Dataset Description and PCA Analysis

Your Name

2024-11-15

1. Dataset Description

1.1 Overview

The dataset contains multivariate data, including quantitative, binary, and categorical variables. It has 12,330 rows and 18 columns, representing user online shopping behavior. ## 1.2 Feature Breakdown

Below is a table representation with all the key features:

Feature Name	Type	Description
Administrative	Quantitative	Number of administrative pages visited
Administrative_Duration	Quantitative	Time spent on administrative pages in seconds
Informational	Quantitative	Number of informational pages visited
Informational_Duration	Quantitative	Time spent on informational pages in seconds
ProductRelated	Quantitative	Number of product-related pages visited
ProductRelated_Duration	Quantitative	Time spent on product-related pages in seconds
BounceRates	Quantitative	Percentage of visitors leaving after one page
ExitRates	Quantitative	Percentage of sessions exiting from each page
PageValues	Quantitative	Average value attributed to a page
SpecialDay	Quantitative	Metric indicating proximity to significant holidays
Weekend	Binary	Indicates if the session occurred on a weekend (1 = Yes, 0 = No)
Revenue	Binary	Indicates if the session resulted in a purchase (1 = Yes, 0 = No)
VisitorType	Categorical	Visitor category (e.g., Returning, New, Other)
Month	Categorical	Month of the visit (e.g., Jan, Feb)
OperatingSystems	Categorical	Visitor2019s operating system
Browser	Categorical	Browser used by the visitor
Region	Categorical	Visitor2019s geographical region
TrafficType	Categorical	Type of traffic source leading to the visit

1.3 Initial Observations

```
# Load necessary libraries
library(ggplot2)   # For data visualization
library(readr)     # For reading datasets
library(lattice)   # For lattice-based visualizations
library(reshape2)  # For reshaping data
```

```r
library(dplyr)      # For data manipulation
library(pander)
```

```r
data <- readRDS('/Users/despoinaiapona/Downloads/processed_dataset.rds')

# Checking for missing values
sapply(data, function(x) sum(is.na(x)))
```

```
##          Administrative AdministrativeDuration          Informational
##                       0                       0                      0
##  InformationalDuration         ProductRelated ProductRelatedDuration
##                       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
```

```r
# Split numeric and categorical variables
numeric_vars <- data %>% select(where(is.numeric))
categorical_vars <- data %>% select(where(is.factor))
```

```r
# Summarize numeric variables
numeric_summary <- summary(numeric_vars)

# Display the summary
pander(numeric_summary, caption = "Summary of Numeric Variables")
```

Table 2: Summary of Numeric Variables (continued below)

Administrative	AdministrativeDuration	Informational
Min. : 0.000	Min. : 0.00	Min. : 0.000
1st Qu.: 0.000	1st Qu.: 0.00	1st Qu.: 0.000
Median : 1.000	Median : 10.05	Median : 0.000
Mean : 2.321	Mean : 79.24	Mean : 0.577
3rd Qu.: 4.000	3rd Qu.: 94.00	3rd Qu.: 0.000
Max. :24.000	Max. :2629.25	Max. :24.000

Table 3: Table continues below

InformationalDuration	ProductRelated	ProductRelatedDuration
Min. : 0.00	Min. : 0.00	Min. : 0.0
1st Qu.: 0.00	1st Qu.: 7.00	1st Qu.: 187.7
Median : 0.00	Median : 18.00	Median : 602.0

InformationalDuration	ProductRelated	ProductRelatedDuration
Mean : 36.28	Mean : 30.53	Mean : 1161.3
3rd Qu.: 0.00	3rd Qu.: 37.00	3rd Qu.: 1459.5
Max. :2050.43	Max. :705.00	Max. :43171.2

BounceRates	ExitRates	PageValues
Min. :0.000000	Min. :0.00000	Min. : 0.000
1st Qu.:0.000000	1st Qu.:0.01467	1st Qu.: 0.000
Median :0.003089	Median :0.02581	Median : 0.000
Mean :0.023104	Mean :0.04380	Mean : 5.899
3rd Qu.:0.018182	3rd Qu.:0.04839	3rd Qu.: 0.000
Max. :0.200000	Max. :0.20000	Max. :255.569

```
# Summarize categorical variables
categorical_summary <- summary(categorical_vars)

# Display the summary
pander(categorical_summary, caption = "Summary of Categorical Variables")
```

Table 5: Summary of Categorical Variables (continued below)

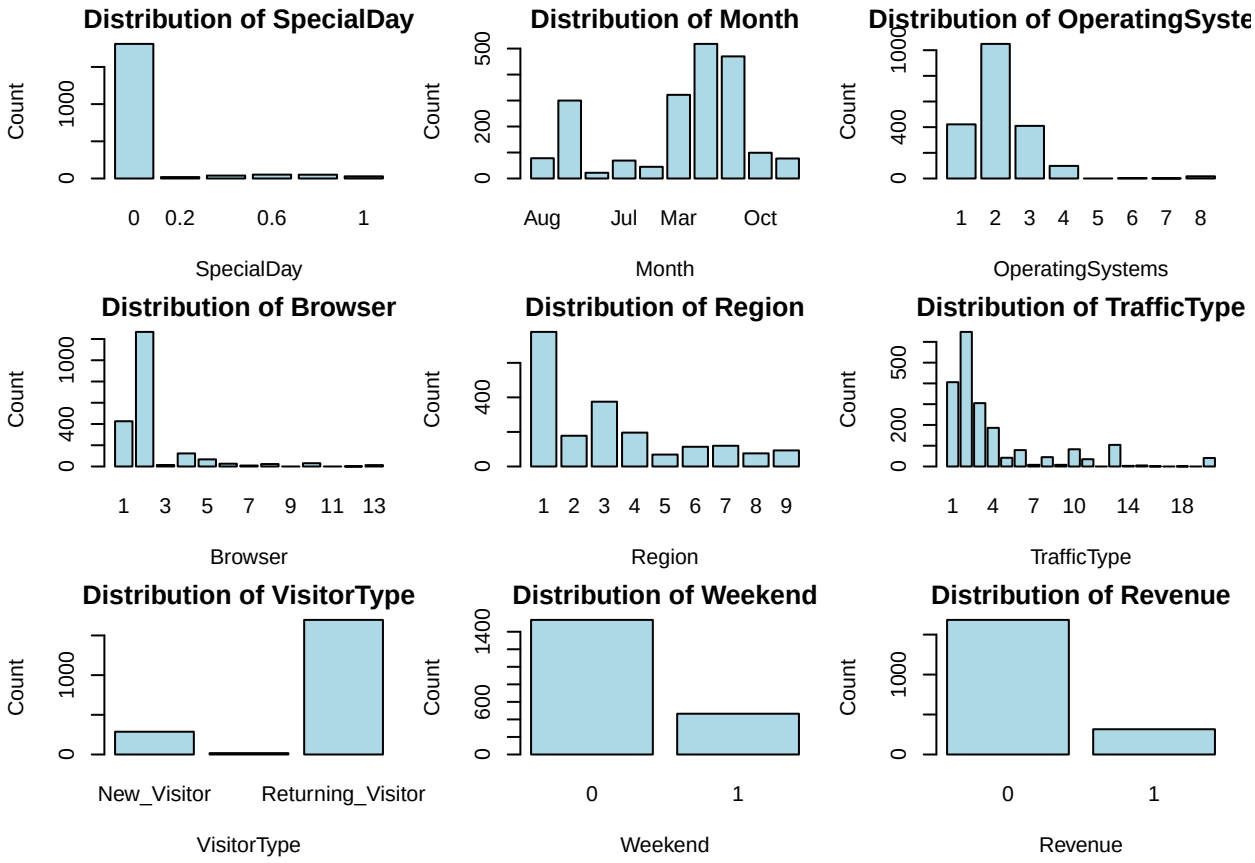
SpecialDay	Month	OperatingSystems	Browser	Region
0 :1810	May :518	2 :1049	2 :1267	1 :779
0.2: 19	Nov :470	1 : 422	1 : 426	3 :375
0.4: 39	Mar :322	3 : 410	4 : 123	4 :196
0.6: 52	Dec :300	4 : 98	5 : 67	2 :178
0.8: 51	Oct : 99	8 : 17	10 : 31	7 :120
1 : 29	Aug : 78	6 : 3	6 : 26	6 :114
NA	(Other):213	(Other): 1	(Other): 60	(Other):238

TrafficType	VisitorType	Weekend	Revenue
2 :648	New_Visitor : 287	0:1535	0:1684
1 :406	Other : 17	1: 465	1: 316
3 :305	Returning_Visitor:1696	NA	NA
4 :186	NA	NA	NA
13 :104	NA	NA	NA
10 : 83	NA	NA	NA
(Other):268	NA	NA	NA

1.4 Visualizations

1.4.1 Barplots for Categorical Variables

```
# Set the layout for 3 rows and 3 columns (9 plots per page)
par(mfrow = c(3, 3), mar = c(4, 4, 2, 1))  # Adjust margins

# Create barplots
for (var_name in names(categorical_vars)) {
  barplot(table(categorical_vars[[var_name]]),
          main = paste("Distribution of", var_name),
          xlab = var_name,
          ylab = "Count",
          col = "lightblue")
}
```

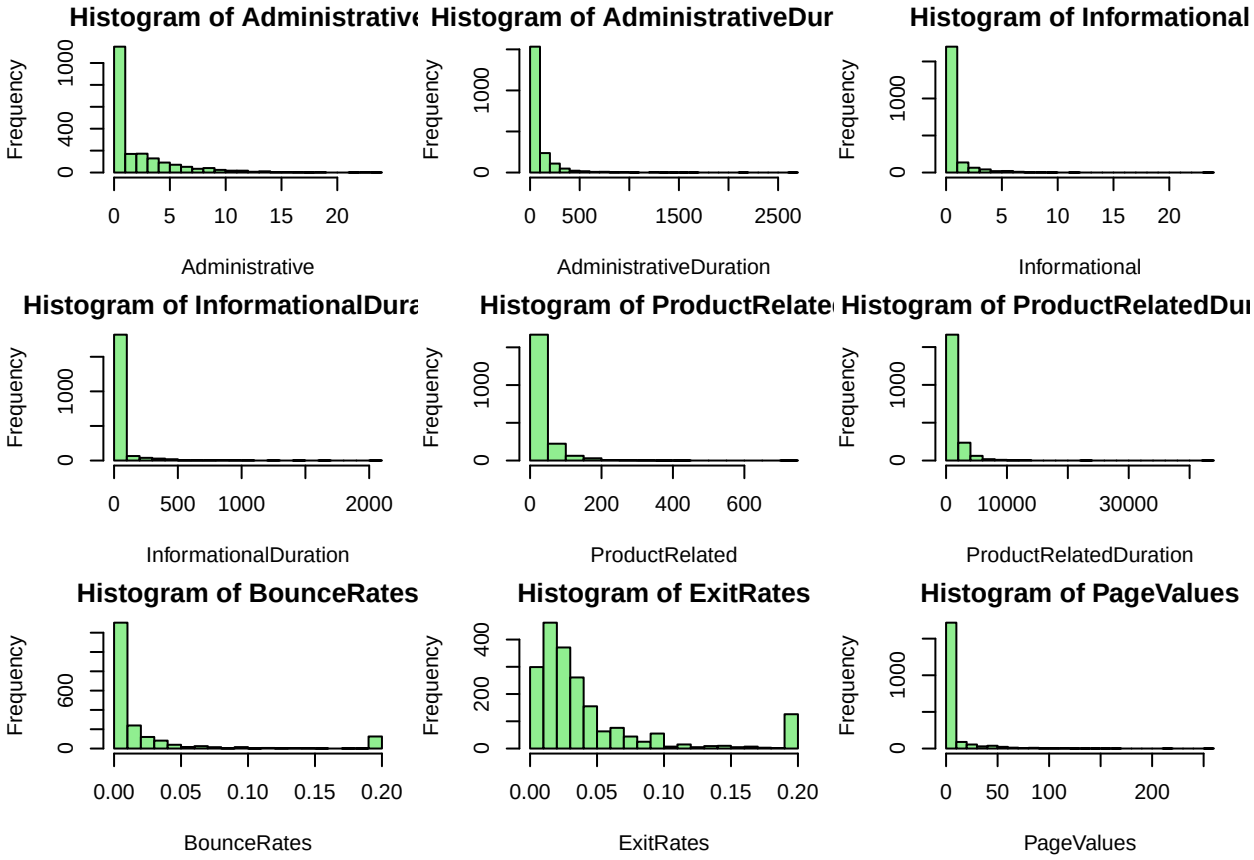


1.4.1.1 General Observations

- Most sessions happen on regular days, with little activity on holidays.
- There are seasonal patterns, with high points in May and November.
- Most users go with Operating System 2 and Browser 2.
- Traffic comes mostly from Region 1 and Traffic Type 2.

- Returning visitors account for a significant part of site sessions, indicating that users enjoy the content and experience.
- The majority of sessions are on weekdays, and only a small part lead to sales.

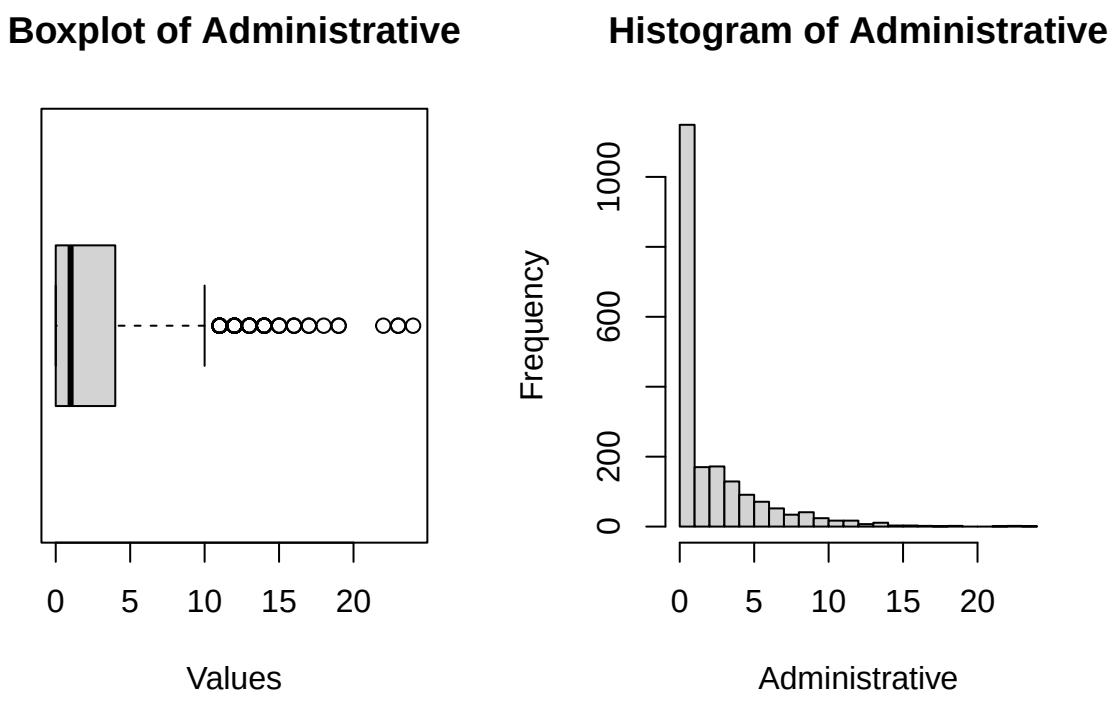
```r
# Set layout for 3 rows and 3 columns (9 plots per page)
par(mfrow = c(3, 3), mar = c(4, 4, 2, 1))  # Adjust margins

# Create histograms for each numeric variable
for (var_name in names(numeric_vars)) {
  hist(numeric_vars[[var_name]],
       main = paste("Histogram of", var_name),
       xlab = var_name,
       col = "lightgreen",
       breaks = 20)
}
```



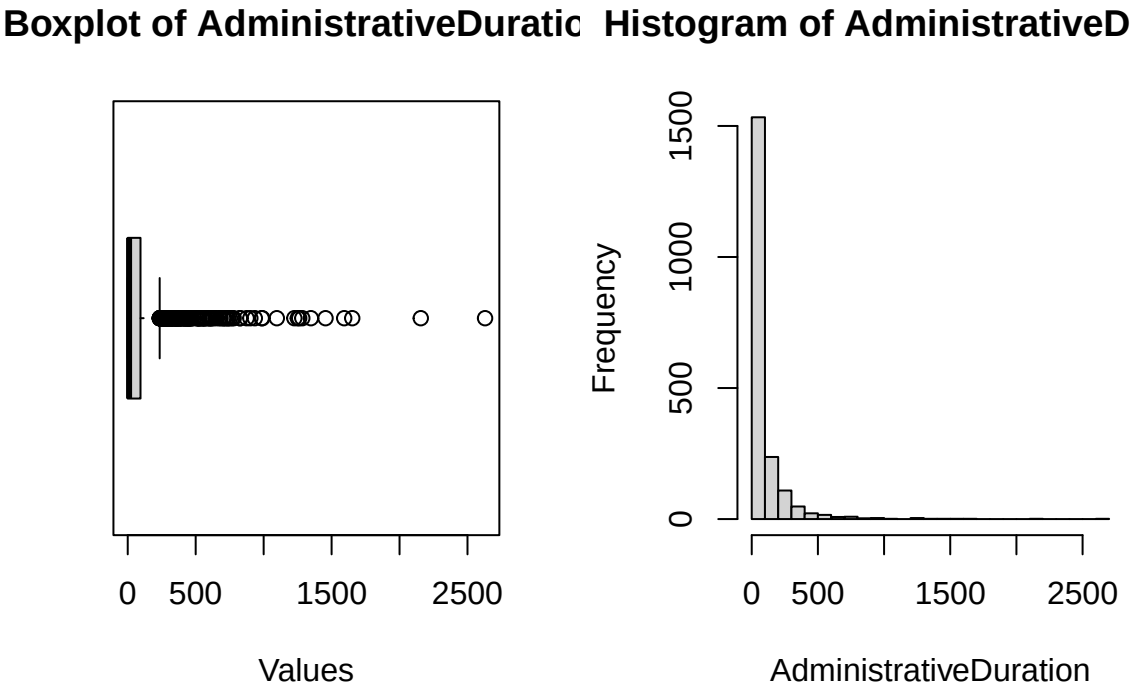

```r
for (i in 1:ncol(numeric_vars)) {
  var_name <- colnames(numeric_vars)[i]

  par(mfrow = c(1, 2), mar = c(5, 4, 4, 2) + 0.1, oma = c(0, 0, 2, 0))

  # Boxplot
  boxplot(numeric_vars[[i]], horizontal = TRUE,
```

```
          main = paste("Boxplot of", var_name),
          xlab = "Values")

  # Histogram
  hist(numeric_vars[[i]], breaks = 20,
       main = paste("Histogram of", var_name),
       xlab = var_name,
       ylab = "Frequency")

  # Add mean and median to the title
  mtext(paste("Variable:", var_name, "| Mean:", round(mean(numeric_vars[[i]]), 2),
              "| Median:", round(median(numeric_vars[[i]]), 2)),
        outer = TRUE, cex = 1, line = -1)
}
```

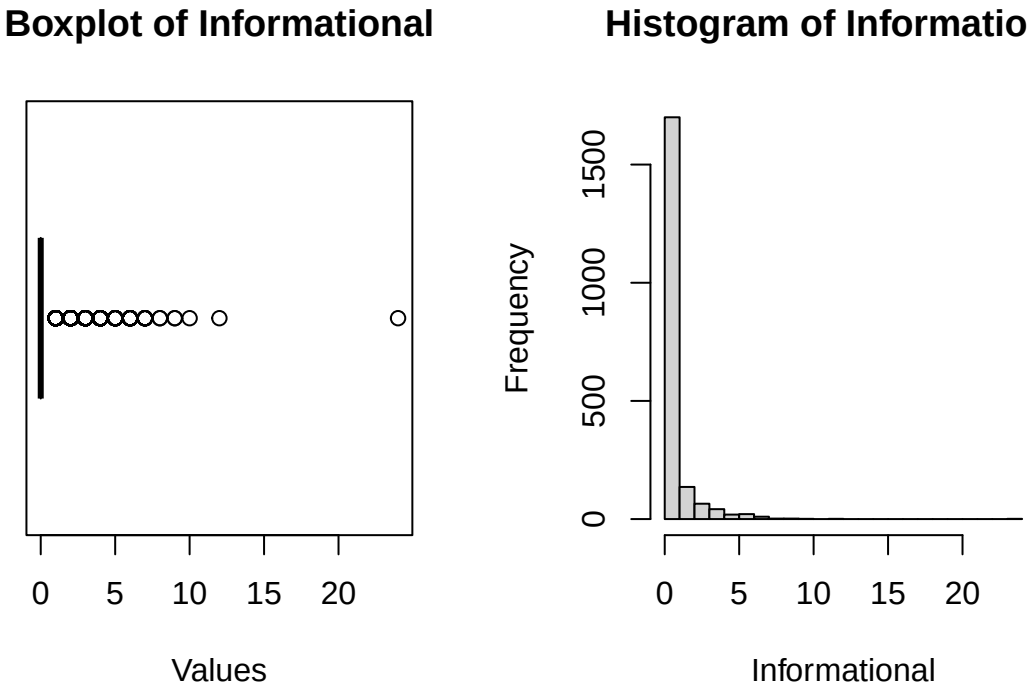
Variable: Administrative | Mean: 2.32 | Median: 1



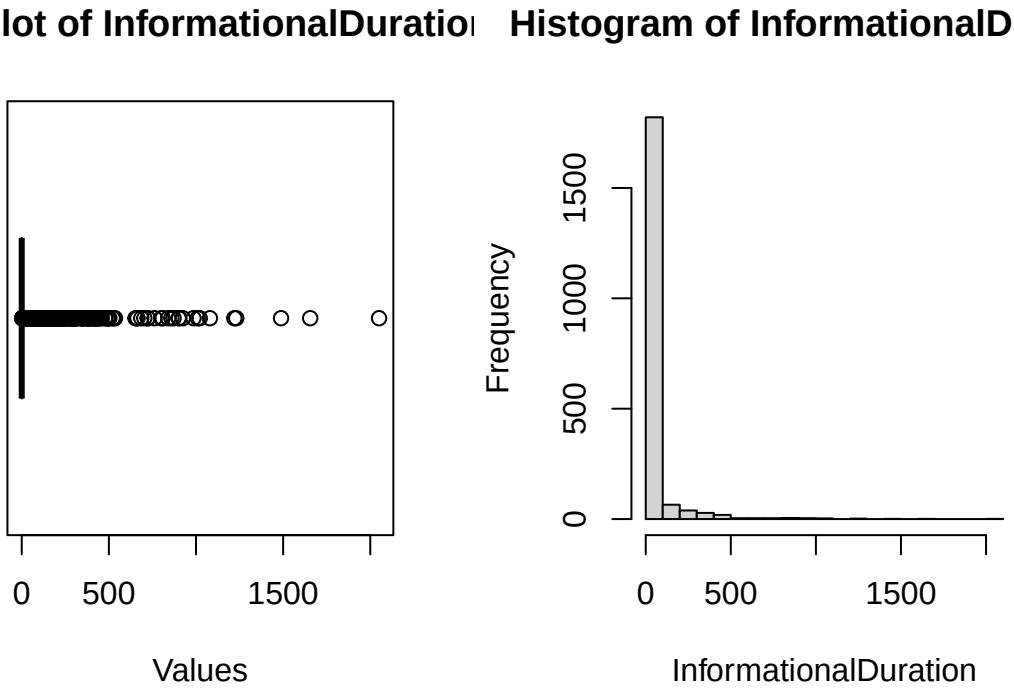
Variable: AdministrativeDuration | Mean: 79.24 | Median: 10.05



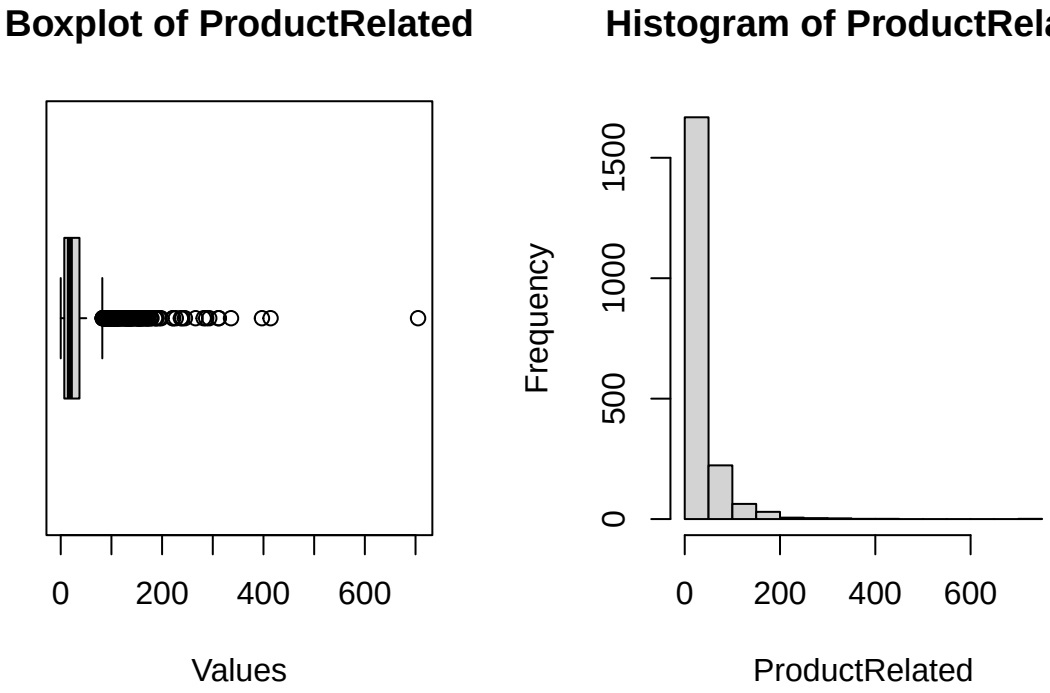
Variable: Informational | Mean: 0.58 | Median: 0



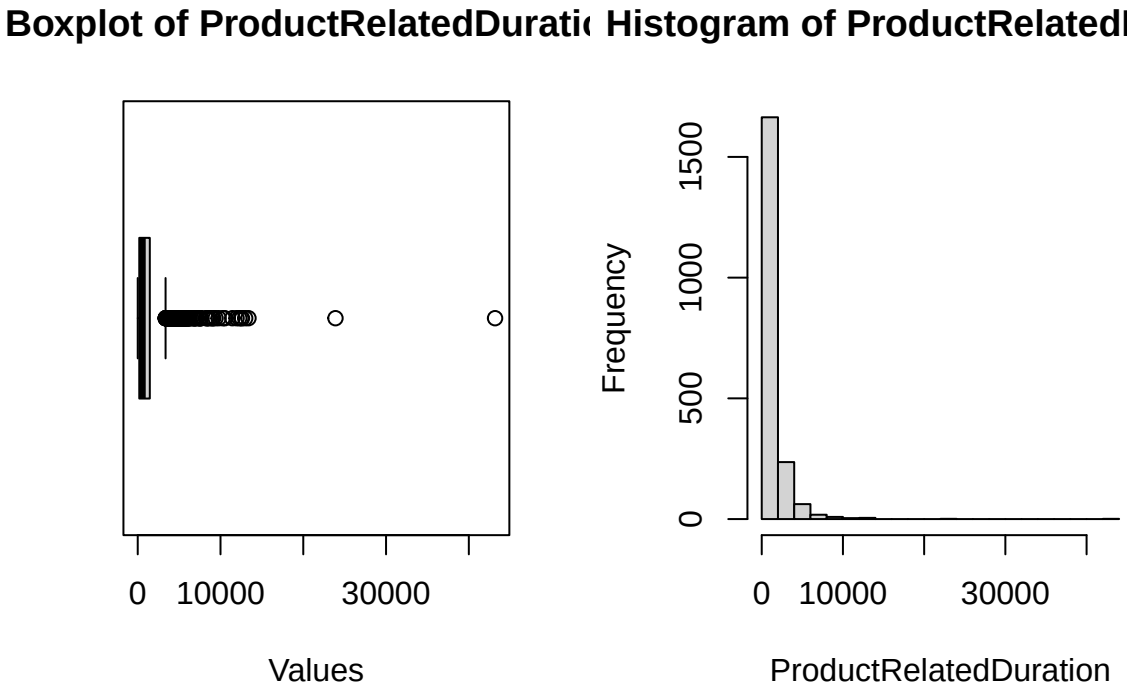
Variable: InformationalDuration | Mean: 36.28 | Median: 0



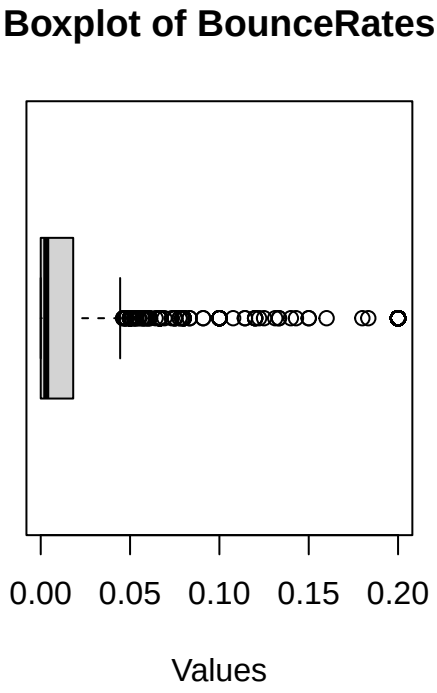
Variable: ProductRelated | Mean: 30.53 | Median: 18

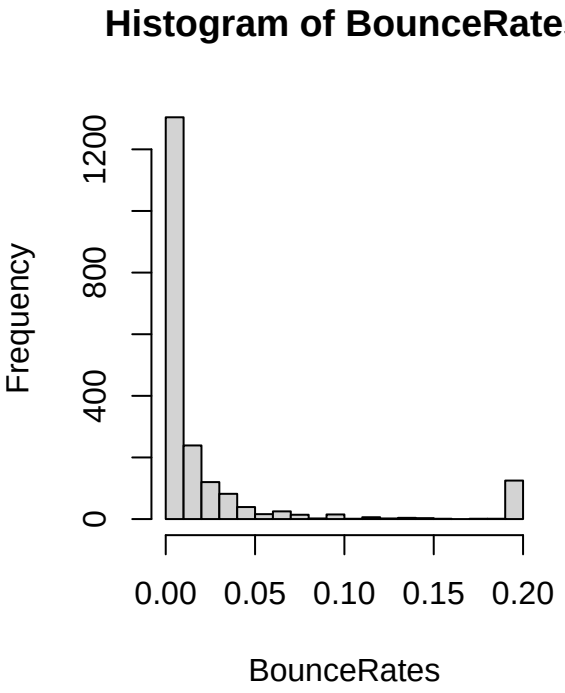


Variable: ProductRelatedDuration | Mean: 1161.28 | Median: 601.97

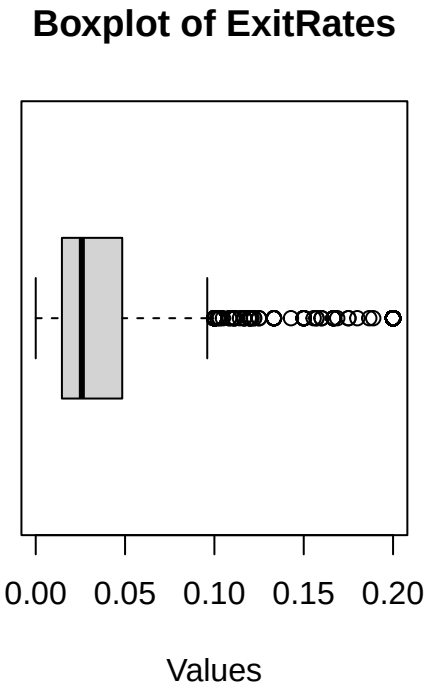


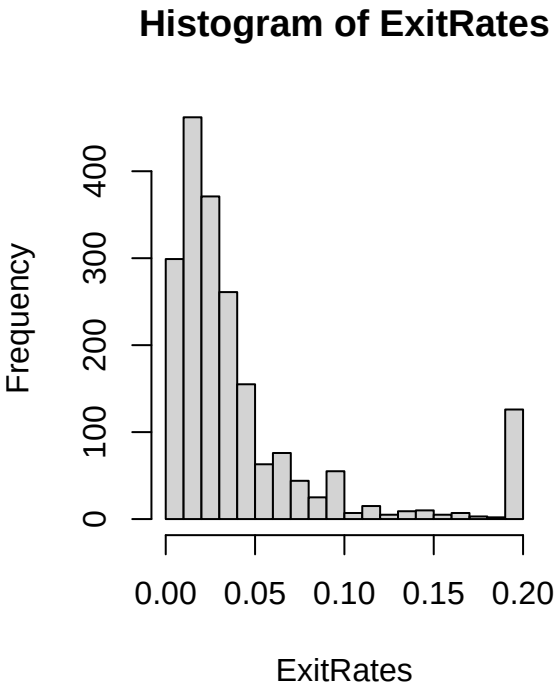
Variable: BounceRates | Mean: 0.02 | Median: 0



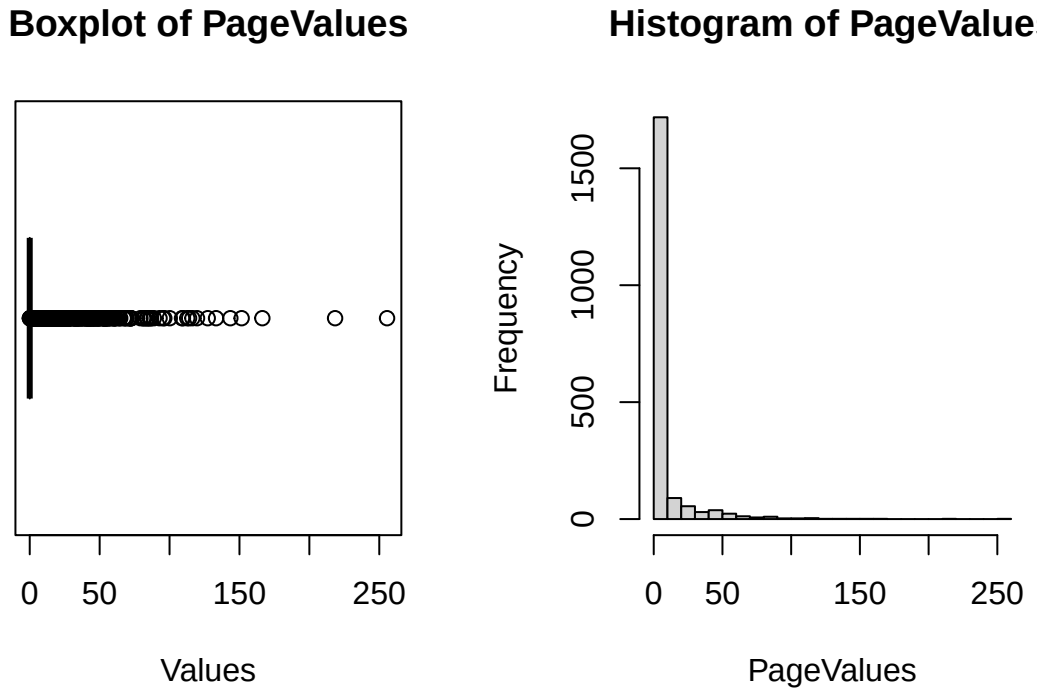


Variable: ExitRates | Mean: 0.04 | Median: 0.03

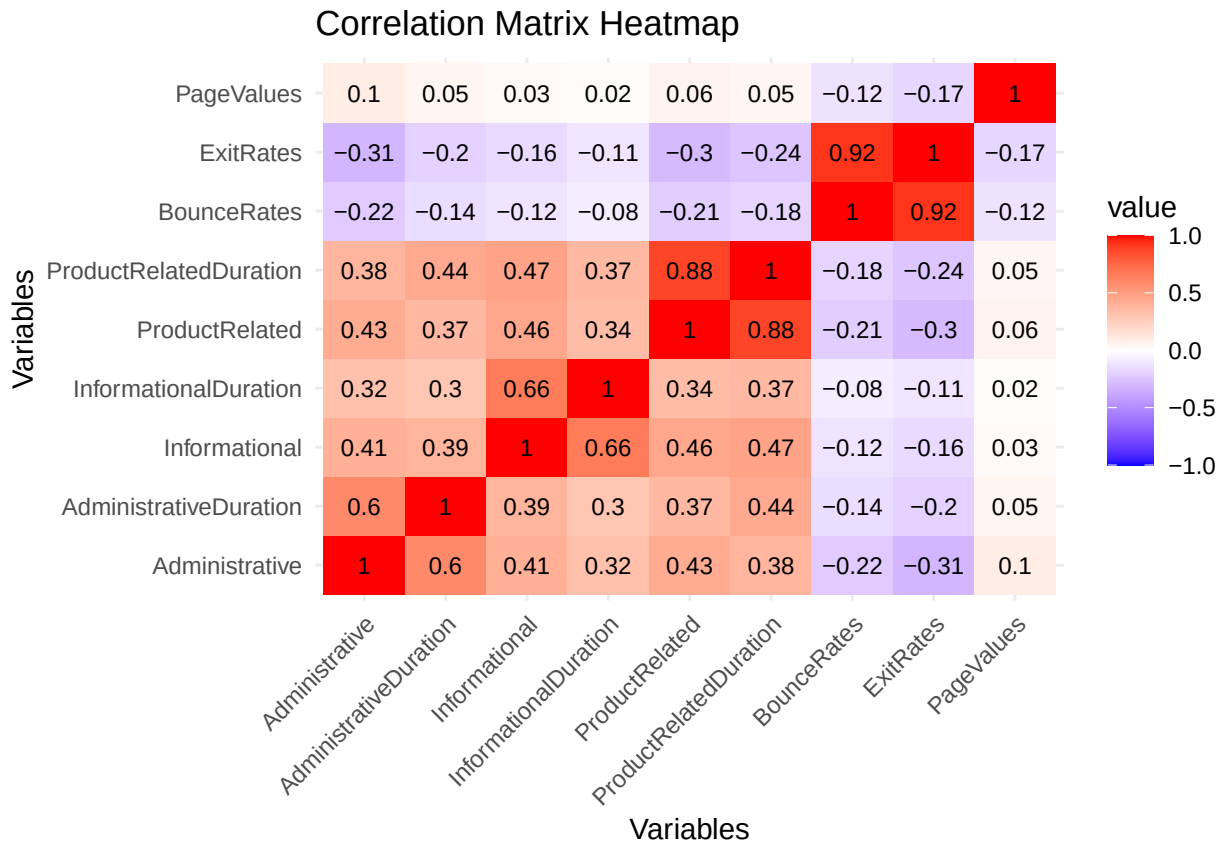




Variable: PageValues | Mean: 5.9 | Median: 0




```
# Compute the correlation matrix
correlation_matrix <- cor(numeric_vars)

correlation_melted <- melt(correlation_matrix)

# Heatmap of the correlation matrix with values
ggplot(correlation_melted, aes(Var1, Var2, fill = value)) +
  geom_tile() +  # Create the heatmap tiles
  geom_text(aes(label = round(value, 2)), color = "black", size = 3) +  # Add the correlation values in
  scale_fill_gradient2(low = "blue", high = "red", mid = "white", midpoint = 0, limit = c(-1, 1)) +  #
  labs(title = "Correlation Matrix Heatmap", x = "Variables", y = "Variables") +  # Title and labels
  theme_minimal() +  # Clean theme
  theme(axis.text.x = element_text(angle = 45, hjust = 1))  # Rotate x-ax
```




```r
# PCA with R built in function
pca <- prcomp(numeric_vars, center = TRUE, scale. = TRUE)
pca$rotation
```

```
##                                PC1        PC2         PC3         PC4
## Administrative          0.37105796  0.0329912  0.21953028 -0.05220820
## AdministrativeDuration  0.34714509  0.1235379  0.19485757  0.01541194
## Informational           0.37346416  0.2609986  0.04525560 -0.39448582
## InformationalDuration   0.31872058  0.2662497  0.07498729 -0.54298773
## ProductRelated          0.41740872  0.1120424 -0.22096803  0.47422249
## ProductRelatedDuration  0.41707403  0.1563865 -0.21780465  0.45935918
## BounceRates            -0.25010949  0.6305022  0.16507323  0.16667502
## ExitRates              -0.29019922  0.6081152  0.11151647  0.11549280
## PageValues              0.07613712 -0.2007321  0.87761893  0.26375189
##                                PC5          PC6          PC7          PC8
## Administrative          0.52507075  0.713679807 -0.045214693 -0.1457811805
## AdministrativeDuration  0.60278060 -0.662670738 -0.049419726  0.1448946839
## Informational          -0.23809702 -0.024825486  0.759799087  0.0007661561
## InformationalDuration  -0.33796380 -0.026948406 -0.641251513  0.0299799561
## ProductRelated         -0.20213008  0.153751548 -0.040885215  0.6700840381
## ProductRelatedDuration -0.18765289 -0.123524046 -0.071149018 -0.6888790978
## BounceRates             0.04411895  0.098547988 -0.010363055  0.1339391720
## ExitRates               0.02890485  0.006699396  0.007341775 -0.1234750628
## PageValues             -0.33345823 -0.039138054  0.010958484 -0.0100583735
##                                PC9
```

```
## Administrative          0.044425841
## AdministrativeDuration  0.010389639
## Informational          -0.011634553
## InformationalDuration   0.004633944
## ProductRelated          0.155956055
## ProductRelatedDuration -0.114255943
## BounceRates            -0.674668205
## ExitRates               0.709947449
## PageValues              0.034384375
```

```r
# PCA step by step

eig_result <- eigen(correlation_matrix)

# Get the eigenvalues and eigenvectors
Lambda <- eig_result$values  # Eigenvalues
T <- eig_result$vectors      # Eigenvectors

# Sort eigenvalues and eigenvectors in descending order
sorted_indices <- order(Lambda, decreasing = TRUE)  # Indices for sorting

# Apply the sorting
Lambda_sorted <- Lambda[sorted_indices]  # Sorted eigenvalues
T_sorted <- T[, sorted_indices]          # Sorted eigenvectors

# Show the sorted eigenvalues
Lambda_sorted
```

```
## [1] 3.57835089 1.72376992 0.96781746 0.93079294 0.88913071 0.40830014 0.32216982
## [8] 0.11170116 0.06796695
```

```r
# Show the eigenvectors
T_sorted
```

```
##              [,1]       [,2]        [,3]        [,4]        [,5]         [,6]
##  [1,]  0.37105796 -0.0329912  0.21953028 -0.05220820  0.52507075  0.713679807
##  [2,]  0.34714509 -0.1235379  0.19485757  0.01541194  0.60278060 -0.662670738
##  [3,]  0.37346416 -0.2609986  0.04525560 -0.39448582 -0.23809702 -0.024825486
##  [4,]  0.31872058 -0.2662497  0.07498729 -0.54298773 -0.33796380 -0.026948406
##  [5,]  0.41740872 -0.1120424 -0.22096803  0.47422249 -0.20213008  0.153751548
##  [6,]  0.41707403 -0.1563865 -0.21780465  0.45935918 -0.18765289 -0.123524046
##  [7,] -0.25010949 -0.6305022  0.16507323  0.16667502  0.04411895  0.098547988
##  [8,] -0.29019922 -0.6081152  0.11151647  0.11549280  0.02890485  0.006699396
##  [9,]  0.07613712  0.2007321  0.87761893  0.26375189 -0.33345823 -0.039138054
##               [,7]          [,8]         [,9]
##  [1,]  0.045214693 -0.1457811805  0.044425841
##  [2,]  0.049419726  0.1448946839  0.010389639
##  [3,] -0.759799087  0.0007661561 -0.011634553
##  [4,]  0.641251513  0.0299799561  0.004633944
##  [5,]  0.040885215  0.6700840381  0.155956055
##  [6,]  0.071149018 -0.6888790978 -0.114255943
##  [7,]  0.010363055  0.1339391720 -0.674668205
##  [8,] -0.007341775 -0.1234750628  0.709947449
##  [9,] -0.010958484 -0.0100583735  0.034384375
```