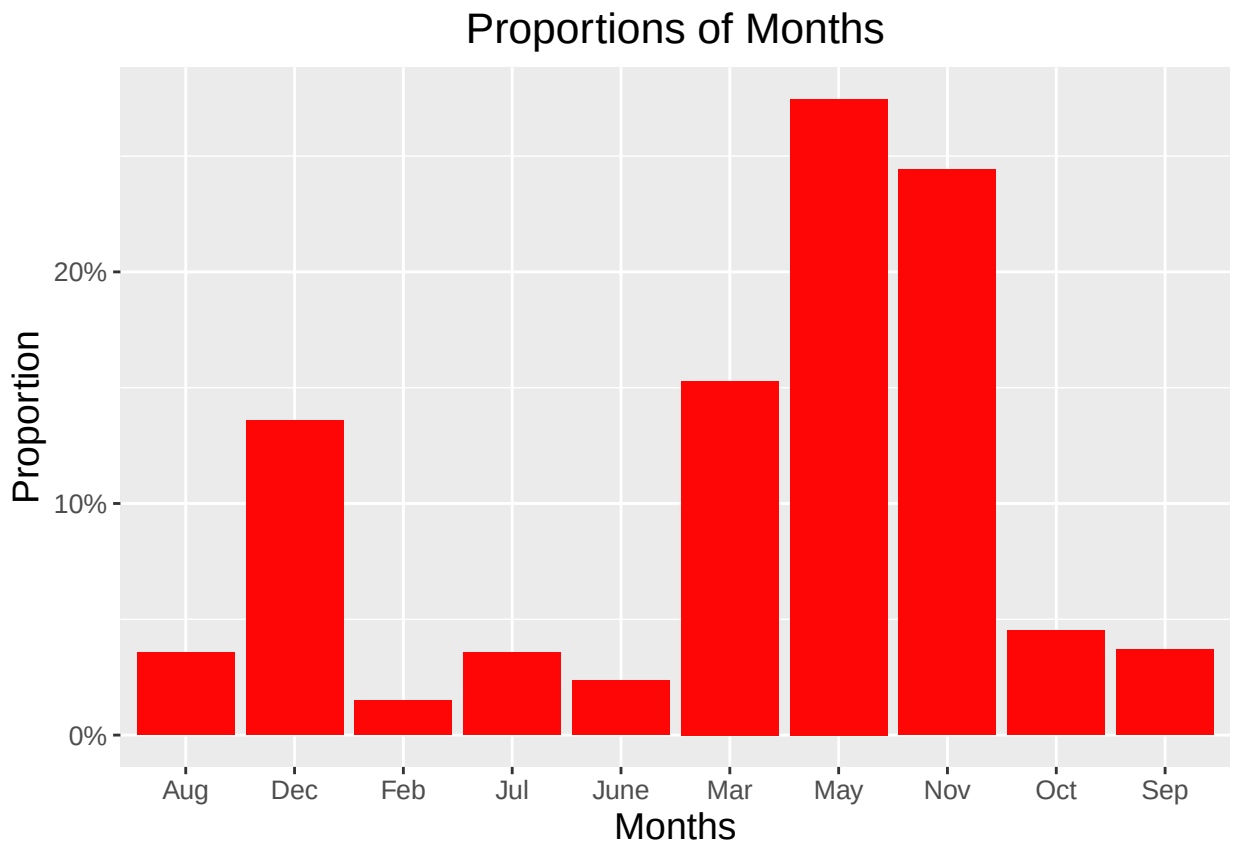



```
#15.61 % of the values in the Revenue attribute were True



#Visualizing the Weekend variable

kira_weekend <- kira %>%
  filter(Weekend != "NA") %>%
  group_by(Weekend) %>%
  count() %>%
  ungroup() %>%
  arrange(desc(Weekend)) %>%
  mutate( percentage = round(n/sum(n), 4)*100, lab.pos = cumsum(percentage)- 0.5 * percentage)
ggplot(kira_weekend, aes(x = "", y= percentage, fill = Weekend)) +
  geom_bar(stat = "identity")+
  coord_polar("y", start = 200) +
  geom_text(aes(y = lab.pos, label = paste(percentage,"%", sep = "")), col = "black") +
  theme_void() + scale_fill_brewer(palette = "pastel2") + labs(title= "Pie Chart of the Weekend Variable
  theme(plot.title = element_text(hjust = 0.4, size = 15))
```

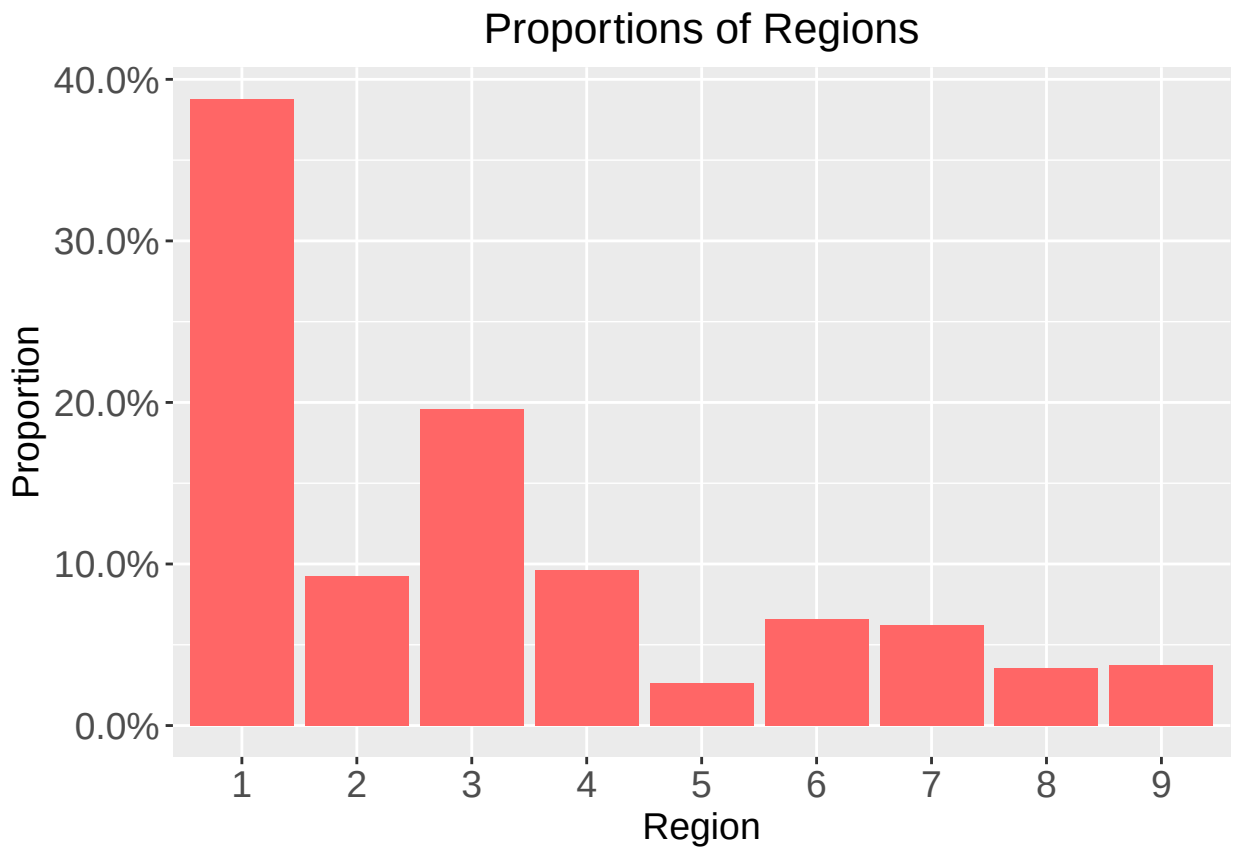
```
## Warning in pal_name(palette, type): Unknown palette pastel2
```



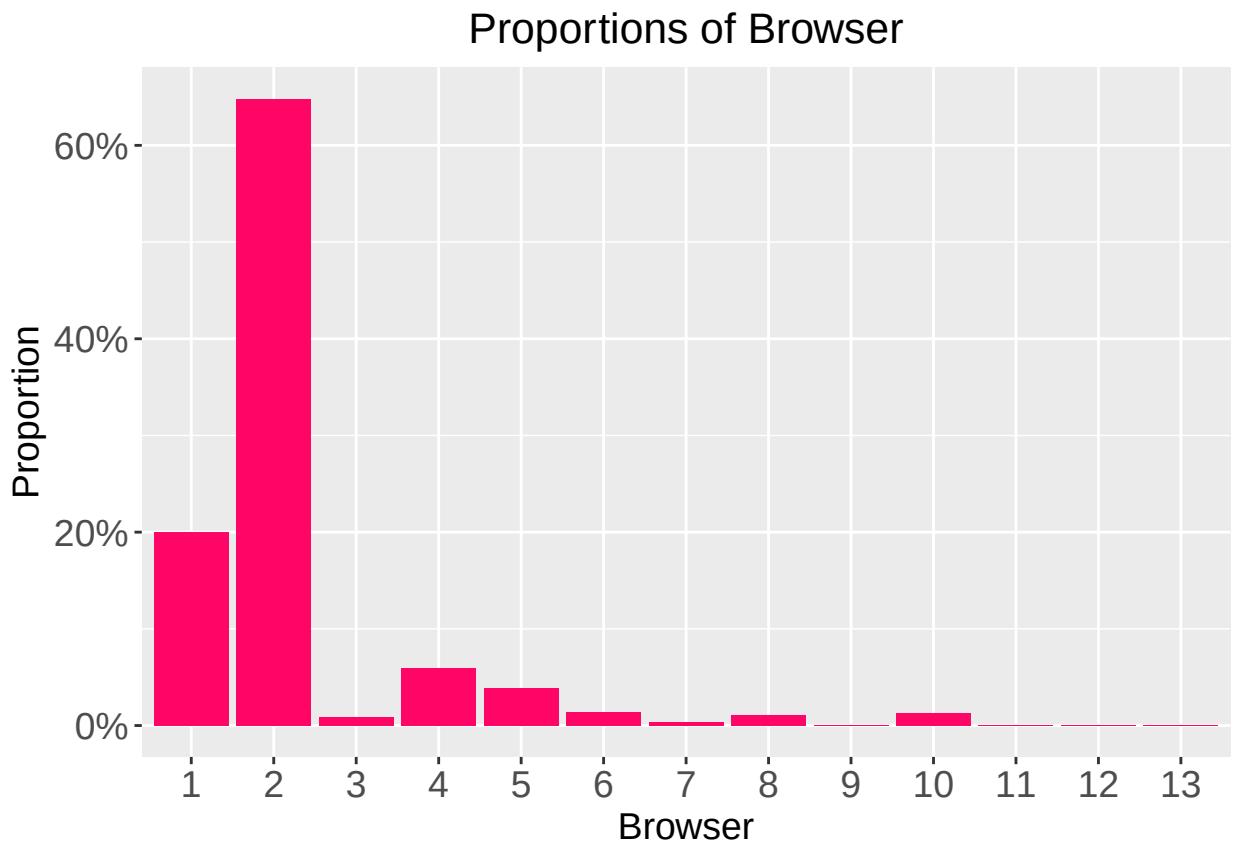

```
# plot a bar chart to visualize the proportion of values in Month column
ggplot(kira, aes(Month)) +
  geom_bar(aes(y = (..count..)/sum(..count..)), fill = "#FF0606") +
  scale_y_continuous(labels=scales::percent) +
  labs(title= 'Proportions of Months', x='Months', y="Proportion") +
  theme(axis.title = element_text(size = 14),
        axis.text = element_text(size=10),
        plot.title = element_text(hjust = 0.5, size = 16))
```



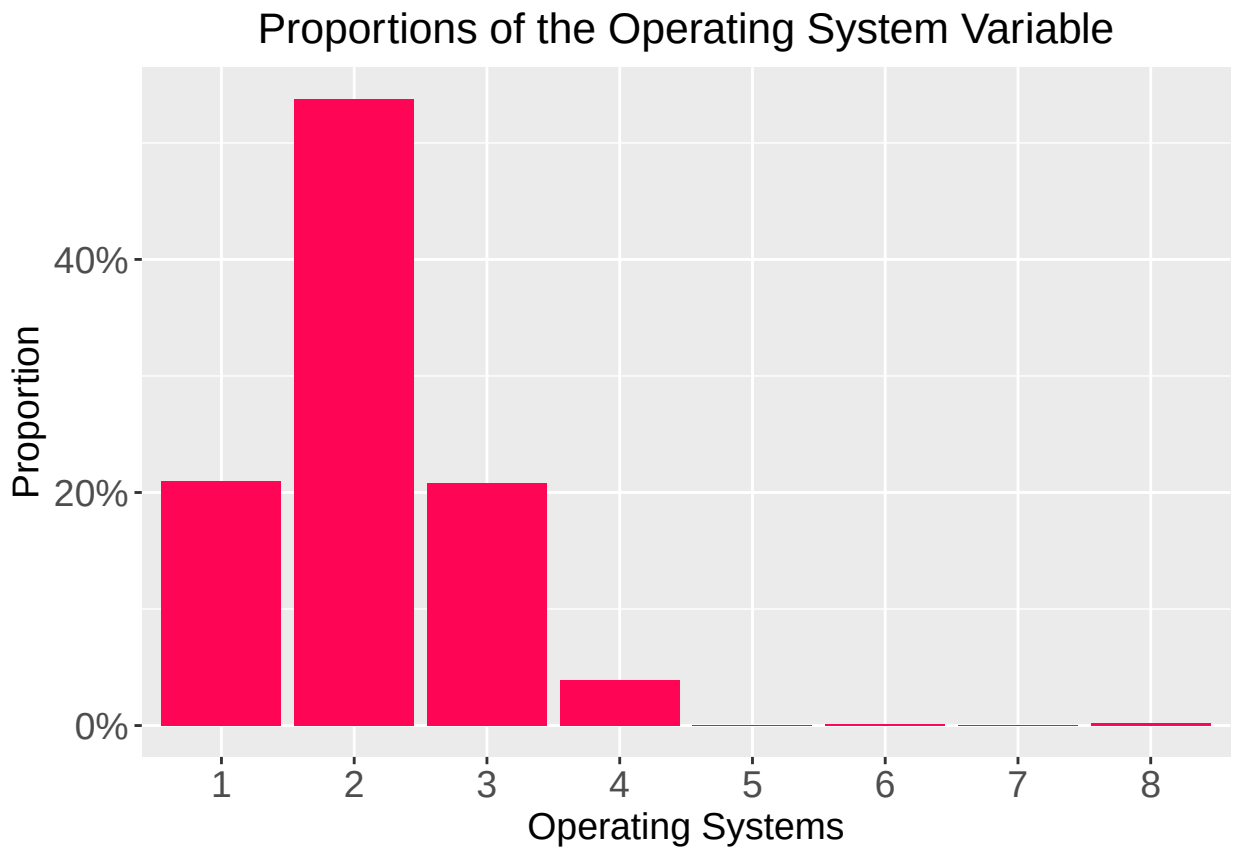

```
#May, November and March respectively had the most numbers of visitors
#February had the least number of visitors


# plot a bar chart to visualize the proportion of values in Region column
ggplot(kira, aes(Region)) +
  geom_bar(aes(y = (..count..)/sum(..count..)), fill = "#FF6666") +
  scale_y_continuous(labels=scales::percent) +
  labs(title= 'Proportions of Regions', x='Region', y="Proportion") +
  theme(axis.title = element_text(size = 14),
        axis.text = element_text(size=14),
        plot.title = element_text(hjust = 0.5, size = 16))
```



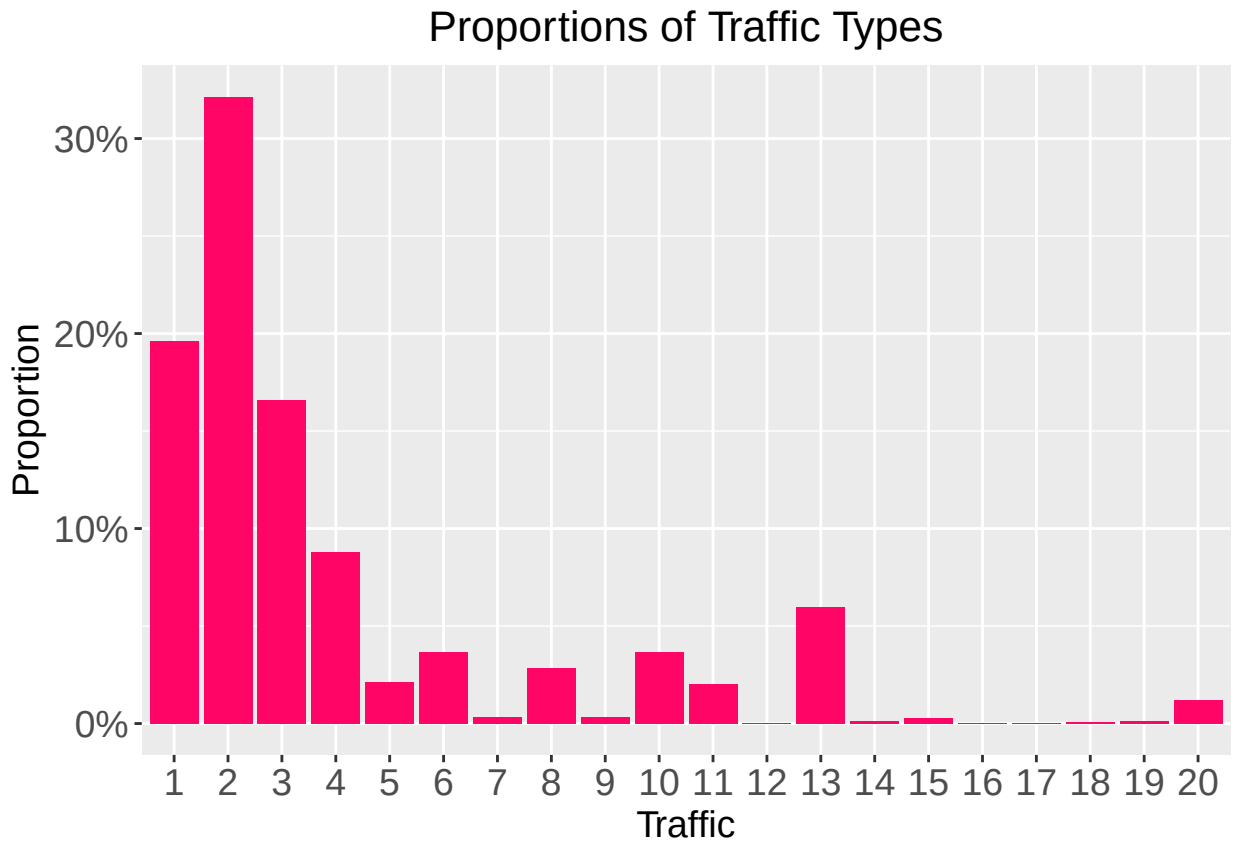

```
#Region 1 is the most represented in the dataset whilst region 5 is the least represented



# plot a bar chart to visualize the proportion of values in Browser column
ggplot(kira, aes(Browser)) +
  geom_bar(aes(y = (..count..)/sum(..count..)), fill = "#FF0566") +
  scale_y_continuous(labels=scales::percent) +
  labs(title= 'Proportions of Browser', x='Browser', y="Proportion") +
  theme(axis.title = element_text(size = 14),
        axis.text = element_text(size=14),
        plot.title = element_text(hjust = 0.5, size = 16))
```




```
# plot a bar chart to visualize the proportion of values in Operating System column
ggplot(kira, aes(Operating_Systems)) +
  geom_bar(aes(y = (..count..)/sum(..count..)), fill = "#FF0556") +
  scale_y_continuous(labels=scales::percent) +
  labs(title= 'Proportions of the Operating System Variable', x='Operating Systems', y="Proportion") +
  theme(axis.title = element_text(size = 14),
        axis.text = element_text(size=14),
        plot.title = element_text(hjust = 0.5, size = 16))
```



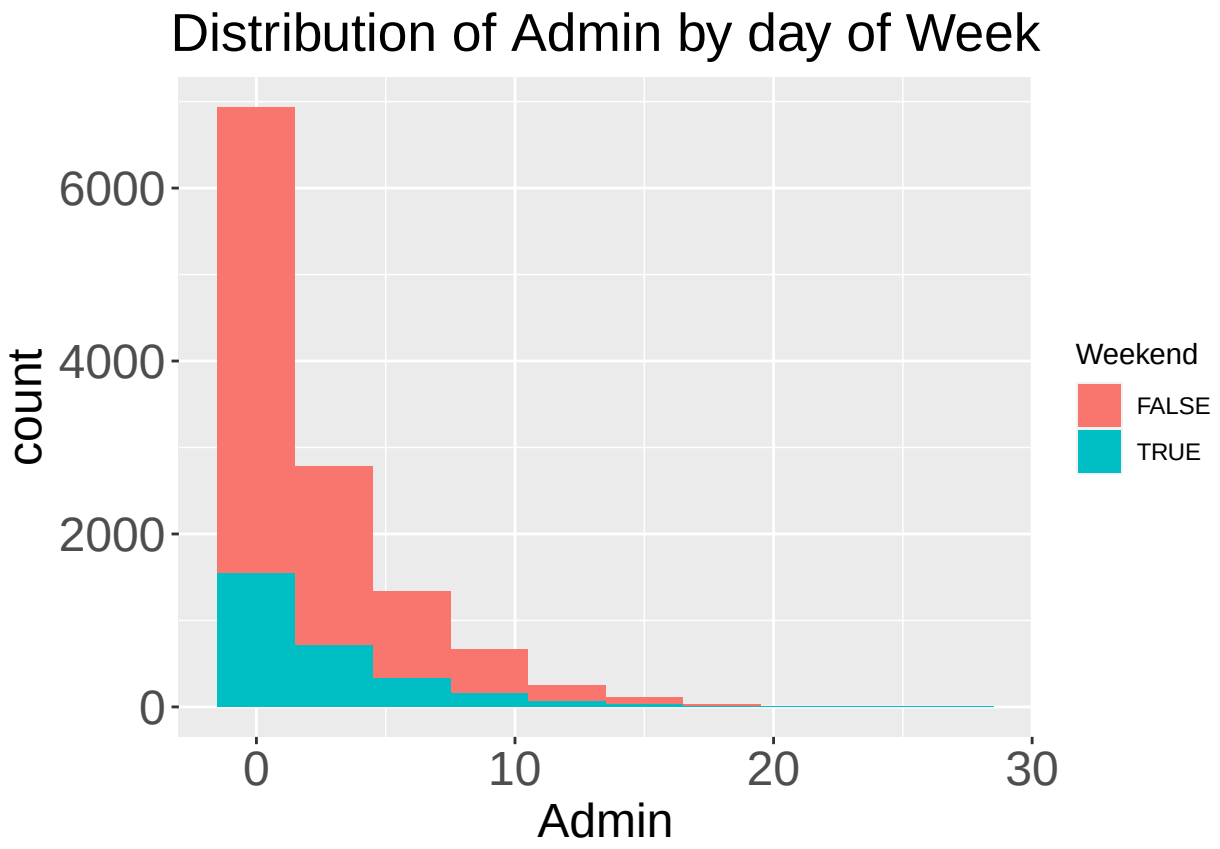

```
# plot a bar chart to visualize the proportion of values in Traffic column
ggplot(kira, aes(Traffic)) +
  geom_bar(aes(y = (..count..)/sum(..count..)), fill = "#FF0566") +
  scale_y_continuous(labels=scales::percent) +
  labs(title= 'Proportions of Traffic Types', x='Traffic', y="Proportion") +
  theme(axis.title = element_text(size = 14),
        axis.text = element_text(size=14),
        plot.title = element_text(hjust = 0.5, size = 16))
```



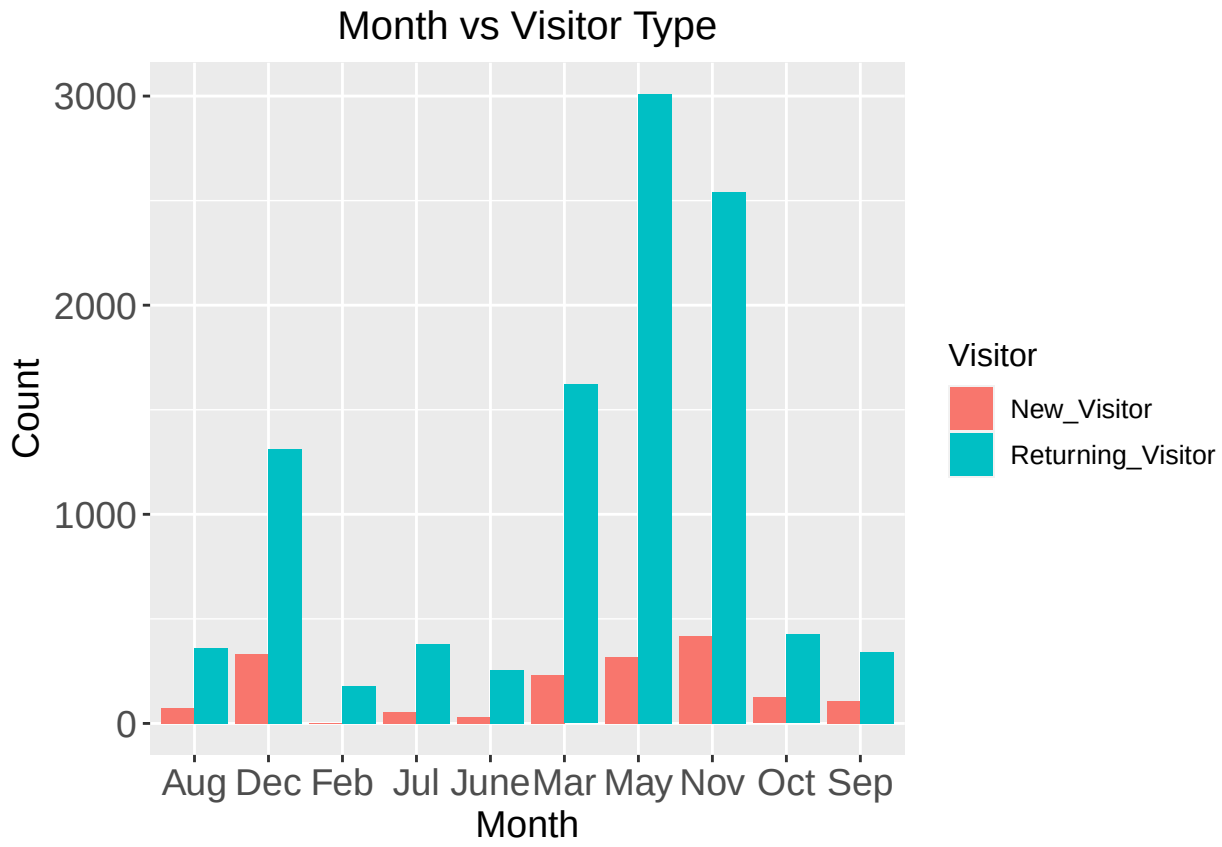

```
#Traffic type 2, 1,3,4 constituted the majority of all traffic types
```

(b) Bivariate Analysis

```
#Plot of Admin by Day of Week
ggplot(kira, aes( x= Admin, fill= Weekend)) + geom_histogram(bins = 10) +
  ggtitle("Distribution of Admin by day of Week") +
  theme(axis.text = element_text(size=18),
        axis.title = element_text(size = 18),
        plot.title = element_text(hjust = 0.5, size = 20))
```




```
# Relationship between the visitor type and the month

options(repr.plot.width = 15, repr.plot.height = 8)
ggplot(kira, aes(x = Month, fill = Visitor)) +
  geom_bar(position = "dodge") +
  labs(title = "Month vs Visitor Type", x = "Month", y = "Count") +
  theme(axis.text = element_text(size=14),
        axis.title = element_text(size = 14),
        plot.title = element_text(hjust = 0.5, size = 15),
        legend.title = element_text(size=12),
        legend.text = element_text(size=10))
```

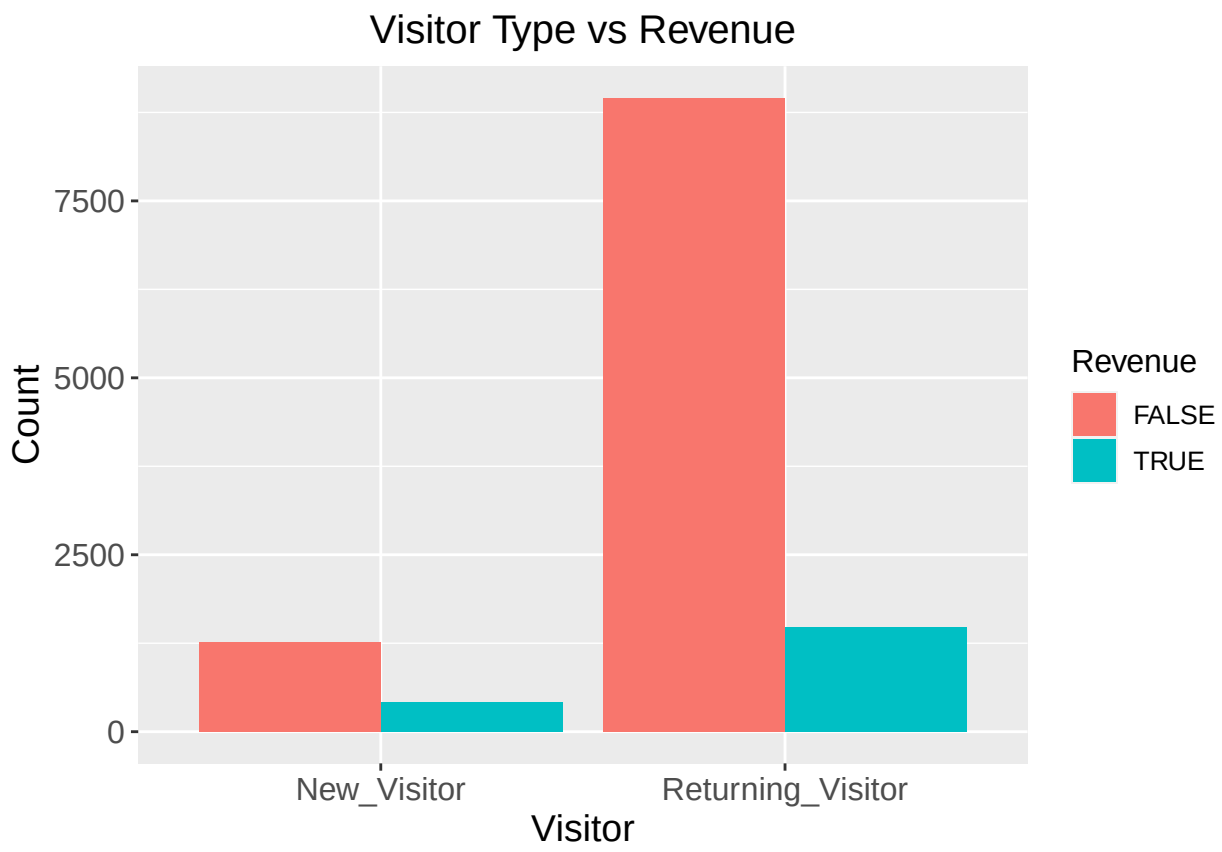



```
# May, November, March and December respectively had the highest numbers of
#Returning visitors
#The Month of February had nearly zero New visitors


# find the means of 'admin_duration', informational_duration', 'productrelated_duration',
#'pagevalues' 'exitrates', and 'bouncerates' per month
month_stats <- kira %>%
  select(Month, Admin, Info_Duration, Prod_Related_Duration, Page_Values, Exit_Rates, Bounce_Rates)%>%
  group_by(Month)%>%
  summarise_all(mean)
month_stats
```

```
## # A tibble: 10 x 7
##    Month Admin Info_Duration Prod_Related_Duration Page_Values Exit_Rates
##    <chr> <dbl>         <dbl>                 <dbl>       <dbl>      <dbl>
##  1 Aug   3.14          35.5                  1273.       5.94      0.0377
##  2 Dec   2.24          39.7                  1148.       6.31      0.0392
##  3 Feb   0.549          2.34                  476.       0.900     0.0728
##  4 Jul   2.42          45.5                  1218.       4.10      0.0453
##  5 June  2.31          19.9                  1197.       3.44      0.0567
##  6 Mar   1.94          31.5                   836.       4.08      0.0408
##  7 May   1.99          27.5                   993.       5.49      0.0472
##  8 Nov   2.64          44.0                  1777.       7.17      0.0370
##  9 Oct   3.72          38.7                  1117.       8.65      0.0290
## 10 Sep   3.33          35.7                  1253.       7.56      0.0303
```

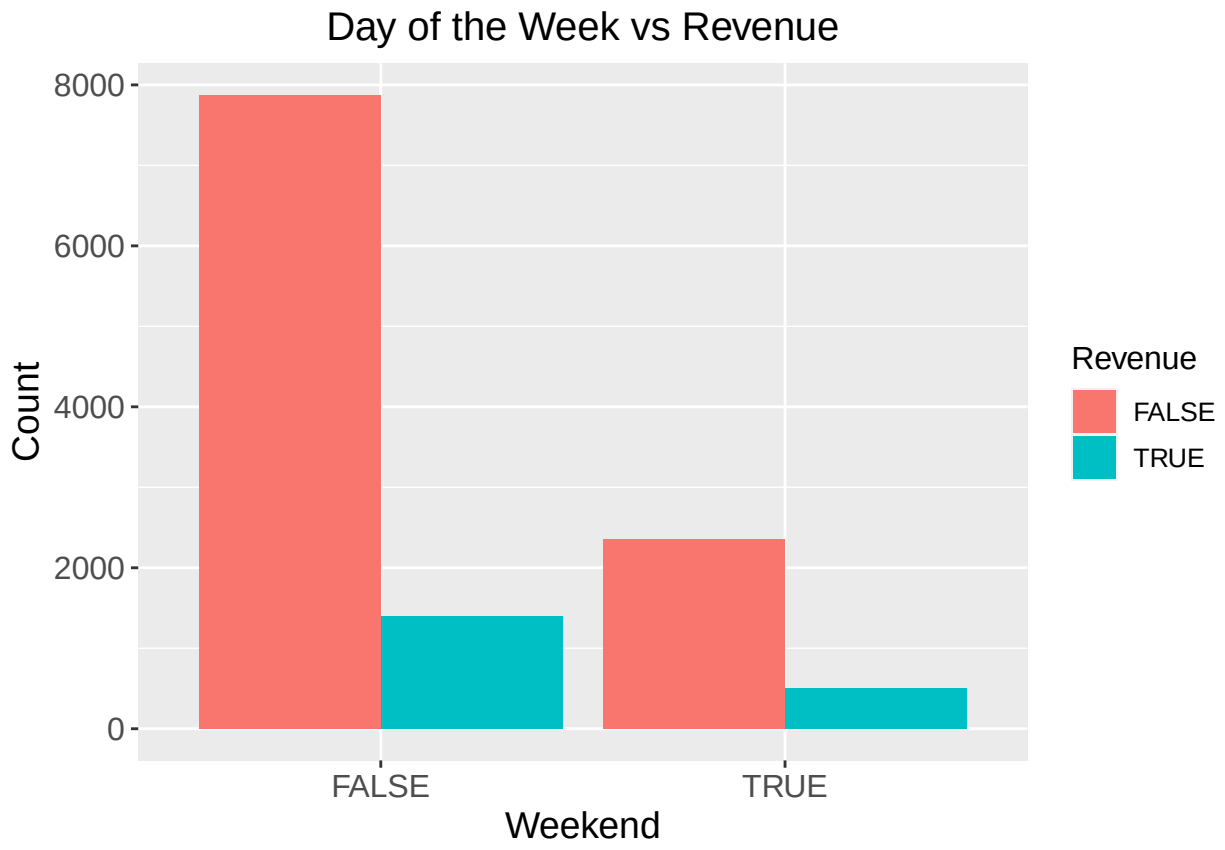
```
## # ... with 1 more variable: Bounce_Rates <dbl>
```

```
# Relationship between the visitor type and Revenue

options(repr.plot.width = 15, repr.plot.height = 8)
ggplot(kira, aes(x = Visitor, fill = Revenue)) +
  geom_bar(position = "dodge") +
  labs(title = "Visitor Type vs Revenue", x = "Visitor", y = "Count") +
  theme(axis.text = element_text(size=12),
        axis.title = element_text(size = 14),
        plot.title = element_text(hjust = 0.5, size = 15),
        legend.title = element_text(size=12),
        legend.text = element_text(size=10))
```

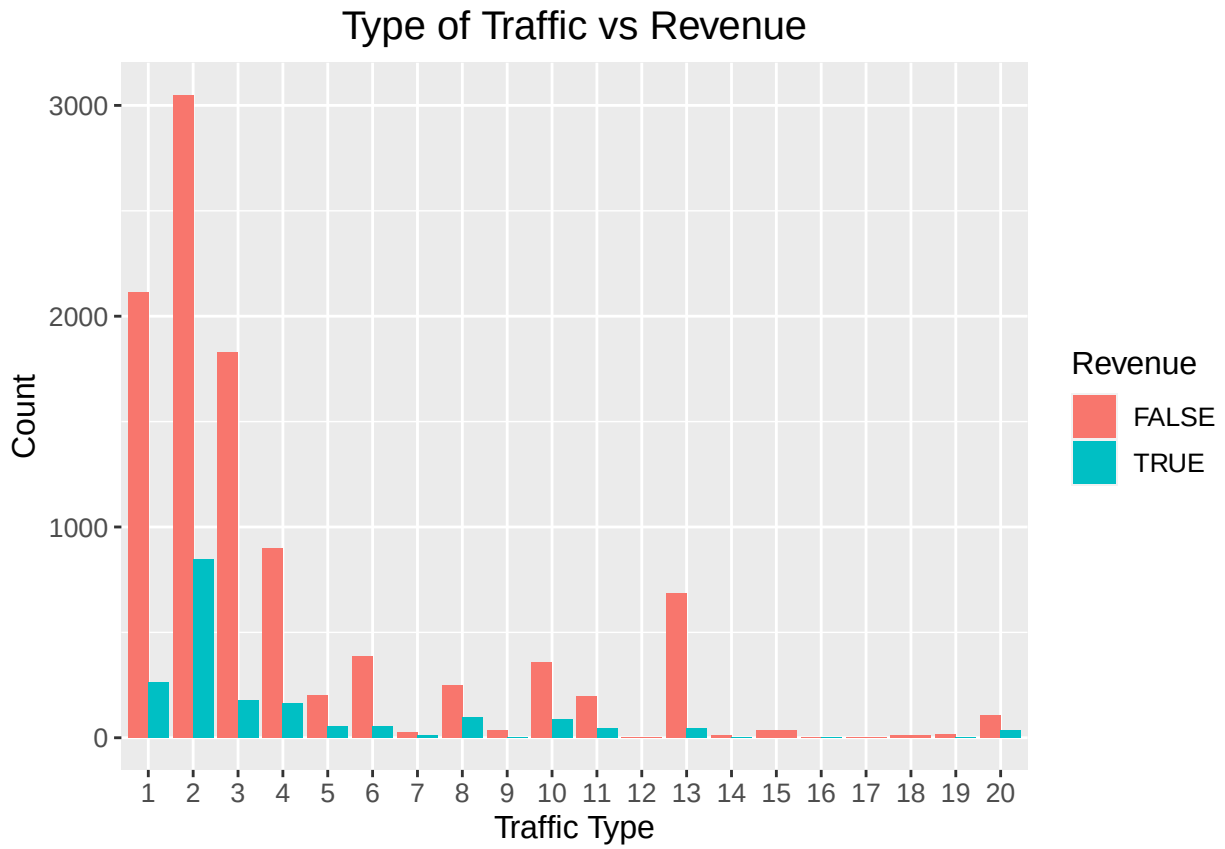



```
#Returning Visitors had the highest False Revenue


# Relationship between the Weekday and Revenue

options(repr.plot.width = 15, repr.plot.height = 8)
ggplot(kira, aes(x = Weekend, fill = Revenue)) +
  geom_bar(position = "dodge") +
  labs(title = "Day of the Week vs Revenue", x = "Weekend", y = "Count") +
  theme(axis.text = element_text(size=12),
        axis.title = element_text(size = 14),
```

```
        plot.title = element_text(hjust = 0.5, size = 15),
        legend.title = element_text(size=12),
        legend.text = element_text(size=10))
```

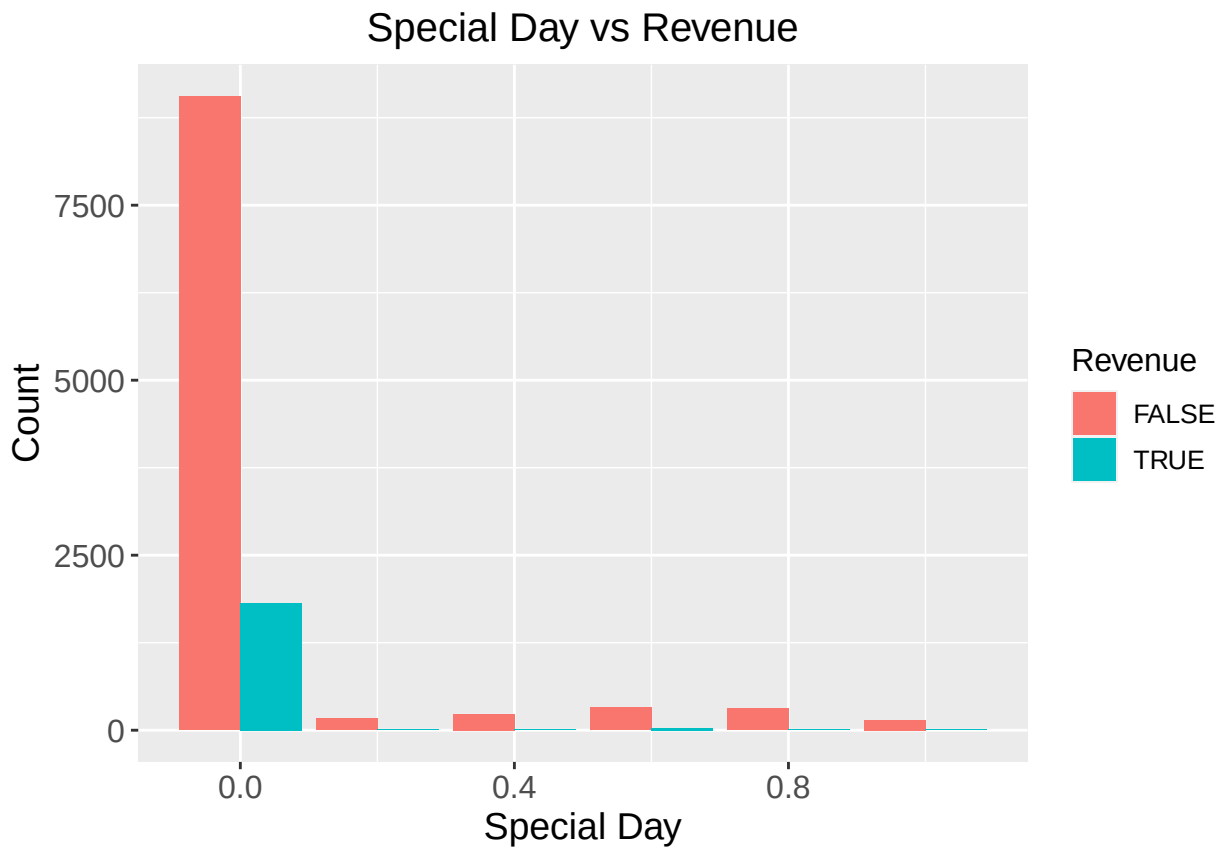



```
#Most of the Revenue is earned on weekdays and not weekends


# Relationship between the Traffic type and Revenue

options(repr.plot.width = 18, repr.plot.height = 10)
ggplot(kira, aes(x = Traffic, fill = Revenue)) +
  geom_bar(position = "dodge") +
  labs(title = "Type of Traffic vs Revenue", x = "Traffic Type", y = "Count") +
  theme(axis.text = element_text(size=10),
        axis.title = element_text(size = 12),
        plot.title = element_text(hjust = 0.5, size = 15),
        legend.title = element_text(size=12),
        legend.text = element_text(size=10))
```

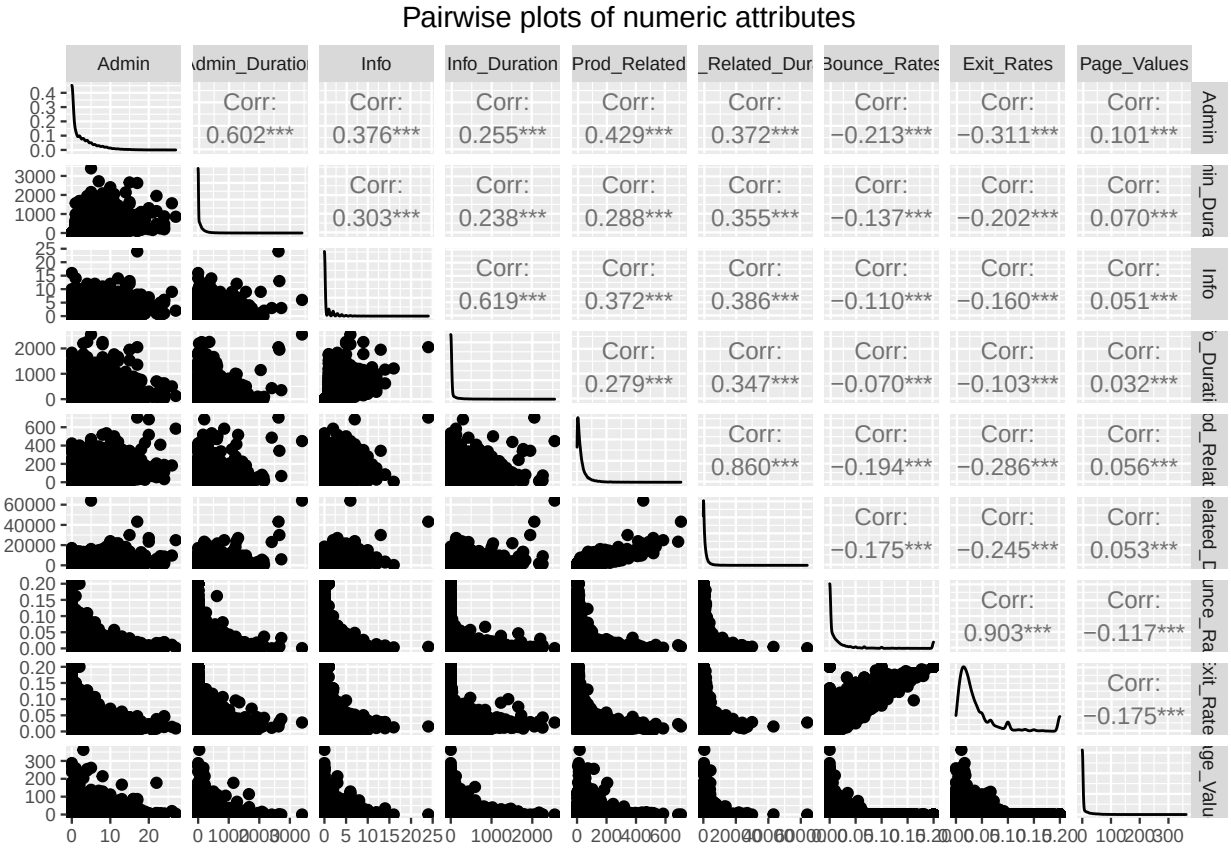



```
#Traffic 2, 1 and 3 generated the most revenues
```

```
# Relationship between the Special Day and Revenue

options(repr.plot.width = 15, repr.plot.height = 8)
ggplot(kira, aes(x = Special_Day, fill = Revenue)) +
  geom_bar(position = "dodge") +
  labs(title = "Special Day vs Revenue", x = "Special Day", y = "Count") +
  theme(axis.text = element_text(size=12),
        axis.title = element_text(size = 14),
        plot.title = element_text(hjust = 0.5, size = 15),
        legend.title = element_text(size=12),
        legend.text = element_text(size=10))
```




```
# Plotting pair plots for numeric columns

options(repr.plot.width = 18, repr.plot.height = 18)
ggpairs(numeric, upper = list(continuous = wrap("cor", size = 3))) +
  labs(title = "Pairwise plots of numeric attributes") +
  theme_grey(base_size = 09) +
  theme(plot.title = element_text(hjust = 0.5))
```



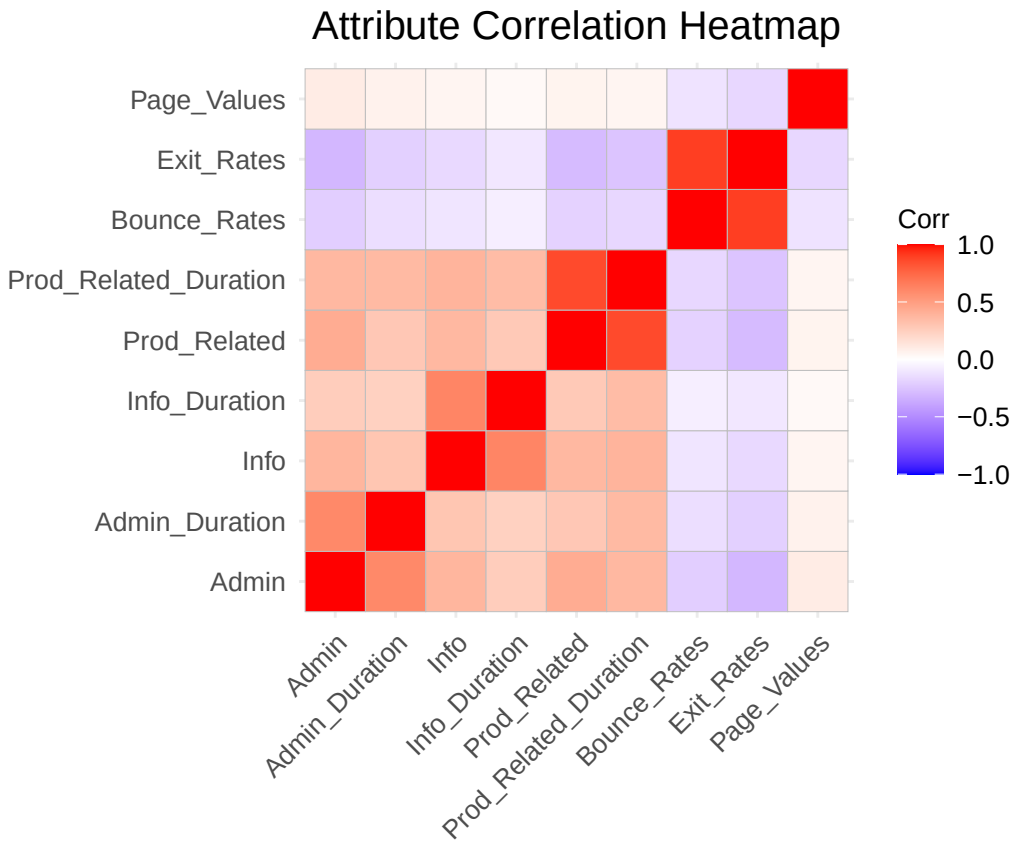
There is a very strong positive correlation between:

1. Bounce rates and exit rates
2. Product related site visits and product related duration
3. There is a strong positive correlation between:

Administrative site visits and administrative duration Informational site visits and informational duration

There are weak correlations between the rest of the variables

```
#Correlation Heatmap
options(repr.plot.width = 15, repr.plot.height = 10)
ggcorrplot(cor(numeric), tl.cex = 10) +
  labs(title = "Attribute Correlation Heatmap") +
  theme(axis.title = element_text(size = 14),
        plot.title = element_text(hjust = 0.5, size = 15),
        legend.title = element_text(size=10),
        legend.text = element_text(size=10))
```



4. Model Development a) K-Means Clustering

K Means Clustering is an unsupervised learning algorithm that groups similar clusters in a dataset by partitioning the dataset into K distinct,non-overlapping clusters

Implementation

1. Data Preprocessing

```
#K-Means is an unsupervised machine learning. We, therefore, won't need the label
#column as the algorithm will help us determine what the labels would be

#Separating the dependent variable from the independent
kira_indep <- kira[, 1:17]


#Changing the Factors in the Dataset to integers

#kira_indep[, 12:15] <- lapply(kira_indep[, 12:15], as.integer)
```

2. Categorical Columns Encoding

```
# one hot encode the categorical variables  to ensure they are
#in a format machine learning can comprehend
dummy <- dummyVars("~ Month + Operating_Systems + Browser + Region + Traffic + Visitor + Weekend", data=
encoded <- data.frame(predict(dummy, newdata = kira_indep))
kira_indep <- cbind(kira_indep[ , 1:10], encoded)
```

3. Normalizing the Data

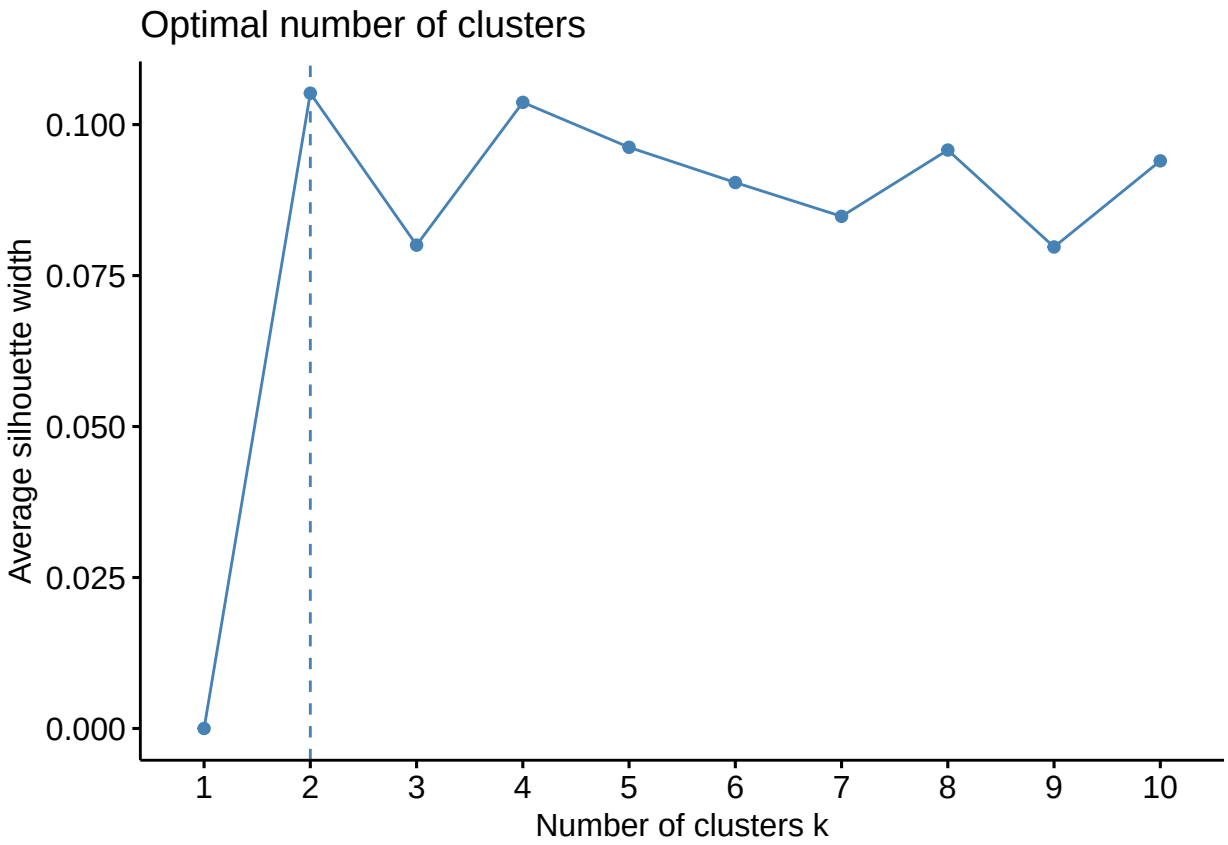
```
#We need to scale the data to ensure the distances between data points are
#aligned and in a machine learning comprehensible format.

prep <- preProcess(kira_indep, method=c("range"))
norm.kira <- predict(prep, kira_indep)


#Preview the first few records
head(norm.kira)
```

```
##   Admin Admin_Duration Info Info_Duration Prod_Related Prod_Related_Duration
## 1     0   0.0002941393    0  0.0003920992  0.001418440          1.563122e-05
## 2     0   0.0002941393    0  0.0003920992  0.002836879          1.016029e-03
## 3     0   0.0000000000    0  0.0000000000  0.001418440          0.000000e+00
## 4     0   0.0002941393    0  0.0003920992  0.002836879          5.731448e-05
## 5     0   0.0002941393    0  0.0003920992  0.014184397          9.824223e-03
## 6     0   0.0002941393    0  0.0003920992  0.026950355          2.426226e-03
##   Bounce_Rates Exit_Rates Page_Values Special_Day MonthAug MonthDec MonthFeb
## 1   1.00000000   1.000000           0           0        0        0        1
## 2   0.00000000   0.500000           0           0        0        0        1
## 3   1.00000000   1.000000           0           0        0        0        1
## 4   0.25000000   0.700000           0           0        0        0        1
## 5   0.10000000   0.250000           0           0        0        0        1
## 6   0.07894737   0.122807           0           0        0        0        1
##   MonthJul MonthJune MonthMar MonthMay MonthNov MonthOct MonthSep
## 1        0         0        0        0        0        0        0
## 2        0         0        0        0        0        0        0
## 3        0         0        0        0        0        0        0
## 4        0         0        0        0        0        0        0
## 5        0         0        0        0        0        0        0
## 6        0         0        0        0        0        0        0
##   Operating_Systems.1 Operating_Systems.2 Operating_Systems.3
## 1                   1                   0                   0
## 2                   0                   1                   0
## 3                   0                   0                   0
## 4                   0                   0                   1
## 5                   0                   0                   1
## 6                   0                   1                   0
##   Operating_Systems.4 Operating_Systems.5 Operating_Systems.6
## 1                   0                   0                   0
## 2                   0                   0                   0
## 3                   1                   0                   0
## 4                   0                   0                   0
## 5                   0                   0                   0
## 6                   0                   0                   0
##   Operating_Systems.7 Operating_Systems.8 Browser.1 Browser.2 Browser.3
## 1                   0                   0         1         0         0
## 2                   0                   0         0         1         0
## 3                   0                   0         1         0         0
## 4                   0                   0         0         1         0
## 5                   0                   0         0         0         1
## 6                   0                   0         0         1         0
```

```
##   Browser.4 Browser.5 Browser.6 Browser.7 Browser.8 Browser.9 Browser.10
## 1         0         0         0         0         0         0          0
## 2         0         0         0         0         0         0          0
## 3         0         0         0         0         0         0          0
## 4         0         0         0         0         0         0          0
## 5         0         0         0         0         0         0          0
## 6         0         0         0         0         0         0          0
##   Browser.11 Browser.12 Browser.13 Region.1 Region.2 Region.3 Region.4 Region.5
## 1          0          0          0        1        0        0        0        0
## 2          0          0          0        1        0        0        0        0
## 3          0          0          0        0        0        0        0        0
## 4          0          0          0        0        1        0        0        0
## 5          0          0          0        1        0        0        0        0
## 6          0          0          0        1        0        0        0        0
##   Region.6 Region.7 Region.8 Region.9 Traffic.1 Traffic.2 Traffic.3 Traffic.4
## 1        0        0        0        0         1         0         0         0
## 2        0        0        0        0         0         1         0         0
## 3        0        0        0        1         0         0         1         0
## 4        0        0        0        0         0         0         0         1
## 5        0        0        0        0         0         0         0         1
## 6        0        0        0        0         0         0         1         0
##   Traffic.5 Traffic.6 Traffic.7 Traffic.8 Traffic.9 Traffic.10 Traffic.11
## 1         0         0         0         0         0          0          0
## 2         0         0         0         0         0          0          0
## 3         0         0         0         0         0          0          0
## 4         0         0         0         0         0          0          0
## 5         0         0         0         0         0          0          0
## 6         0         0         0         0         0          0          0
##   Traffic.12 Traffic.13 Traffic.14 Traffic.15 Traffic.16 Traffic.17 Traffic.18
## 1          0          0          0          0          0          0          0
## 2          0          0          0          0          0          0          0
## 3          0          0          0          0          0          0          0
## 4          0          0          0          0          0          0          0
## 5          0          0          0          0          0          0          0
## 6          0          0          0          0          0          0          0
##   Traffic.19 Traffic.20 VisitorNew_Visitor VisitorReturning_Visitor
## 1          0          0                  0                        1
## 2          0          0                  0                        1
## 3          0          0                  0                        1
## 4          0          0                  0                        1
## 5          0          0                  0                        1
## 6          0          0                  0                        1
##   WeekendFALSE WeekendTRUE
## 1            1           0
## 2            1           0
## 3            1           0
## 4            1           0
## 5            0           1
## 6            1           0
```

4 Determining the optimal number of Clusters

```
#Using the silhouette method
set.seed(101)
fviz_nbclust(norm.kira, FUN = kmeans, method = "silhouette")
```




```
#According to the Silhouette method, the optimal clusters is 2.

# Implementing the L-means clustering algorithm with k is 2

means.silo <- kmeans(norm.kira, 2, nstart = 25)

#print(means.silo)

#Checking number of records in each cluster

means.silo$size
```

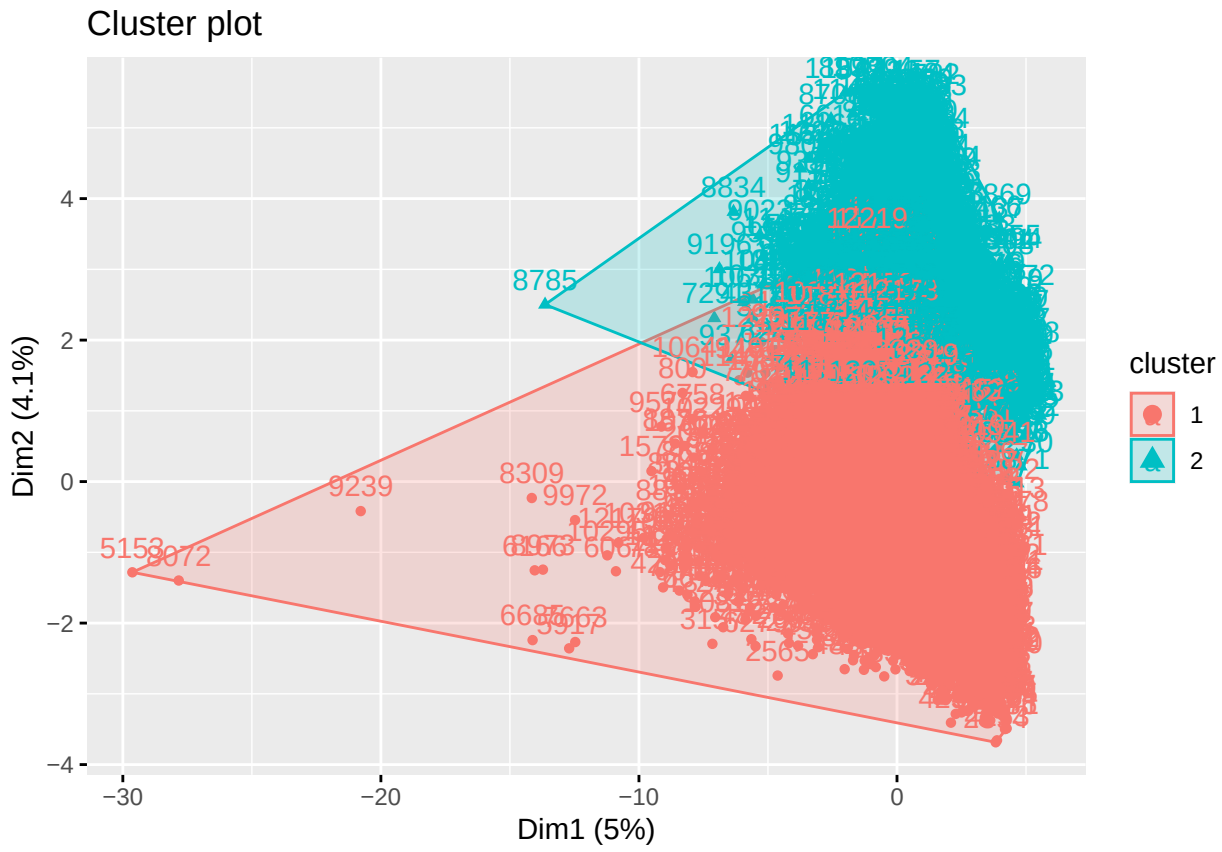
```
## [1] 9547 2571
```

```
# viewing the cluster center datapoints by each attribute
means.silo$centers
```

```
##        Admin Admin_Duration       Info Info_Duration Prod_Related
## 1 0.08852888     0.02441004 0.02170403    0.01481089   0.04860983
## 2 0.08068629     0.02413322 0.01975561    0.01151186   0.03467481
```

```
##   Prod_Related_Duration Bounce_Rates Exit_Rates Page_Values Special_Day
## 1            0.02026956   0.09676651  0.2021862  0.01642786  0.06496282
## 2            0.01406929   0.12093908  0.2247769  0.01540744  0.05281991
##     MonthAug  MonthDec   MonthFeb   MonthJul  MonthJune  MonthMar  MonthMay
## 1 0.03540379 0.1332356 0.01204567 0.03582277 0.02367236 0.1553368 0.2824971
## 2 0.03695060 0.1462466 0.02605990 0.03500583 0.02255932 0.1439129 0.2454298
##    MonthNov   MonthOct   MonthSep Operating_Systems.1 Operating_Systems.2
## 1 0.2416466 0.04158374 0.03875563          0.02922384         0.681365874
## 2 0.2543757 0.05912096 0.03033839          0.87981330         0.005834306
##   Operating_Systems.3 Operating_Systems.4 Operating_Systems.5
## 1        0.2640620090           0.0207395        0.0006284697
## 2        0.0003889537           0.1089070        0.0000000000
##   Operating_Systems.6 Operating_Systems.7 Operating_Systems.8    Browser.1
## 1         0.001990154         0.000000000         0.001990154 0.0001047449
## 2         0.000000000         0.002722676         0.002333722 0.9408790354
##   Browser.2  Browser.3   Browser.4   Browser.5   Browser.6   Browser.7
## 1  0.822981 0.01099822 0.075835341 0.047973185 0.017806641 0.005132502
## 2  0.000000 0.00000000 0.001166861 0.002333722 0.001555815 0.000000000
##    Browser.8    Browser.9  Browser.10   Browser.11  Browser.12   Browser.13
## 1 0.00000000 0.0001047449 0.016654446 0.0006284697 0.001047449 0.0007332146
## 2 0.05250875 0.0000000000 0.001166861 0.0000000000 0.000000000 0.0003889537
##    Region.1   Region.2  Region.3   Region.4   Region.5   Region.6   Region.7
## 1 0.3797004 0.09489892 0.1922070 0.09343249 0.02922384 0.06735100 0.06881743
## 2 0.4192921 0.08401400 0.2096461 0.10540646 0.01478024 0.06067678 0.03850642
##     Region.8   Region.9 Traffic.1 Traffic.2 Traffic.3  Traffic.4  Traffic.5
## 1 0.03446109 0.03990782 0.2073950 0.3171677 0.1450718 0.08725254 0.02073950
## 2 0.03928433 0.02839362 0.1540257 0.3376118 0.2430961 0.09062622 0.02372618
##    Traffic.6   Traffic.7  Traffic.8   Traffic.9 Traffic.10 Traffic.11
## 1 0.04158374 0.003561328 0.02597675 0.002199644 0.03498481 0.02042526
## 2 0.01750292 0.002333722 0.03656165 0.007779074 0.04395177 0.01983664
##     Traffic.12  Traffic.13  Traffic.14  Traffic.15   Traffic.16   Traffic.17
## 1 0.0001047449 0.075835341 0.001361684 0.001256939 0.0003142348 0.0000000000
## 2 0.0000000000 0.001166861 0.000000000 0.009334889 0.0000000000 0.0003889537
##    Traffic.18  Traffic.19 Traffic.20 VisitorNew_Visitor
## 1 0.001047449 0.001466429 0.01225516          0.1348067
## 2 0.000000000 0.001166861 0.01089070          0.1579152
##   VisitorReturning_Visitor WeekendFALSE WeekendTRUE
## 1                0.8651933    0.7794071   0.2205929
## 2                0.8420848    0.7110074   0.2889926
```

```
#Visualizing the Clusters

#clusplot(norm.kira, means.silo$cluster, color= T, shade = T, labels =0, lines = 0  )

fviz_cluster(means.silo, data = norm.kira)
```

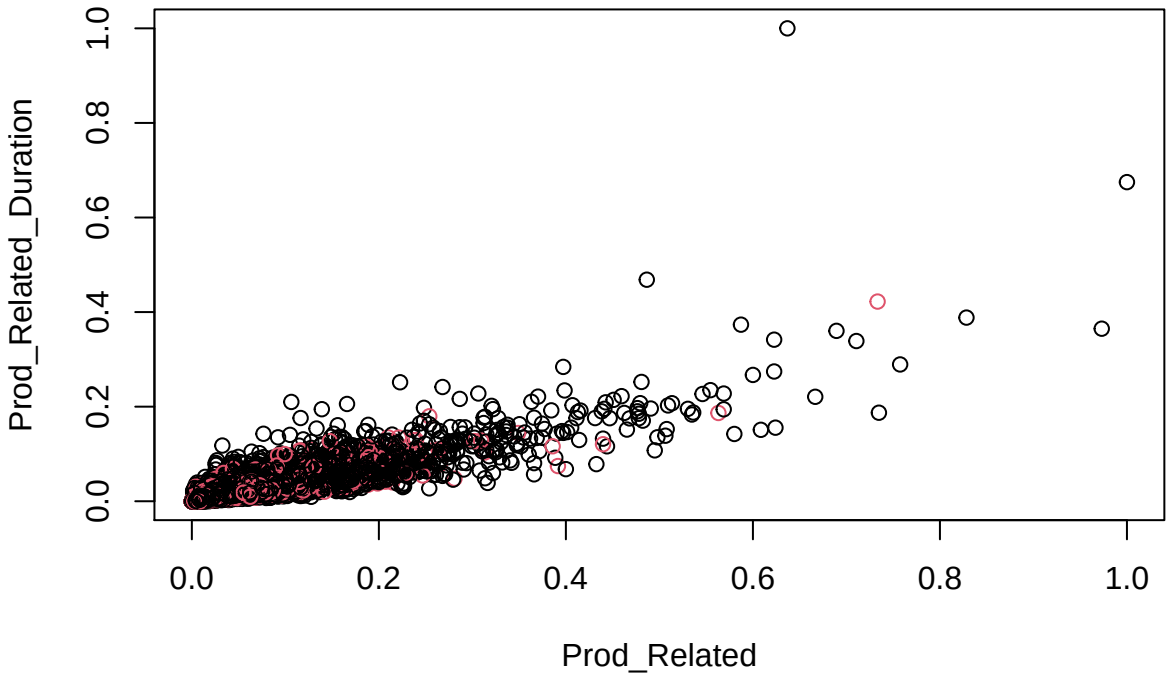



```
#Distribution of revenue classes
table(means.silo$cluster, kira$Revenue)
```

```
##
##      FALSE TRUE
##    1  8043 1504
##    2  2183  388
```

```
#Our Model classifies False revenues correctly with an accuracy of 84.2%
#
```

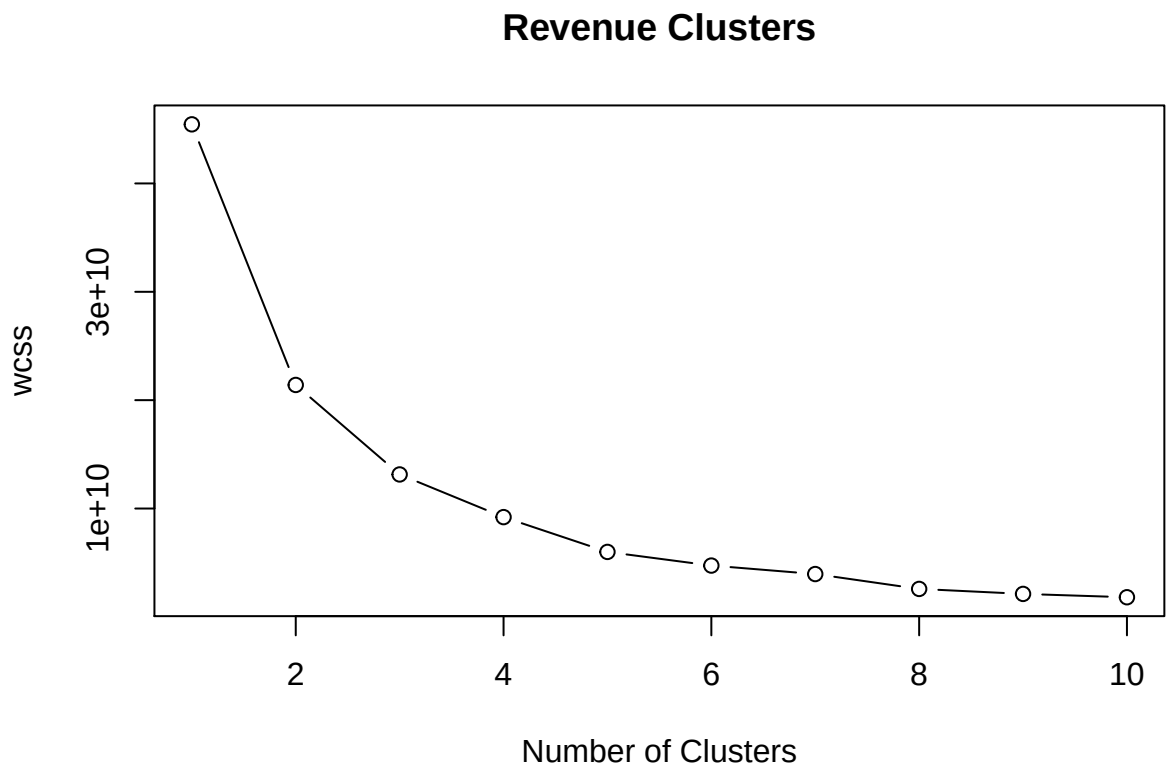
```
# Comparing variables & the distribution of their data points in the cluster
# Product Related, vs Product Related Duration

plot(norm.kira[, 5:6], col = means.silo$cluster)
```



Would a different method of determining Clusters change the results?

Determining optimal number of clusters using the elbow method

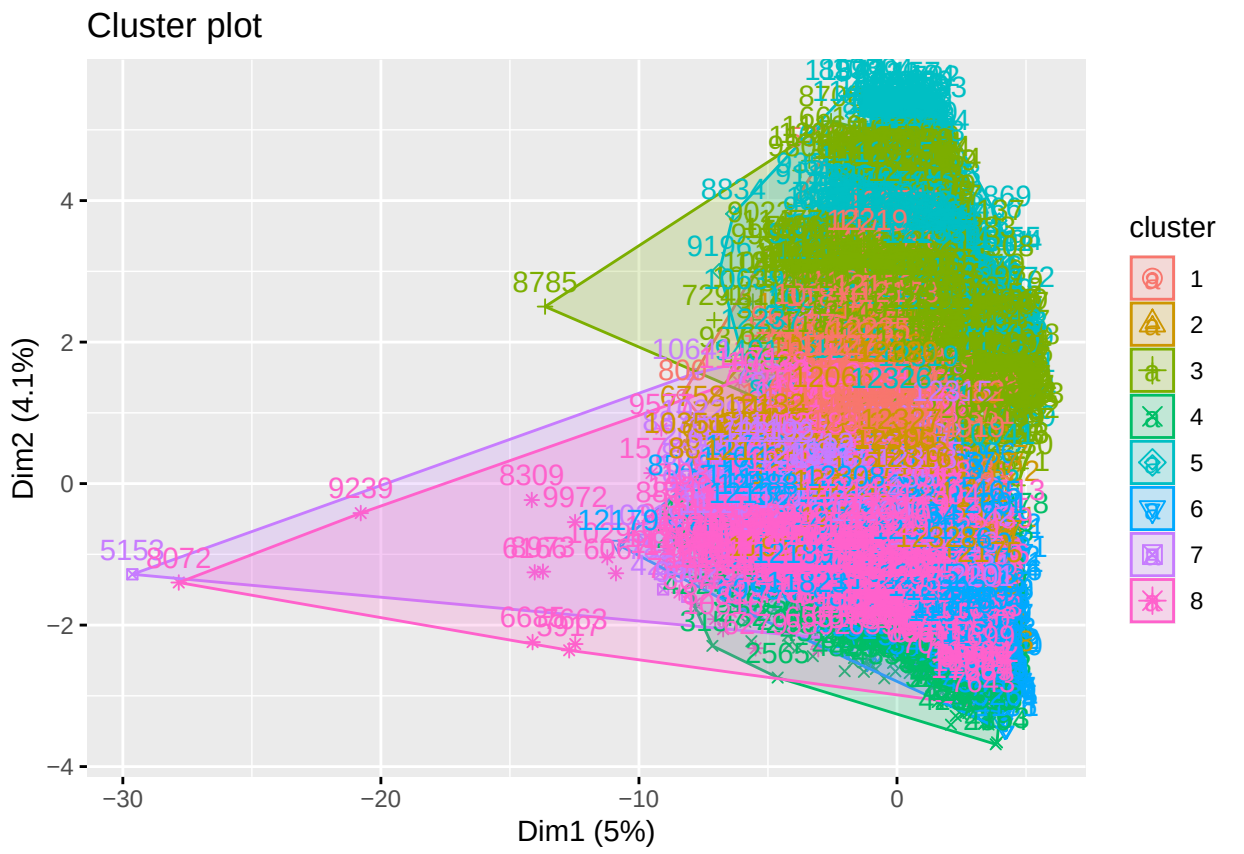
```
set.seed(101)

wcss <- vector()
for (i in 1:10) wcss[i] <- sum(kmeans(kira_indep, i)$withinss)
plot(1:10, wcss, type = "b",main = paste('Revenue Clusters'), xlab = "Number of Clusters",
     ylab = "wcss")
```




```
#According to the elbow method the optimal number of k is 8
```

Checking Results with K=8

```
wcss.means <- kmeans(norm.kira, 8, nstart = 25)


fviz_cluster(wcss.means, data = norm.kira)
```




```
#WIth 8 centroids our model does not do a good job in classifying the labels
```

Challenging the Solution

Implementing a second Clustering Model and comparing our results

4 (b)Hierarchical Clustering

It addresses one of the key disadvantages of K-means clustering by pre-specifying the number of clusters K. We therefore do not have to commit to a particular choice of K

1. Computing the Euclidean distance btn datapoints

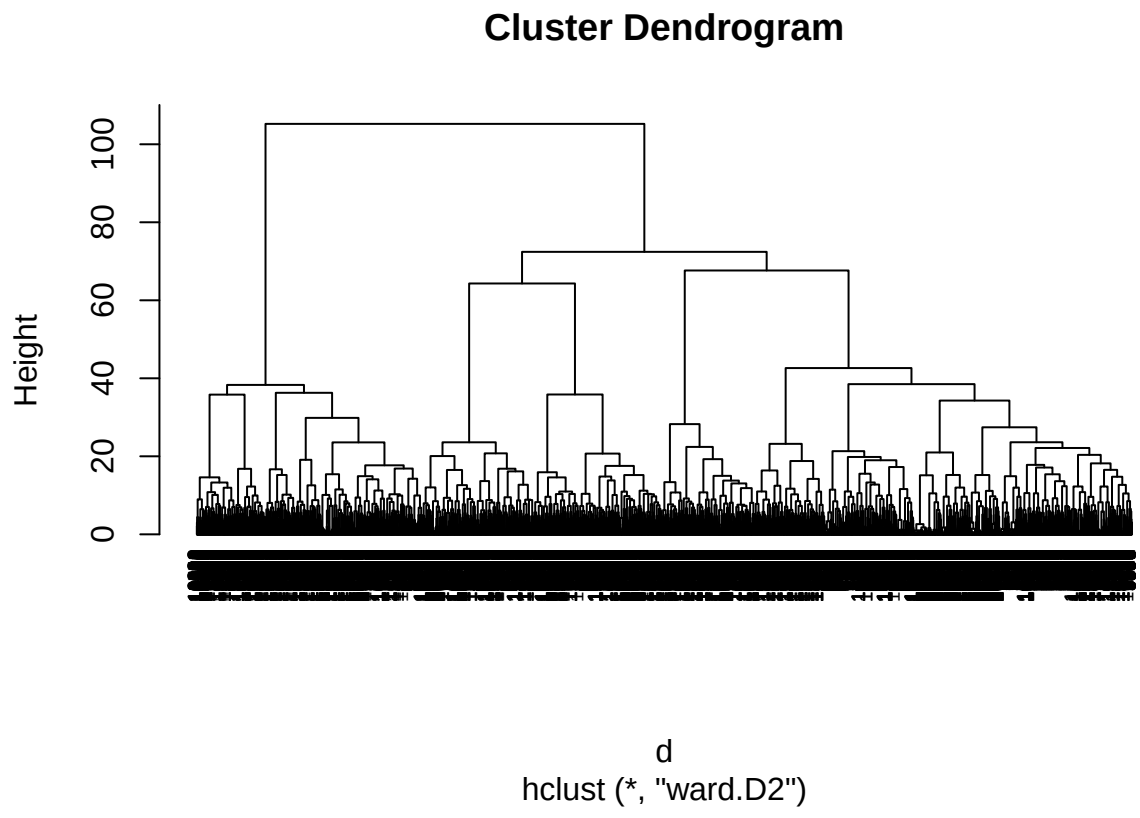
```
# We use the dist() function
#Important that the Data is Scaled to ensure the datapoints
#We will use the already scaled data
d <- dist( norm.kira, method = "euclidean")
```

2. perform hierarchical clustering using the Ward's method

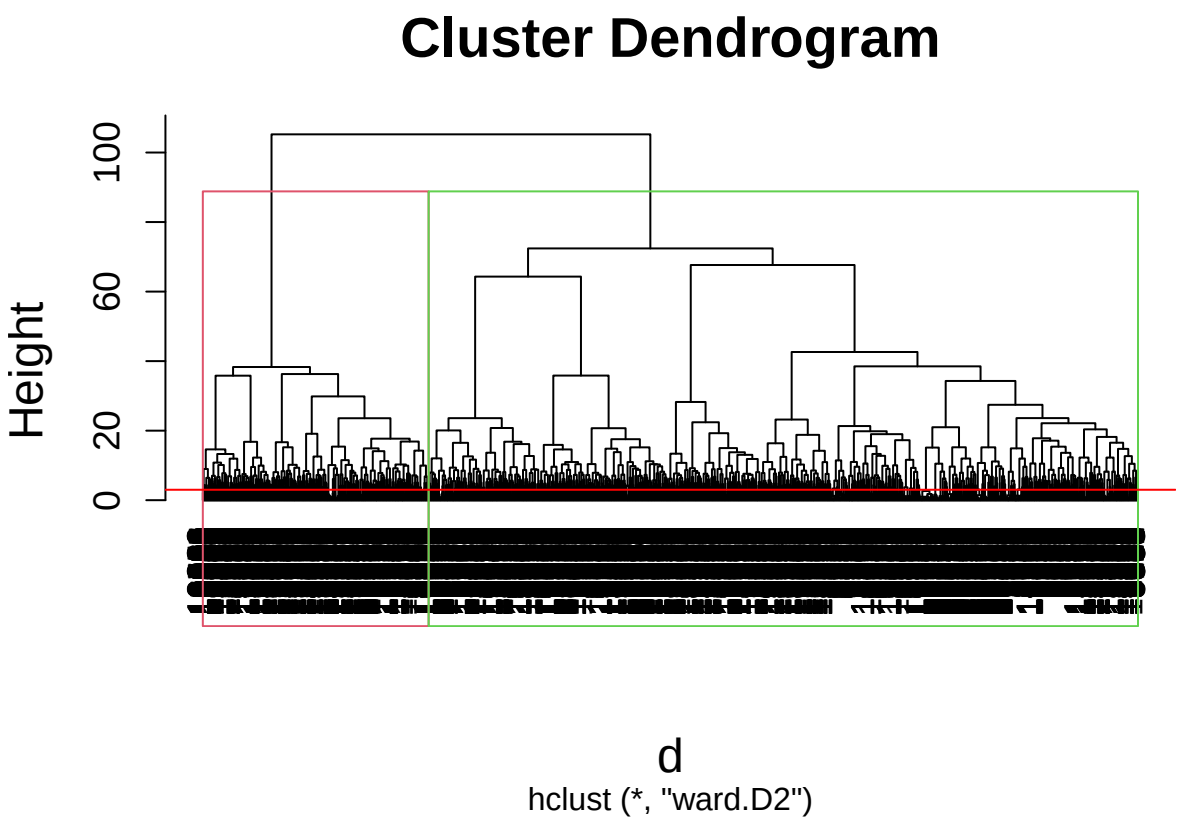
```
res.hc <- hclust(d, method = "ward.D2" )
```

3. plot the obtained dendrogram

```
options(repr.plot.width = 15, repr.plot.height = 8)

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




```
#We have 2 clusters according to the Dendrogram
# see the clusters on the dendrogram using R's abline() function to draw the cut line and
# superimpose rectangular compartments for each cluster on the tree with the rect.hclust() function

options(repr.plot.width = 15, repr.plot.height = 15)
plot(res.hc, cex = 1, hang =-1, cex.main = 1.75, cex.lab=1.5, cex.axis=1.2)
rect.hclust(res.hc ,k = 2,  border = 2:5)
abline(h = 3, col = 'red')
```



Number of Records in each cluster:

```
# find the number of records in each cluster
cut <- cutree(res.hc, k = 2)
table (cut)
```

```
## cut
##    1    2
## 2926 9192
```

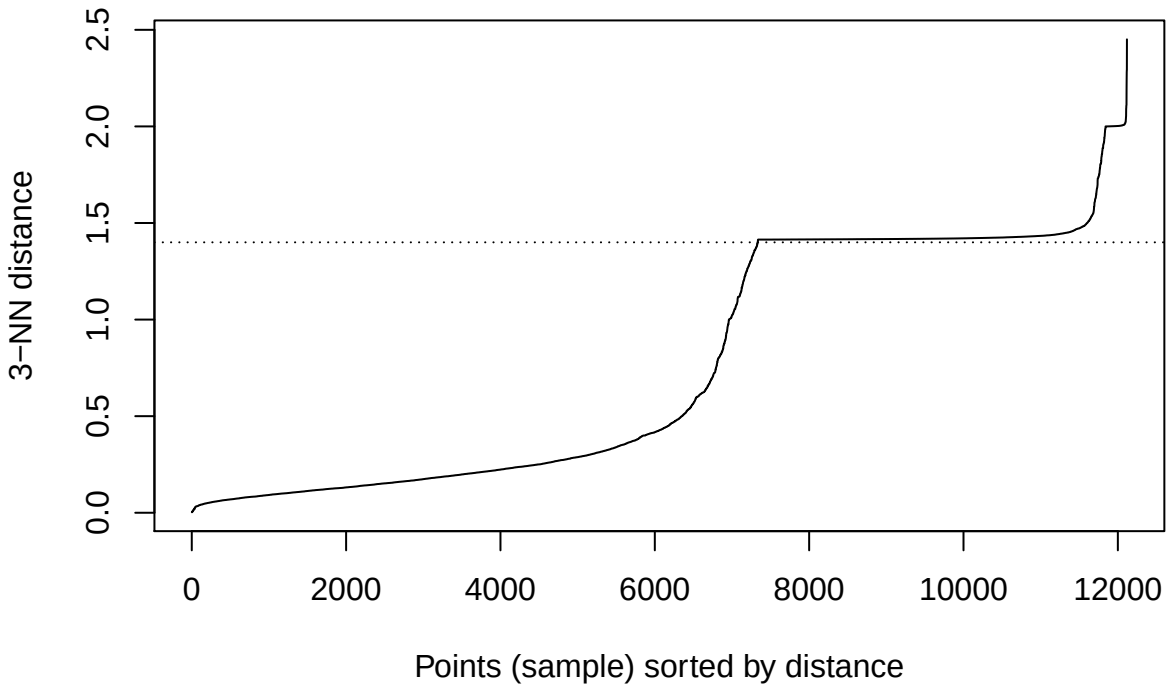
Density Based Spatial Clustering of Applications with Noise (DBSCAN) c) Would dbscan yield better results?

In theory, yes. DBSCAN works well with data with many outliers.

Implementation

1. Obtaining the Optimal eps value

```
kNNdistplot(norm.kira, k=3)
abline(h= 1.40, lty=3)
```

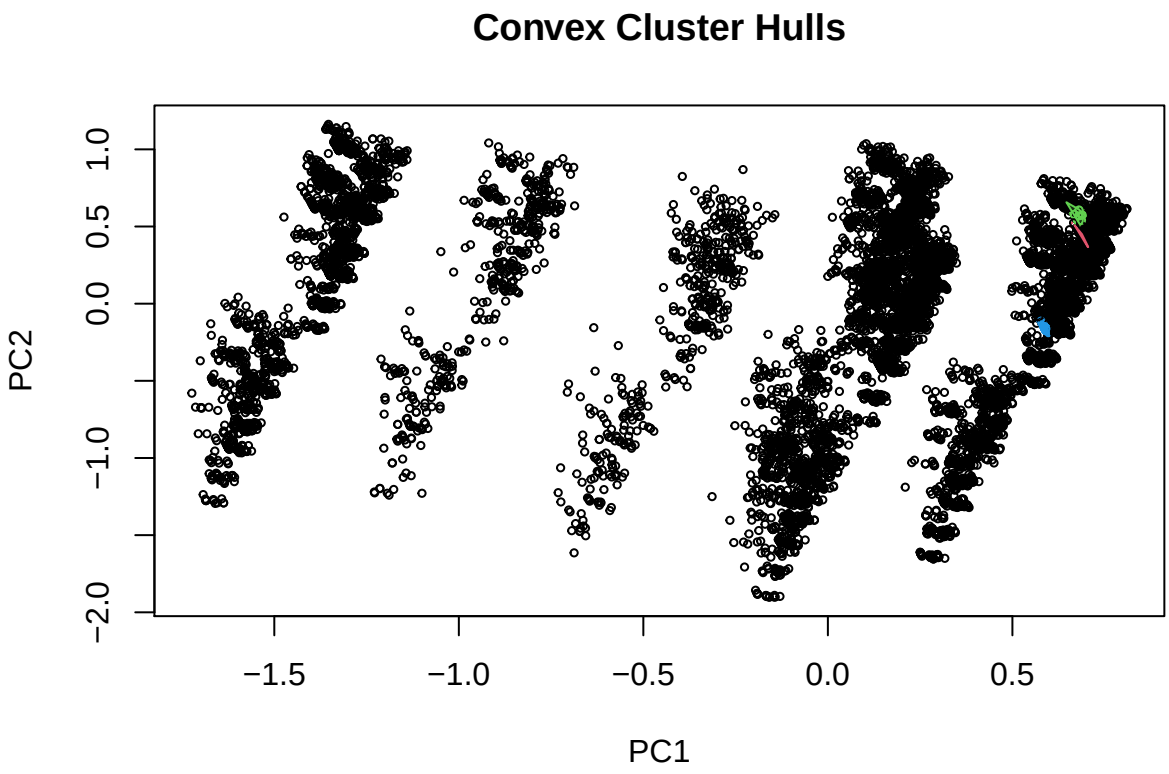


2. Applying the DBSCAN algorithm

```
#We adjust the Minimum distance between points till we get sensible classes
dbscan.model <- dbscan(norm.kira, eps = 1.40, MinPts = 84 )

print(dbscan.model)
```

```
## dbscan Pts=12118 MinPts=84 eps=1.4
##            0  1  2   3
## border 11763  2 24   0
## seed       0 87 61 181
## total  11763 89 85 181
```

```
#Plotting the CLusters

hullplot(norm.kira, dbscan.model$cluster)
```




```
#0 means Noise- All those points that could not be clustered are recorded as noise
#97% of the data points are classed as noise

#CLuster 1 has 89 data points, cluster 2 has 85 data points and cluster 3 has 181 data points
```

```
#Density Based CLustering with FPC
#set.seed(101)
#f <- fpc::dbscan(norm.kira, eps= 0.4, MinPts = 4)

#f

#CLuster Visualization
#fviz_cluster(f, norm.kira, geom ="point")
```

Follow Up Questions

1. Did we have the right data? Yes, we did.

2 Did we ask the right question? Yes, we did.

Conclusion

In this project, we sought to build an unsupervised learning model that would accurately classify the revenue attribute. Such a model would need to have accuracy levels of 95%

Unfortunately, the best our model, the K-Means algorithm could deliver is around 85%. We therefore did not meet our objective

The DBSACAN model does not perform as well as we expected

Recommendation Dimensionality Reduction - Perhaps a reduction of the dimension would allow for better generalization of the data and consequently better classification. PCA would be a potential solution