# Stats 140SL Midterm

Hana Yerin Lim

2/2/2021

## Data Preparation

```
library(readr)
library(ggplot2)
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
```

```
data <- read_csv("myMTdata.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   Administrative = col_double(),
##   Administrative_Duration = col_double(),
##   Informational = col_double(),
##   Informational_Duration = col_double(),
##   ProductRelated = col_double(),
##   ProductRelated_Duration = col_double(),
##   BounceRates = col_double(),
##   ExitRates = col_double(),
##   PageValues = col_double(),
##   SpecialDay = col_double(),
##   Month = col_character(),
##   OperatingSystems = col_double(),
##   Browser = col_double(),
##   Region = col_double(),
##   TrafficType = col_double(),
##   VisitorType = col_character(),
##   Weekend = col_logical(),
##   Revenue = col_logical()
## )
```

# Data Description

## 1.

### a) how many non-numeric fields are present?

```
table(unlist(lapply(data, is.numeric)))
```

```
##
## FALSE  TRUE
##     4    14
```

There are 4 non-numeric fields present in the dataset.

### b) how many numeric fields are present, clearly identify which are discrete and which should be treated as continuous?

```
numeric_cols <- data[which(unlist(lapply(data, is.numeric)))]
sort(unlist(lapply(numeric_cols, function(x) sum(!duplicated(x)))))
```

```
##              SpecialDay        OperatingSystems                  Region
##                       6                       7                       9
##                 Browser           Informational             TrafficType
##                      11                      12                      16
##          Administrative          ProductRelated  Informational_Duration
##                      19                     146                     181
##              PageValues             BounceRates Administrative_Duration
##                     246                     312                     491
##              ExitRates ProductRelated_Duration
##                     661                    1012
```

There are 14 numeric fields present in the dataset. In order to observe the amounts of nonduplicated values per column, I wrote the code that counts the non-duplicated values for each columns. Based on the dimension of the dataset (1130 rows, 18 columns), around **7** fields are considered continuous because of the amounts of their unique values.

### c) how many observations were in your dataset

```
# check to see if there are any NA values
# data[which(is.na(data)), ]
# There is no NA value so no rows will be removed
dim(data)
```

```
## [1] 1130   18
```

There are 1130 observations in the dataset.

**a) Duration was measured in 3 different ways, please construct a total duration variable and provide an appropriate statistical summary for your duration variable (you can define a "statistical summary" for us)**

```
newdata <- data %>% mutate(total_duration = Administrative_Duration + Informational_Duration + ProductR
summary(newdata$total_duration)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     0.0   207.7   655.0  1292.3  1604.2 17782.9
```

The mean of the total duration on the website is 1292.3 minutes, the 25th percentile is 207.7 minutes, the 75th percentile is 1604.2 minutes, the minimum is 0 minute, and the maximum is 17782.9 minutes.

**b) The field "Browser" has numerous values, but two of the values dominate. Please recode/reconstruct "Browser" in such a way that there are only three possible values – the two dominant values and all other. Then, tell us whether there is evidence that duration differs by the value of your new "Browser" variable.**

```
# Table before reconstruction
newdata1 <- newdata
table(newdata1$Browser)
```

**i)**

```
##
##   1   2   3   4   5   6   7   8  10  11  13
## 220 716  10  73  49  14   6  14  19   2   7
```

```
newdata1[which(newdata1$Browser >= 3),]$Browser <- 3
# Table after reconstruction
table(newdata1$Browser)
```

```
##
##   1   2   3
## 220 716 194
```

Reconstructed "Browser" that there are only three possible values: 1, 2, and 3.

**ii)** H0: Duration does not differ by the value of the new Browser variable. Ha: Duration differs by the value of the new Browser variable.

```
summary(aov(Browser ~ total_duration, data = newdata1))
```

```
##                   Df Sum Sq Mean Sq F value Pr(>F)
## total_duration     1    1.2  1.2129   3.319 0.0687 .
## Residuals       1128  412.2  0.3654
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Based on the ANOVA test, the p-value is 0.0687. When the significance value is 0.05, we fail to reject the null hypothesis and thus there is no evidence that the duration differs by the value of the new Browser variable. However, when the significance value is 0.1, we reject the null hypothesis and thus we can conclude that there is an evidence to support that duration differs by the value of the new Browser variable.

**c) Which Month/VisitorType (omit VisitorType = Other) combination has the highest proportion of Revenue = TRUE?**

```
newdata1 %>% filter(VisitorType != "Other") %>%
  group_by(VisitorType, Month, Revenue) %>%
  summarise(rev_true = n()) %>%
  mutate(prop = rev_true / sum(rev_true)) %>%
  filter(Revenue == "TRUE") %>%
  arrange(desc(prop))
```

```
## `summarise()` regrouping output by 'VisitorType', 'Month' (override with `.groups` argument)
```

```
## # A tibble: 17 x 5
## # Groups:   VisitorType, Month [17]
##    VisitorType       Month Revenue rev_true   prop
##    <chr>             <chr> <lgl>      <int>  <dbl>
##  1 Returning_Visitor Sep   TRUE          10 0.357
##  2 New_Visitor       Oct   TRUE           4 0.333
##  3 New_Visitor       May   TRUE           9 0.290
##  4 Returning_Visitor Nov   TRUE          70 0.285
##  5 New_Visitor       June  TRUE           2 0.25
##  6 New_Visitor       Sep   TRUE           3 0.25
##  7 Returning_Visitor Aug   TRUE           7 0.219
##  8 New_Visitor       Dec   TRUE           8 0.2
##  9 New_Visitor       Nov   TRUE          10 0.2
## 10 Returning_Visitor Jul   TRUE           7 0.184
## 11 Returning_Visitor Oct   TRUE           5 0.147
## 12 New_Visitor       Aug   TRUE           1 0.143
## 13 Returning_Visitor June  TRUE           2 0.111
## 14 Returning_Visitor Mar   TRUE          12 0.096
## 15 New_Visitor       Mar   TRUE           2 0.0952
## 16 Returning_Visitor May   TRUE          25 0.0947
## 17 Returning_Visitor Dec   TRUE          10 0.0758
```

Returning Visitor and the month of September combination has the highest proportion of the Revenue when it is true by the proportion of 0.35714286.

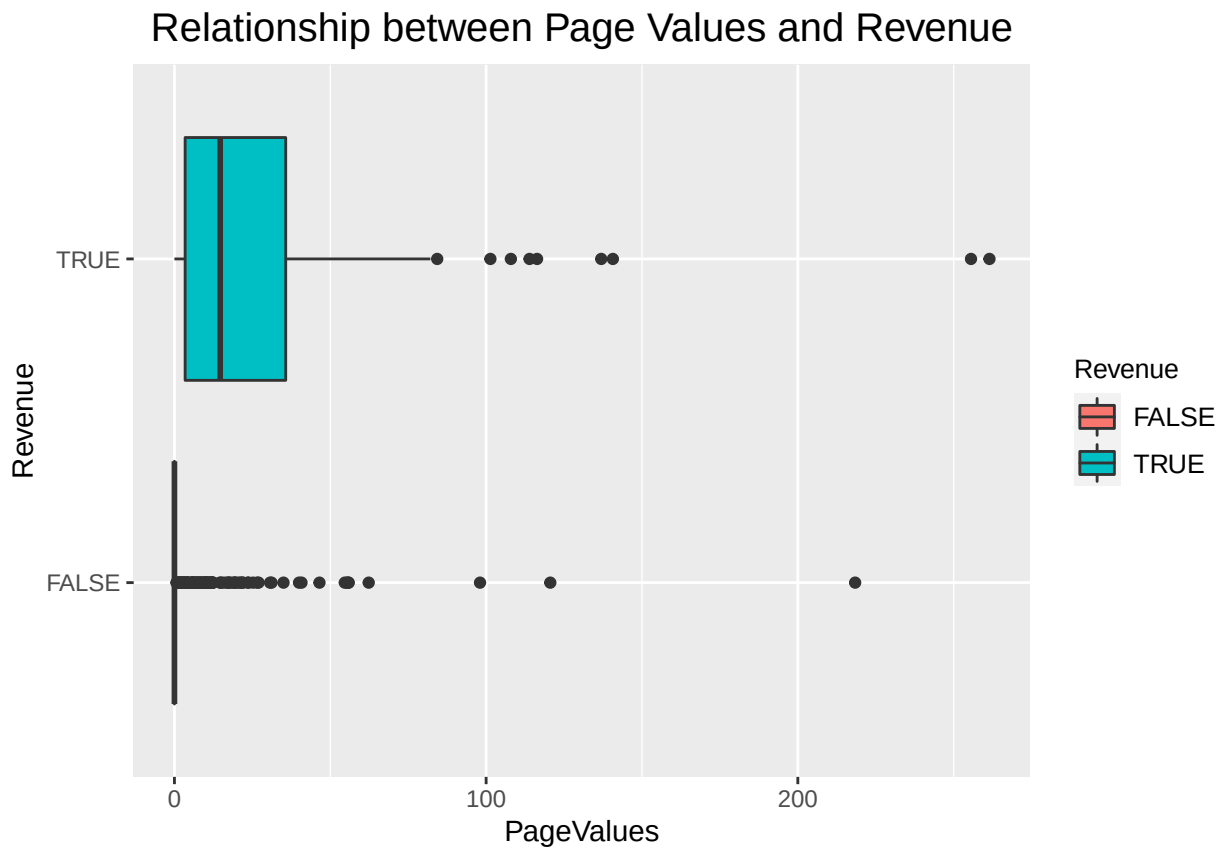
## 2.

```
library(gridExtra)
```

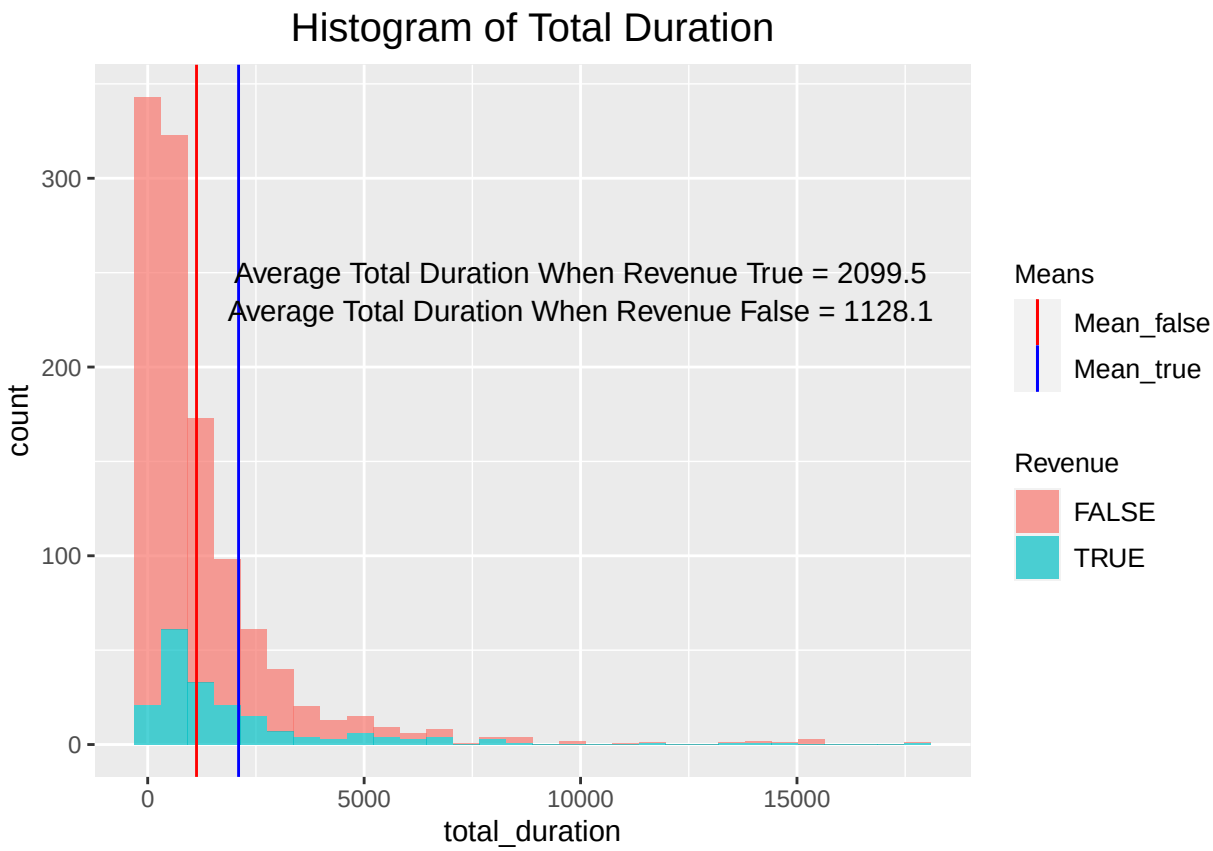
```
##
## Attaching package: 'gridExtra'
```

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

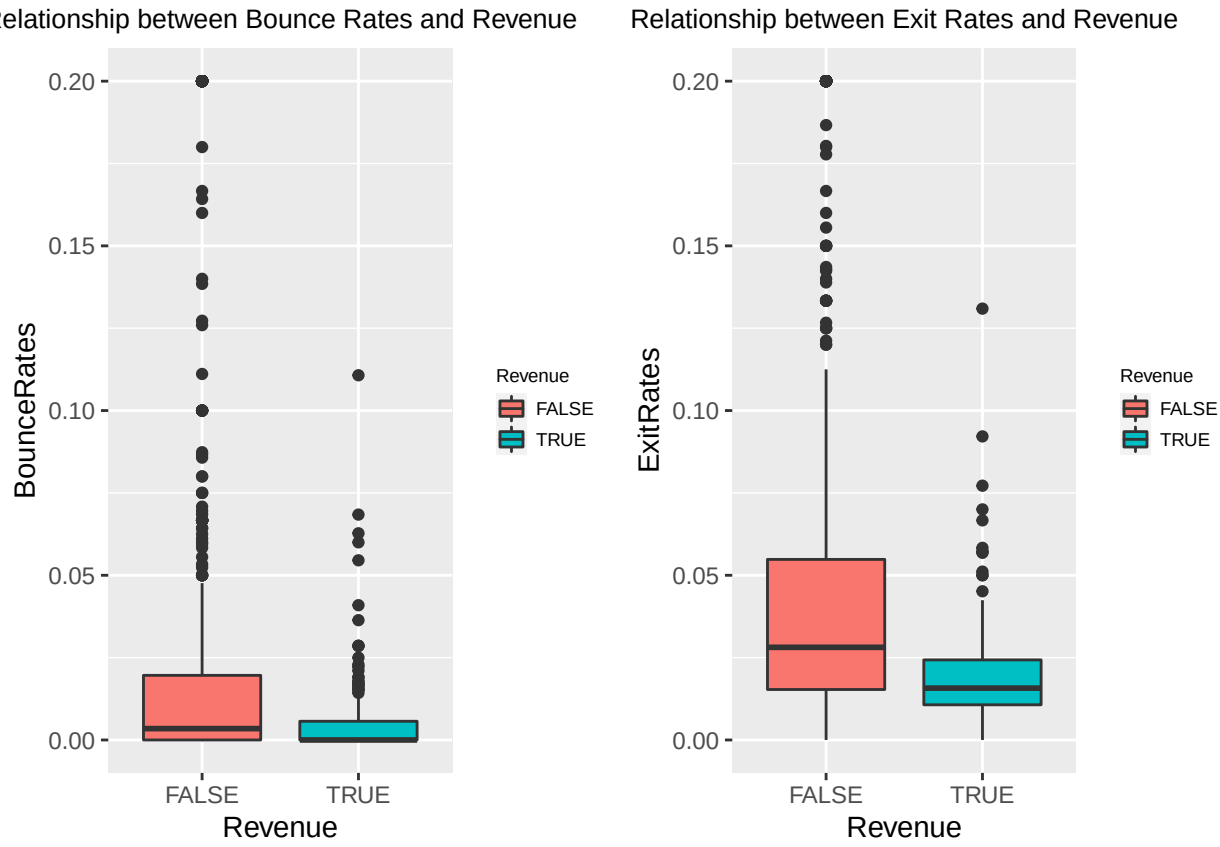
```
# Page values
ggplot(data = newdata1, aes(PageValues, Revenue, fill = Revenue)) +
  geom_boxplot() +
  ggtitle("Relationship between Page Values and Revenue") +
  theme(plot.title = element_text(hjust = 0.5, size = 15), legend.title = element_text(size = 10), leger
```




```
# Total duration
rev_t <- newdata1 %>% filter(Revenue == "TRUE")
rev_f <- newdata1 %>% filter(Revenue == "FALSE")

ggplot(newdata1, aes(total_duration, fill = Revenue)) +
  geom_histogram(bins = 30, alpha = 0.7) +
  geom_vline(aes(xintercept = mean(rev_t$total_duration), color = "Mean_true")) +
  geom_vline(aes(xintercept = mean(rev_f$total_duration), color = "Mean_false")) +
  scale_color_manual(name = "Means", values = c(Mean_true = "blue", Mean_false = "red")) +
  annotate("text", x = c(10000,10000), y = c(250, 230), label = c("Average Total Duration When Revenue
  ggtitle("Histogram of Total Duration") +
  theme(plot.title = element_text(hjust = 0.5, size = 15), legend.title = element_text(size = 10), leger
```




```
# Bounce and exit rates
bounce_rev <- ggplot(newdata1, aes(x = Revenue, y = BounceRates, fill = Revenue)) + geom_boxplot() +
  ggtitle("Relationship between Bounce Rates and Revenue") +
  theme(plot.title = element_text(hjust = 0.5, size = 10), legend.title = element_text(size = 7),
              legend.text = element_text(size = 7), legend.key.size = unit(0.8,"line"))

exit_rev <- ggplot(newdata1, aes(x = Revenue, y = ExitRates, fill = Revenue)) + geom_boxplot() +
  ggtitle("Relationship between Exit Rates and Revenue") +
  theme(plot.title = element_text(hjust = 0.5, size = 10), legend.title = element_text(size = 7),
              legend.text = element_text(size = 7), legend.key.size = unit(0.8,"line"))

grid.arrange(bounce_rev, exit_rev, ncol = 2)
```




```
# Months
newdata1$Month <- factor(newdata1$Month, levels = c("Feb",  "Mar",  "May", "June", "Jul" , "Aug" , "Sep
ct_mo <- suppressWarnings(ggplot(newdata1 %>% group_by(Month, Revenue) %>% summarise(count = n()), aes(
  geom_bar(stat = "identity") +
  geom_text(aes(label = count), vjust=1, position = position_dodge(0.9)) +
  ggtitle("The Amount of Generated Revenue Per Months") +
  theme(plot.title = element_text(hjust = 0.5, size = 15), legend.title = element_text(size = 10),
                legend.text = element_text(size = 10)) +
  labs(x = "Months"))
```

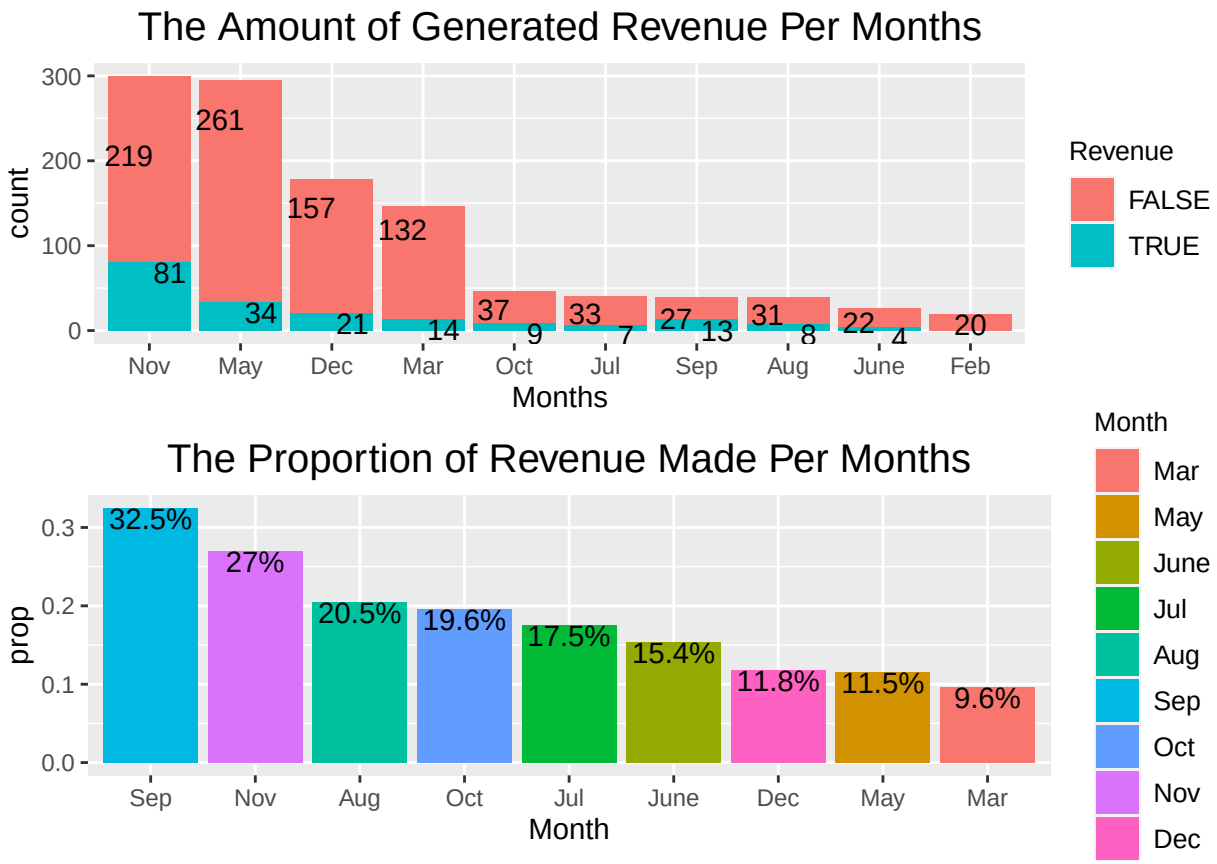
```
## `summarise()` regrouping output by 'Month' (override with `.groups` argument)
```

```
proportion <- newdata1 %>%
  group_by(Month, Revenue) %>%
  summarise(rev_true = n()) %>%
  mutate(prop = rev_true / sum(rev_true)) %>%
  filter(Revenue == "TRUE") %>%
  arrange(desc(prop))
```

```
## `summarise()` regrouping output by 'Month' (override with `.groups` argument)
```

```
prop_mo <- ggplot(proportion, aes(x = reorder(Month, -prop), y = prop, fill = Month)) + geom_bar(stat =
  geom_text(aes(label = paste0(round(prop,3)*100, "%")), vjust=1, position = position_dodge(0.9)) +
  labs(x = "Month") + ggtitle("The Proportion of Revenue Made Per Months") +
```

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

grid.arrange(ct_mo, prop_mo)
```




Description is in video presentation.

### 3.

```
library(reshape2)
glm_mod <- glm(Revenue ~ ., family = "binomial", data = newdata1)
summary(glm_mod)
```

```
##
## Call:
## glm(formula = Revenue ~ ., family = "binomial", data = newdata1)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -5.5365  -0.5042  -0.3500  -0.1411   2.7511
##
## Coefficients: (1 not defined because of singularities)
##                               Estimate Std. Error z value Pr(>|z|)
```

```
## (Intercept)                  -1.505e+01  4.958e+02  -0.030   0.9758
## Administrative                3.358e-03  3.863e-02   0.087   0.9307
## Administrative_Duration      -3.303e-04  7.672e-04  -0.430   0.6669
## Informational                 1.975e-02  8.643e-02   0.229   0.8192
## Informational_Duration        7.590e-05  7.667e-04   0.099   0.9211
## ProductRelated                1.622e-03  4.486e-03   0.362   0.7176
## ProductRelated_Duration       8.597e-05  1.124e-04   0.765   0.4444
## BounceRates                  -5.594e-01  8.873e+00  -0.063   0.9497
## ExitRates                    -1.732e+01  7.470e+00  -2.318   0.0204 *
## PageValues                    8.144e-02  8.240e-03   9.883   <2e-16 ***
## SpecialDay                   -2.055e-01  8.335e-01  -0.247   0.8053
## MonthMar                      1.285e+01  4.958e+02   0.026   0.9793
## MonthMay                      1.285e+01  4.958e+02   0.026   0.9793
## MonthJune                     1.339e+01  4.958e+02   0.027   0.9785
## MonthJul                      1.374e+01  4.958e+02   0.028   0.9779
## MonthAug                      1.409e+01  4.958e+02   0.028   0.9773
## MonthSep                      1.385e+01  4.958e+02   0.028   0.9777
## MonthOct                      1.344e+01  4.958e+02   0.027   0.9784
## MonthNov                      1.399e+01  4.958e+02   0.028   0.9775
## MonthDec                      1.296e+01  4.958e+02   0.026   0.9792
## OperatingSystems              1.190e-02  1.200e-01   0.099   0.9210
## Browser                      -1.150e-01  1.785e-01  -0.644   0.5195
## Region                       -1.437e-02  4.309e-02  -0.333   0.7388
## TrafficType                   2.481e-02  2.694e-02   0.921   0.3570
## VisitorTypeOther              2.311e+00  1.156e+00   2.000   0.0455 *
## VisitorTypeReturning_Visitor -6.210e-02  2.717e-01  -0.229   0.8192
## WeekendTRUE                  -2.245e-01  2.461e-01  -0.912   0.3617
## total_duration                       NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1026.81  on 1129  degrees of freedom
## Residual deviance:  700.01  on 1103  degrees of freedom
## AIC: 754.01
##
## Number of Fisher Scoring iterations: 15
```

```
df <- newdata1[, -which(lapply(newdata1, is.numeric) == FALSE)]
subset(melt(cor(df)), value > 0.5 & value < 1)
```

```
##                       Var1                    Var2     value
## 2   Administrative_Duration          Administrative 0.6073522
## 16           Administrative Administrative_Duration 0.6073522
## 34   Informational_Duration           Informational 0.5487622
## 48            Informational  Informational_Duration 0.5487622
## 66  ProductRelated_Duration          ProductRelated 0.9196129
## 75           total_duration          ProductRelated 0.9132137
## 80           ProductRelated ProductRelated_Duration 0.9196129
## 90           total_duration ProductRelated_Duration 0.9932849
## 98                ExitRates             BounceRates 0.9138687
## 112            BounceRates               ExitRates 0.9138687
## 215          ProductRelated          total_duration 0.9132137
```

```
## 216 ProductRelated_Duration          total_duration 0.9932849
```

One of the efficient methods to identify the fields that are significant for understanding user generated revenue is to create a summary of generalized linear model(glm). Glm function calculates the significance of each variables to Revenue variable, standard error, z-value, deviance, and P-values, which could be used to find the significant predictors. **Page values, Exit Rates, and Visitor Type (other)** are the most significant fields out of 18 fields provided in the dataset. The reason is that the p-values for these fields are less than the significant values, which represent the significance effects on the revenue variable. Furthermore, when the correlation is calculated, we can observe that ExitRates variable is highly correlated with BounceRates, which indicates that BounceRates also play a significant role on the revenue. Therefore, it is important to think about how the page values, bounce and exit rates, and specifically "other" visitor type cause any difference in revenue, and how revenue is created each month.

In addition to the current and previous question, the amounts of users in November, May, December, and March are relatively higher than the amounts in other months. This might be due to the holiday seasons such as Thanksgiving (September), Memorial day, Mother's day (May), Christmas, end of the year sales (December), and much more. However, we can observe from the proportion barplot that the proportions of revenue true during these months (December, May, Mar) are the lowest three. So, it is important to drive a solution to minimize the false revenue.