Week 13 Unsupervised Learning PART 2

Brian Onchweri

5/30/2022

#Kira Plastinina Online Shop Analysis

##Specifying The Question 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.

##Metrics of success 1.Successfully Perform clustering stating insights drawn from the analysis and visualizations below.

2.Upon implementation, successfully giving comparisons between the approaches i.e. K-Means clustering vs Hierarchical clustering highlighting the strengths and limitations of each approach in the context of the analysis below.

Findings from this analysis should help inform the team in formulating the marketing and sales strategies of the brand.

##Understanding the context We will be using data collected from an E-Commerce site. E-commerce is the buying and selling of goods and services, or the transmitting of funds or data, over an electronic network, primarily the internet.

##Experimental design i)Loading the data ii)Check the Data iii)Perform Data Cleaning iv)Perform Exploratory Data Analysis (Univariate, Bivariate & Multivariate) v)Implement the Solution vi)Conclusion

#Loading libraries

```
library(dplyr)
```

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

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

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

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

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

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

```
library(psych)
library(ggplot2)
```

```
##
## Attaching package: 'ggplot2'

## The following objects are masked from 'package:psych':
##
##     %+%, alpha
```

```
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
library(tidyverse)
```

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

## v tibble  3.1.7     v purrr   0.3.4
## v tidyr   1.2.0     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x ggplot2::%+%()        masks psych::%+%()
## x ggplot2::alpha()      masks psych::alpha()
## x data.table::between() masks dplyr::between()
## x tidyr::extract()      masks magrittr::extract()
## x dplyr::filter()       masks stats::filter()
## x data.table::first()   masks dplyr::first()
## x dplyr::lag()          masks stats::lag()
## x data.table::last()    masks dplyr::last()
## x purrr::set_names()    masks magrittr::set_names()
## x purrr::transpose()    masks data.table::transpose()
```

```
library(dummy)
```

```
## dummy 0.1.3
```

```
## dummyNews()
```

```
library(caret)
```

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

```
##
## Attaching package: 'caret'
```

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

#Reading the url

```
shop = read.csv("http://bit.ly/EcommerceCustomersDataset")
```

```
# checking the head of our data
head(shop)
```

```
##   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 the Data structure of the Dataset
str(shop)
```

```
## '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 ...
```

```
# checking the number of observations and features
dim(shop)
```

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

```
#Checking for missing values
colSums(is.na(shop))
```

```
##          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
```

```
#Checking the percentage of missing values in the Dataset. We can observe that we have missing values f

pMiss <- function(x){sum(is.na(x))/length(x)*100}
apply(shop,2,pMiss)
```

```
##          Administrative Administrative_Duration           Informational
##               0.1135442               0.1135442               0.1135442
##  Informational_Duration          ProductRelated ProductRelated_Duration
##               0.1135442               0.1135442               0.1135442
##             BounceRates              ExitRates             PageValues
##               0.1135442               0.1135442               0.0000000
```

```
##              SpecialDay                  Month        OperatingSystems
##               0.0000000              0.0000000               0.0000000
##                 Browser                 Region            TrafficType
##               0.0000000              0.0000000               0.0000000
##             VisitorType                Weekend                Revenue
##               0.0000000              0.0000000               0.0000000
```

```
#Doing without missing values
shop = na.omit(shop)

# Checking for missing values

colSums(is.na(shop))
```

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

```
#Checking the number of rows we're working with
nrow(shop)
```

```
## [1] 12316
```

```
# Checking for duplicates
duplicates <- shop[duplicated(shop),]
dim(duplicates)
```

```
## [1] 117  18
```

```
# removing duplicates
shop <- shop[!duplicated(shop),]
dim(shop)
```

```
## [1] 12199    18
```

```
# Check for unique values in month and visitor type columns
unique(shop$Month);
```

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

```
unique(shop$VisitorType);
```

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

##Anomalies can be detected in some of the columns where we have negative intergers where its not ideal to have such. For instance "Informational_duration" has a value of -1. This is not ideal since we cannot have negative time.

```
# Check the number of records with this anomaly

anomaly <- shop %>% select(c(Administrative_Duration, Administrative, Informational_Duration, Informati

anomaly
```

```
##    Administrative_Duration Administrative Informational_Duration Informational
## 1                       -1              0                     -1             0
## 2                       -1              0                     -1             0
## 3                       -1              1                     -1             0
## 4                       -1              0                     -1             0
## 5                       -1              0                     -1             0
## 6                       -1              0                     -1             0
## 7                       -1              0                     -1             0
## 8                       -1              0                     -1             0
## 9                       -1              0                     -1             0
## 10                      -1              0                     -1             0
## 11                      -1              0                     -1             0
## 12                      -1              0                     -1             0
## 13                      -1              0                     -1             0
## 14                      -1              0                     -1             0
## 15                      -1              0                     -1             0
## 16                      -1              0                     -1             0
## 17                      -1              0                     -1             0
## 18                      -1              0                     -1             0
## 19                      -1              0                     -1             0
## 20                      -1              0                     -1             0
## 21                      -1              0                     -1             0
## 22                      -1              0                     -1             0
## 23                      -1              0                     -1             0
## 24                      -1              0                     -1             0
## 25                      -1              0                     -1             0
## 26                      -1              0                     -1             0
## 27                      -1              0                     -1             0
## 28                      -1              0                     -1             0
## 29                      -1              1                     -1             0
## 30                      -1              0                     -1             0
## 31                      -1              0                     -1             0
## 32                      -1              0                     -1             0
## 33                      -1              0                     -1             0
##    ProductRelated_Duration ProductRelated
## 1                       -1              1
## 2                       -1              1
## 3                       -1              1
```

```
## 4                        -1              1
## 5                        -1              1
## 6                        -1              1
## 7                        -1              1
## 8                        -1              1
## 9                        -1              1
## 10                       -1              1
## 11                       -1              1
## 12                       -1              1
## 13                       -1              1
## 14                       -1              1
## 15                       -1              1
## 16                       -1              1
## 17                       -1              1
## 18                       -1              1
## 19                       -1              1
## 20                       -1              1
## 21                       -1              1
## 22                       -1              1
## 23                       -1              1
## 24                       -1              1
## 25                       -1              1
## 26                       -1              1
## 27                       -1              1
## 28                       -1              1
## 29                       -1              1
## 30                       -1              1
## 31                       -1              1
## 32                       -1              1
## 33                       -1              1
```

```
# Dropping the records with these anomalies.
shop <- shop %>% filter(Administrative_Duration != -1, Informational_Duration != -1, ProductRelated_Dura
```

```
# checking the remaining observations in our data
dim(shop)
```

```
## [1] 12164    18
```

##Checking for Outliers

```
outlier_tool <- function(x){
  out <- boxplot.stats(x)$out
  return((length(out)/  12164)*100)
}
```

##counting number of outliers per column

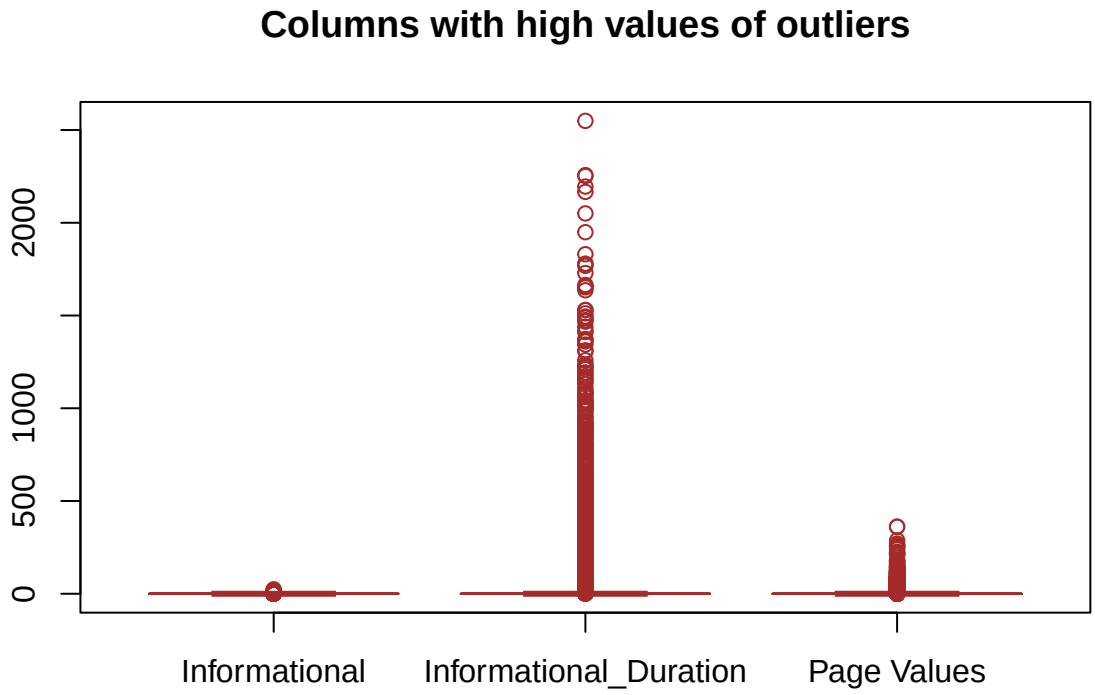
```
sapply(shop[,c(1:9)], outlier_tool)
```

```
##          Administrative Administrative_Duration           Informational
##                3.321276                9.363696               21.621177
```

```
##  Informational_Duration          ProductRelated ProductRelated_Duration
##                19.763236                8.278527                7.809931
##             BounceRates               ExitRates              PageValues
##                11.649129               10.637948                22.443275
```

##Plotting a boxplot to check for these outliers

```
# Plot boxplots of columns with high % of outliers
boxplot(shop$Informational, shop$Informational_Duration, shop$PageValues,
main = "Columns with high values of outliers",
names = c("Informational", "Informational_Duration", "Page Values"),
col = c("orange","blue"),
border = "brown",
notch = TRUE)
```



#Univariate Analysis

```
# checking the summary statistics of each column
summary(shop)
```

```
##  Administrative   Administrative_Duration Informational
##  Min.   : 0.000   Min.   :   0.00         Min.   : 0.0000
##  1st Qu.: 0.000   1st Qu.:   0.00         1st Qu.: 0.0000
##  Median : 1.000   Median :  10.00         Median : 0.0000
##  Mean   : 2.347   Mean   :  81.92         Mean   : 0.5103
##  3rd Qu.: 4.000   3rd Qu.:  95.00         3rd Qu.: 0.0000
```

```
##  Max.   :27.000   Max.   :3398.75         Max.   :24.0000
##  Informational_Duration ProductRelated   ProductRelated_Duration
##  Min.   :   0.00        Min.   :  0.00   Min.   :    0.0
##  1st Qu.:   0.00        1st Qu.:  8.00   1st Qu.:  196.5
##  Median :   0.00        Median : 18.00   Median :  613.2
##  Mean   :  34.94        Mean   : 32.15   Mean   : 1211.0
##  3rd Qu.:   0.00        3rd Qu.: 38.00   3rd Qu.: 1482.0
##  Max.   :2549.38        Max.   :705.00   Max.   :63973.5
##   BounceRates         ExitRates         PageValues       SpecialDay
##  Min.   :0.000000   Min.   :0.00000   Min.   :  0.00   Min.   :0.00000
##  1st Qu.:0.000000   1st Qu.:0.01417   1st Qu.:  0.00   1st Qu.:0.00000
##  Median :0.002865   Median :0.02500   Median :  0.00   Median :0.00000
##  Mean   :0.020001   Mean   :0.04108   Mean   :  5.97   Mean   :0.06202
##  3rd Qu.:0.016318   3rd Qu.:0.04804   3rd Qu.:  0.00   3rd Qu.:0.00000
##  Max.   :0.200000   Max.   :0.20000   Max.   :361.76   Max.   :1.00000
##     Month           OperatingSystems    Browser          Region
##  Length:12164       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   :2.125    Mean   : 2.358   Mean   :3.153
##                     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.   : 1.000   Length:12164       Mode :logical   Mode :logical
##  1st Qu.: 2.000   Class :character   FALSE:9311      FALSE:10256
##  Median : 2.000   Mode  :character   TRUE :2853      TRUE :1908
##  Mean   : 4.076
##  3rd Qu.: 4.000
##  Max.   :20.000
```

```
# descriptive statistics of our columns
describe(shop)
```

```
## Warning in FUN(newX[, i], ...): no non-missing arguments to min; returning Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to min; returning Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning -Inf

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning -Inf

##                          vars     n    mean      sd median trimmed    mad min
## Administrative              1 12164    2.35    3.33   1.00    1.66   1.48   0
## Administrative_Duration     2 12164   81.92  177.73  10.00   43.06  14.83   0
## Informational               3 12164    0.51    1.28   0.00    0.18   0.00   0
## Informational_Duration      4 12164   34.94  141.65   0.00    3.76   0.00   0
## ProductRelated              5 12164   32.15   44.63  18.00   23.14  19.27   0
## ProductRelated_Duration     6 12164 1210.99 1921.59 613.24  835.59 747.59   0
## BounceRates                 7 12164    0.02    0.04   0.00    0.01   0.00   0
## ExitRates                   8 12164    0.04    0.05   0.03    0.03   0.02   0
## PageValues                  9 12164    5.97   18.68   0.00    1.34   0.00   0
## SpecialDay                 10 12164    0.06    0.20   0.00    0.00   0.00   0
## Month*                     11 12164    6.17    2.38   7.00    6.36   1.48   1
```

```
## OperatingSystems          12 12164    2.12    0.91   2.00    2.06   0.00   1
## Browser                   13 12164    2.36    1.71   2.00    2.00   0.00   1
## Region                    14 12164    3.15    2.40   3.00    2.79   2.97   1
## TrafficType               15 12164    4.08    4.02   2.00    3.23   1.48   1
## VisitorType*              16 12164    2.71    0.69   3.00    2.89   0.00   1
## Weekend                   17 12164     NaN      NA     NA     NaN     NA Inf
## Revenue                   18 12164     NaN      NA     NA     NaN     NA Inf
##                              max    range  skew kurtosis    se
## Administrative             27.00    27.00  1.94     4.62  0.03
## Administrative_Duration  3398.75  3398.75  5.58    49.97  1.61
## Informational              24.00    24.00  4.01    26.56  0.01
## Informational_Duration   2549.38  2549.38  7.53    75.23  1.28
## ProductRelated            705.00   705.00  4.33    31.01  0.40
## ProductRelated_Duration 63973.52 63973.52  7.25   136.43 17.42
## BounceRates                0.20     0.20  3.21     9.71  0.00
## ExitRates                  0.20     0.20  2.26     4.79  0.00
## PageValues               361.76   361.76  6.34    64.75  0.17
## SpecialDay                 1.00     1.00  3.28     9.78  0.00
## Month*                    10.00     9.00 -0.83    -0.37  0.02
## OperatingSystems           8.00     7.00  2.03    10.29  0.01
## Browser                   13.00    12.00  3.22    12.56  0.02
## Region                     9.00     8.00  0.98    -0.16  0.02
## TrafficType               20.00    19.00  1.96     3.45  0.04
## VisitorType*               3.00     2.00 -2.04     2.21  0.01
## Weekend                    -Inf     -Inf    NA       NA    NA
## Revenue                    -Inf     -Inf    NA       NA    NA
```

```
# Frequency distribution of the categorical variables
sapply(shop[, c(11:18)], table)
```

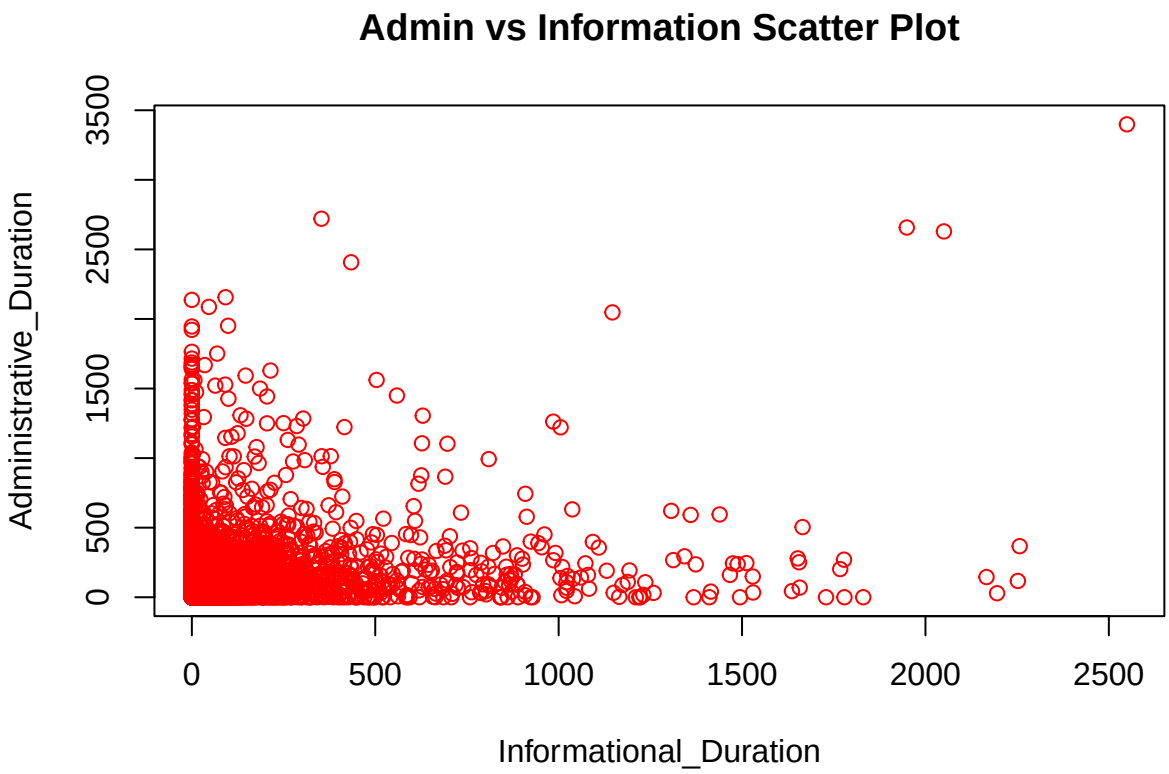
```
## $Month
##
##  Aug  Dec  Feb  Jul June  Mar  May  Nov  Oct  Sep
##  433 1706  169  431  285 1842 3321 2980  549  448
##
## $OperatingSystems
##
##    1    2    3    4    5    6    7    8
## 2539 6519 2523  476    6   19    7   75
##
## $Browser
##
##    1    2    3    4    5    6    7    8    9   10   11   12   13
## 2418 7859  104  727  464  174   49  134    1  162    6   10   56
##
## $Region
##
##    1    2    3    4    5    6    7    8    9
## 4701 1122 2374 1164  315  800  755  431  502
##
## $TrafficType
##
##    1    2    3    4    5    6    7    8    9   10   11   12   13   14   15   16
## 2373 3905 2002 1064  259  440   40  343   41  450  247    1  727   13   36    3
```

```
##   17   18   19   20
##    1   10   17  192
##
## $VisitorType
##
##       New_Visitor             Other Returning_Visitor
##              1693                81             10390
##
## $Weekend
##
## FALSE  TRUE
##  9311  2853
##
## $Revenue
##
## FALSE  TRUE
## 10256  1908
```

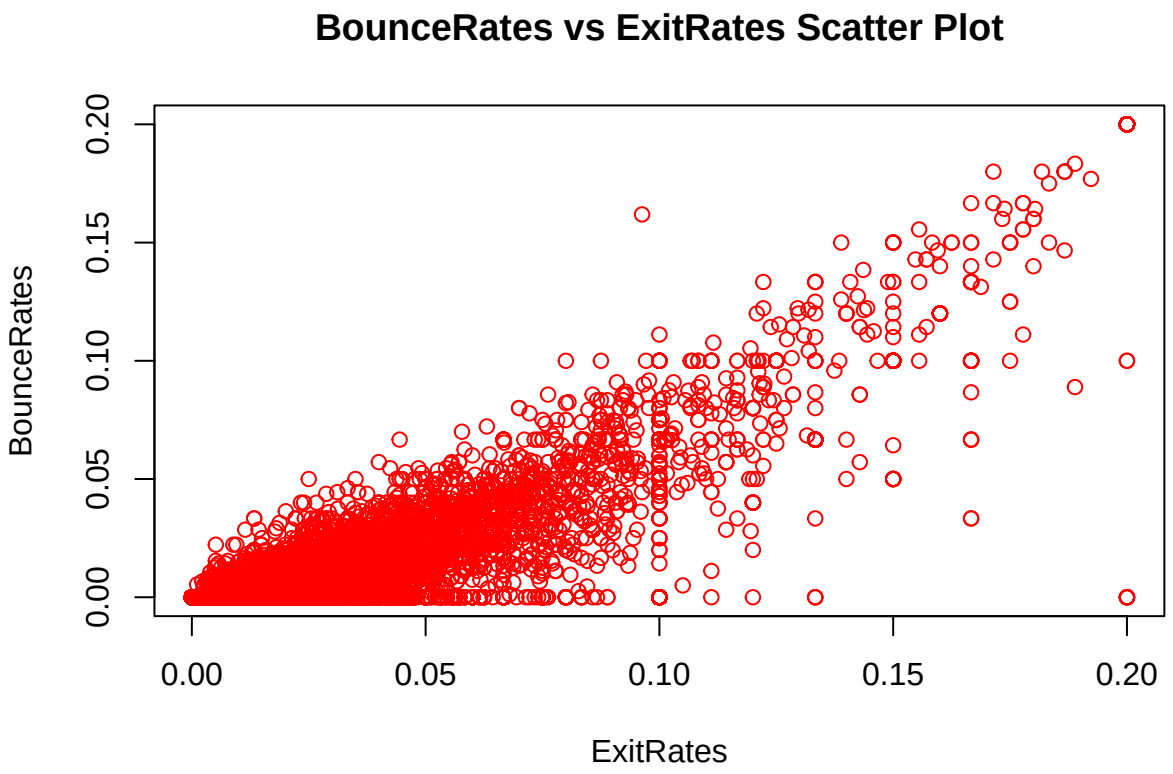
```
# Creating histogram plots to visually view the categorical variables
par(mfrow=c(4,1))
#for(i in 11:18) {
 #   counts <- table(shop[,i])
 #   name <- names(shop)[i]
 #   barplot(counts, main=name, col = heat.colors(20))}
```

#Multivariate Analysis

```
# Let's plot scatter plots
plot(Administrative_Duration ~ Informational_Duration, dat = shop,
      col = "red",
      main = "Admin vs Information Scatter Plot")
```




```
# Let's plot scatter plots
plot(BounceRates ~ ExitRates, dat = shop,
      col = "red",
      main = "BounceRates vs ExitRates Scatter Plot")
```




```
# Number of visits to product related pages per month
product_stats <- shop %>% select(ProductRelated, ProductRelated_Duration, Month)%>%group_by(Month)%>% su
product_stats[order(product_stats$ProductRelated, decreasing = TRUE),]
```

```
## # A tibble: 10 x 3
##    Month ProductRelated ProductRelated_Duration
##    <chr>          <dbl>                   <dbl>
##  1 Nov             46.3                   1769.
##  2 Aug             38.3                   1273.
##  3 Jul             36.5                   1220.
##  4 June            36.4                   1226.
##  5 Oct             33.6                   1117.
##  6 Sep             33.1                   1253.
##  7 Dec             28.3                   1125.
##  8 May             26.8                    995.
##  9 Mar             20.5                    841.
## 10 Feb             12.1                    513.
```

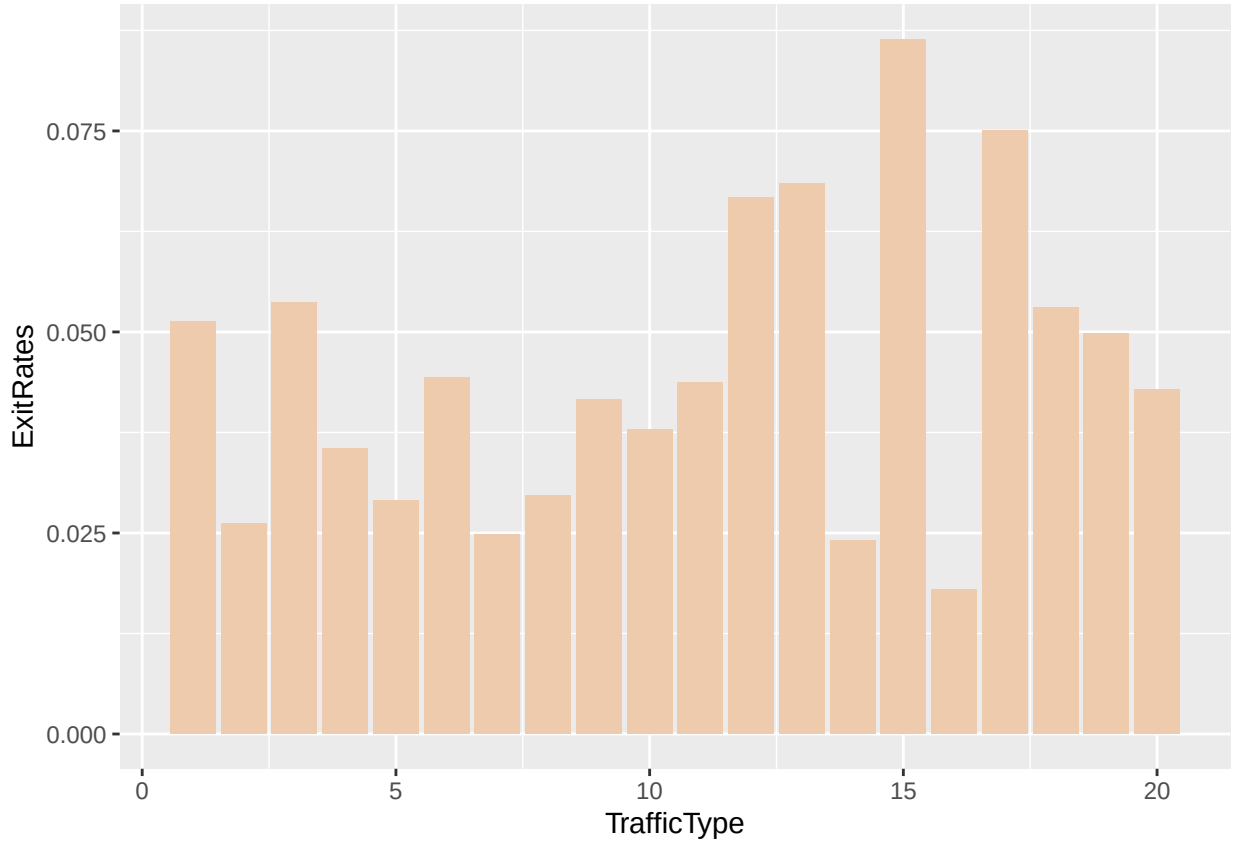
```
# Getting the  bounce rates and exit rates among visitor groups
visitor <- shop %>% select(VisitorType, ExitRates, BounceRates)%>% group_by(VisitorType)%>%summarise_al

visitor
```

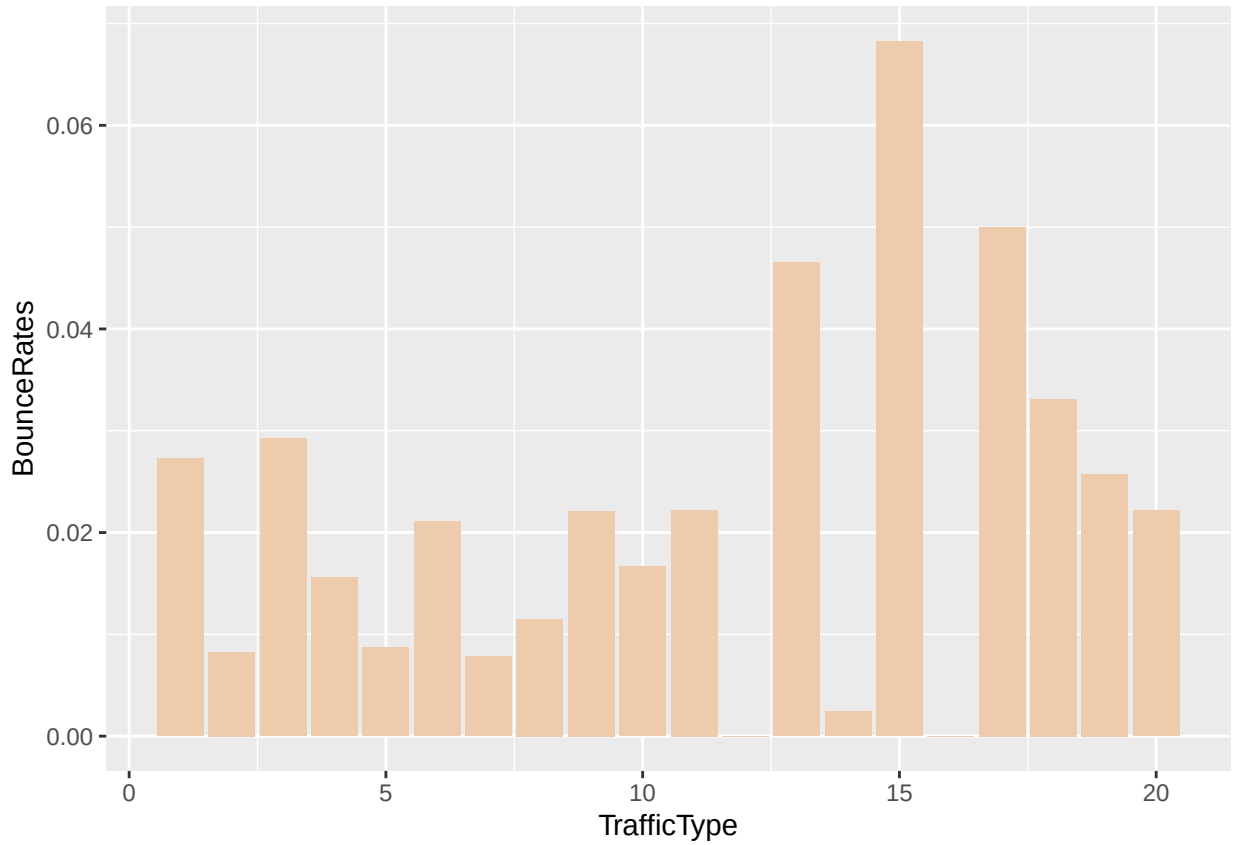
```
## # A tibble: 3 x 3
##   VisitorType       ExitRates BounceRates
```

```
##   <chr>                <dbl>       <dbl>
## 1 New_Visitor          0.0206     0.00515
## 2 Other                0.0566     0.0306
## 3 Returning_Visitor    0.0443     0.0223
```

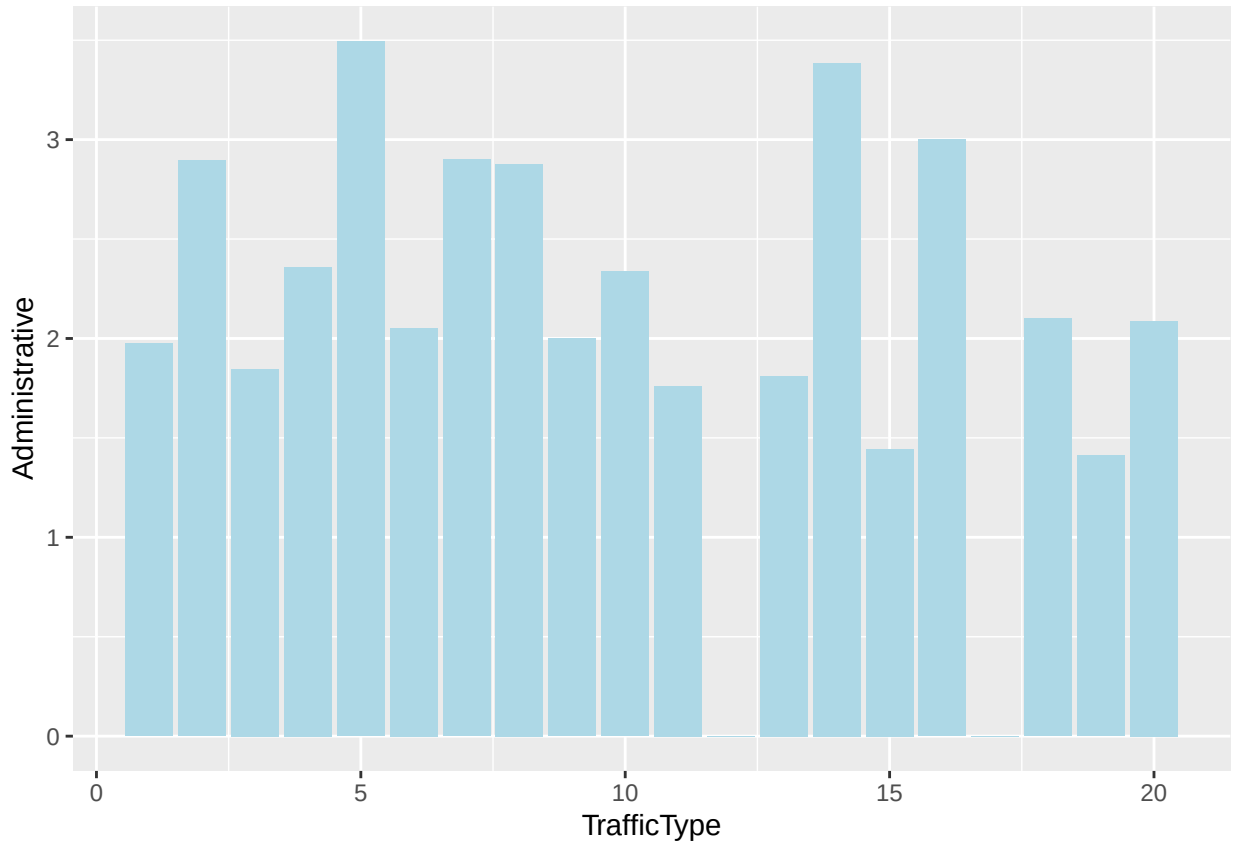
```r
# Creating a plot to show the ExitRate and BounceRatesin relation to the traffic type.
traffic <- shop %>% select(TrafficType, ExitRates, BounceRates)%>% group_by(TrafficType)%>% summarise_a
par(mfrow = c(1,2))
ggplot(traffic, aes(x=TrafficType, y = ExitRates))+
  geom_bar(stat = "identity", fill="peachpuff2")
```



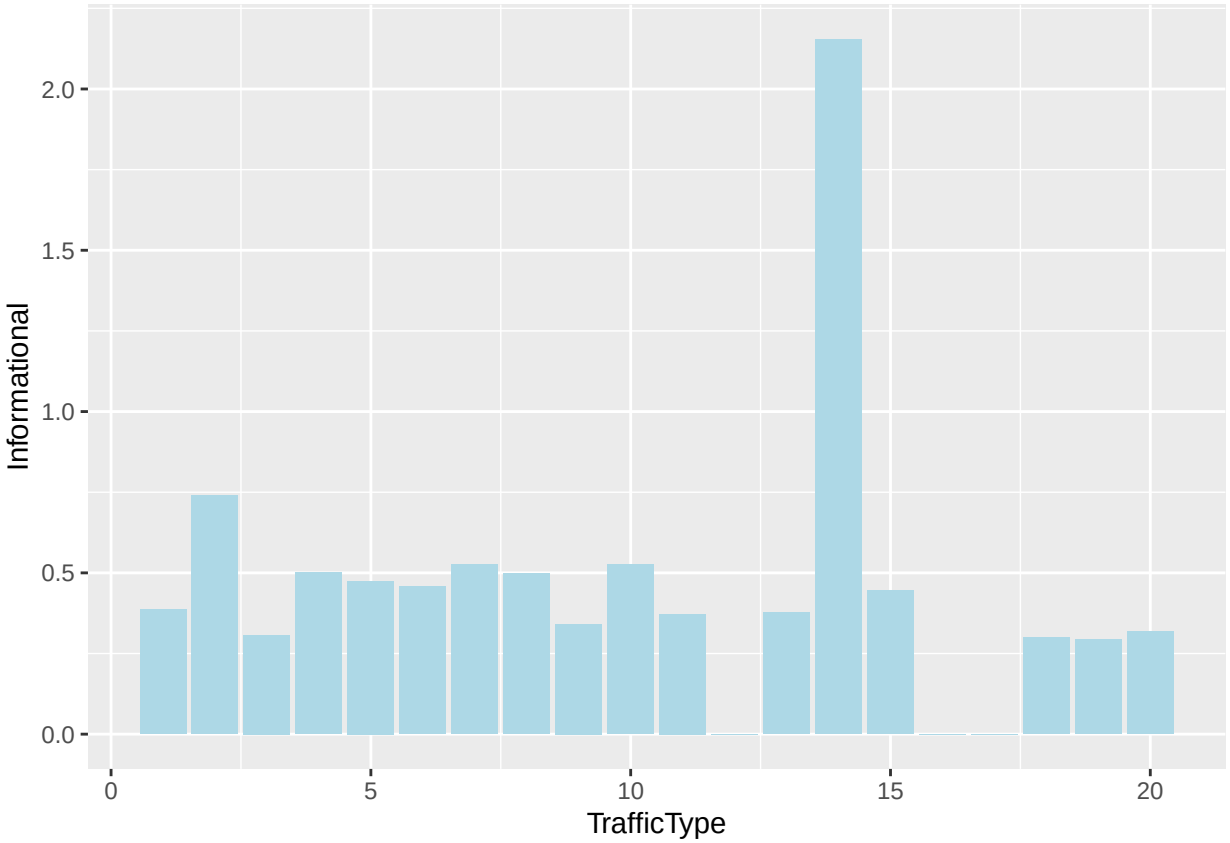

```r
ggplot(traffic, aes(x=TrafficType, y = BounceRates))+
  geom_bar(stat = "identity", fill="peachpuff2")
```



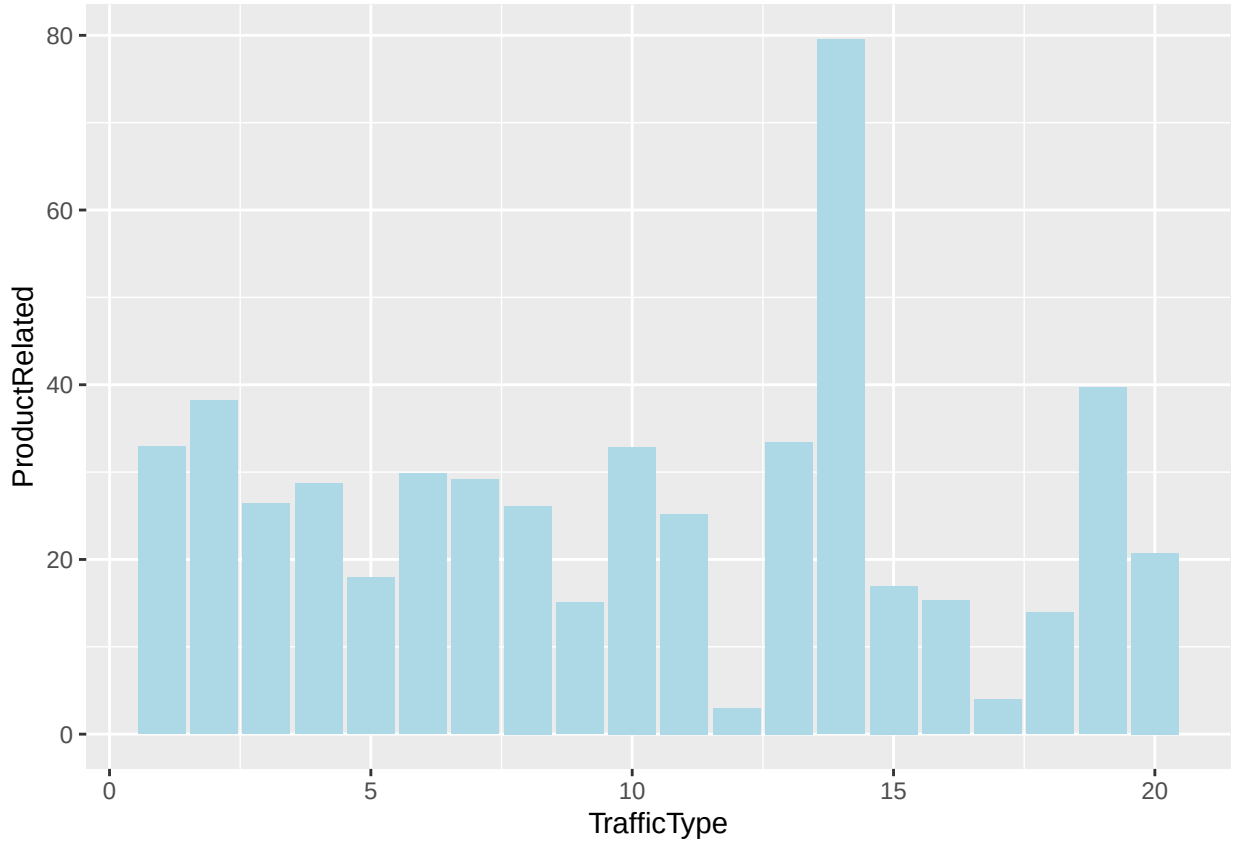

```
# Creating a plot to show the Administrative, ProductRelated and Informational relation to the traffic
traffic_page<- shop %>% select(TrafficType, Administrative,Informational,ProductRelated)%>% group_by(Tra
par(mfrow = c(1,3))
ggplot(traffic_page, aes(x=TrafficType, y = Administrative))+
  geom_bar(stat = "identity", fill="lightblue")
```



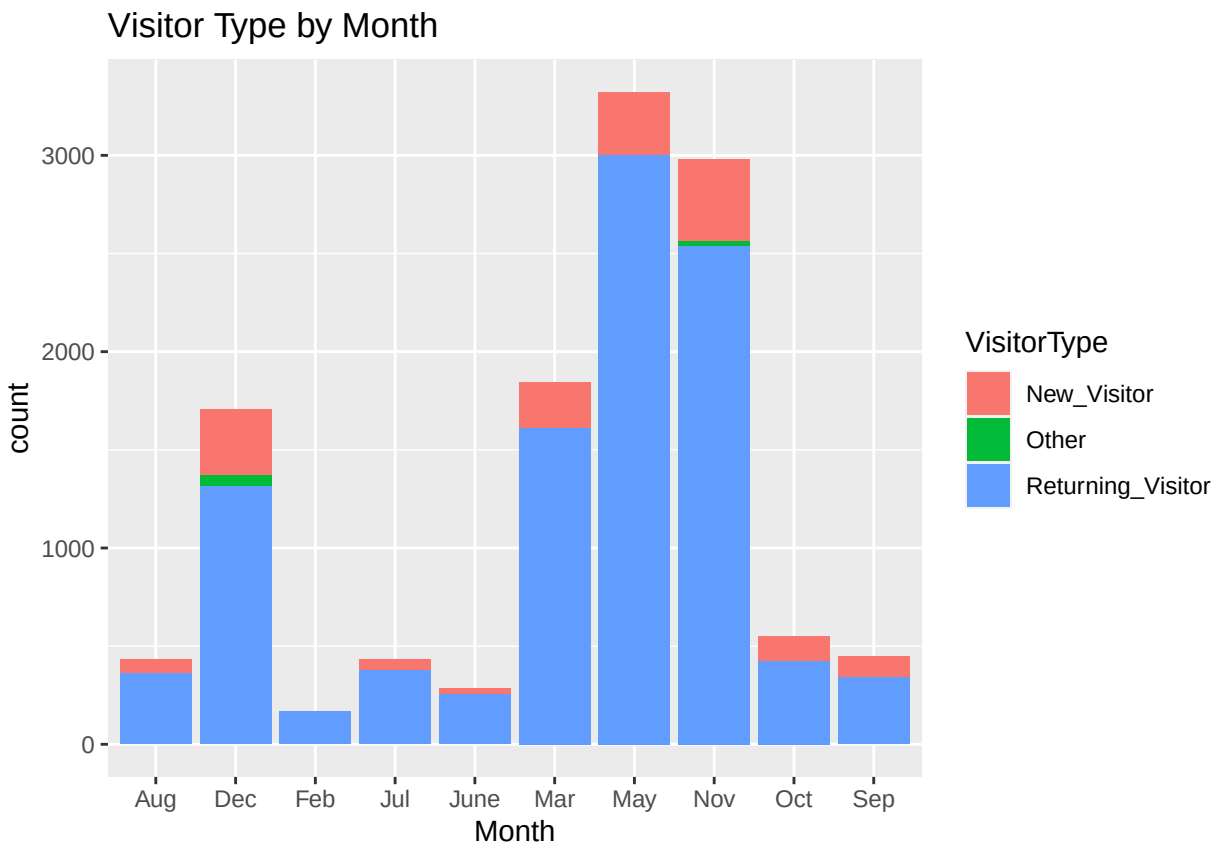

```
ggplot(traffic_page, aes(x=TrafficType, y = Informational))+
  geom_bar(stat = "identity", fill="lightblue")
```




```
ggplot(traffic_page, aes(x=TrafficType, y = ProductRelated))+
  geom_bar(stat = "identity", fill="lightblue")
```



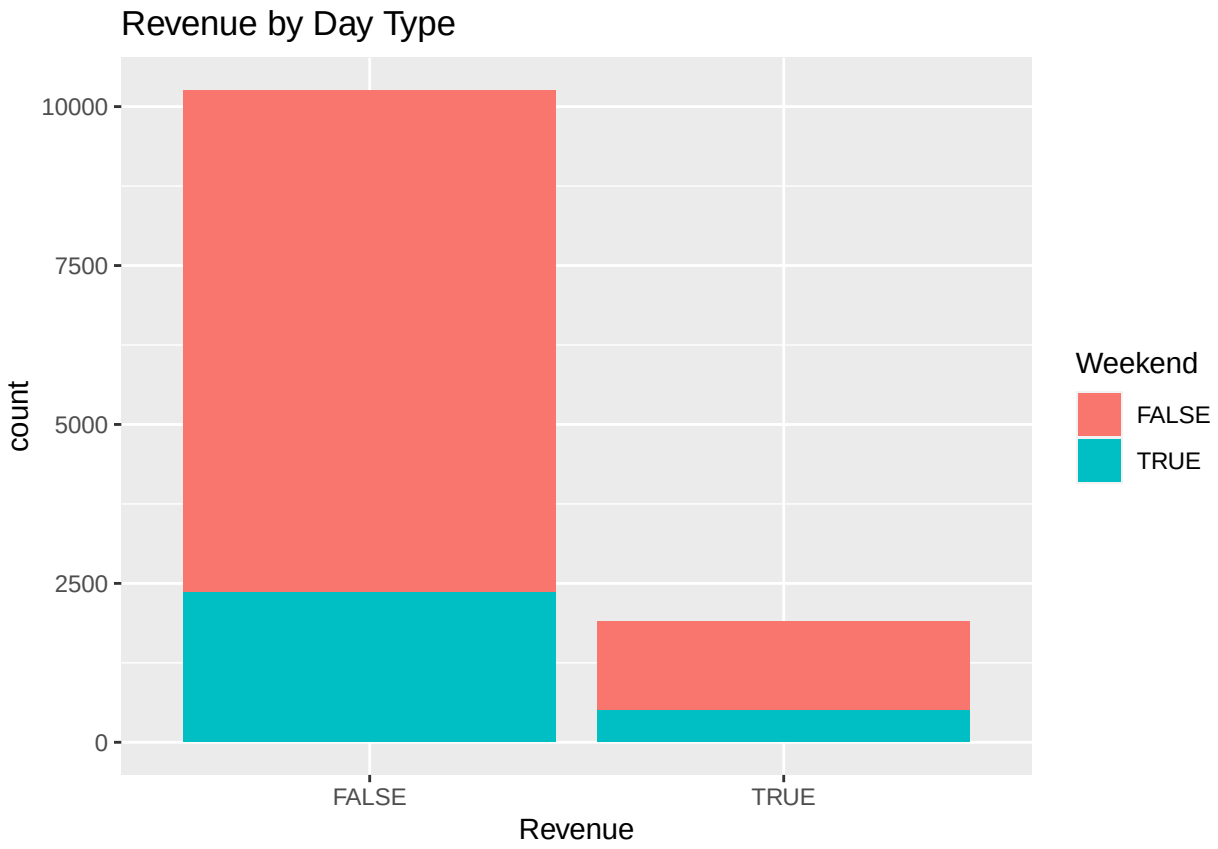

```
# Stacked bar chart: Visitor Type vs Month
shop %>%
    ggplot(aes(Month)) +
    geom_bar(aes(fill = VisitorType))+
    labs(title = "Visitor Type by Month")
```



From above we note that 'Feb' and 'June' are the least busy months.

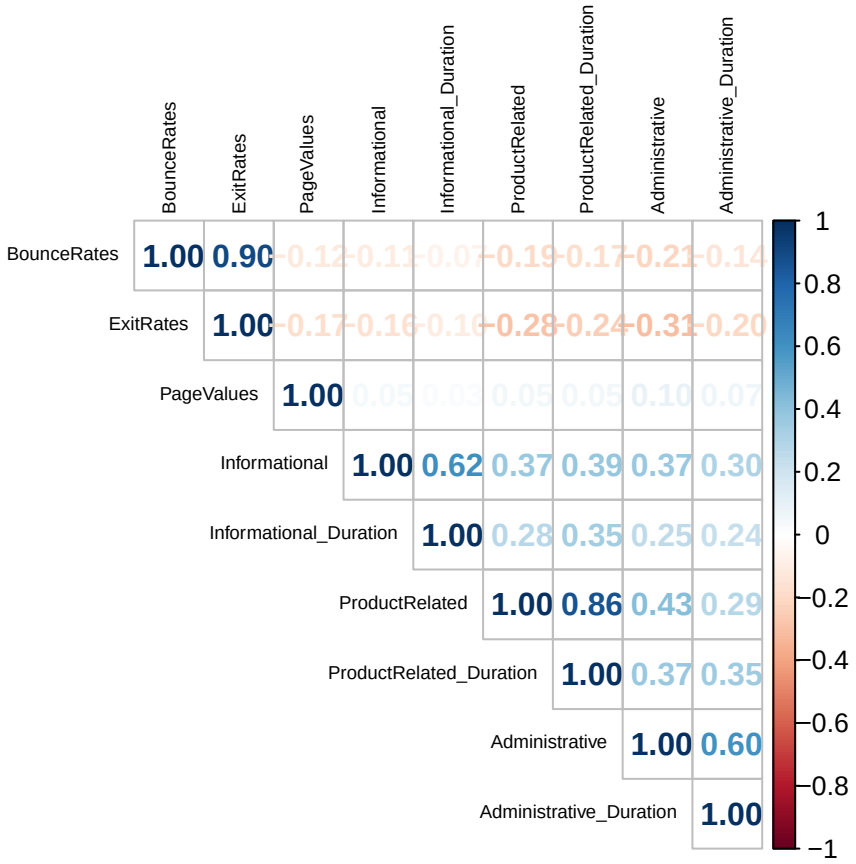
May, Nov, March, and December are the busy months.The company can maximize on this and plan ad campaigns during this time.

```
# Stacked bar chart: Revenue vs Day Type
shop %>%
    ggplot(aes(Revenue)) +
    geom_bar(aes(fill = Weekend))+
    labs(title = "Revenue by Day Type")
```




```
#install.packages("corrplot")

# calculating correlations and plotting a correlation plot
corrplot(corr = cor(shop[, c(1:9)]), method = "number", type = "upper", order = "hclust", tl.col = "bla
```



#Implementing the solution ##Encoding categorical variables

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

dummy_shop = dummyVars(" ~ .", data = shop)

df = data.frame(predict(dummy_shop, newdata = shop))
```

```
# checking the data types
sapply(df, class)
```

```
##              Administrative     Administrative_Duration
##                   "numeric"                   "numeric"
##               Informational      Informational_Duration
##                   "numeric"                   "numeric"
##              ProductRelated     ProductRelated_Duration
##                   "numeric"                   "numeric"
##                BounceRates                   ExitRates
##                   "numeric"                   "numeric"
##                 PageValues                  SpecialDay
##                   "numeric"                   "numeric"
##                   MonthAug                    MonthDec
##                   "numeric"                   "numeric"
##                   MonthFeb                    MonthJul
##                   "numeric"                   "numeric"
##                  MonthJune                    MonthMar
```

```
##                     "numeric"                     "numeric"
##                      MonthMay                      MonthNov
##                     "numeric"                     "numeric"
##                      MonthOct                      MonthSep
##                     "numeric"                     "numeric"
##             OperatingSystems                       Browser
##                     "numeric"                     "numeric"
##                        Region                   TrafficType
##                     "numeric"                     "numeric"
##       VisitorTypeNew_Visitor             VisitorTypeOther
##                     "numeric"                     "numeric"
## VisitorTypeReturning_Visitor                 WeekendFALSE
##                     "numeric"                     "numeric"
##                   WeekendTRUE                 RevenueFALSE
##                     "numeric"                     "numeric"
##                   RevenueTRUE
##                     "numeric"
```

```
glimpse(df)
```

```
## Rows: 12,164
## Columns: 31
## $ Administrative               <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2, 0,~
## $ Administrative_Duration      <dbl> 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0~
## $ Informational                <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ Informational_Duration       <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ ProductRelated               <dbl> 1, 2, 2, 10, 19, 2, 3, 3, 16, 7, 6, 2, 23~
## $ ProductRelated_Duration      <dbl> 0.000000, 64.000000, 2.666667, 627.500000~
## $ BounceRates                  <dbl> 0.200000000, 0.000000000, 0.050000000, 0.~
## $ ExitRates                    <dbl> 0.200000000, 0.100000000, 0.140000000, 0.~
## $ PageValues                   <dbl> 0.00000, 0.00000, 0.00000, 0.00000, 0.000~
## $ SpecialDay                   <dbl> 0.0, 0.0, 0.0, 0.0, 0.0, 0.8, 0.4, 0.0, 0~
## $ MonthAug                     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ MonthDec                     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ MonthFeb                     <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,~
## $ MonthJul                     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ MonthJune                    <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ MonthMar                     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ MonthMay                     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ MonthNov                     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ MonthOct                     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ MonthSep                     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ OperatingSystems             <dbl> 1, 2, 3, 3, 2, 2, 2, 1, 1, 1, 2, 3, 1, 1,~
## $ Browser                      <dbl> 1, 2, 2, 3, 2, 2, 4, 1, 1, 1, 5, 2, 1, 1,~
## $ Region                       <dbl> 1, 1, 2, 1, 1, 2, 1, 3, 4, 1, 1, 3, 9, 1,~
## $ TrafficType                  <dbl> 1, 2, 4, 4, 3, 3, 2, 3, 3, 3, 3, 3, 3, 4,~
## $ VisitorTypeNew_Visitor       <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ VisitorTypeOther             <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
## $ VisitorTypeReturning_Visitor <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,~
## $ WeekendFALSE                 <dbl> 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0,~
## $ WeekendTRUE                  <dbl> 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1,~
## $ RevenueFALSE                 <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,~
## $ RevenueTRUE                  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,~
```

```r
# We will remove the Revenue column it is the class label, we will store it in another variable
df_copy <- df[, -c(30:31)]
df.class<- shop[, "Revenue"]

df_copy_copy <- df[, -c(30,31)]
```

```r
# Previewing the dataset with dummies
head(df_copy)
```

```
##   Administrative Administrative_Duration Informational Informational_Duration
## 1              0                       0             0                      0
## 2              0                       0             0                      0
## 3              0                       0             0                      0
## 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              2                2.666667  0.05000000 0.1400000          0
## 4             10              627.500000  0.02000000 0.0500000          0
## 5             19              154.216667  0.01578947 0.0245614          0
## 6              2               37.000000  0.00000000 0.1000000          0
##   SpecialDay MonthAug MonthDec MonthFeb MonthJul MonthJune MonthMar MonthMay
## 1        0.0        0        0        1        0         0        0        0
## 2        0.0        0        0        1        0         0        0        0
## 3        0.0        0        0        1        0         0        0        0
## 4        0.0        0        0        1        0         0        0        0
## 5        0.0        0        0        1        0         0        0        0
## 6        0.8        0        0        1        0         0        0        0
##   MonthNov MonthOct MonthSep OperatingSystems Browser Region TrafficType
## 1        0        0        0                1       1      1           1
## 2        0        0        0                2       2      1           2
## 3        0        0        0                3       2      2           4
## 4        0        0        0                3       3      1           4
## 5        0        0        0                2       2      1           3
## 6        0        0        0                2       2      2           3
##   VisitorTypeNew_Visitor VisitorTypeOther VisitorTypeReturning_Visitor
## 1                      0                0                            1
## 2                      0                0                            1
## 3                      0                0                            1
## 4                      0                0                            1
## 5                      0                0                            1
## 6                      0                0                            1
##   WeekendFALSE WeekendTRUE
## 1            1           0
## 2            1           0
## 3            1           0
## 4            0           1
## 5            1           0
## 6            1           0
```

```r
# scaling
df_scaled <- scale(df_copy)
# check the output
summary(df_scaled)
```

```
##  Administrative   Administrative_Duration Informational
##  Min.   :-0.704   Min.   :-0.4609         Min.   :-0.3995
##  1st Qu.:-0.704   1st Qu.:-0.4609         1st Qu.:-0.3995
##  Median :-0.404   Median :-0.4047         Median :-0.3995
##  Mean   : 0.000   Mean   : 0.0000         Mean   : 0.0000
##  3rd Qu.: 0.496   3rd Qu.: 0.0736         3rd Qu.:-0.3995
##  Max.   : 7.396   Max.   :18.6624         Max.   :18.3893
##  Informational_Duration ProductRelated    ProductRelated_Duration
##  Min.   :-0.2467        Min.   :-0.7203   Min.   :-0.6302
##  1st Qu.:-0.2467        1st Qu.:-0.5410   1st Qu.:-0.5279
##  Median :-0.2467        Median :-0.3170   Median :-0.3111
##  Mean   : 0.0000        Mean   : 0.0000   Mean   : 0.0000
##  3rd Qu.:-0.2467        3rd Qu.: 0.1311   3rd Qu.: 0.1410
##  Max.   :17.7512        Max.   :15.0749   Max.   :32.6617
##   BounceRates         ExitRates         PageValues        SpecialDay
##  Min.   :-0.44877   Min.   :-0.9005   Min.   :-0.3195   Min.   :-0.3104
##  1st Qu.:-0.44877   1st Qu.:-0.5899   1st Qu.:-0.3195   1st Qu.:-0.3104
##  Median :-0.38448   Median :-0.3526   Median :-0.3195   Median :-0.3104
##  Mean   : 0.00000   Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000
##  3rd Qu.:-0.08264   3rd Qu.: 0.1524   3rd Qu.:-0.3195   3rd Qu.:-0.3104
##  Max.   : 4.03863   Max.   : 3.4832   Max.   :19.0449   Max.   : 4.6939
##     MonthAug          MonthDec          MonthFeb          MonthJul
##  Min.   :-0.1921   Min.   :-0.4039   Min.   :-0.1187   Min.   :-0.1917
##  1st Qu.:-0.1921   1st Qu.:-0.4039   1st Qu.:-0.1187   1st Qu.:-0.1917
##  Median :-0.1921   Median :-0.4039   Median :-0.1187   Median :-0.1917
##  Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000
##  3rd Qu.:-0.1921   3rd Qu.:-0.4039   3rd Qu.:-0.1187   3rd Qu.:-0.1917
##  Max.   : 5.2048   Max.   : 2.4758   Max.   : 8.4244   Max.   : 5.2173
##    MonthJune          MonthMar          MonthMay          MonthNov
##  Min.   :-0.1549   Min.   :-0.4224   Min.   :-0.6128   Min.   :-0.5696
##  1st Qu.:-0.1549   1st Qu.:-0.4224   1st Qu.:-0.6128   1st Qu.:-0.5696
##  Median :-0.1549   Median :-0.4224   Median :-0.6128   Median :-0.5696
##  Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000
##  3rd Qu.:-0.1549   3rd Qu.:-0.4224   3rd Qu.: 1.6317   3rd Qu.:-0.5696
##  Max.   : 6.4558   Max.   : 2.3671   Max.   : 1.6317   Max.   : 1.7555
##     MonthOct          MonthSep       OperatingSystems     Browser
##  Min.   :-0.2174   Min.   :-0.1955   Min.   :-1.2396   Min.   :-0.7940
##  1st Qu.:-0.2174   1st Qu.:-0.1955   1st Qu.:-0.1373   1st Qu.:-0.2091
##  Median :-0.2174   Median :-0.1955   Median :-0.1373   Median :-0.2091
##  Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000
##  3rd Qu.:-0.2174   3rd Qu.:-0.1955   3rd Qu.: 0.9650   3rd Qu.:-0.2091
##  Max.   : 4.5994   Max.   : 5.1137   Max.   : 6.4765   Max.   : 6.2239
##      Region          TrafficType       VisitorTypeNew_Visitor
##  Min.   :-0.89608   Min.   :-0.76583   Min.   :-0.4021
##  1st Qu.:-0.89608   1st Qu.:-0.51688   1st Qu.:-0.4021
##  Median :-0.06355   Median :-0.51688   Median :-0.4021
##  Mean   : 0.00000   Mean   : 0.00000   Mean   : 0.0000
##  3rd Qu.: 0.35272   3rd Qu.:-0.01897   3rd Qu.:-0.4021
```

```
##  Max.   : 2.43405   Max.   : 3.96428   Max.   : 2.4868
##  VisitorTypeOther   VisitorTypeReturning_Visitor  WeekendFALSE
##  Min.   :-0.08187   Min.   :-2.4200              Min.   :-1.8065
##  1st Qu.:-0.08187   1st Qu.: 0.4132              1st Qu.: 0.5535
##  Median :-0.08187   Median : 0.4132              Median : 0.5535
##  Mean   : 0.00000   Mean   : 0.0000              Mean   : 0.0000
##  3rd Qu.:-0.08187   3rd Qu.: 0.4132              3rd Qu.: 0.5535
##  Max.   :12.21313   Max.   : 0.4132              Max.   : 0.5535
##   WeekendTRUE
##  Min.   :-0.5535
##  1st Qu.:-0.5535
##  Median :-0.5535
##  Mean   : 0.0000
##  3rd Qu.:-0.5535
##  Max.   : 1.8065
```

```
# Lets normalize the data and see if the results change.
# Normalize
df_norm <- as.data.frame(apply(df_copy, 2, function(x) (x - min(x))/(max(x)-min(x))))
# summary of normalized data
summary(df_norm)
```

```
##  Administrative    Administrative_Duration Informational
##  Min.   :0.00000   Min.   :0.000000        Min.   :0.00000
##  1st Qu.:0.00000   1st Qu.:0.000000        1st Qu.:0.00000
##  Median :0.03704   Median :0.002942        Median :0.00000
##  Mean   :0.08691   Mean   :0.024103        Mean   :0.02126
##  3rd Qu.:0.14815   3rd Qu.:0.027952        3rd Qu.:0.00000
##  Max.   :1.00000   Max.   :1.000000        Max.   :1.00000
##  Informational_Duration ProductRelated    ProductRelated_Duration
##  Min.   :0.00000        Min.   :0.00000   Min.   :0.000000
##  1st Qu.:0.00000        1st Qu.:0.01135   1st Qu.:0.003072
##  Median :0.00000        Median :0.02553   Median :0.009586
##  Mean   :0.01371        Mean   :0.04560   Mean   :0.018929
##  3rd Qu.:0.00000        3rd Qu.:0.05390   3rd Qu.:0.023165
##  Max.   :1.00000        Max.   :1.00000   Max.   :1.000000
##   BounceRates        ExitRates         PageValues       SpecialDay
##  Min.   :0.00000   Min.   :0.00000   Min.   :0.0000   Min.   :0.00000
##  1st Qu.:0.00000   1st Qu.:0.07087   1st Qu.:0.0000   1st Qu.:0.00000
##  Median :0.01433   Median :0.12500   Median :0.0000   Median :0.00000
##  Mean   :0.10001   Mean   :0.20542   Mean   :0.0165   Mean   :0.06202
##  3rd Qu.:0.08159   3rd Qu.:0.24020   3rd Qu.:0.0000   3rd Qu.:0.00000
##  Max.   :1.00000   Max.   :1.00000   Max.   :1.0000   Max.   :1.00000
##     MonthAug         MonthDec         MonthFeb          MonthJul
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.00000   Min.   :0.00000
##  1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:0.00000
##  Median :0.0000   Median :0.0000   Median :0.00000   Median :0.00000
##  Mean   :0.0356   Mean   :0.1402   Mean   :0.01389   Mean   :0.03543
##  3rd Qu.:0.0000   3rd Qu.:0.0000   3rd Qu.:0.00000   3rd Qu.:0.00000
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.00000   Max.   :1.00000
##    MonthJune          MonthMar         MonthMay        MonthNov
##  Min.   :0.00000   Min.   :0.0000   Min.   :0.000   Min.   :0.000
##  1st Qu.:0.00000   1st Qu.:0.0000   1st Qu.:0.000   1st Qu.:0.000
##  Median :0.00000   Median :0.0000   Median :0.000   Median :0.000
```

```
##  Mean   :0.02343   Mean   :0.1514   Mean   :0.273   Mean   :0.245
##  3rd Qu.:0.00000   3rd Qu.:0.0000   3rd Qu.:1.000   3rd Qu.:0.000
##  Max.   :1.00000   Max.   :1.0000   Max.   :1.000   Max.   :1.000
##     MonthOct          MonthSep       OperatingSystems    Browser
##  Min.   :0.00000   Min.   :0.00000   Min.   :0.0000   Min.   :0.00000
##  1st Qu.:0.00000   1st Qu.:0.00000   1st Qu.:0.1429   1st Qu.:0.08333
##  Median :0.00000   Median :0.00000   Median :0.1429   Median :0.08333
##  Mean   :0.04513   Mean   :0.03683   Mean   :0.1606   Mean   :0.11313
##  3rd Qu.:0.00000   3rd Qu.:0.00000   3rd Qu.:0.2857   3rd Qu.:0.08333
##  Max.   :1.00000   Max.   :1.00000   Max.   :1.0000   Max.   :1.00000
##      Region        TrafficType      VisitorTypeNew_Visitor VisitorTypeOther
##  Min.   :0.0000   Min.   :0.00000   Min.   :0.0000         Min.   :0.000000
##  1st Qu.:0.0000   1st Qu.:0.05263   1st Qu.:0.0000         1st Qu.:0.000000
##  Median :0.2500   Median :0.05263   Median :0.0000         Median :0.000000
##  Mean   :0.2691   Mean   :0.16191   Mean   :0.1392         Mean   :0.006659
##  3rd Qu.:0.3750   3rd Qu.:0.15789   3rd Qu.:0.0000         3rd Qu.:0.000000
##  Max.   :1.0000   Max.   :1.00000   Max.   :1.0000         Max.   :1.000000
##  VisitorTypeReturning_Visitor  WeekendFALSE     WeekendTRUE
##  Min.   :0.0000                Min.   :0.0000   Min.   :0.0000
##  1st Qu.:1.0000                1st Qu.:1.0000   1st Qu.:0.0000
##  Median :1.0000                Median :1.0000   Median :0.0000
##  Mean   :0.8542                Mean   :0.7655   Mean   :0.2345
##  3rd Qu.:1.0000                3rd Qu.:1.0000   3rd Qu.:0.0000
##  Max.   :1.0000                Max.   :1.0000   Max.   :1.0000
```

#Cluster Analysis

```
library(cluster)    # clustering algorithms
library(factoextra) # clustering algorithms & visualization
```

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

```
#Using Elbow plot  method, Searching for the optimal number of clusters
#fviz_nbclust(df_norm, kmeans, method = "wss") +
    #geom_vline(xintercept = 4, linetype = 2)+
  #labs(subtitle = "Elbow plot")
```

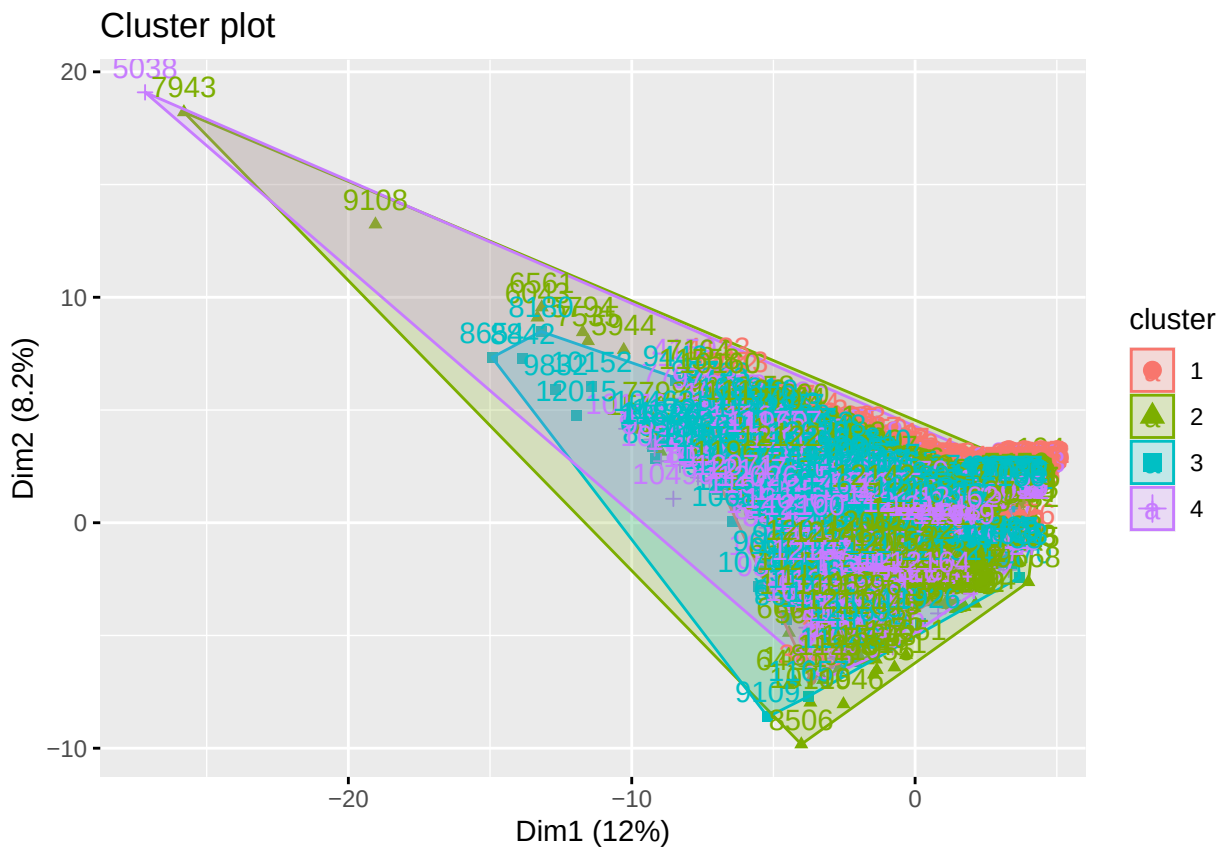
```
# Compute k-means clustering with k = 4
set.seed(123)
final <- kmeans(df_norm, 4, nstart = 25)
#print(final)
```

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

final$size
```

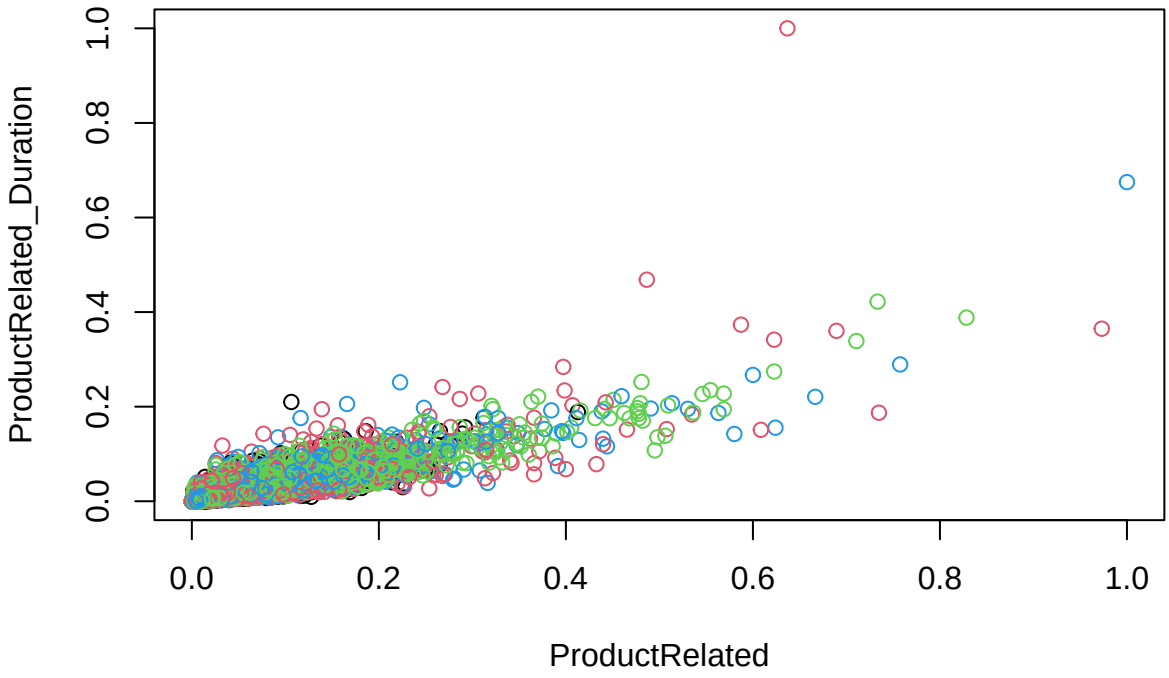
```
## [1] 2607 4515 2189 2853
```

```
# visualize the results
fviz_cluster(final, data = df)
```




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

plot(df_norm[, 5:6], col = final$cluster)
```

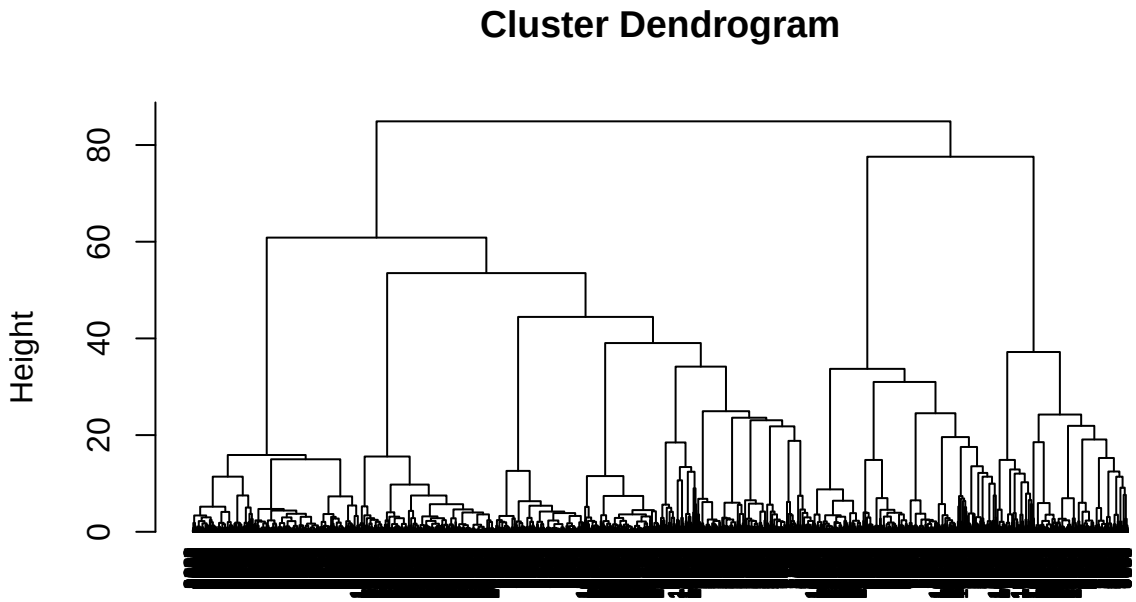



```
#shop %>%
#  mutate(Cluster = final$cluster) %>%
#  group_by(Cluster) %>%
#  summarise_all("mean")
```

```
#First we use the dist() to compute the Euclidean distance btwn observation points
shop_dist = dist(df_norm, method = "euclidean")

#Set the hclust() dissimilairty matrix
#We then apply hierarchical clustering using the Ward's method
shop_hc = hclust(shop_dist, method = "ward.D2")

#Plot the obtained dendrogram
plot(shop_hc, cex = 0.6, hang = -1)
```




```
# cutting the clusters into 4 groups
group<-cutree(shop_hc,k=6)
# viewing the clustered groups
table(group)
```

```
## group
##    1    2    3    4    5    6
## 2979 2376 1709 1161 2159 1780
```

```
# creating a table
hclust<-dplyr::mutate(shop,clusters=group)
head(hclust)
```

```
##   Administrative Administrative_Duration Informational Informational_Duration
## 1              0                       0             0                      0
## 2              0                       0             0                      0
## 3              0                       0             0                      0
## 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              2                 2.666667  0.05000000 0.1400000          0
## 4             10               627.500000  0.02000000 0.0500000          0
```

```
## 5             19              154.216667  0.01578947 0.0245614          0
## 6              2               37.000000  0.00000000 0.1000000          0
##   SpecialDay Month OperatingSystems Browser Region TrafficType
## 1        0.0   Feb                1       1      1           1
## 2        0.0   Feb                2       2      1           2
## 3        0.0   Feb                3       2      2           4
## 4        0.0   Feb                3       3      1           4
## 5        0.0   Feb                2       2      1           3
## 6        0.8   Feb                2       2      2           3
##         VisitorType Weekend Revenue clusters
## 1 Returning_Visitor   FALSE   FALSE        1
## 2 Returning_Visitor   FALSE   FALSE        1
## 3 Returning_Visitor   FALSE   FALSE        1
## 4 Returning_Visitor    TRUE   FALSE        2
## 5 Returning_Visitor   FALSE   FALSE        1
## 6 Returning_Visitor   FALSE   FALSE        1
```

#Conclusion

i) The months with the highest activity are May, November, March and December.The company should consider posting ads during this time.

ii)Most visitors to the site are located in region 1 and 3 more empasis should be directed to this region

iii)Visitors to the site are mostly returning visitors.

iv) Most of the traffic is concentrated on weekdays rather than on weekends. Most adverts should be running on weekdays.

v)The traffic types 15 and 17 have the highest Exit and Bounce Rates.

#Comparison Between K-Means and Hierachical Clustering

Hierachical clustering is far easy to implement as we dont to specify number of clusters.

Hierachical clustering outputs a cluster which is a structure and makes more sense. Due to this aspect its more informative. Its also easy to implement.