WEEK_12_IP_R

Ian_Tirok

August 26, 2021

#1. Defining the Question

##a.Specifying the data analytic question Identify which individuals are most likely to click on ads. ##b) Defining the metric of success When we are able to identify the individuals likely to click on ads based on the information and data given. ##c) Understanding the 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 my services as a Data Science Consultant to help her identify which individuals are most likely to click on her ads. ##d) Recording the experimental design The process will entail: * Define the question, the metric for success, the context, experimental design taken. * Read and explore the given dataset. * Perform data cleanin - check for duplicates and missing data * Find and deal with outliers, anomalies, and missing data within the dataset. * Perform univariate and bivariate analysis recording my observations. ##e) Data Relevance The data provided is very relevant and has a wide range of variables to consider in achieving our main objective of this project. [Advertising dataset](http://bit.ly/IPAdvertisingData)

```
install.packages("data.table")
```

```
## Installing package into 'C:/Users/hp/Documents/R/win-library/4.0'
## (as 'lib' is unspecified)
```

```
## package 'data.table' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\hp\AppData\Local\Temp\RtmpAtTR9P\downloaded_packages
```

```
library("data.table")
```

```
#Reading the dataset

df = fread('http://bit.ly/IPAdvertisingData')
```

```
#Lets check the head of the dataframe
head(df)
```

```
##    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
```

```
#Lets check the tail of the dataframe

tail(df, n=6)
```

```
##    Daily Time Spent on Site Age Area Income Daily Internet Usage
## 1:                    43.70  28    63126.96               173.01
## 2:                    72.97  30    71384.57               208.58
## 3:                    51.30  45    67782.17               134.42
## 4:                    51.63  51    42415.72               120.37
## 5:                    55.55  19    41920.79               187.95
## 6:                    45.01  26    29875.80               178.35
##                            Ad Topic Line          City Male
## 1:        Front-line bifurcated ability  Nicholasland    0
## 2:        Fundamental modular algorithm     Duffystad    1
## 3:      Grass-roots cohesive monitoring   New Darlene    1
## 4:         Expanded intangible solution South Jessica    1
## 5: Proactive bandwidth-monitored policy   West Steven    0
## 6:      Virtual 5thgeneration emulation   Ronniemouth    0
##                    Country           Timestamp Clicked on Ad
## 1:                 Mayotte 2016-04-04 03:57:48             1
## 2:                 Lebanon 2016-02-11 21:49:00             1
## 3: Bosnia and Herzegovina 2016-04-22 02:07:01             1
## 4:                Mongolia 2016-02-01 17:24:57             1
## 5:               Guatemala 2016-03-24 02:35:54             0
## 6:                  Brazil 2016-06-03 21:43:21             1
```

```
#Lets find out number of rows in our dataset

nrow(df)
```

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

```
#Lets find out number of collumns in our dataset

ncol(df)
```

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

```
#Lets find out the dimensions of our dataset

dim(df)
```

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

The shape of our dataframe is 1000 rows and 10 columns

```
#Checking the structure of our dataset.

str(df)
```

```
## Classes 'data.table' and '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 ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

DATA CLEANING

```
#Standardizing column names
clean_names <- function(.df, unique = FALSE) {
  n <- if (is.df.frame(.df)) colnames(.df) else .df
  n <- gsub("%+", "_pct_", n)
  n <- gsub("\\$+", "_dollars_", n)
  n <- gsub("\\++", "_plus_", n)
  n <- gsub("-+", "_minus_", n)
  n <- gsub("\\*+", "_star_", n)
  n <- gsub("#+", "_cnt_", n)
  n <- gsub("&+", "_and_", n)
  n <- gsub("@+", "_at_", n)
  n <- gsub("[^a-zA-Z0-9_]+", "_", n)
  n <- gsub("([A-Z][a-z])", "_\\1", n)
  n <- tolower(trimws(n))

  n <- gsub("(^_+|_+$)", "", n)

  n <- gsub("_+", "_", n)
  if (unique) n <- make.unique(n, sep = "_")

  if (is.df.frame(.df)) {
    colnames(.df) <- n
    .df
  } else {
    n
  }
}
```

Dealing with missing data

```
#Lets find out how many missing records are in our data

colSums(is.na(df))
```

```
## Daily Time Spent on Site                      Age              Area Income
##                        0                        0                        0
##     Daily Internet Usage            Ad Topic Line                     City
##                        0                        0                        0
##                     Male                  Country                Timestamp
##                        0                        0                        0
##            Clicked on Ad
##                        0
```

```
#Sum of missing records
sum(is.na(df))
```

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

There seems to be no null records. we can procceed

Lets deal with duplicate records

```
#Checking for any duplicates

duplicated(df)
```

```
##    [1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##   [13] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##   [25] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##   [37] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##   [49] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##   [61] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##   [73] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##   [85] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##   [97] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [109] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [121] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [133] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [145] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [157] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [169] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [181] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [193] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [205] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [217] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [229] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [241] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [253] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [265] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [277] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [289] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [301] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [313] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [325] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [337] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [349] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [361] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [373] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [385] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [397] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [409] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [421] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [433] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [445] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [457] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [469] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [481] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [493] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [505] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [517] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [529] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [541] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [553] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [565] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [577] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [589] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [601] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [613] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [625] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [637] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [649] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [661] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [673] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [685] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [697] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [709] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [721] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [733] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [745] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [757] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [769] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [781] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [793] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [805] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [817] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [829] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [841] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [853] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [865] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [877] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [889] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [901] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [913] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [925] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [937] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [949] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [961] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [973] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [985] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
##  [997] FALSE FALSE FALSE FALSE
```

The above output does not give us clear information as to whether there are any duplicates. lets try to count them

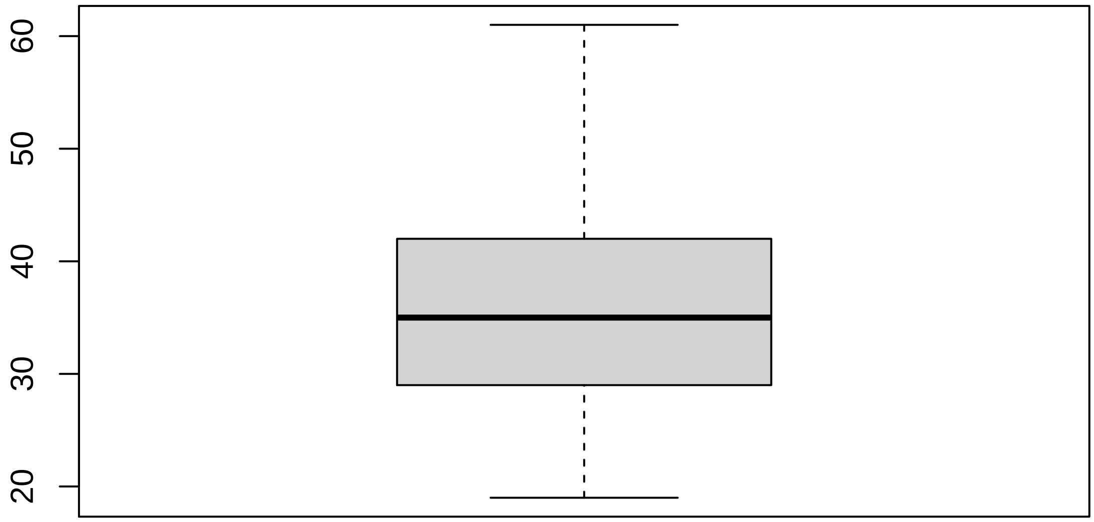
```
#Counting any duplicates

anyDuplicated(df)
```

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

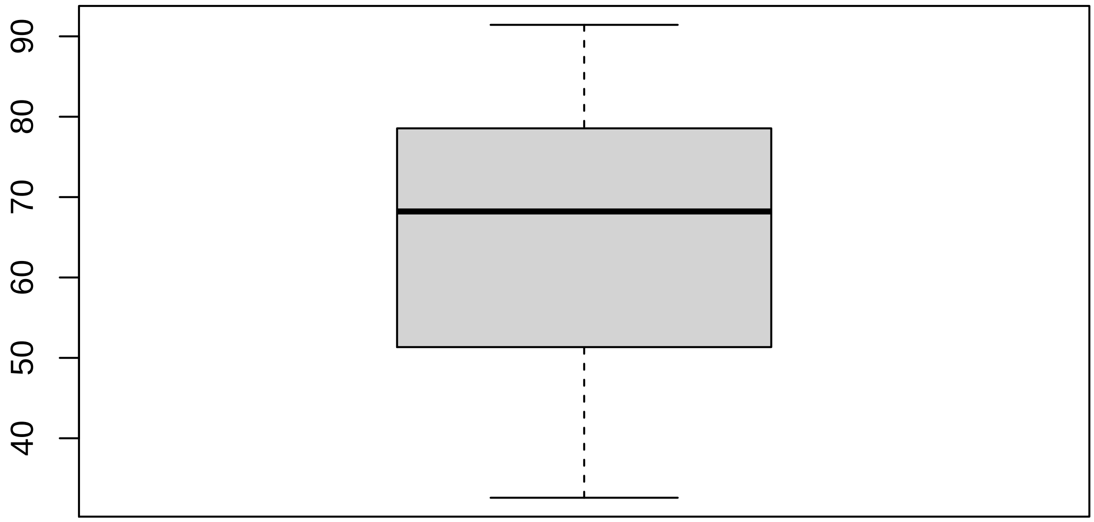
There doesnt seem to be any duplicates

Lets check for outliers

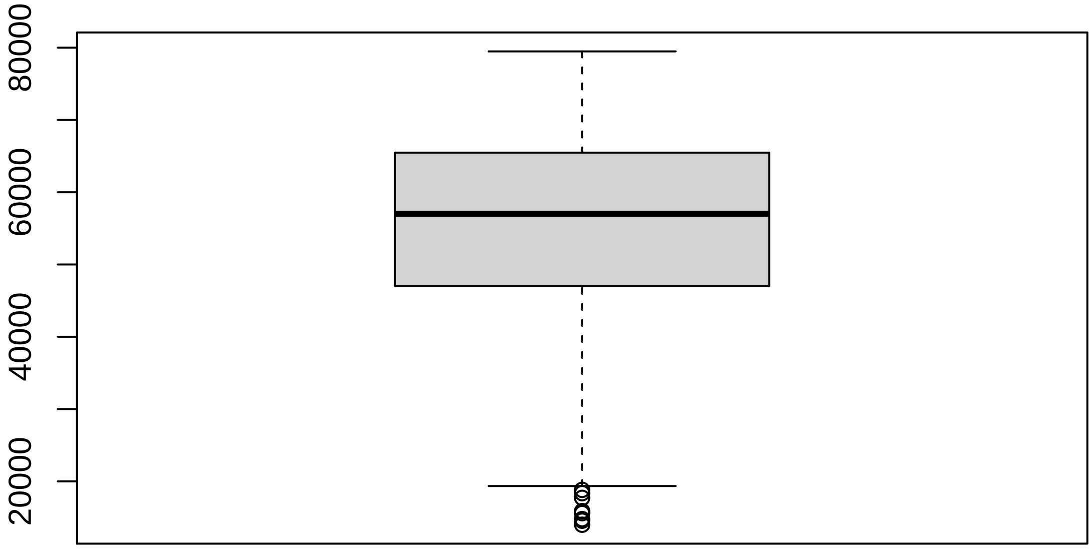
```
#Checking the outliers in the Age Column.
boxplot(df$Age)
```



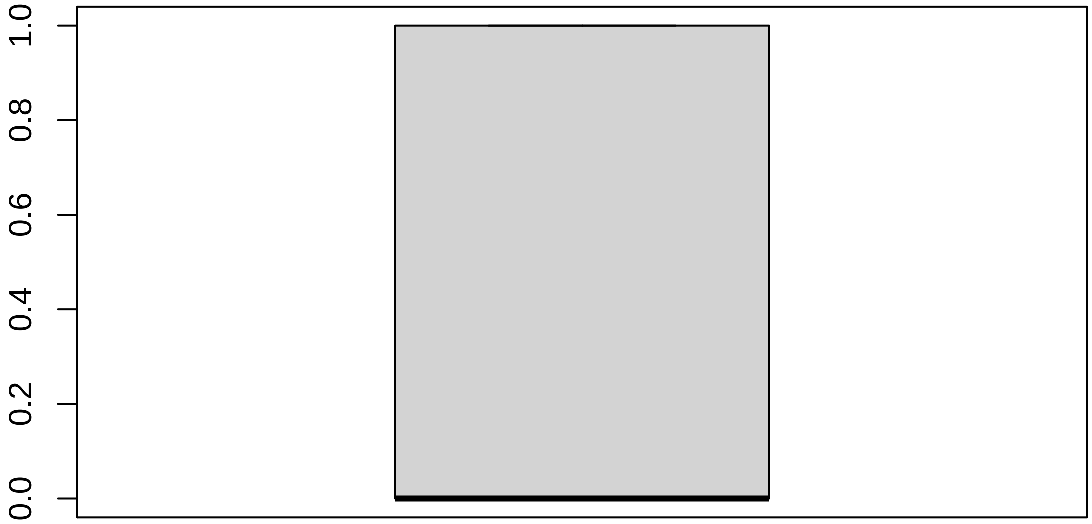
```
#Checking the outliers in the Time spent on site Column.
boxplot(df$'Daily Time Spent on Site')
```



```
#Checking the outliers in the Area income Column.

boxplot(df$'Area Income')
```



```
#Checking the outliers in the Male Column.

boxplot(df$'Male')
```



```
head(df)
```

```
##    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
```

Lets attempt to split the time stamp column to get more information

```
install.packages("dplyr")
```

```
## Installing package into ’C:/Users/hp/Documents/R/win-library/4.0’
## (as ’lib’ is unspecified)
```

```
## package ’dplyr’ successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\hp\AppData\Local\Temp\RtmpAtTR9P\downloaded_packages
```

EXPLAROTARY DATA ANALYSIS

Measures of dispersion

```
dt.mean <- mean(df$'Daily Time Spent on Site')
dt.mean
```

```
## [1] 65.0002
```

```
dt.median <- median(df$'Daily Time Spent on Site')
dt.median
```

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

```
getmode <- function(v) {
   uniqv <- unique(v)
   uniqv[which.max(tabulate(match(v, uniqv)))]
}
dt.mode <- getmode(df$'Daily Time Spent on Site')
dt.mode
```

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

Daily Time Spent on Site Measures of central tendency

Mean - 65.002 Median - 68.215 Mode - 62.26

```
#Checking the mean, median and mode of the age column
age.mean <- mean(df$Age)
age.mean
```

```
## [1] 36.009
```

```
age.median <- median(df$Age)
age.median
```

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

```
getmode <- function(v) {
   uniqv <- unique(v)
   uniqv[which.max(tabulate(match(v, uniqv)))]
}
age.mode <- getmode(df$Age)
age.mode
```

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

Age Measures of central tendency

Mean - 36.009 Median - 35 Mode - 31

```
#Checking the mean, median and mode of the area income column
ai.mean <- mean(df$'Area Income')
ai.mean
```

```
## [1] 55000
```

```
ai.median <- median(df$'Area Income')
ai.median
```

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

```
getmode <- function(v) {
   uniqv <- unique(v)
   uniqv[which.max(tabulate(match(v, uniqv)))]
}
ai.mode <- getmode(df$'Area Income')
ai.mode
```

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

Area income Measures of central tendency

Mean - 55000.00008 Median - 57012.3 Mode -61833.9

```
#Checking the mean, median and mode of the daily internet usage column
diu.mean <- mean(df$'Daily Internet Usage')
diu.mean
```

```
## [1] 180.0001
```

```
diu.median <- median(df$'Daily Internet Usage')
diu.median
```

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

```
getmode <- function(v) {
   uniqv <- unique(v)
   uniqv[which.max(tabulate(match(v, uniqv)))]
}
diu.mode <- getmode(df$'Daily Internet Usage')
diu.mode
```

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

Daily Internet Usage Measures of central tendency

Mean - 180.0001 Median - 183.13 Mode -167.22

```
#Checking the mode of the country column
getmode <- function(v) {
   uniqv <- unique(v)
   uniqv[which.max(tabulate(match(v, uniqv)))]
}
country.mode <- getmode(df$Country)
country.mode
```

```
## [1] "Czech Republic"
```

Czech Republic is the most frequent country.

```
#Checking the mode of the city column
getmode <- function(v) {
   uniqv <- unique(v)
   uniqv[which.max(tabulate(match(v, uniqv)))]
}
city.mode <- getmode(df$City)
city.mode
```

```
## [1] "Lisamouth"
```

Lisamouth is the most frequent city.

```
#Checking the mode of the sex column
getmode <- function(v) {
   uniqv <- unique(v)
   uniqv[which.max(tabulate(match(v, uniqv)))]
}
Male.mode <- getmode(df$Male)
Male.mode
```

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

```
#Checking the mode of the Daily.Internet.Usage column
getmode <- function(v) {
   uniqv <- unique(v)
   uniqv[which.max(tabulate(match(v, uniqv)))]
}
diu.mode <- getmode(df$'Daily Internet Usage')
diu.mode
```

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

Majority of the site visitors had a daily internet usage of 167.22

```
#Checking the mode of the Year column
getmode <- function(v) {
   uniqv <- unique(v)
   uniqv[which.max(tabulate(match(v, uniqv)))]
}
yr.mode <- getmode(df$Timestamp
                   )
yr.mode
```

```
## [1] "2016-03-27 00:53:11"
```

The site had most traffic on 27th March 2016 at 53 minutes past midnight

Measures of dispersion Range

```
#Finding the Range
range(df$'Daily Time Spent on Site')
```

```
## [1] 32.60 91.43
```

```
range(df$'Age')
```

```
## [1] 19 61
```

```
range(df$'Area Income')
```

```
## [1] 13996.5 79484.8
```

```
range(df$'Daily Internet Usage')
```

```
## [1] 104.78 269.96
```

```
range(df$'Male')
```

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

```
range(df$'Clicked on Ad')
```

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

The time spent on the site ranges from 32 minutes to 91 minutes

The range of the age is 19 to 61 years old old

The range of the area income is between 13,996 to 79,848 dollars

The range of daily internet usage is betweem 104 minutes to 269 minutes

The range of gender is 1 because there are only 2 possibilities

The range of whether ad was clicked or not is 1. Because there are only 2 possibilites

Standard Deviation

```
#Finding the Standard Deviation

sd(df$'Daily Time Spent on Site')
```

```
## [1] 15.85361
```

```
sd(df$'Age')
```

```
## [1] 8.785562
```

```
sd(df$'Area Income')
```

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

```
sd(df$'Daily Internet Usage')
```

```
## [1] 43.90234
```

```
sd(df$'Male')
```

```
## [1] 0.4998889
```

```
sd(df$'Clicked on Ad')
```

```
## [1] 0.5002502
```

Standard deviation of Daily time spent on site is 15.85361 Standard deviation of Age is 8.785562 Standard deviation of Area income is 13414.63 Standard deviation of Daily internet usage is 43.90234 Standard deviation of Gender is 0.4998889 Standard deviation of Clicked on Ad is 0.5002502

Calculating Variance

```
#Finding the Standard Deviation

var(df$'Daily Time Spent on Site')
```

```
## [1] 251.3371
```

```
var(df$'Age')
```

```
## [1] 77.18611
```

```
var(df$'Area Income')
```

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

```
var(df$'Daily Internet Usage')
```

```
## [1] 1927.415
```

```
var(df$'Male')
```

```
## [1] 0.2498889
```

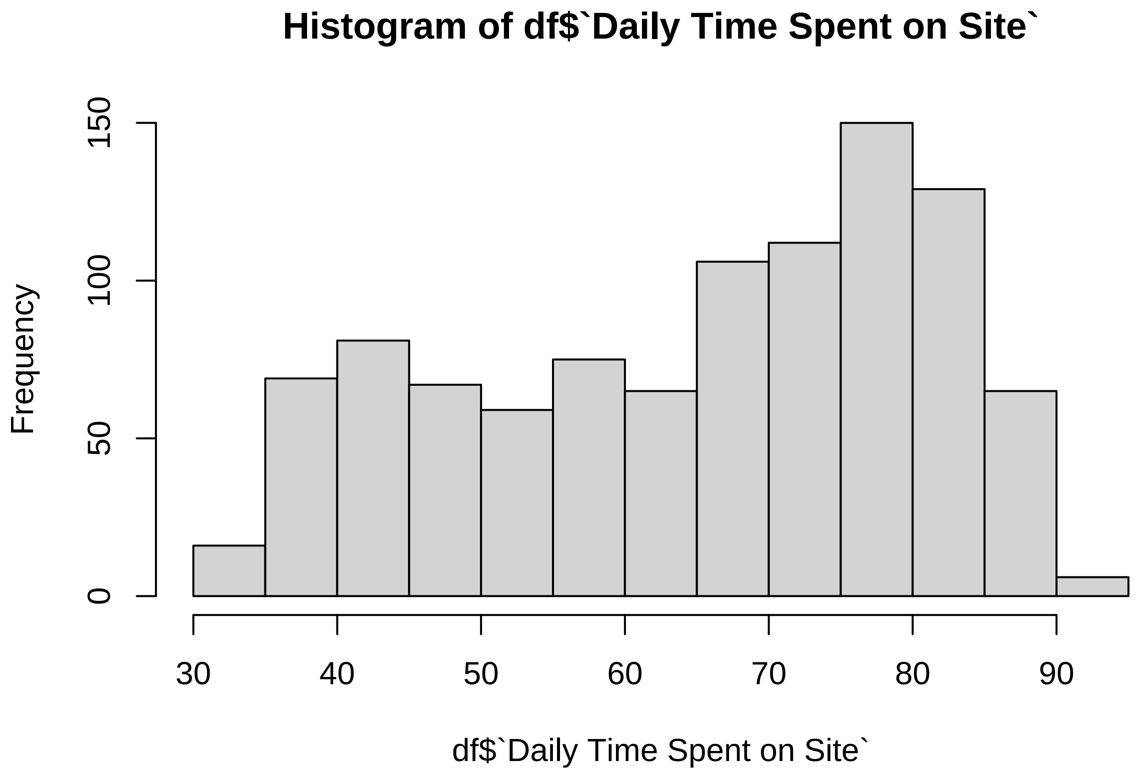
```
var(df$'Clicked on Ad')
```

```
## [1] 0.2502503
```

Variance of Daily time spent on site is 251.3371 Variance of Age is 77.18611 Variance of Area income is 179952406 Variance of Daily internet usage is 1927.415 Variance of Gender is 0.2498889 Variance of Clicked on Ad is 0.2502503

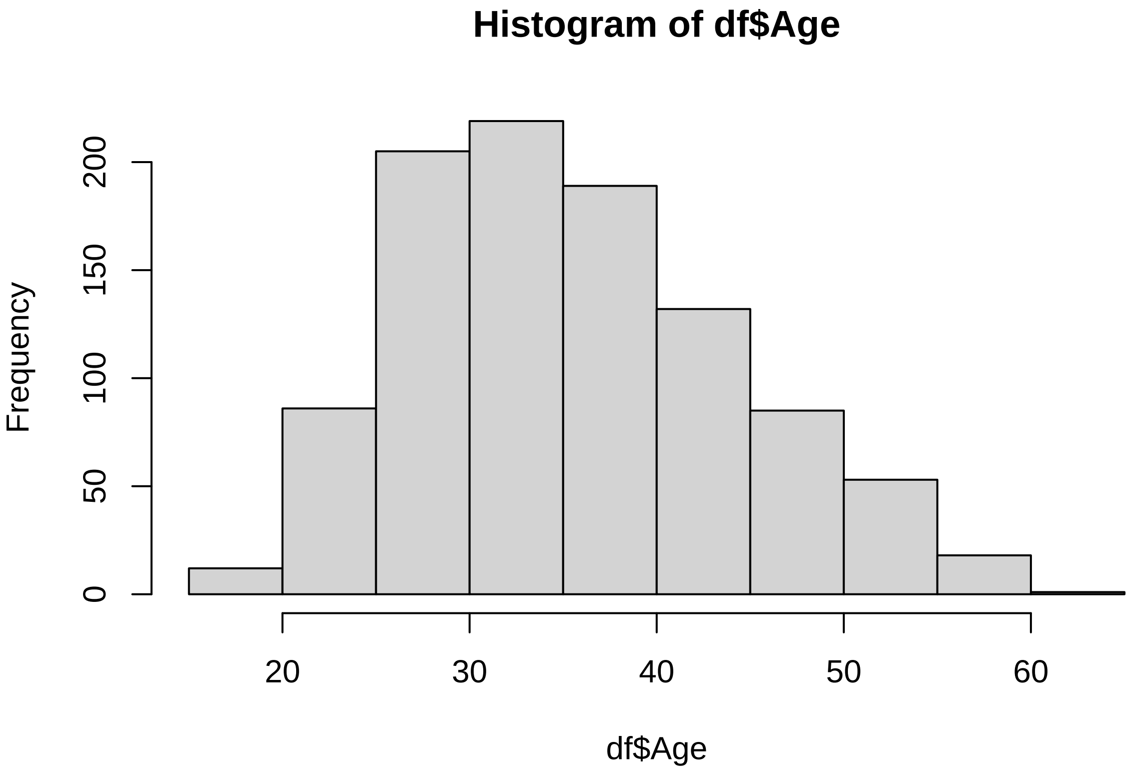
Univariate analysis

```
# Histogram for age
hist(df$`Daily Time Spent on Site`)
```

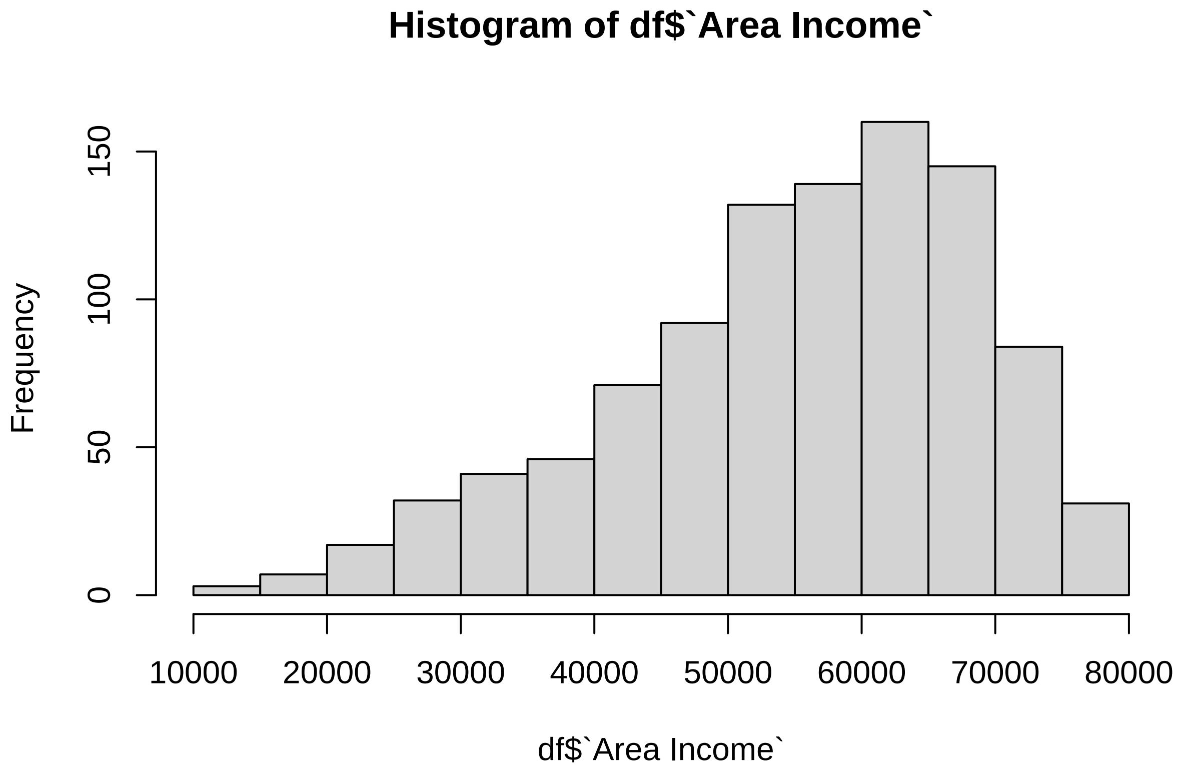


Most people spend about 75 minutes on the website

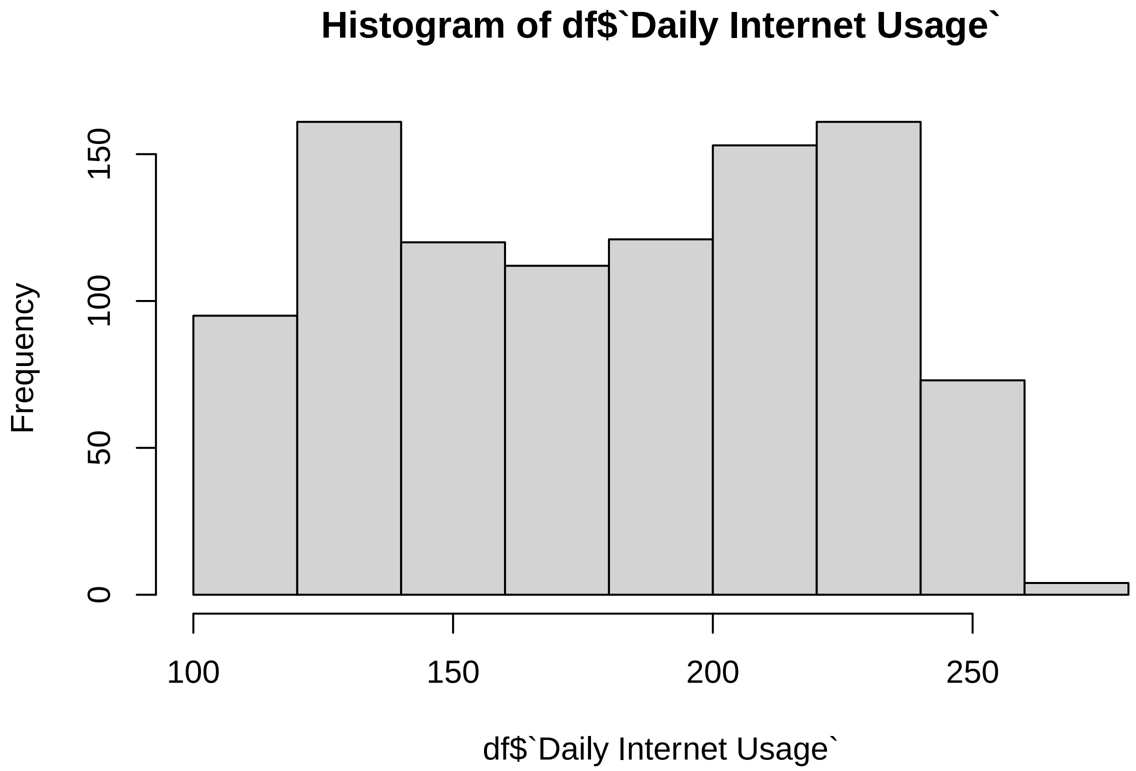
```
# Histogram for age
hist(df$`Age`)
```



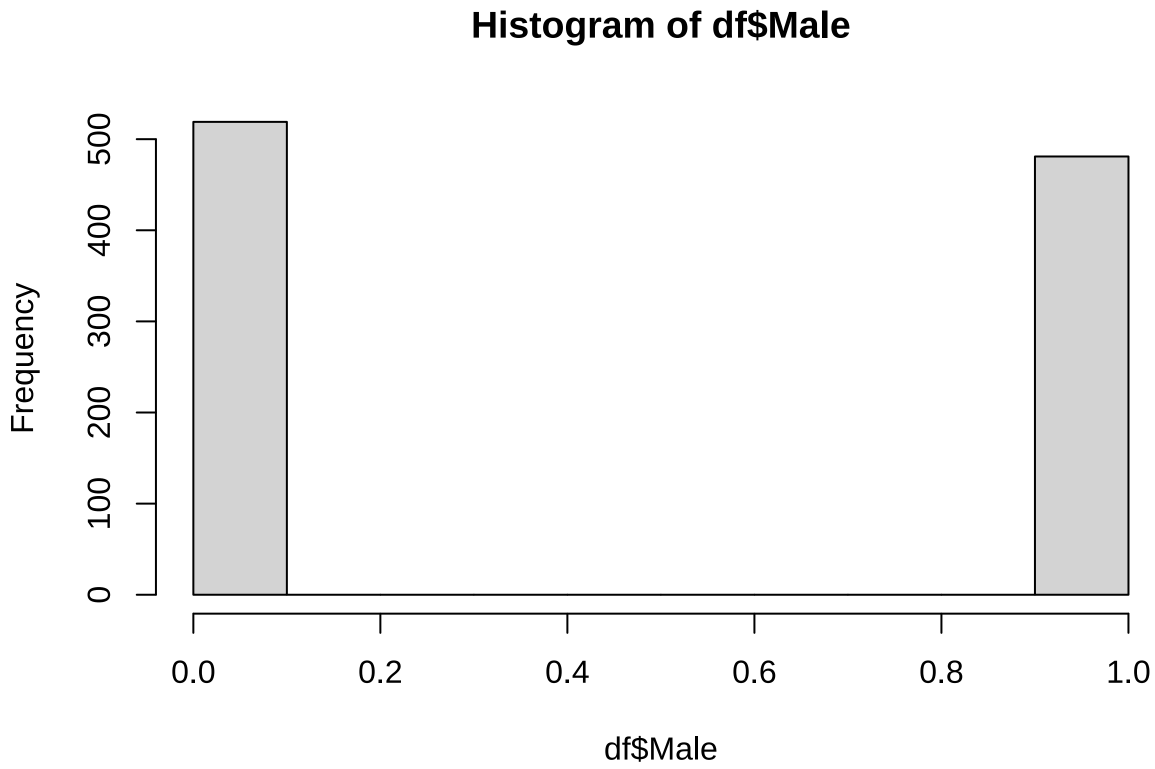
```
# Histogram for area income
hist(df$`Area Income`)
```



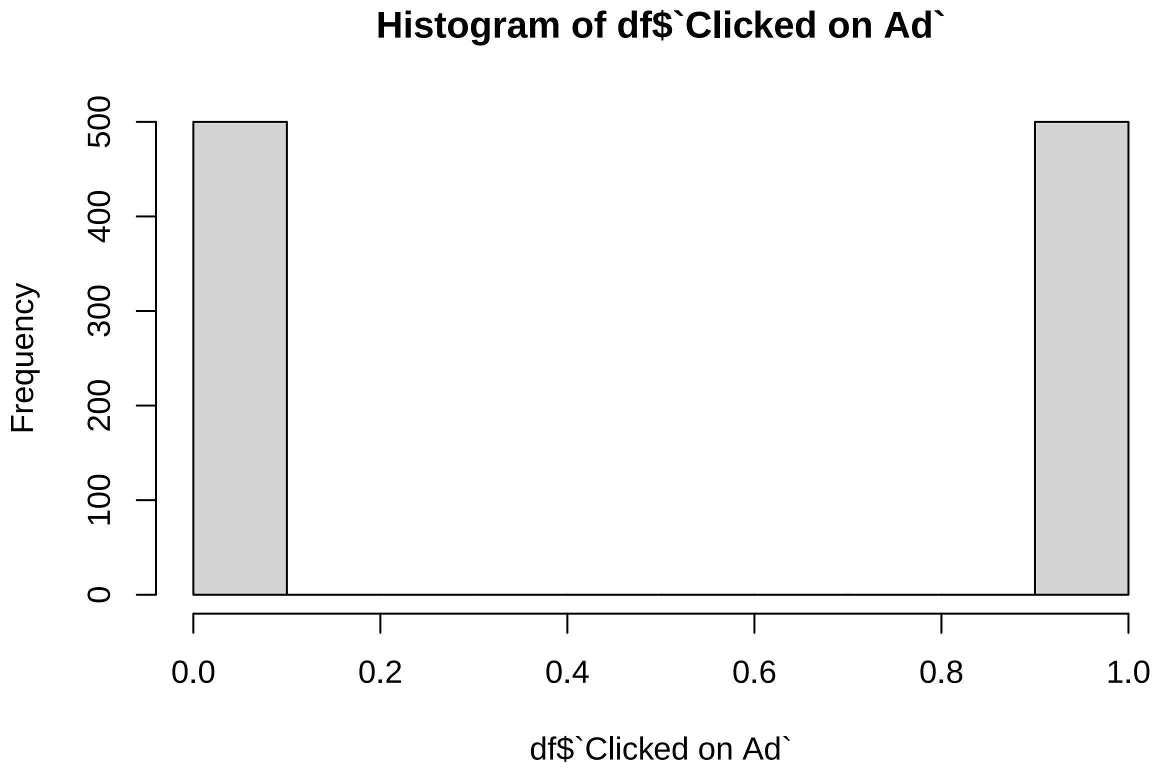
```
# Histogram for area income
hist(df$`Daily Internet Usage`)
```



```
# Histogram on gender distribution
# 1 is male and 0 is female
hist(df$`Male`)
```



```
# Histogram on gender distribution
# 1 is clicked on ad and 0 represents people that did not click on the ad
hist(df$`Clicked on Ad`)
```



BIVARIATE ANALYSIS **Income and Click on Ad distribution**

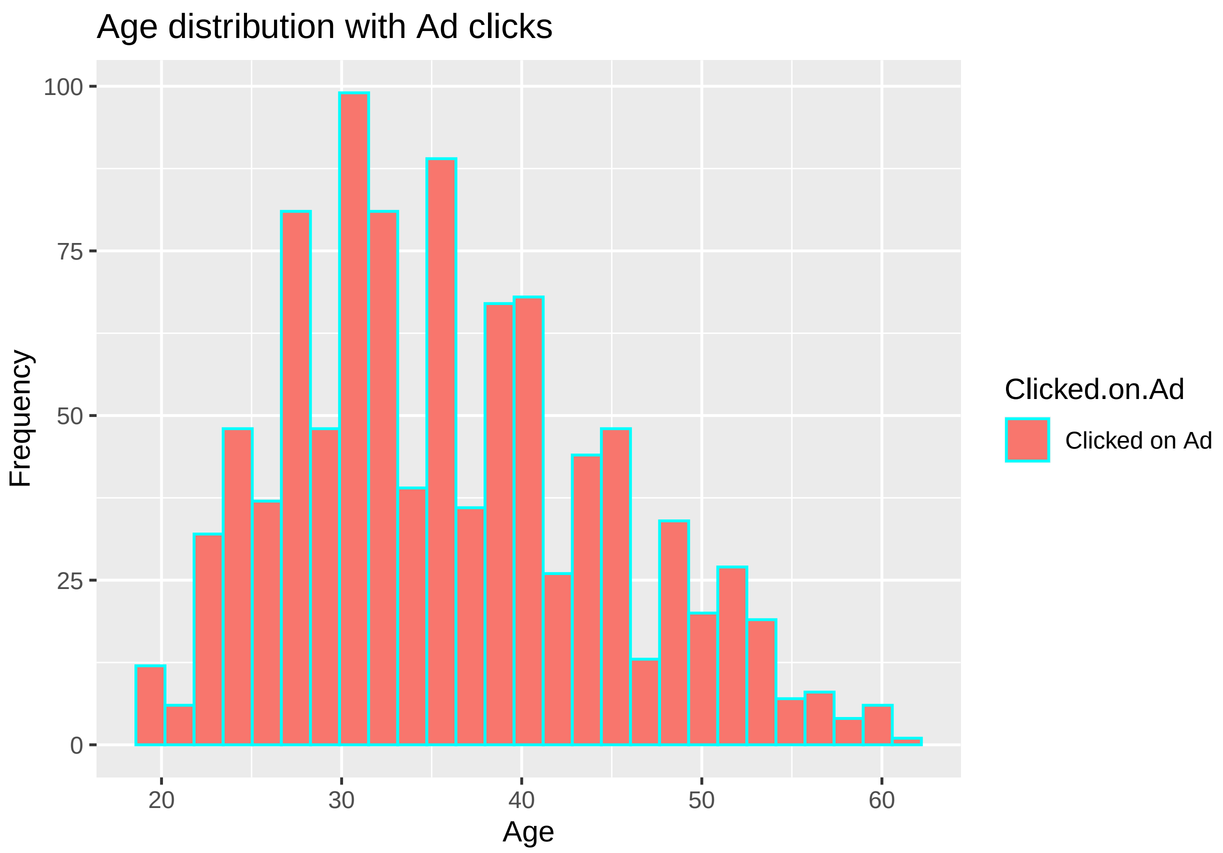
```
install.packages("ggplot2")
```

```
## Installing package into 'C:/Users/hp/Documents/R/win-library/4.0'
## (as 'lib' is unspecified)
```

```
## package 'ggplot2' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\hp\AppData\Local\Temp\RtmpAtTR9P\downloaded_packages
```

```
library("ggplot2")
```

```
library("ggplot2")
ggplot(data = df, aes(x = Age, fill = 'Clicked on Ad'))+
    geom_histogram(bins = 27, color = 'cyan') +
    labs(title = 'Age distribution with Ad clicks', x = 'Age', y = 'Frequency', fill = 'Clicked.on.Ad')
        scale_color_brewer(palette = 'Set2')
```



```
###Bivariate analysis
#Covariance of age and click on ad
age <- df$Age
click <- df$'Clicked on Ad'
gender <- df$Male
cov(age, click)
```

```
## [1] 2.164665
```

The covariance of age to click is positive meaning they're positively related

```
income <- df$'Area Income'
click <- df$'Clicked on Ad'
cov(income, click)
```

```
## [1] -3195.989
```

As income increases, the click on ad reduces - a negative covariance

```
#Using correlation coefficient
cor(age, click)
```

```
## [1] 0.4925313
```

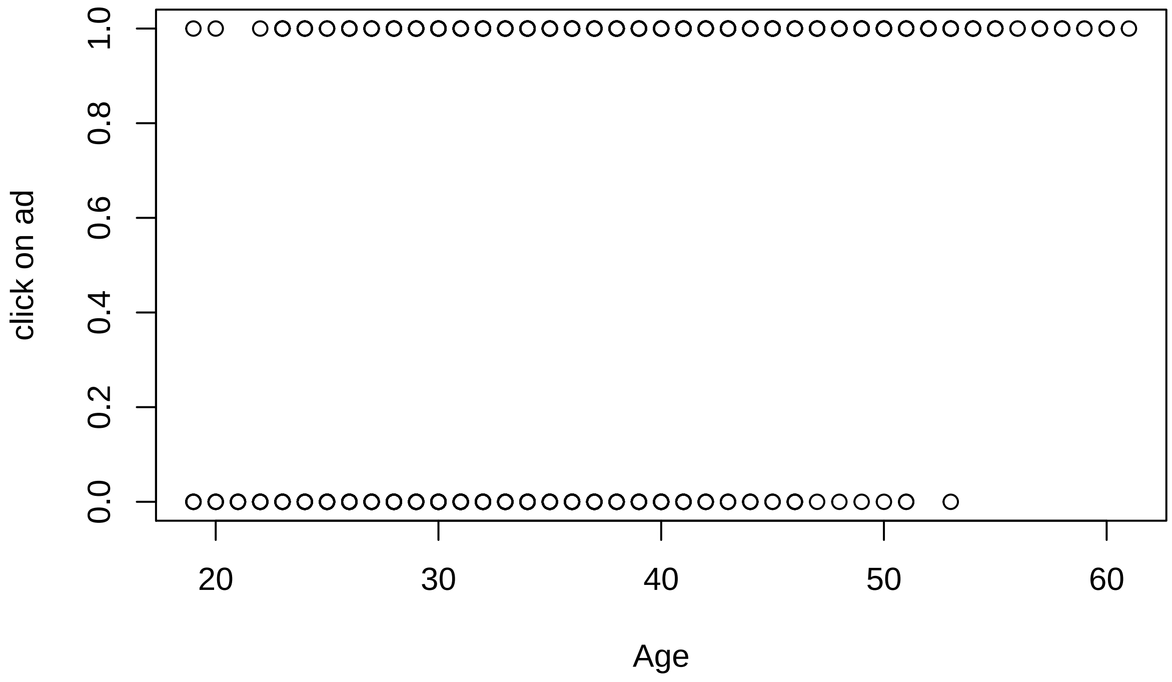
There is a medium positive correlation between age and click on ad

```
cor(income, click)
```

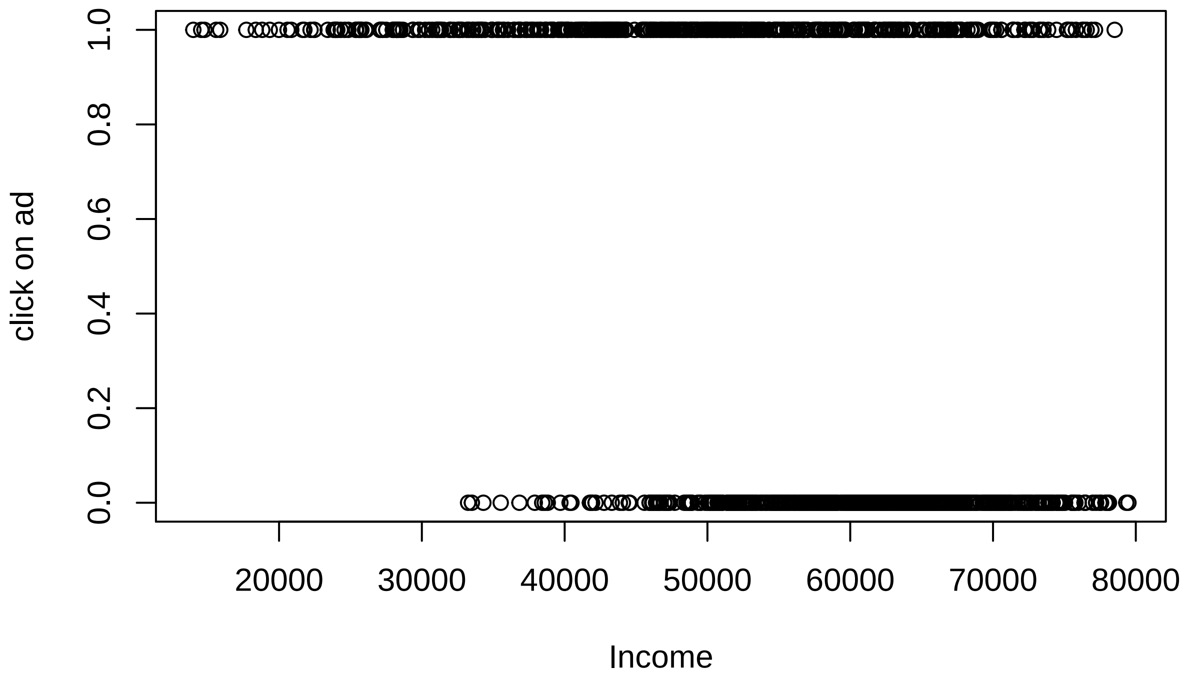
```
## [1] -0.4762546
```

There is a medium negative correlation between age and click on ad

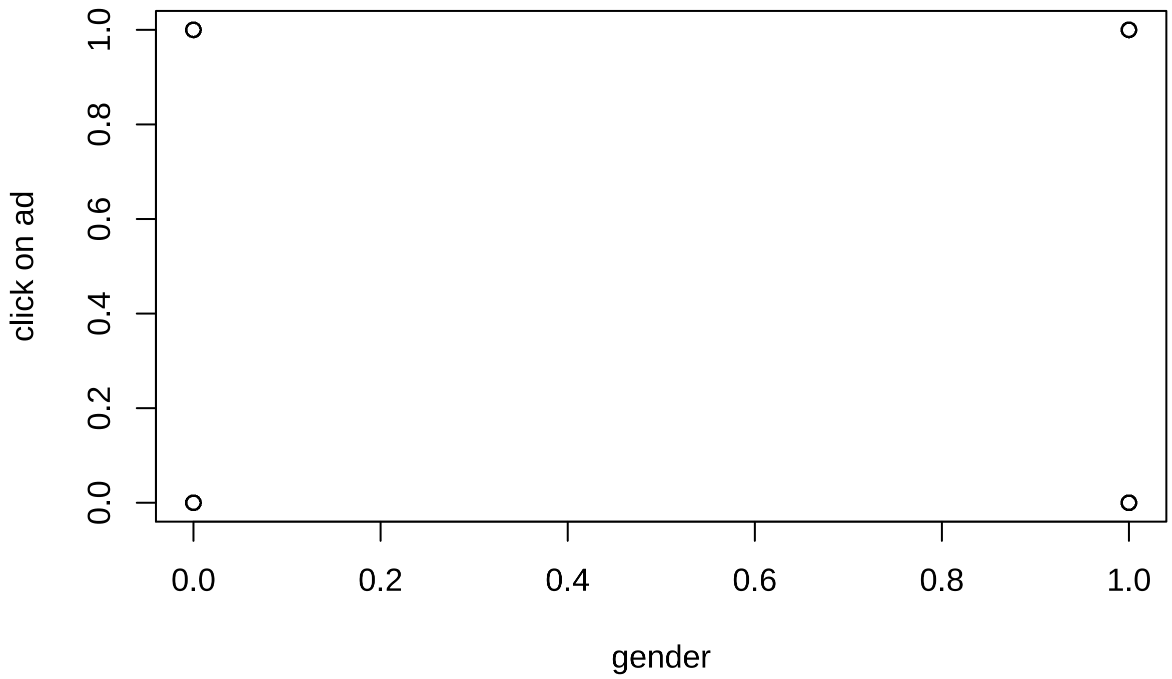
```
#
# Creating a scatter plot
plot(age, click, xlab="Age", ylab="click on ad")
```



```
plot(income, click, xlab="Income", ylab="click on ad")
```

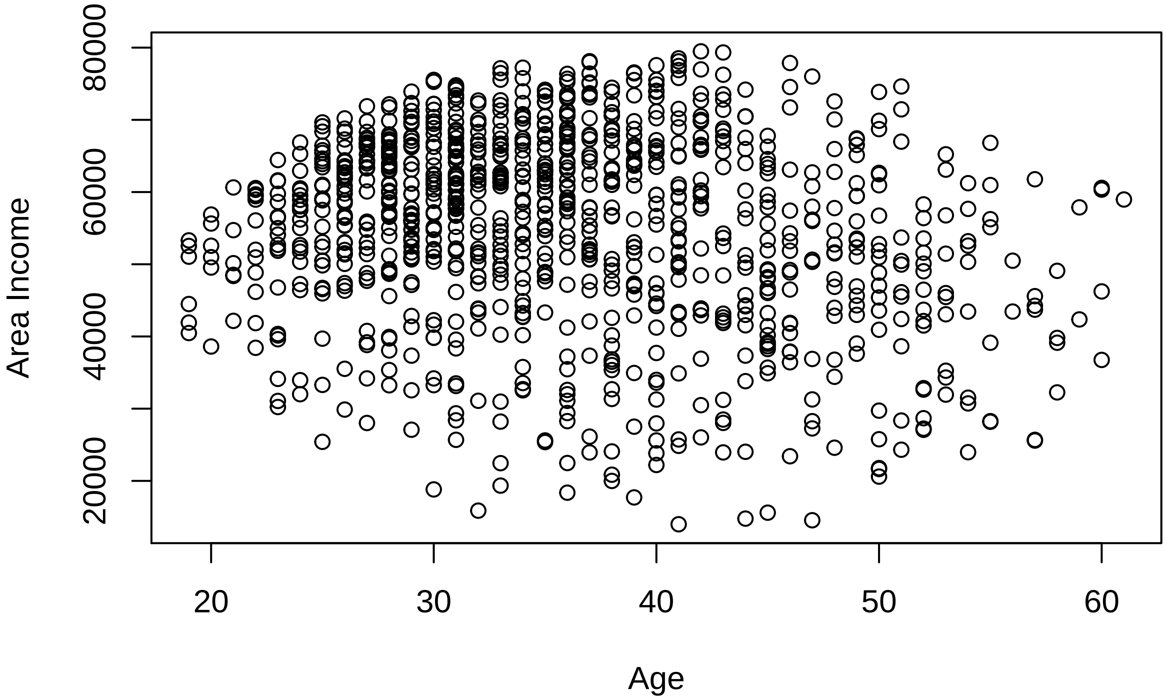


```
plot(gender, click, xlab="gender", ylab="click on ad")
```



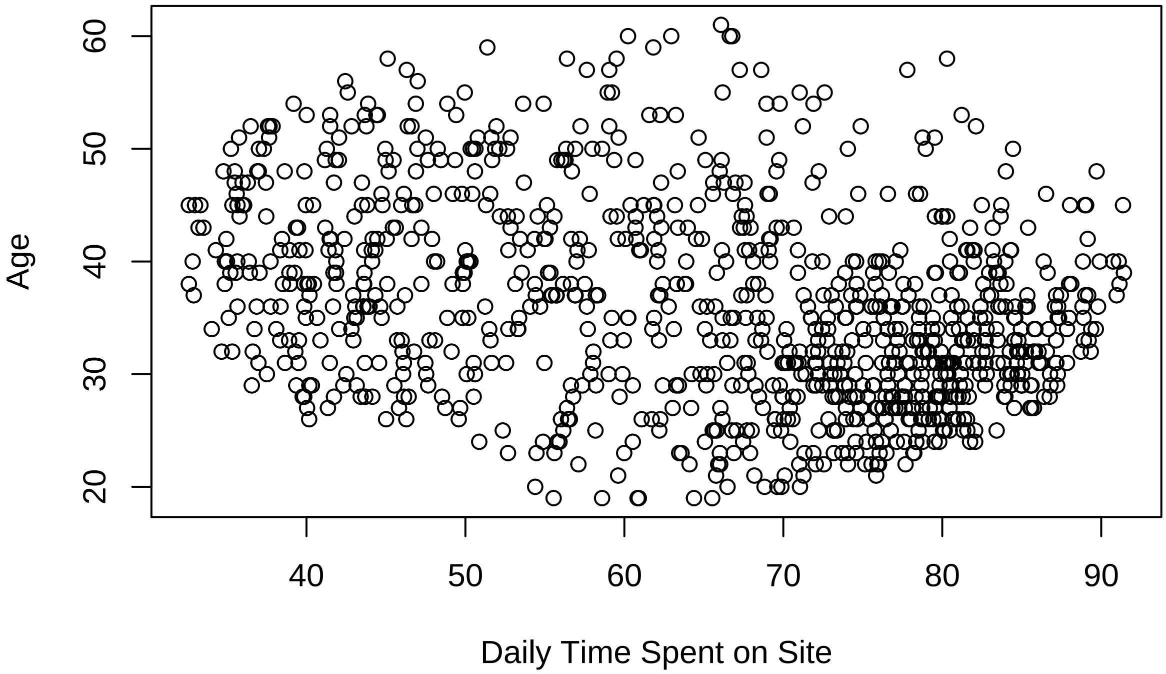
```
# Both plots reveal positive correlation
```

```
#A scatter plot of Age vs Area Income.
plot(df$Age, df$'Area Income', xlab="Age", ylab="Area Income")
```



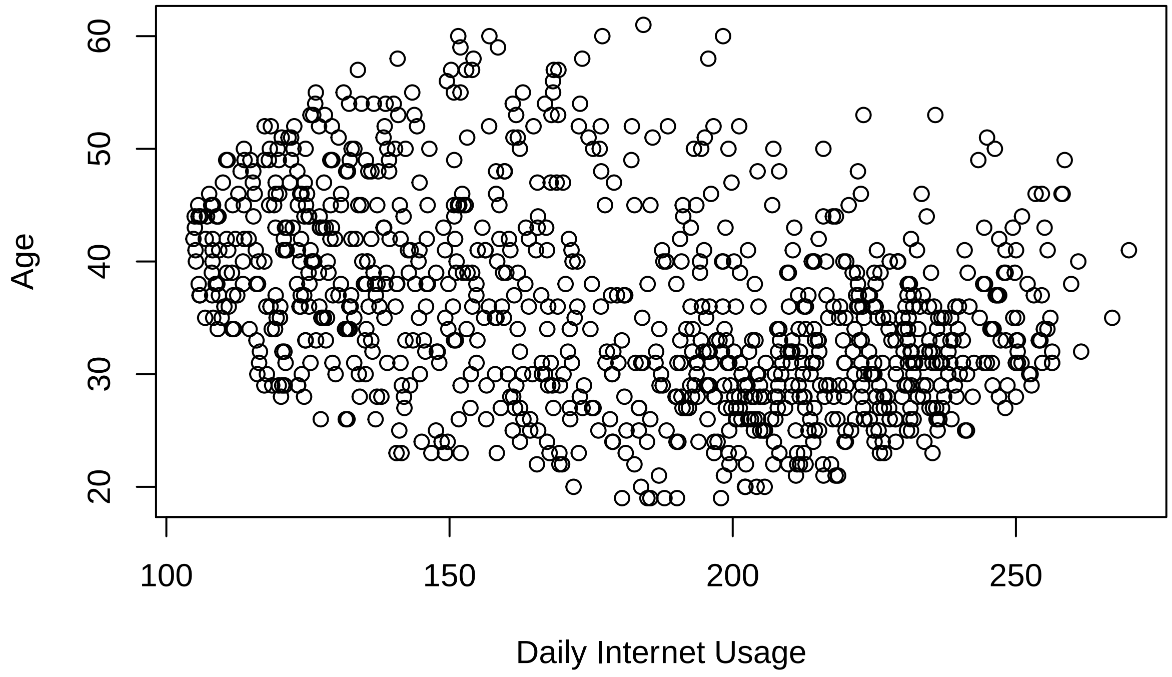
There seems to be no correlation between the two variables.

```
#A scatter plot of Age vs Daily Time Spent on Site.
plot(df$'Daily Time Spent on Site', df$Age, xlab="Daily Time Spent on Site", ylab="Age")
```



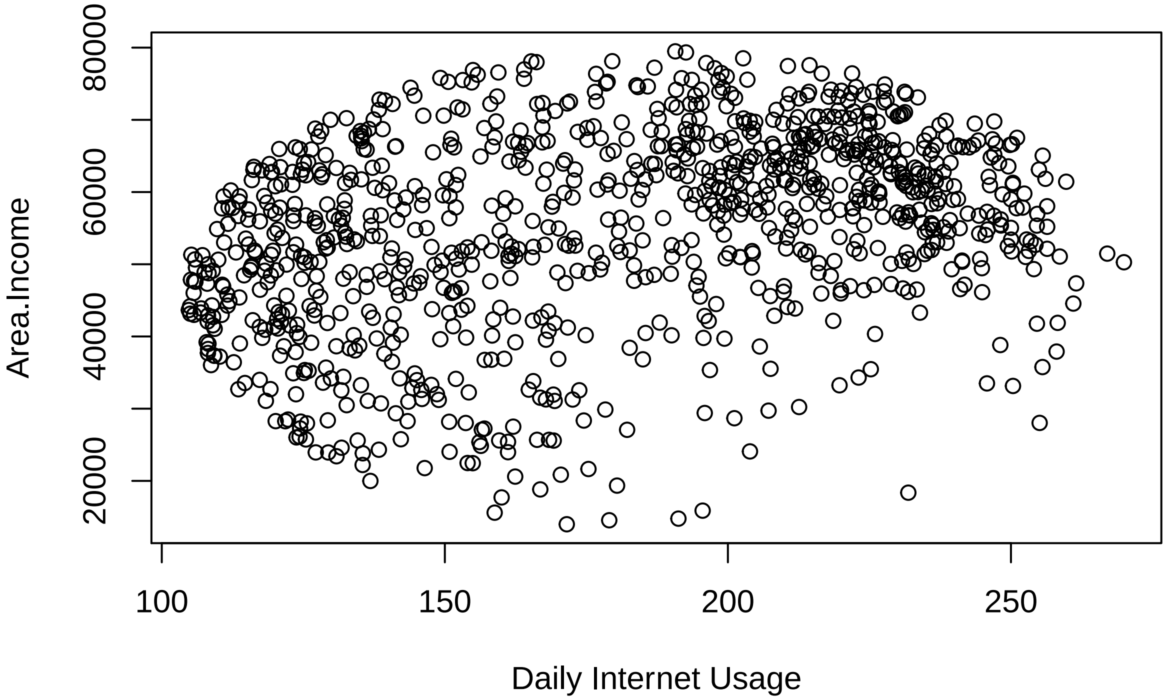
The younger people seem to spend more time on the website than older people

```
#A scatter plot of Age vs Daily.Internet.Usage.
plot(df$'Daily Internet Usage', df$Age, xlab="Daily Internet Usage", ylab="Age")
```



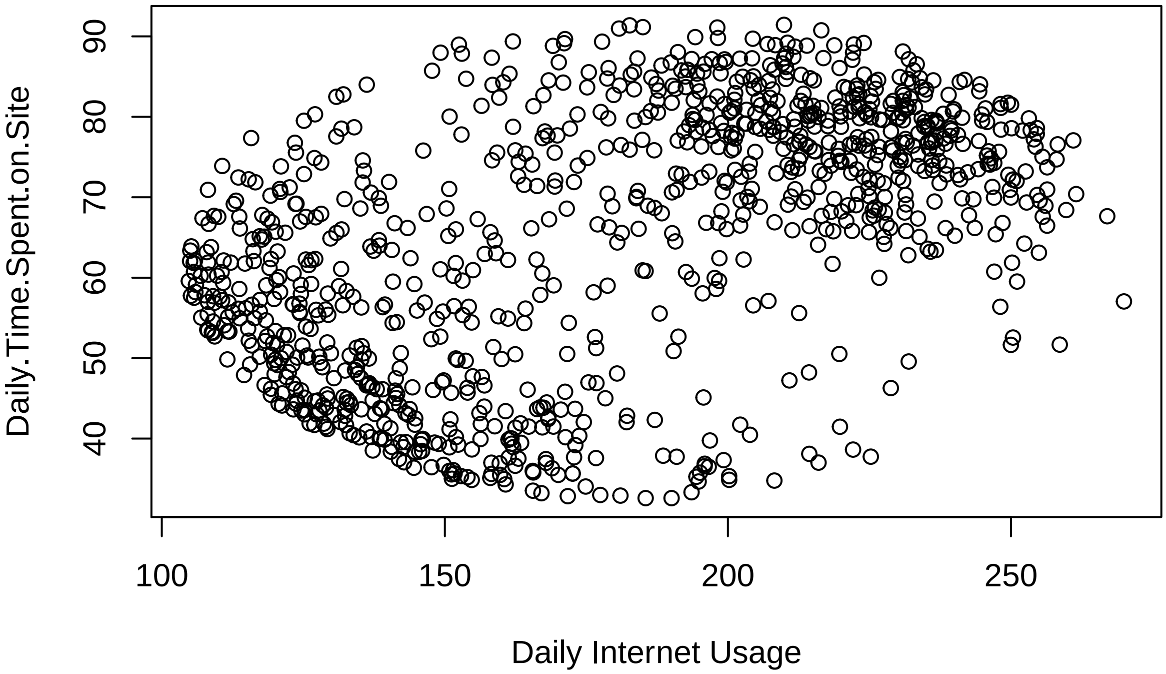
The younger people seem to spend more time on the internet

```
#A scatter plot of Area.Income vs Daily.Internet.Usage.
plot(df$'Daily Internet Usage', df$'Area Income', xlab="Daily Internet Usage", ylab="Area.Income")
```



People with higher income seem to spend more time on the internet

```
#A scatter plot of Daily.Internet.Usage vs Daily Time Spent on Site.
plot(df$'Daily Internet Usage', df$'Daily Time Spent on Site', xlab="Daily Internet Usage", ylab="Daily
```



There does not seem to be a clear correlation between the 2 variables

```
tinytex::install_tinytex()
```

```
## Warning: Detected an existing tlmgr at C:
## \Users\hp\AppData\Roaming\TinyTeX\bin\win32\tlmgr.bat. It seems TeX Live has
## been installed (check tinytex::tinytex_root()). You are recommended to uninstall
## it, although TinyTeX should work well alongside another LaTeX distribution if a
## LaTeX document is compiled through tinytex::latexmk().

## Warning in download.file(url, output, ..., method = method): URL 'https://
## yihui.org/gh/tinytex/tools/pkgs-custom.txt': status was 'SSL connect error'

## Warning in download.file(url, output, ..., method = method): URL 'https://
## yihui.org/gh/tinytex/tools/tinytex.profile': status was 'SSL connect error'

## Starting to install TinyTeX to C:\Users\hp\AppData\Roaming/TinyTeX. It will take a few minutes.

## Next you may see two error dialog boxes about the missing luatex.dll, and an error message like "Use

## Please quit and reopen your R session and IDE (if you are using one, such as RStudio or Emacs) and cl
```

```
tinytex:::is_tinytex()
```

```
## [1] TRUE
```

CONCLUSION

1. Gender - Majority of the website visitors are female
2. Age - Majority of the website visitors are between the ages 30 and 40 years
3. City - Lisamouth is the city where most traffic originates from
4. Country - Czech republic is the country where most traffic originates from
5. Date - The website had most visitors on 27th March 2016 at 53 minutes past midnight
6. Most people spend about 75 minutes on the website

Recomendations

1. Make more targeted ads for women
2. Make more targeted ads for people between the ages of 30 - 40 years
3. Expand target and try to capture markets other than Lisamouth city
4. Expand targeted ads and try to capture other countres apart from Czech Republic
5. Find out why the date 27th March 2016 had most traffic
6. Find out whether users are comfortable speding upto 75 minutes on the website. And why they spend this much time