Project1

2022-10-31

R Markdown

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

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

```
#install.packages('GGally')
#install.packages('pheatmap')
#install.packages('funModeling')
library(ggplot2)
library(gridExtra)
library(GGally)
```

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

```
library(caret)
```

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

```
library(data.table)
library(ggpubr)
library(ROSE)
```

```
## Loaded ROSE 0.0-4
```

```
library(class)
library(pheatmap)
library(dplyr)
```

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

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

## The following object is masked from 'package:gridExtra':
##
##     combine
```

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

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

```
library(funModeling)
```

```
## Loading required package: Hmisc

## Loading required package: survival

##
## Attaching package: 'survival'

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

## 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

## funModeling v.1.9.4 :)
## Examples and tutorials at livebook.datascienceheroes.com
##  / Now in Spanish: librovivodecienciadedatos.ai

##
## Attaching package: 'funModeling'

## The following object is masked from 'package:GGally':
##
##     range01
```

```
#Reading the data
data <- read.csv("/Users/adityak/Documents/NEU/Coursework/Fall22/IE 6600 - Computation and Visualizatio
str(data)
```

```
## 'data.frame':    12330 obs. of  18 variables:
##  $ Administrative         : int  0 0 0 0 0 0 0 1 0 0 ...
##  $ Administrative_Duration: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Informational          : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Informational_Duration : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ ProductRelated         : int  1 2 1 2 10 19 1 0 2 3 ...
##  $ ProductRelated_Duration: num  0 64 0 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 ...
```

```
#Structure of the data
```

```
#Missing Value Analysis
sapply(data, function(x) sum(is.na(x)))
```

```
##          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
```

```
data <- na.omit(data)
str(data)
```

```
## 'data.frame':    12330 obs. of  18 variables:
##  $ Administrative         : int  0 0 0 0 0 0 0 1 0 0 ...
##  $ Administrative_Duration: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Informational          : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Informational_Duration : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ ProductRelated         : int  1 2 1 2 10 19 1 0 2 3 ...
##  $ ProductRelated_Duration: num  0 64 0 2.67 627.5 ...
##  $ BounceRates            : num  0.2 0 0.2 0.05 0.02 ...
##  $ ExitRates              : num  0.2 0.1 0.2 0.14 0.05 ...
##  $ PageValues             : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ SpecialDay             : num  0 0 0 0 0 0 0.4 0 0.8 0.4 ...
```

```
##  $ Month                  : chr  "Feb" "Feb" "Feb" "Feb" ...
##  $ OperatingSystems       : int  1 2 4 3 3 2 2 1 2 2 ...
##  $ Browser                : int  1 2 1 2 3 2 4 2 2 4 ...
##  $ Region                 : int  1 1 9 2 1 1 3 1 2 1 ...
##  $ TrafficType            : int  1 2 3 4 4 3 3 5 3 2 ...
##  $ VisitorType            : chr  "Returning_Visitor" "Returning_Visitor" "Returning_Visitor" "Return
##  $ Weekend                : logi  FALSE FALSE FALSE FALSE TRUE FALSE ...
##  $ Revenue                : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
```

```
unique(data$Month)
```

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

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.

```
#fix the structure of the data

data$Revenue <- gsub(FALSE, 0, data$Revenue)
data$Revenue <- gsub(TRUE, 1, data$Revenue)
data$Weekend <- gsub(TRUE, 1, data$Weekend)
data$Weekend <- gsub(FALSE, 0, data$Weekend)

data$Month <- factor(data$Month,
  levels = c("Feb", "Mar", "May", "June", "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"),
  ordered = TRUE)
data$OperatingSystems <- factor(data$OperatingSystems)
data$Browser <- factor(data$Browser)
data$Region <- factor(data$Region)
data$TrafficType <- factor(data$TrafficType)
data$VisitorType <- factor(data$VisitorType)
data$Revenue <- factor(data$Revenue)
data$Weekend <- factor(data$Weekend)
str(data)
```

```
## 'data.frame':    12330 obs. of  18 variables:
##  $ Administrative         : int  0 0 0 0 0 0 0 1 0 0 ...
##  $ Administrative_Duration: num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Informational          : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Informational_Duration : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ ProductRelated         : int  1 2 1 2 10 19 1 0 2 3 ...
##  $ ProductRelated_Duration: num  0 64 0 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                  : Ord.factor w/ 10 levels "Feb"<"Mar"<"May"<..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ OperatingSystems       : Factor w/ 8 levels "1","2","3","4",..: 1 2 4 3 3 2 2 1 2 2 ...
##  $ Browser                : Factor w/ 13 levels "1","2","3","4",..: 1 2 1 2 3 2 4 2 2 4 ...
##  $ Region                 : Factor w/ 9 levels "1","2","3","4",..: 1 1 9 2 1 1 3 1 2 1 ...
##  $ TrafficType            : Factor w/ 20 levels "1","2","3","4",..: 1 2 3 4 4 3 3 5 3 2 ...
##  $ VisitorType            : Factor w/ 3 levels "New_Visitor",..: 3 3 3 3 3 3 3 3 3 3 ...
##  $ Weekend                : Factor w/ 2 levels "0","1": 1 1 1 1 2 1 1 2 1 1 ...
##  $ Revenue                : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
```

```r
#Descriptive analysis

summary(data[,c(1:10)])
```

```
##  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 :   7.50         Median : 0.0000
##  Mean   : 2.315   Mean   :  80.82         Mean   : 0.5036
##  3rd Qu.: 4.000   3rd Qu.:  93.26         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.:  7.00   1st Qu.:  184.1
##  Median :   0.00        Median : 18.00   Median :  598.9
##  Mean   :  34.47        Mean   : 31.73   Mean   : 1194.8
##  3rd Qu.:   0.00        3rd Qu.: 38.00   3rd Qu.: 1464.2
##  Max.   :2549.38        Max.   :705.00   Max.   :63973.5
##   BounceRates         ExitRates         PageValues        SpecialDay
##  Min.   :0.000000   Min.   :0.00000   Min.   :  0.000   Min.   :0.00000
##  1st Qu.:0.000000   1st Qu.:0.01429   1st Qu.:  0.000   1st Qu.:0.00000
##  Median :0.003112   Median :0.02516   Median :  0.000   Median :0.00000
##  Mean   :0.022191   Mean   :0.04307   Mean   :  5.889   Mean   :0.06143
##  3rd Qu.:0.016813   3rd Qu.:0.05000   3rd Qu.:  0.000   3rd Qu.:0.00000
##  Max.   :0.200000   Max.   :0.20000   Max.   :361.764   Max.   :1.00000
```

```r
table(data$Revenue)
```

```
##
##     0     1
## 10422  1908
```

```r
table(data$Weekend)
```

```
##
##    0    1
## 9462 2868
```

```r
table(data$VisitorType)
```

```
##
##       New_Visitor             Other Returning_Visitor
##              1694                85             10551
```

```r
table(data$TrafficType)
```

```
##
##    1    2    3    4    5    6    7    8    9   10   11   12   13   14   15   16
## 2451 3913 2052 1069  260  444   40  343   42  450  247    1  738   13   38    3
##   17   18   19   20
##    1   10   17  198
```

```r
table(data$Region)
```

```
##
##    1    2    3    4    5    6    7    8    9
## 4780 1136 2403 1182  318  805  761  434  511
```

```r
table(data$Browser)
```

```
##
##    1    2    3    4    5    6    7    8    9   10   11   12   13
## 2462 7961  105  736  467  174   49  135    1  163    6   10   61
```

```r
table(data$OperatingSystems)
```

```
##
##    1    2    3    4    5    6    7    8
## 2585 6601 2555  478    6   19    7   79
```

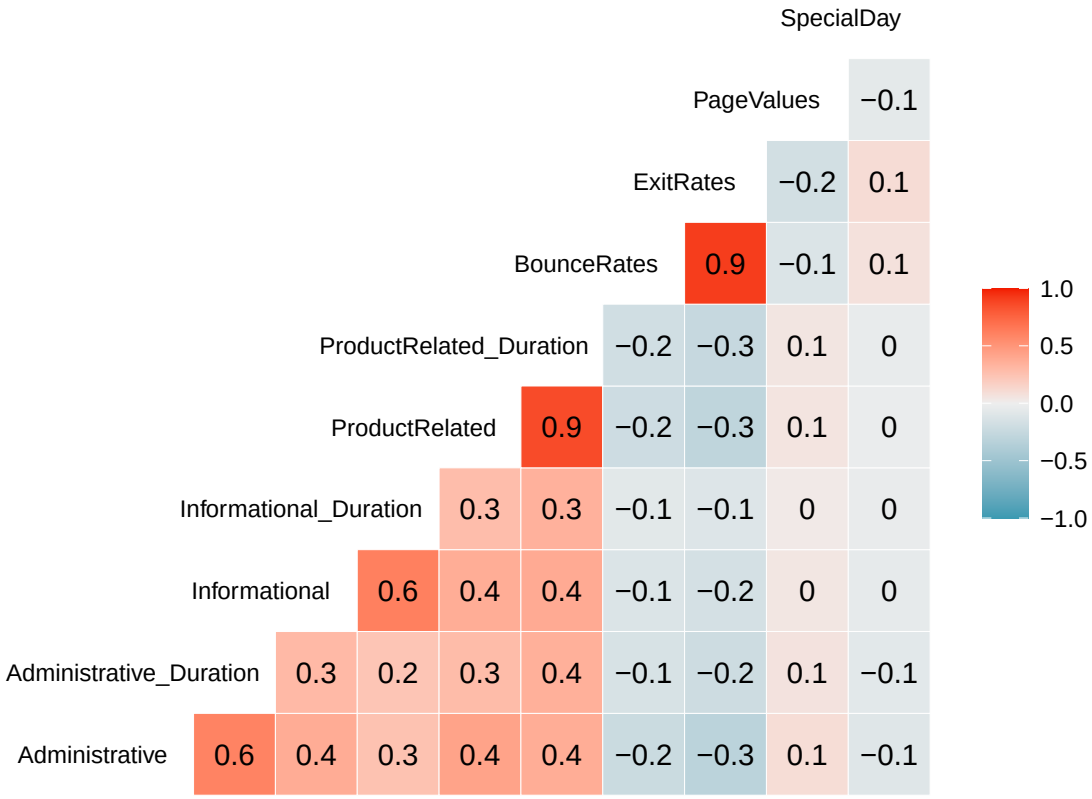
```r
table(data$Month)
```

```
##
##  Feb  Mar  May June  Jul  Aug  Sep  Oct  Nov  Dec
##  184 1907 3364  288  432  433  448  549 2998 1727
```

```r
#Visualization 1 - Correlation Heatmap
#Correlation Analysis & Plotting

library(corrplot)
```

```
## corrplot 0.92 loaded
```

```r
corr_map <- ggcorr(data[, 1:10], method=c("everything", "pearson"), label=TRUE,
                   hjust = .90, size = 3, layout.exp = 2)
corr_map
```



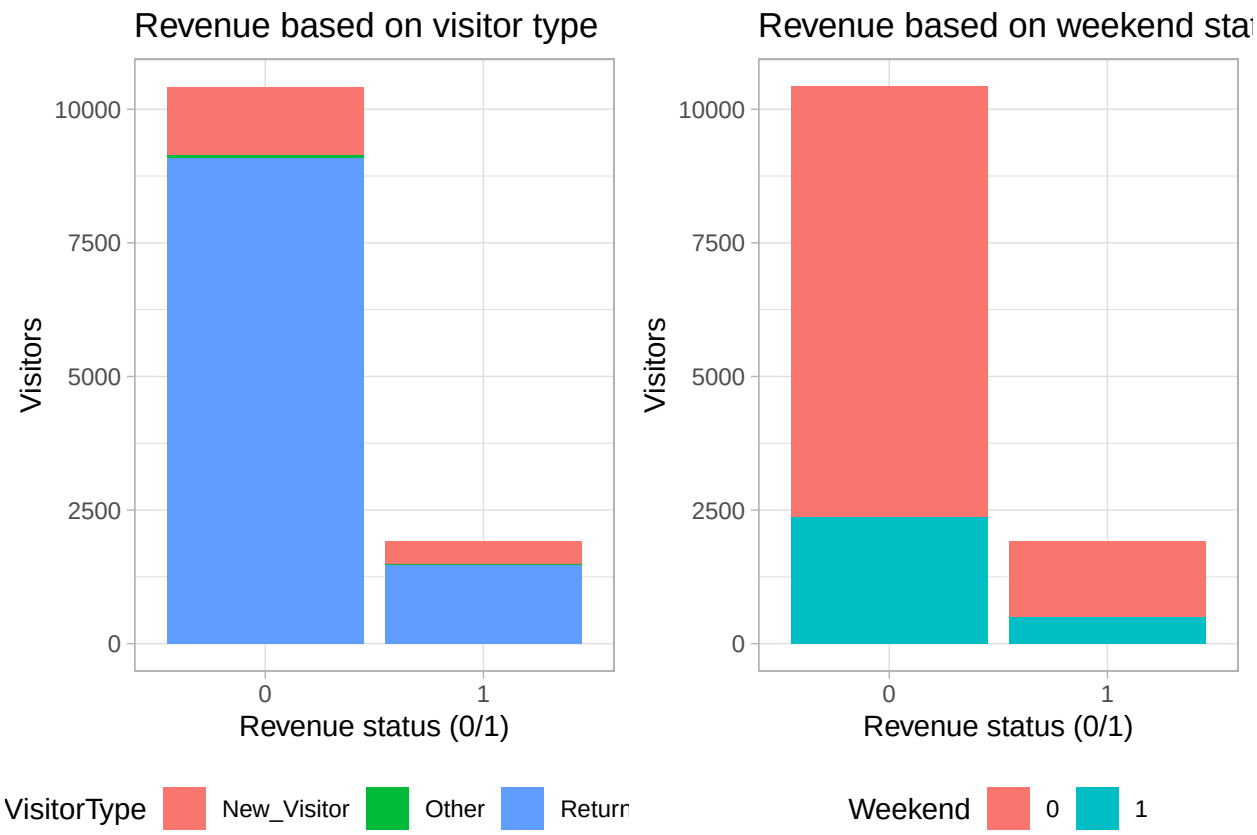

```
#The column pairs BounceRates & ExitRates and ProductRelated & ProductRelated_Duration have
#very high correlation as bounce rate refers to % of visitors who exit the page that they
#entered through and ExitRates refers to the percentage of pageviews on the website that end
#at that specific page.
#ProductRelated & ProductRelated_Duration are also highly correlated as both the columns deal
#with a related product.
```

```
#Visualization 2 - Stacked & Grouped Bar Chart
library(gridExtra)
table(data$Revenue, data$VisitorType)
```

```
##
##     New_Visitor Other Returning_Visitor
##   0        1272    69              9081
##   1         422    16              1470
```

```
options(repr.plot.width = 10, repr.plot.height = 6)
p1 <- ggplot(data = data, mapping = aes(x = Revenue)) +
  geom_bar(mapping = aes(fill = VisitorType)) + theme_light() +
  ggtitle("Revenue based on visitor type") + xlab("Revenue status (0/1)") +
  ylab("Visitors") + theme(legend.position = "bottom")
options(repr.plot.width = 10, repr.plot.height = 6)
p2 <- ggplot(data = data, mapping = aes(x = Revenue)) + geom_bar(mapping = aes(fill = Weekend)) +
  theme_light() + ggtitle("Revenue based on weekend status") + xlab("Revenue status (0/1)") +
  ylab("Visitors") + theme(legend.position = "bottom")
```

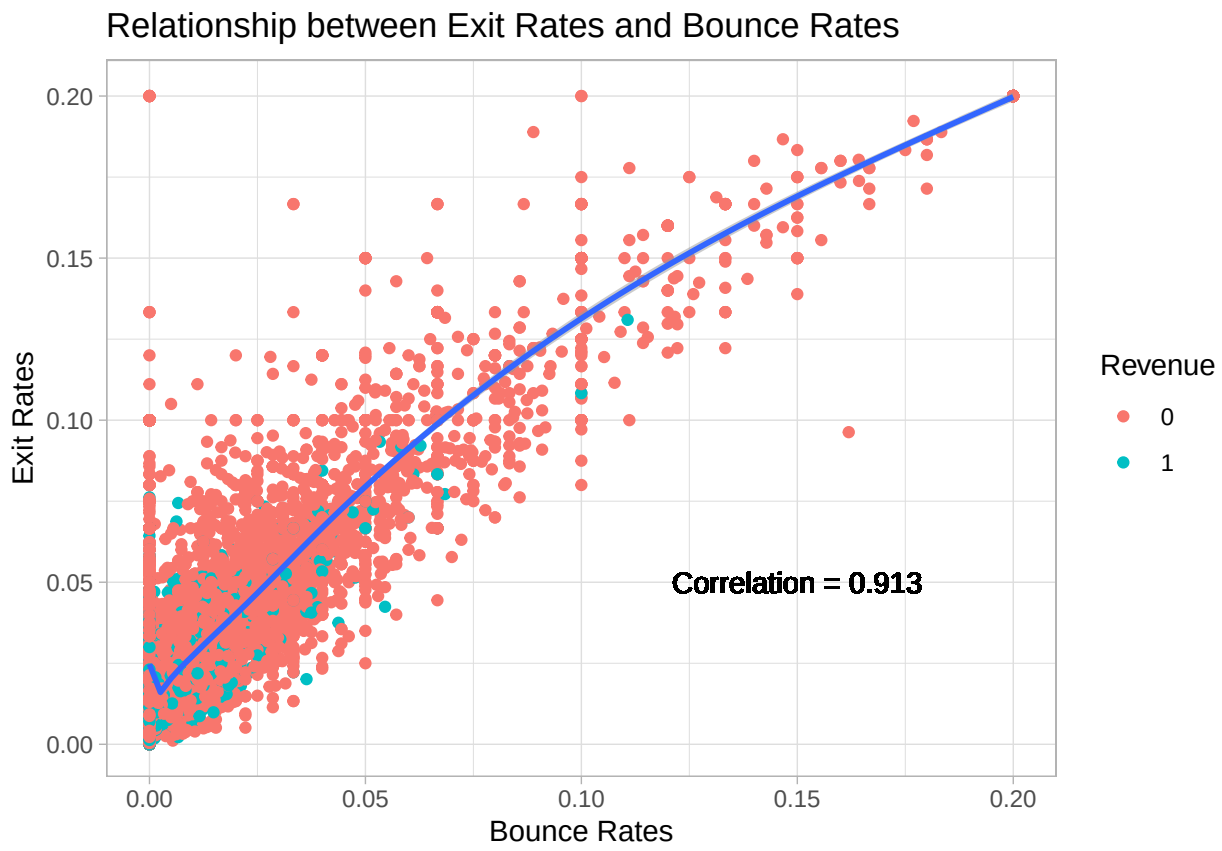
```
grid.arrange(p1,p2, nrow = 1)
```




```
#From the first stacked bar chart, it can be observed the returning customers are almost 5 times
#more than the new customers, the reason behind this could be that majority of the returning customers
#who have already purchased a product, will dig deeper in the website for similar products and then
#complete the transaction as opposed to new customers who don't have any shopping experience in the
#website. Another reason could be some of the new customers may feel the registration process
#(entering name, email, address and card details) to be a tedious process.
#From the first stacked bar chart, the revenue is higher on the weekends as compared to the weekdays.
#The reason could be generally on weekends people tend to have more time to shop without any hurry
#whereas on weekdays people are very busy majorly due to work and other reasons.
```

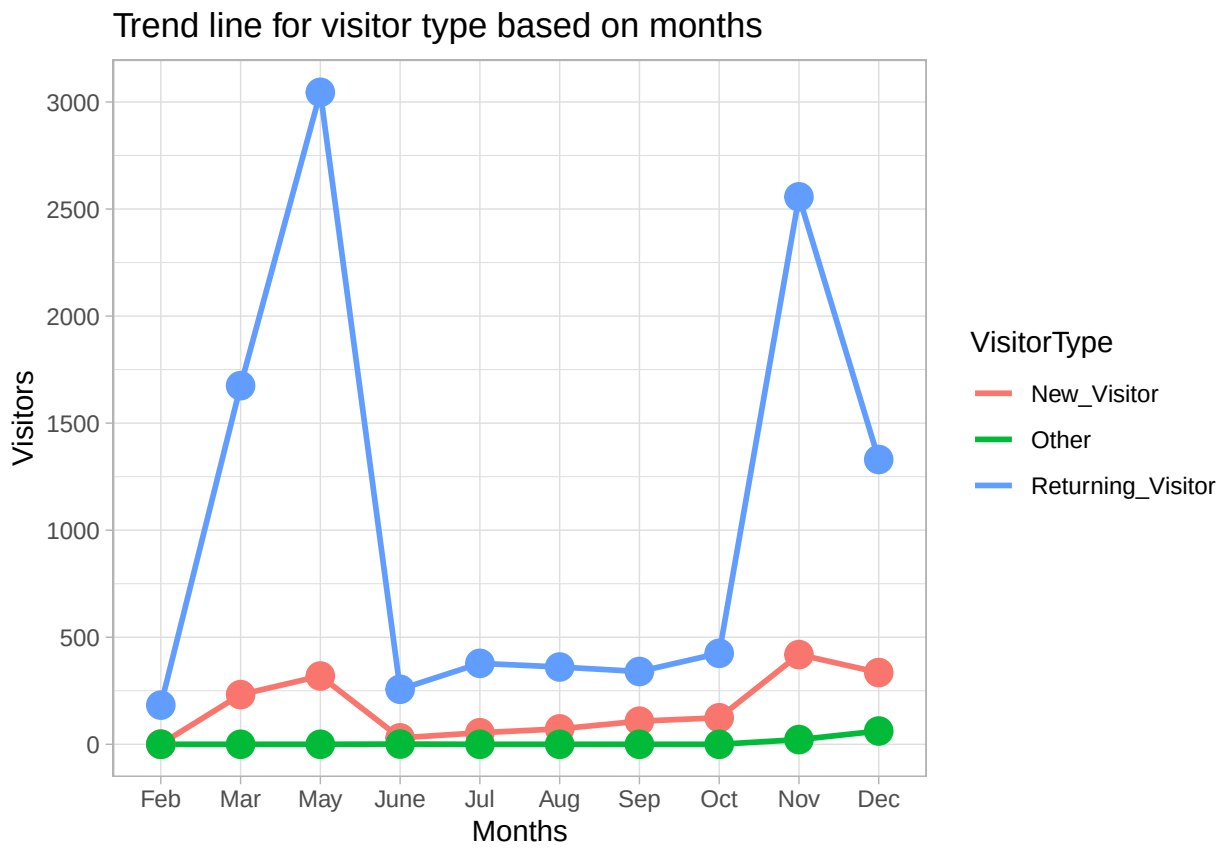
```
#Visualization 3 - Scatter Line Plot for Correlation
#Relationship between Exit Rates and Bounce Rates

library(ggplot2)

options(repr.plot.width = 8, repr.plot.height = 5)
ggplot(data = data, mapping = aes(x = BounceRates, y = ExitRates)) +
  geom_point(mapping = aes(color = Revenue)) + geom_smooth(se = TRUE, alpha = 0.5) +
  theme_light() + ggtitle("Relationship between Exit Rates and Bounce Rates") +
  xlab("Bounce Rates") + ylab("Exit Rates") +
  geom_text(mapping = aes(x = 0.15, y = 0.05, label = "Correlation = 0.913"))
```

```
## `geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```




```
#Exit rates and bounce rates have a positive correlation because as the % of visitors
#who exit the page without triggering any additional tasks increase, the exit rate which
#is the % of page views for that end on a specific page also increases.
```

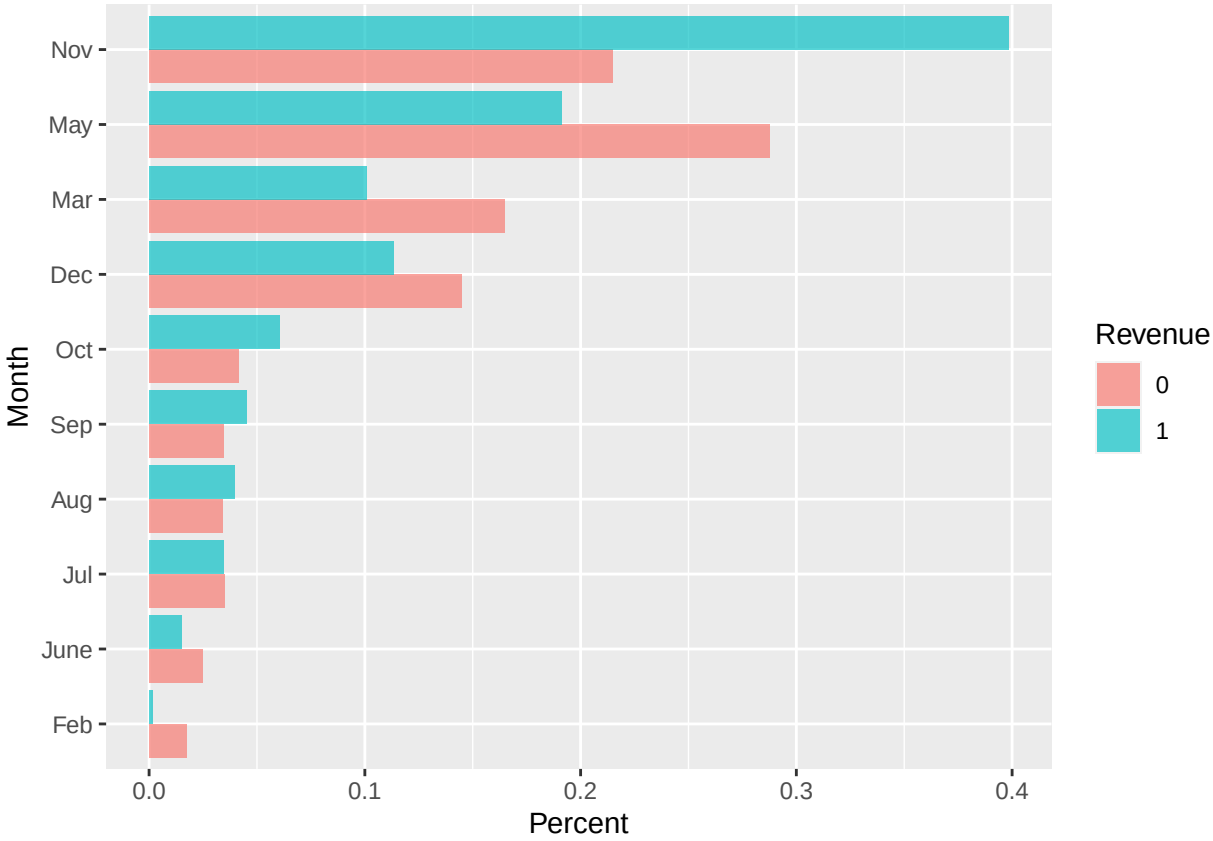
```
#Visualization 4 - Line Graph
#Trend line for revenue status based on months and trend line for visitor type based on months
options(repr.plot.width = 8, repr.plot.height = 5)

trend <- data.frame(table(data$Month, data$Revenue))
names(trend) <- c("Months", "Revenue", "Frequency")


trend <- data.frame(table(data$VisitorType, data$Month))
names(trend) <- c("VisitorType", "Month", "Frequency")
ggplot(data = trend, mapping = aes(x = Month, y = Frequency)) +
  geom_line(mapping = aes(color = VisitorType, group = VisitorType), lwd = 1) +
  geom_point(mapping = aes(color = VisitorType, group = VisitorType, size = 0.1),
             show.legend = FALSE) + theme_light() +
  scale_y_continuous(breaks = seq(from = 0, to = 4000, by = 500)) +
  ggtitle("Trend line for visitor type based on months") +
  xlab("Months") + ylab("Visitors")
```




```
#We observe that all the types of customers i.e. New visitors, returning visitors and others
#are all together contributing a revenue in common trends on a monthly basis. From previous
#graphical inferences, we reinstate that typically during vacation and festive months,
#we see a rise in revenue from all the types of customers
```

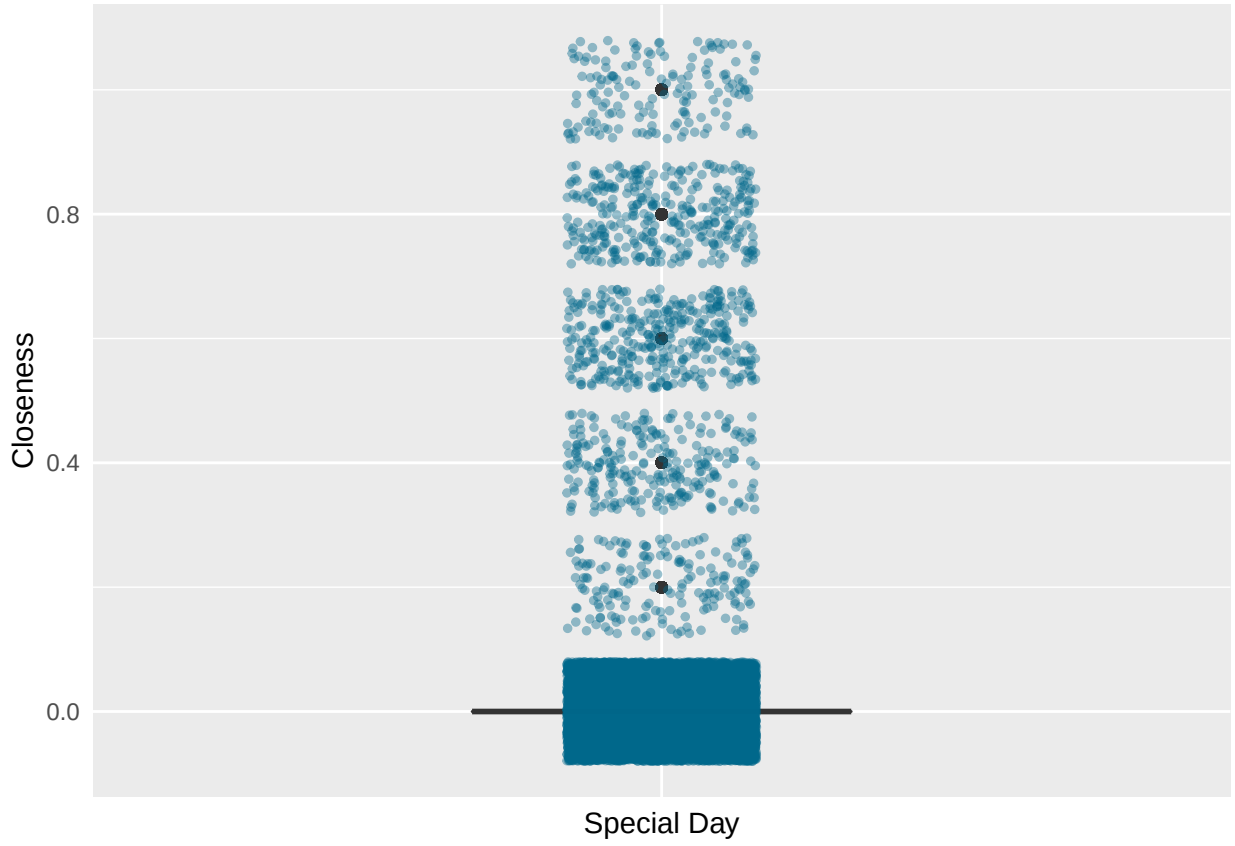
#aes(x = Month, y = perc, fill = Revenue)

```
#Visualization 5 - Inverted & Ordered Grouped Bar Chart
month_table <- table(data$Month, data$Revenue)
month_tab <- as.data.frame(prop.table(month_table, 2))
colnames(month_tab) <-  c("Month", "Revenue", "perc")

ggplot(data = month_tab, aes(x = reorder(Month, perc), y = perc, fill=Revenue)) +
  geom_bar(stat = 'identity', position = 'dodge', alpha = 2/3) +
  xlab("Month")+
  ylab("Percent") + coord_flip()
```




```
#We see very high shopping rates in September, October, and November; months that typically
#correspond to the 'shopping season' in North America. Also, of note is the month of May
#with a lot of visits to the website.
#The shopping rates are particularly high in the months of November to December
#(may be due to Black Friday sale) and March to May(may be due to summer vacation period)
#as compared to other months
```

```
#Visualization 6 - Box Plot with Jitter
plot1 <- ggplot(data, aes(x = factor(1), y = SpecialDay)) + geom_boxplot(width = 0.4, fill = "white") +
  geom_jitter(color = "deepskyblue4", width = 0.1, size = 1, alpha=0.4) + labs(x = "Special Day") +
  labs(y = "Closeness") + theme(axis.text.x = element_blank(), axis.ticks = element_blank())

grid.arrange(plot1)
```




```
#This supports our observation that most customer decisions are not influenced by whether it
#is a special day or not.
#From the box plot, it can be inferred that customers decision to purchase is not affected
#by the proximity of special day(Eg: mother's day or valentine's day) from the shopping date
#as the density of data points is not increasing at the special date approaches.
```

```
#Visualization 7 - Density Graph
plot1 <- ggdensity(data, x = "BounceRates", fill = "thistle2", color = "thistle2", add = "median",
                   rug = TRUE) + labs(y = " ")
```

```
## Warning: geom_vline(): Ignoring ‘mapping‘ because ‘xintercept‘ was provided.
```

```
## Warning: geom_vline(): Ignoring ‘data‘ because ‘xintercept‘ was provided.
```

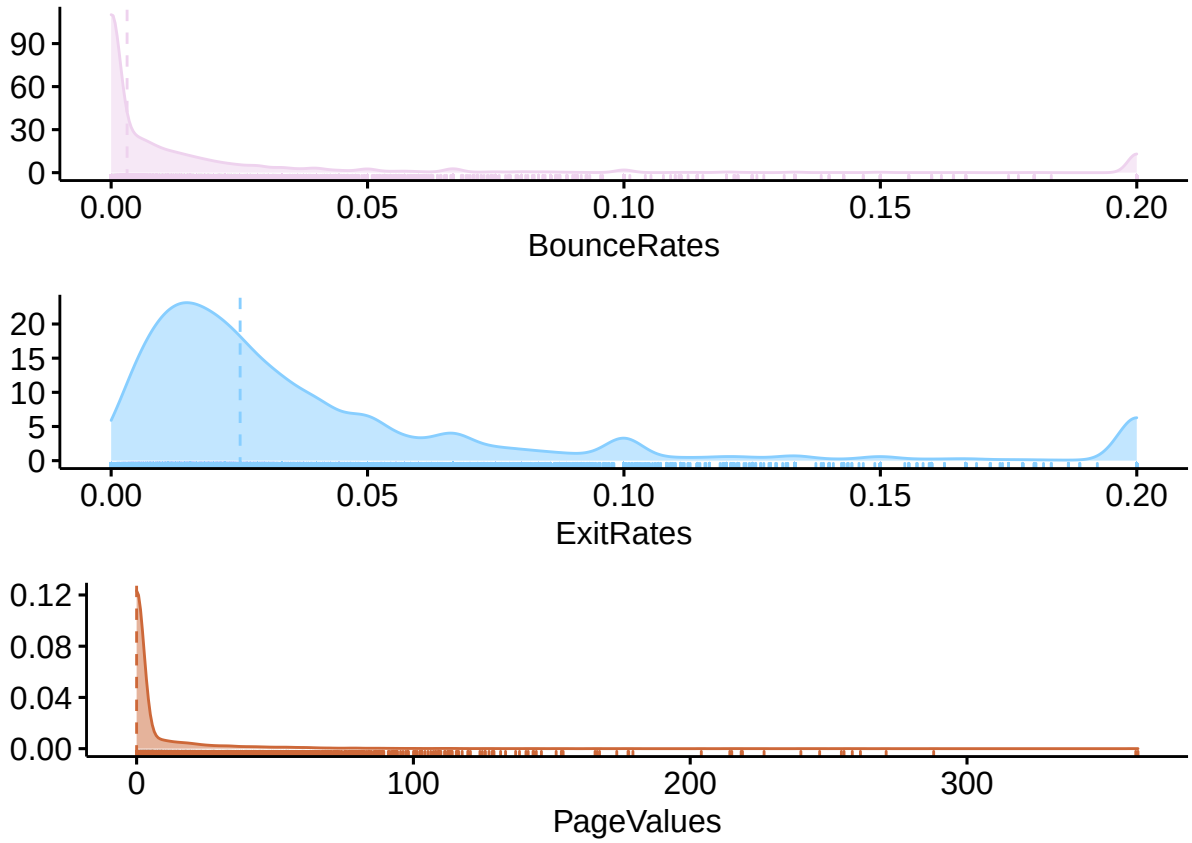
```
plot2 <- ggdensity(data, x = "ExitRates", fill = "skyblue1", color = "skyblue1", add = "median",
                   rug = TRUE) + labs(y = " ")
```

```
## Warning: geom_vline(): Ignoring ‘mapping‘ because ‘xintercept‘ was provided.
## geom_vline(): Ignoring ‘data‘ because ‘xintercept‘ was provided.
```

```
plot3 <- ggdensity(data, x = "PageValues", fill = "sienna3", color = "sienna3", add = "median",
                   rug = TRUE) + labs(y = " ")
```

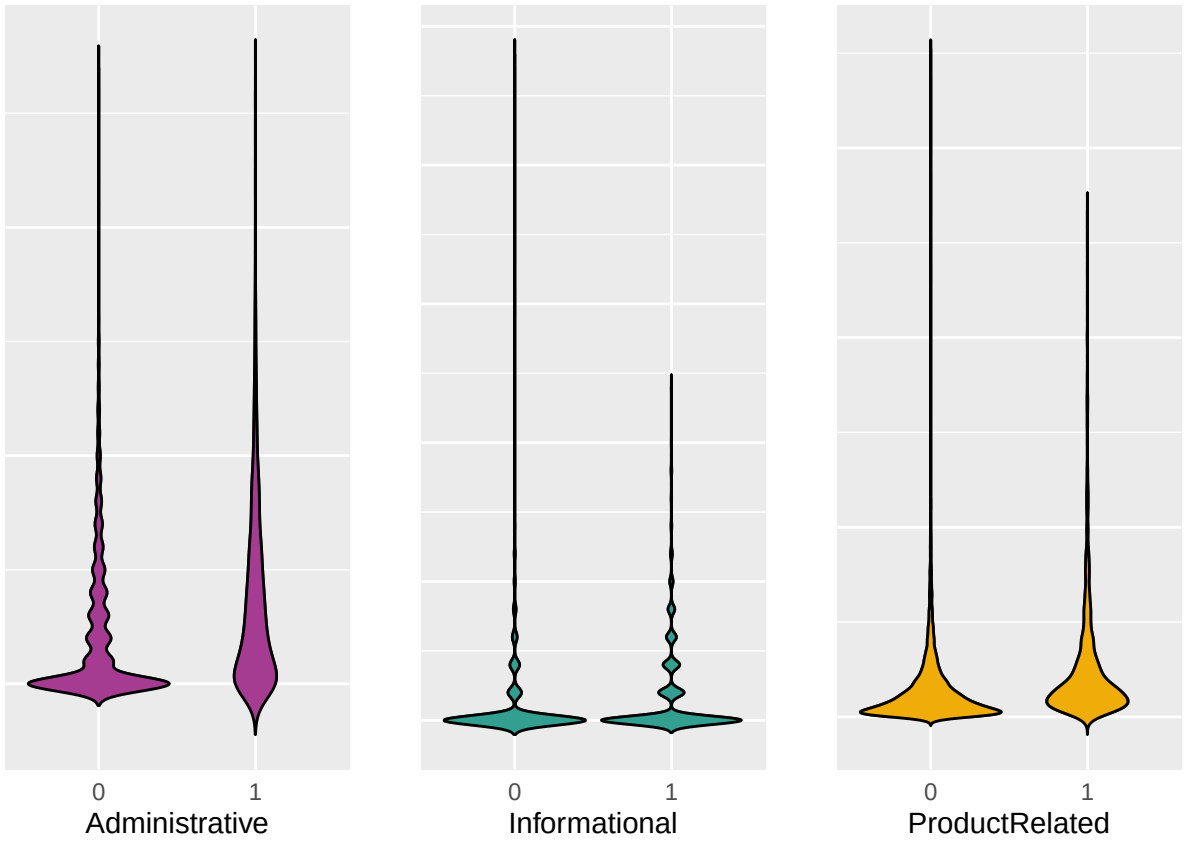
```
## Warning: geom_vline(): Ignoring `mapping` because `xintercept` was provided.
## geom_vline(): Ignoring `data` because `xintercept` was provided.
```

```
grid.arrange(plot1, plot2, plot3, nrow = 3)
```




```
#It can be observed that between the two customer categories of false customers and true customers,
#the exit rate of true customers(who purchased something) is less than that of false customers
#because the true customers stayed on pages with higher probability as they already have prior
#shopping experience and knowledge about the website.
#Similarly, the page values of false customers is way lesser than true customers because they spend
#less time on related pages.
```

```
#Visualization 8 - Violin Plot
plot1 <- ggplot(data, aes(x=Revenue, y=Administrative)) + geom_violin() + geom_violin(trim=FALSE,
fill='#a53c91', color='black') + labs(x = "Administrative") + labs(y = " ") + theme(axis.text.y =
                                                element_blank(), axis.ticks = element_blank())

plot2 <- ggplot(data, aes(x=Revenue, y=Informational)) + geom_violin() + geom_violin(trim=FALSE,
                fill='#329f91', color='black') + labs(x = "Informational") + labs(y = " ") +
                theme(axis.text.y = element_blank(), axis.ticks = element_blank())

plot3 <- ggplot(data, aes(x=Revenue, y=ProductRelated)) + geom_violin() + geom_violin(trim=FALSE,
 fill='#f0ac09', color='black') + labs(x = "ProductRelated") +
              labs(y = " ") + theme(axis.text.y = element_blank(), axis.ticks = element_blank())

grid.arrange(plot1, plot2, plot3, ncol = 3)
```




```
#We may infer that the customers who haven't made a purchase, have spent much more time than
#the customers who have, this could be because the new customers might have had to spend time
#on registration, account creation, feeding details, which as opposed to the existing customers,
#would already have the details stored in the website and would just login to their account saving
#much more time.
# Similarly the false customers(who haven't made any purchase), have spent more time on informational
#page than on the administrative page or product related page. This could be due to lack of experience
#with the website and knowing more about the website page on the informational, which isn't the case
#with the true customers, since they probably are pretty familiar with the website.
# False customers tend to spend more time on Product Related page, when compared to the true customers.
#A valid reason could be, since a true customer has an intention to buy the product, they would spend
#lesser time on the product related page, whereas the false customers tend to compare prices, read
#more about the product in detail, or could be unsure about the decision whether to buy the product
#or not, resulting in spending more time on the Product related page
```