Week 8 IP

Peter Kiragu

7/10/2020

1. Introduction

- The goal of this analysis is to conduct explanatory data analysis to reveals patterns in the data.
- The metric for success is getting meaning information that allows us to understand the variables in our dataset.

1.1 Context

A Kenyan entrepreneur has created an online cryptography course and would want to advertise it on her blog. She currently targets audiences originating from various countries. In the past, she ran ads to advertise a related course on the same blog and collected data in the process. She would now like to employ your services as a Data Science Consultant to help her identify which individuals are most likely to click on her ads.

2. Reading & Previewing Data

```
# First we we need to import the dataset

advert_data <- read.csv("advertising.csv")

# Previewing the top of out data

head(advert_data)
```

```
##   Daily.Time.Spent.on.Site Age Area.Income Daily.Internet.Usage
## 1                    68.95  35    61833.90               256.09
## 2                    80.23  31    68441.85               193.77
## 3                    69.47  26    59785.94               236.50
## 4                    74.15  29    54806.18               245.89
## 5                    68.37  35    73889.99               225.58
## 6                    59.99  23    59761.56               226.74
##                           Ad.Topic.Line           City Male    Country
## 1    Cloned 5thgeneration orchestration    Wrightburgh    0    Tunisia
## 2    Monitored national standardization      West Jodi    1      Nauru
## 3      Organic bottom-line service-desk       Davidton    0 San Marino
## 4 Triple-buffered reciprocal time-frame West Terrifurt    1      Italy
## 5         Robust logistical utilization   South Manuel    0    Iceland
## 6       Sharable client-driven software      Jamieberg    1     Norway
##              Timestamp Clicked.on.Ad
## 1 2016-03-27 00:53:11             0
## 2 2016-04-04 01:39:02             0
## 3 2016-03-13 20:35:42             0
## 4 2016-01-10 02:31:19             0
## 5 2016-06-03 03:36:18             0
## 6 2016-05-19 14:30:17             0
```

```
# Previewing the bottom of out data

tail(advert_data)
```

```
##      Daily.Time.Spent.on.Site Age Area.Income Daily.Internet.Usage
## 995                     43.70  28    63126.96               173.01
## 996                     72.97  30    71384.57               208.58
## 997                     51.30  45    67782.17               134.42
## 998                     51.63  51    42415.72               120.37
## 999                     55.55  19    41920.79               187.95
## 1000                    45.01  26    29875.80               178.35
##                               Ad.Topic.Line          City Male
## 995         Front-line bifurcated ability  Nicholasland    0
## 996         Fundamental modular algorithm     Duffystad    1
## 997       Grass-roots cohesive monitoring   New Darlene    1
## 998          Expanded intangible solution South Jessica    1
## 999  Proactive bandwidth-monitored policy   West Steven    0
## 1000      Virtual 5thgeneration emulation   Ronniemouth    0
##                      Country           Timestamp Clicked.on.Ad
## 995                  Mayotte 2016-04-04 03:57:48             1
## 996                  Lebanon 2016-02-11 21:49:00             1
## 997   Bosnia and Herzegovina 2016-04-22 02:07:01             1
## 998                 Mongolia 2016-02-01 17:24:57             1
## 999                Guatemala 2016-03-24 02:35:54             0
## 1000                  Brazil 2016-06-03 21:43:21             1
```

3. Checking Our Data

```
# Checking the class of the object "advert_data"

class(advert_data)
```

```
## [1] "data.frame"
```

```
# Our object is a data frame

# Checking the dimension of our dataset

dim(advert_data)
```

```
## [1] 1000   10
```

```
# Our dataset has 1000 rows and 10 columns

# Checking the structure of our data frame

str(advert_data)
```

```
## 'data.frame':    1000 obs. of  10 variables:
##  $ Daily.Time.Spent.on.Site: num  69 80.2 69.5 74.2 68.4 ...
##  $ Age                     : int  35 31 26 29 35 23 33 48 30 20 ...
##  $ Area.Income             : num  61834 68442 59786 54806 73890 ...
##  $ Daily.Internet.Usage    : num  256 194 236 246 226 ...
##  $ Ad.Topic.Line           : chr  "Cloned 5thgeneration orchestration" "Monitored national standardi
##  $ City                    : chr  "Wrightburgh" "West Jodi" "Davidton" "West Terrifurt" ...
##  $ Male                    : int  0 1 0 1 0 1 0 1 1 1 ...
##  $ Country                 : chr  "Tunisia" "Nauru" "San Marino" "Italy" ...
##  $ Timestamp               : chr  "2016-03-27 00:53:11" "2016-04-04 01:39:02" "2016-03-13 20:35:42"
##  $ Clicked.on.Ad           : int  0 0 0 0 0 0 0 1 0 0 ...
```

```
# Our data frame has integer, number and character values
```

```
# Getting the names of the columns we will be working with

colnames(advert_data)
```

```
##  [1] "Daily.Time.Spent.on.Site" "Age"
##  [3] "Area.Income"              "Daily.Internet.Usage"
##  [5] "Ad.Topic.Line"            "City"
##  [7] "Male"                     "Country"
##  [9] "Timestamp"                "Clicked.on.Ad"
```

```
# "Daily.Time.Spent.on.Site" , "Age", "Area.Income", "Daily.Internet.Usage", "Ad.Topic.Line"
# "City", "Male", "Country", "Timestamp", "Clicked.on.Ad"
```

4. Cleaning Data

```
# Checking for duplicated values in our data set

anyDuplicated(advert_data)
```

```
## [1] 0
```

```
# Since there are no duplicated values, no action is required
```

```
# Checking if our dataset has any missing values

sum(is.na(advert_data))
```

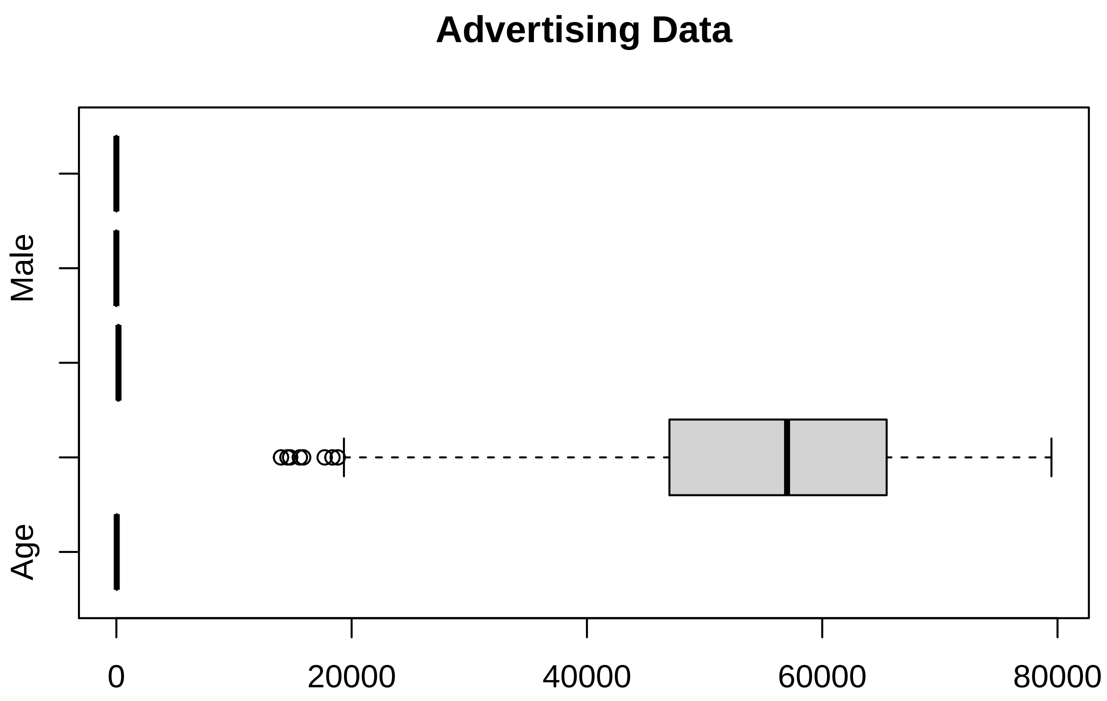
```
## [1] 0
```

```r
# There are no null values in the dataset so no action is required
```

```r
# Checking for outliers in our dataset

# To check for outliers, we only need the numerical columns
# Getting numeric columns from the advert_data
nums <- unlist(lapply(advert_data, is.numeric))

numerical_cols <- advert_data[ ,nums]

head(numerical_cols)
```

```
##   Daily.Time.Spent.on.Site Age Area.Income Daily.Internet.Usage Male
## 1                    68.95  35    61833.90               256.09    0
## 2                    80.23  31    68441.85               193.77    1
## 3                    69.47  26    59785.94               236.50    0
## 4                    74.15  29    54806.18               245.89    1
## 5                    68.37  35    73889.99               225.58    0
## 6                    59.99  23    59761.56               226.74    1
##   Clicked.on.Ad
## 1             0
## 2             0
## 3             0
## 4             0
## 5             0
## 6             0
```

```r
# We can see that we have 6 numeric columns
#

# Plotting the boxplot to visualize the outliers in the dataset

boxplot(numerical_cols[,-1], horizontal=TRUE, main="Advertising Data")
```



```
# Only the Area income column has some outliers of people earning below 20,000
```

4. Exploratory Data Analysis

4.1 Univariate EDA

```
# Getting the mean of the numeric columns

colMeans(numerical_cols)
```

```
## Daily.Time.Spent.on.Site                      Age              Area.Income
##                  65.0002                  36.0090               55000.0001
##     Daily.Internet.Usage                     Male               Clicked.on.Ad
##                 180.0001                   0.4810                   0.5000
```

```
# Creating a function for getting the mode

getmode <- function(v) {
   uniqv <- unique(v)
   uniqv[which.max(tabulate(match(v, uniqv)))]
}

# Getting mode for time spent on site
```

```r
getmode(numerical_cols$Daily.Time.Spent.on.Site)
```

```
## [1] 62.26
```

```r
# Getting mode for age

getmode(numerical_cols$Age)
```

```
## [1] 31
```

```r
# Getting mode for Area Income

getmode(numerical_cols$Area.Income)
```

```
## [1] 61833.9
```

```r
# Getting mode for daily internet usage

getmode(numerical_cols$Daily.Internet.Usage)
```

```
## [1] 167.22
```

```r
# Getting mode of male variable

getmode(numerical_cols$Male)
```

```
## [1] 0
```

```r
# Getting mode for clicked on ad variable

getmode(numerical_cols$Clicked.on.Ad)
```

```
## [1] 0
```

```r
# Finding the median income

median(numerical_cols$Area.Income)
```

```
## [1] 57012.3
```

```r
# Finding median age

median(numerical_cols$Age)
```

```
## [1] 35
```

```r
# Finding median daily internet usage

median(numerical_cols$Daily.Internet.Usage)
```

```
## [1] 183.13
```

```r
# Finding media for time spent on site

median(numerical_cols$Daily.Time.Spent.on.Site)
```

```
## [1] 68.215
```

```r
# Finding min & max area income

min(numerical_cols$Area.Income)
```

```
## [1] 13996.5
```

```r
max(numerical_cols$Area.Income)
```

```
## [1] 79484.8
```

```r
# Finding min & max daily time spent on site

min(numerical_cols$Daily.Time.Spent.on.Site)
```

```
## [1] 32.6
```

```r
max(numerical_cols$Daily.Time.Spent.on.Site)
```

```
## [1] 91.43
```

```r
# Finding min & max daily internet usage

min(numerical_cols$Daily.Internet.Usage)
```

```
## [1] 104.78
```

```r
max(numerical_cols$Daily.Internet.Usage)
```

```
## [1] 269.96
```

```r
# Finding min & max age

min(numerical_cols$Age)
```

```
## [1] 19
```

```r
max(numerical_cols$Age)
```

```
## [1] 61
```

```r
# Getting 1st quantile for age

quantile(numerical_cols$Age, 0.25)
```

```
## 25%
##  29
```

```r
# Getting 2nd quantile for age

quantile(numerical_cols$Age, 0.5)
```

```
## 50%
##  35
```

```r
# Getting 3rd quantile for age

quantile(numerical_cols$Age, 0.75)
```

```
## 75%
##  42
```

```r
# Getting inter-quantile range for age

IQR(numerical_cols$Age)
```

```
## [1] 13
```

```r
# Getting 1st quantile for age

quantile(numerical_cols$Area.Income, 0.25)
```

```
##     25%
## 47031.8
```

```r
# Getting 2nd quantile for age

quantile(numerical_cols$Area.Income, 0.5)
```

```
##     50%
## 57012.3
```

```r
# Getting 3rd quantile for age

quantile(numerical_cols$Area.Income, 0.75)
```

```
##      75%
## 65470.64
```

```r
# Getting inter-quantile range for age

IQR(numerical_cols$Area.Income)
```

```
## [1] 18438.83
```

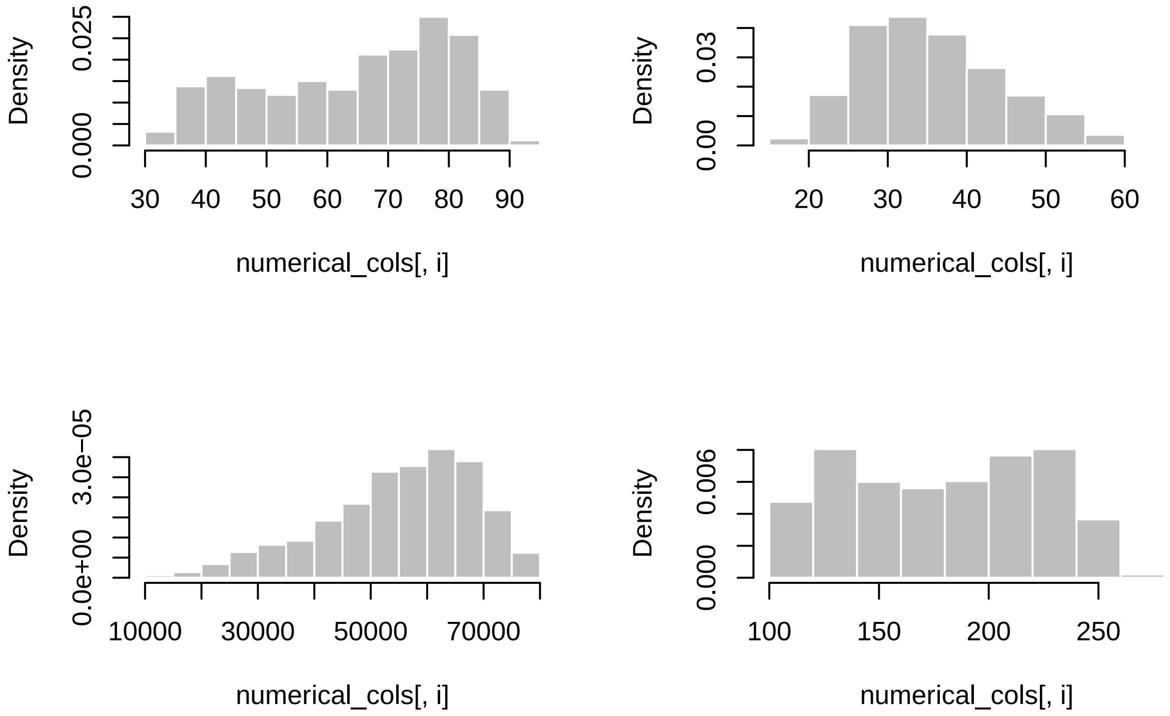
```r
# Finding std deviation

sd(numerical_cols$Area.Income)
```

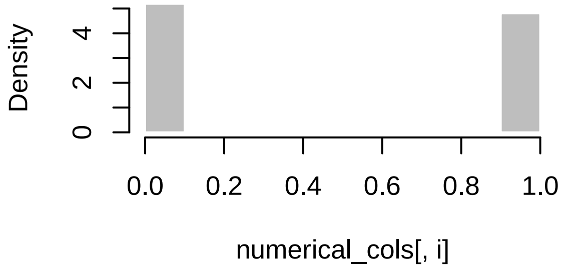
```
## [1] 13414.63
```

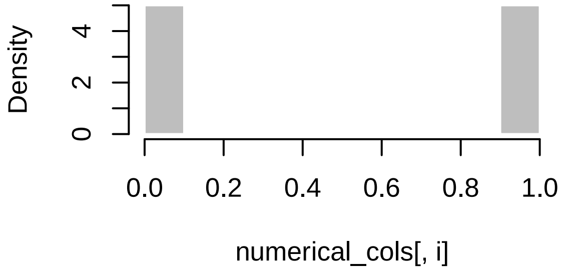
```r
# Getting variance

var(numerical_cols$Area.Income)
```

```
## [1] 179952406
```

```r
# Plotting the histogram for the numerical variables

par(mfrow=c(2, 2))

colnames <- dimnames(numerical_cols)[[2]]
for (i in colnames) {
  hist(numerical_cols[ ,i], main= colnames[i], probability=TRUE, col="gray", border="white")
}
```

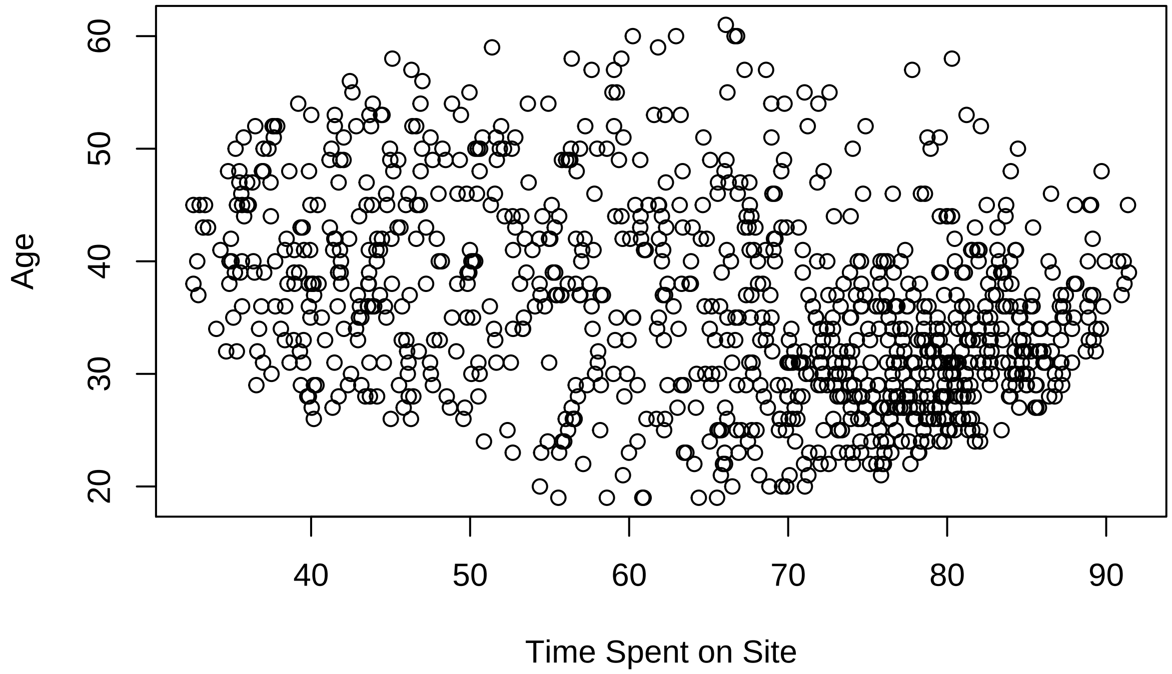




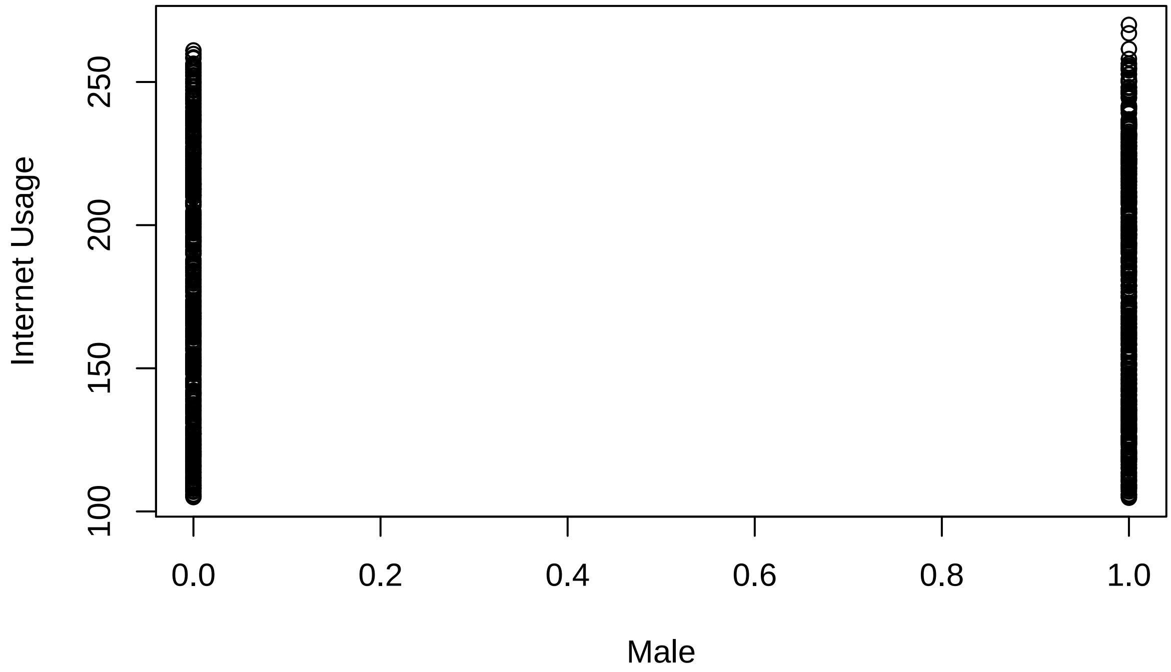


```
# Selecting our columns and assigning variable names to the columns

age <- advert_data$Age

income <- advert_data$Area.Income

male <- advert_data$Male

city <- advert_data$City

time_on_site <- advert_data$Daily.Time.Spent.on.Site

internet_usage <- advert_data$Daily.Internet.Usage

country <- advert_data$Country

clicked_ad <- advert_data$Clicked.on.Ad

topic_line <- advert_data$Ad.Topic.Line

time <- advert_data$Timestamp
```

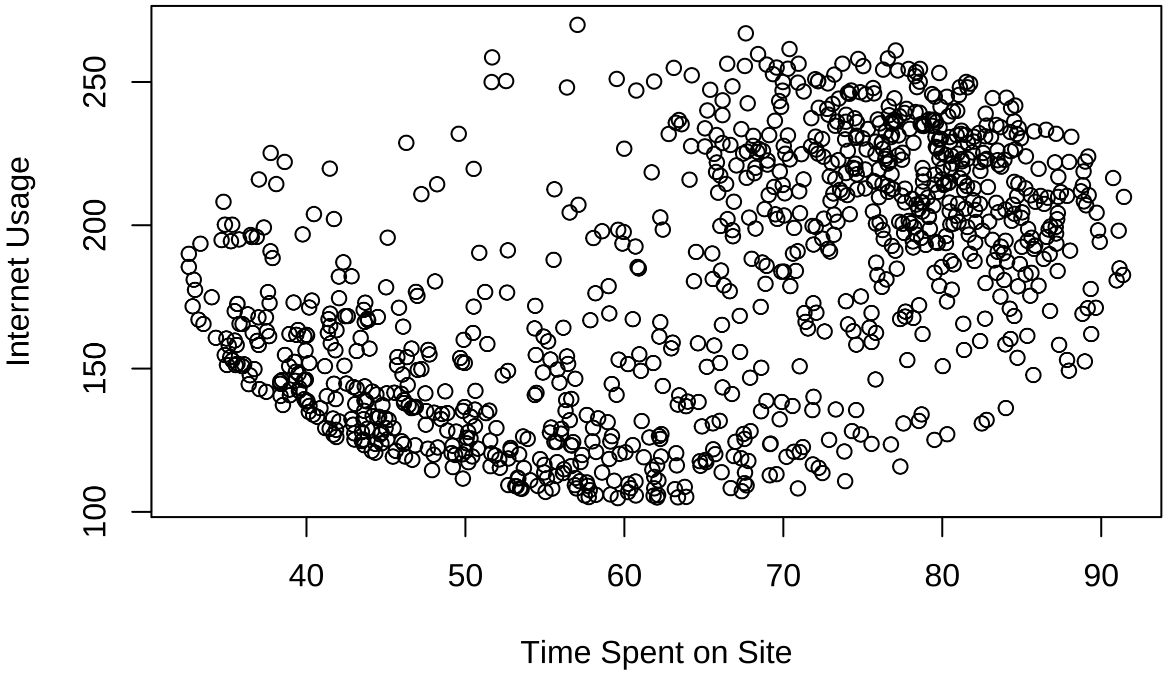
```
# Scatter plot for age against time sment on site

plot(time_on_site, age, xlab = "Time Spent on Site", ylab = "Age")
```



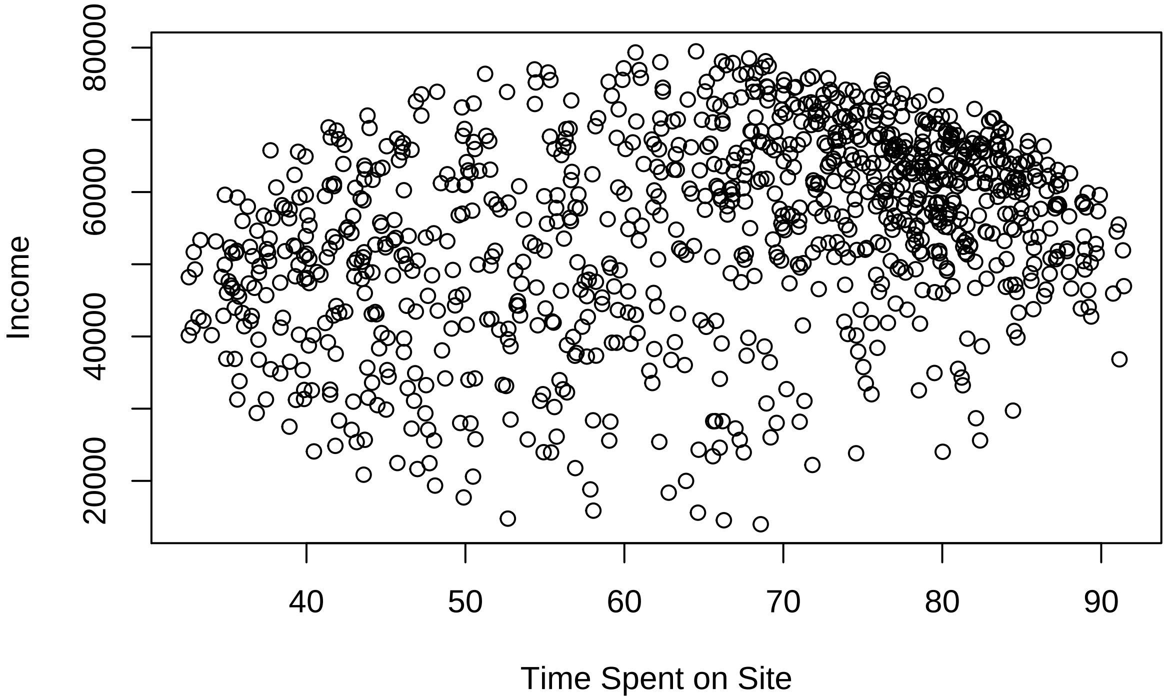
```r
# Scatter plot for internet usage against male variable

plot(male, internet_usage, xlab = "Male", ylab = "Internet Usage")
```



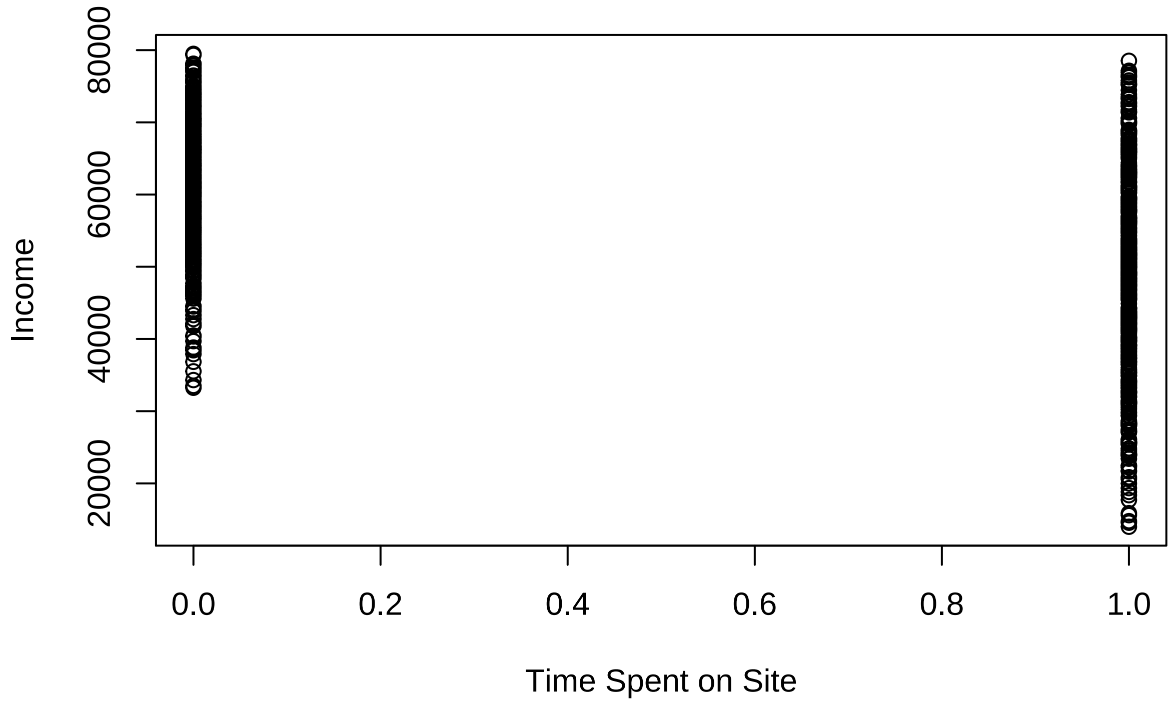
```r
# Scatter plot for internet usage against time spent on site

plot(time_on_site, internet_usage, xlab = "Time Spent on Site", ylab = "Internet Usage")
```



```
# Scatter plot for time income against time spent on site

plot(time_on_site, income, xlab = "Time Spent on Site", ylab = "Income")
```



```
# Scatter plot for income against time spent on site

plot(clicked_ad, income, xlab = "Time Spent on Site", ylab = "Income")
```



```r
# Getting the correlation between our numeric variables

cor(numerical_cols)
```

```
##                          Daily.Time.Spent.on.Site         Age  Area.Income
## Daily.Time.Spent.on.Site               1.00000000 -0.33151334  0.310954413
## Age                                   -0.33151334  1.00000000 -0.182604955
## Area.Income                            0.31095441 -0.18260496  1.000000000
## Daily.Internet.Usage                   0.51865848 -0.36720856  0.337495533
## Male                                  -0.01895085 -0.02104406  0.001322359
## Clicked.on.Ad                         -0.74811656  0.49253127 -0.476254628
##                          Daily.Internet.Usage         Male Clicked.on.Ad
## Daily.Time.Spent.on.Site           0.51865848 -0.018950855   -0.74811656
## Age                               -0.36720856 -0.021044064    0.49253127
## Area.Income                        0.33749553  0.001322359   -0.47625463
## Daily.Internet.Usage               1.00000000  0.028012326   -0.78653918
## Male                               0.02801233  1.000000000   -0.03802747
## Clicked.on.Ad                     -0.78653918 -0.038027466    1.00000000
```

```r
# Getting covariance for our numeric variables

cov(numerical_cols)
```

```
##                          Daily.Time.Spent.on.Site           Age   Area.Income
## Daily.Time.Spent.on.Site              251.3370949 -4.617415e+01  6.613081e+04
## Age                                   -46.1741459  7.718611e+01 -2.152093e+04
## Area.Income                         66130.8109082 -2.152093e+04  1.799524e+08
## Daily.Internet.Usage                  360.9918827 -1.416348e+02  1.987625e+05
## Male                                   -0.1501864 -9.242142e-02  8.867509e+00
## Clicked.on.Ad                          -5.9331431  2.164665e+00 -3.195989e+03
##                          Daily.Internet.Usage        Male Clicked.on.Ad
## Daily.Time.Spent.on.Site         3.609919e+02 -0.15018639 -5.933143e+00
## Age                             -1.416348e+02 -0.09242142  2.164665e+00
## Area.Income                      1.987625e+05  8.86750903 -3.195989e+03
## Daily.Internet.Usage             1.927415e+03  0.61476667 -1.727409e+01
## Male                             6.147667e-01  0.24988889 -9.509510e-03
## Clicked.on.Ad                   -1.727409e+01 -0.00950951  2.502503e-01
```