Business Understanding

My-Duka is an online shop that recently launched their services. As a new company, they would like to use effective and strategic marketing techniques to reach their clientelle.

Specifying the analytic Question

My-duka would like to understand which customers are highly likely to click on an add ontheir site and vice-versa.

Define the Metric for Success

Thorough Data Cleaning Perform Univariate analysis Perform Bivariate Analysis

Experimental design

Data Understanding Univariate Analysis Bivariate Analysis Plotting the summaries Conclusion

```
output:
  pdf_document: default
---

title: "Data Cleaning with R"
author: "Vivian Njau"
date: "2/26/2020"
output: pdf_document
```

R Markdown

Data Cleaning

```
#specify the path where the file is located
library("data.table")
```

obtaining the path to the working directrory

```
getwd()

## [1] "C:/Users/hp/Documents"
```

Loading the datasets

```
library("readr")
df <- read.csv("advertising.csv")
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
```

Previewing the top of the dataset

```
advert_df <- data.frame(df)
head(advert_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
```

Previewing the summary of the dataset

```
summary(advert_df)
```

```
##  Daily.Time.Spent.on.Site      Age         Area.Income
Daily.Internet.Usage
##  Min.   :32.60            Min.   :19.00   Min.   :13996   Min.   :104.8
##  1st Qu.:51.36            1st Qu.:29.00   1st Qu.:47032   1st Qu.:138.8
##  Median :68.22            Median :35.00   Median :57012   Median :183.1
##  Mean   :65.00            Mean   :36.01   Mean   :55000   Mean   :180.0
##  3rd Qu.:78.55            3rd Qu.:42.00   3rd Qu.:65471   3rd Qu.:218.8
##  Max.   :91.43            Max.   :61.00   Max.   :79485   Max.   :270.0
##
##                                  Ad.Topic.Line              City
##  Adaptive 24hour Graphic Interface      :  1   Lisamouth      :  3
##  Adaptive asynchronous attitude         :  1   Williamsport   :  3
##  Adaptive context-sensitive application :  1   Benjaminchester:  2
##  Adaptive contextually-based methodology:  1   East John      :  2
##  Adaptive demand-driven knowledgebase   :  1   East Timothy   :  2
##  Adaptive uniform capability            :  1   Johnstad       :  2
##  (Other)                                :994   (Other)        :986
##       Male                 Country                   Timestamp
Clicked.on.Ad
##  Min.   :0.000   Czech Republic:  9   2016-01-01 02:52:10:  1   Min.
:0.0
##  1st Qu.:0.000   France        :  9   2016-01-01 03:35:35:  1   1st
Qu.:0.0
##  Median :0.000   Afghanistan   :  8   2016-01-01 05:31:22:  1   Median
:0.5
##  Mean   :0.481   Australia     :  8   2016-01-01 08:27:06:  1   Mean
:0.5
##  3rd Qu.:1.000   Cyprus        :  8   2016-01-01 15:14:24:  1   3rd
Qu.:1.0
##  Max.   :1.000   Greece        :  8   2016-01-01 20:17:49:  1   Max.
:1.0
##                  (Other)       :950   (Other)            :994
```

Properties of the dataset

Length

```
length(advert_df)

## [1] 10

#The dataframe has 1000 entries
```

Dimensions

```
dim(advert_df)

## [1] 1000   10

#The dataframe has 1000 row entries and 10 columns
```

Column Names

```
colnames(advert_df)
```

```
##  [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"
```

```
#The ten column names are:
```

Column data types

```
sapply(advert_df, class)
```

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

Data Cleaning

Missing values

```
#Checking the sum of missing values per column
colSums(is.na(advert_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
```

```
#there are no misssing values in the data
```

Duplicates

```
duplicated_rows <- advert_df[duplicated(advert_df),]
duplicated_rows
```

```
##  [1] Daily.Time.Spent.on.Site Age                      Area.Income
##  [4] Daily.Internet.Usage     Ad.Topic.Line            City
##  [7] Male                     Country                  Timestamp
## [10] Clicked.on.Ad
## <0 rows> (or 0-length row.names)
```

```
#there are no duplicate entries in the data
```

Assigning the appropriate datatypes for each column

Changing the timestamp datatype from factor to date_time

```
#changing the timestamp datatype from factor to date_time
advert_df$Timestamp <- as.Date(advert_df$Timestamp, format = "%Y-%m-%s-%h-%m-%s")

#checking the new datatype for the Timestamp column
sapply(advert_df, class)
```

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

Univarite analysis

Daily.Time.Spent.on.Site

```
#This column represents the amount of time that a user spends on the website
# measures of central tendency

# mean
mean(advert_df$Daily.Time.Spent.on.Site)
```

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

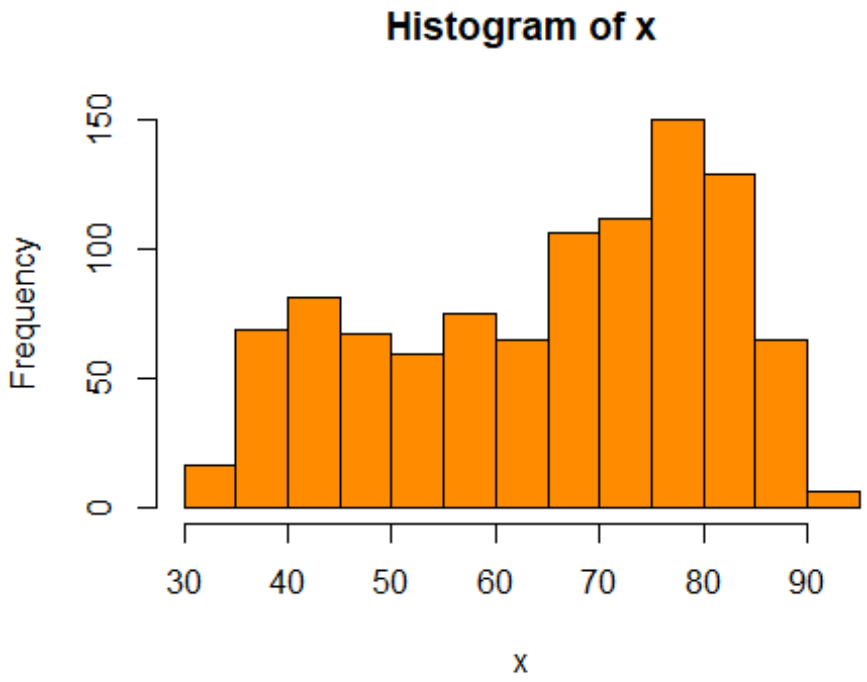
```
# median
median(advert_df$Daily.Time.Spent.on.Site)
```

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

```
# mode
x <- advert_df$Daily.Time.Spent.on.Site
#sort(x)
names(table(x))[table(x)==max(table(x))]
```

```
## [1] "62.26" "75.55" "77.05" "78.76" "84.53"
```

```
#each of the values printed below appear thrice in the dataset

#distribution
hist(x, col=c("darkorange"))
```



The users spend an average 65.002 minutes on the website.

The modal time is “62.26” “75.55” “77.05” “78.76” “84.53”

The median time is 68.215.

The distribution above is left-skewed.

The highest frequency is 80 units of time(minutes).

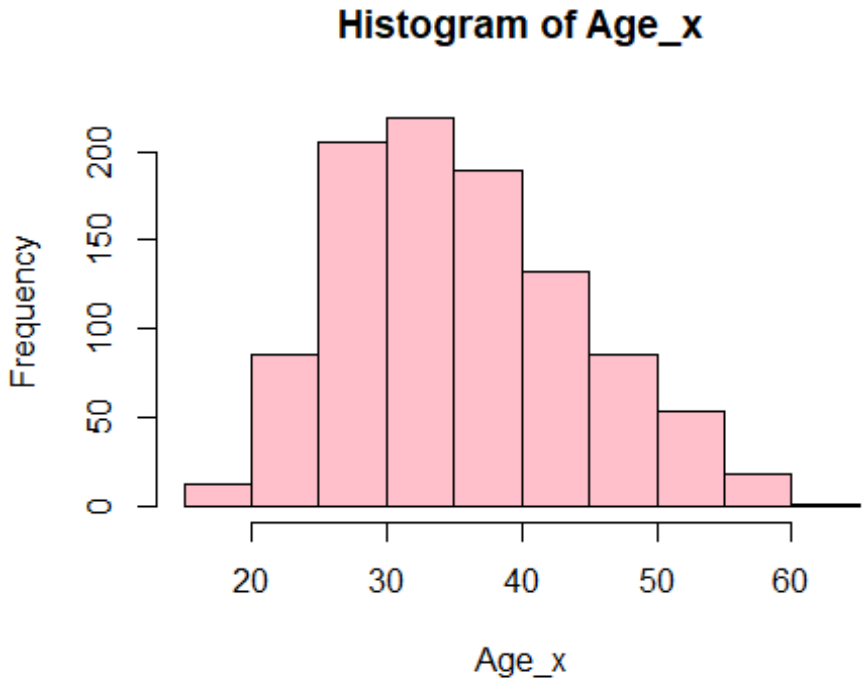
Age

```
# Age of the user
#This column represents the Age of the user
# measures of central tendency

# mean
mean(advert_df$Age)

## [1] 36.009

# median
median(advert_df$Age)

## [1] 35

# mode
Age_x <- advert_df$Age
```

```r
#sort(Age_x)
names(table(Age_x))[table(Age_x)==max(table(Age_x))]
```

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

```r
#each of the values printed below appear thrice in the dataset

#distribution
hist(Age_x, col = c("pink"))
```



The age distribution is right skewed

The respondents on the website are mostly 25-40 years old.

The mean age is 36.

The median age is 35

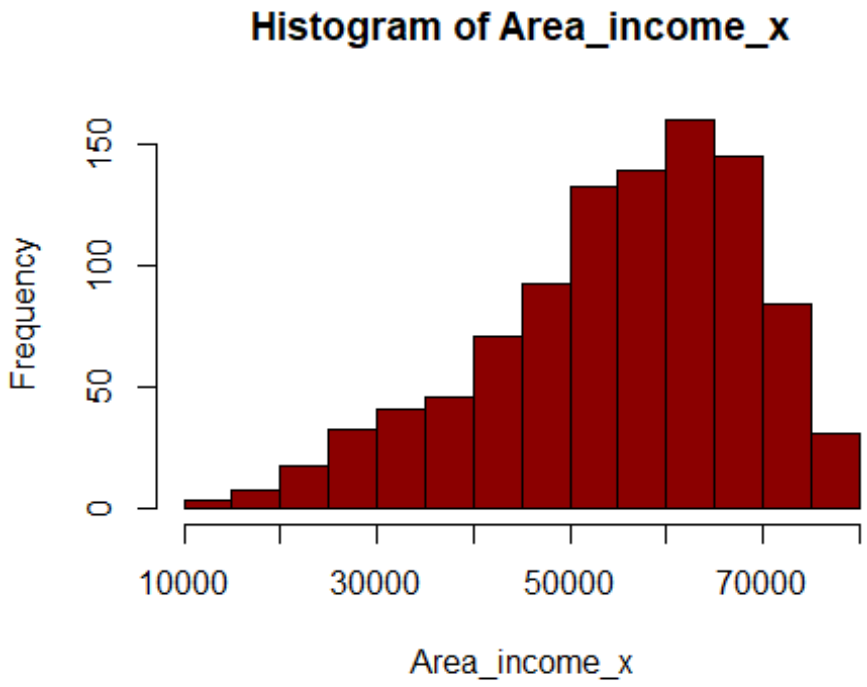
Area.Income

```r
#income

# mean
mean(advert_df$Area.Income)
```

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

```r
# median
median(advert_df$Area.Income)
```

```
## [1] 57012.3

# mode
Area_income_x <- advert_df$Area.Income
#sort(Daily.Internet.Usage_x)
#names(table(Area_income_x))[table(Area_income_x)==max(table(Area_income_x))]
#each of the values printed below appear thrice in the dataset

#distribution
hist(Area_income_x, col = c('darkred'))
```



The income distribution is left skewed

The respondents on the website mostly earn between 55,000 to 70,000.

The mean income is 55,000.

The median income is 57,012.

Daily.Internet.Usage

```
#This column represents the amount of data that the user consumers in a day
# measures of central tendency

# mean
mean(advert_df$Daily.Internet.Usage)

## [1] 180.0001
```

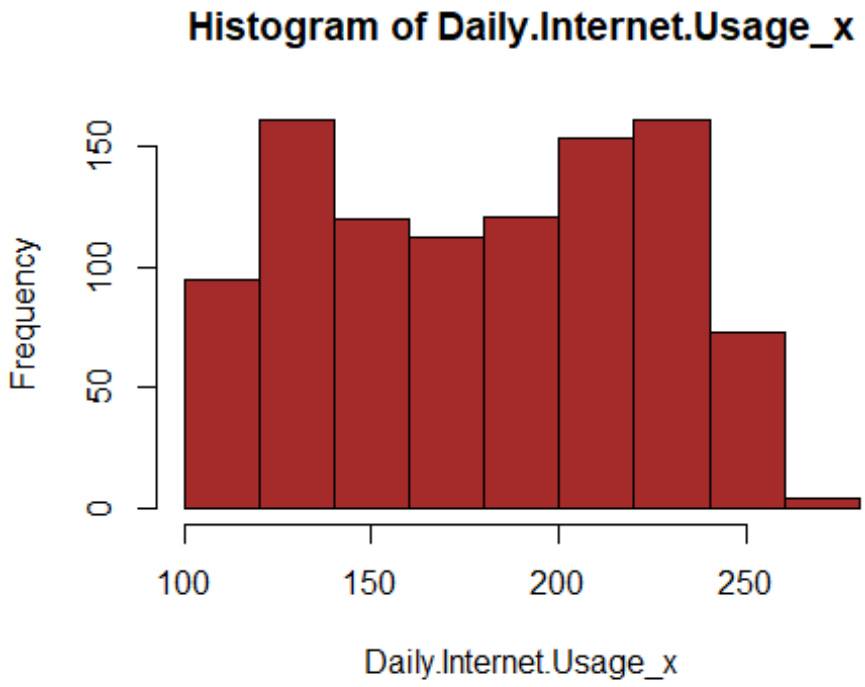
```r
# median
median(advert_df$Daily.Internet.Usage)
```

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

```r
# mode
Daily.Internet.Usage_x <- advert_df$Daily.Internet.Usage
#sort(Daily.Internet.Usage_x)
names(table(Daily.Internet.Usage_x))[table(Daily.Internet.Usage_x)==max(table
(Daily.Internet.Usage_x))]
```

```
##  [1] "113.53" "115.91" "117.3"  "119.3"  "120.06" "125.45" "132.38"
"135.24"
##  [9] "136.18" "138.35" "158.22" "161.16" "162.44" "164.25" "167.22"
"169.4"
## [17] "178.75" "182.65" "190.95" "194.23" "201.15" "211.87" "214.42"
"215.18"
## [25] "219.72" "222.11" "223.16" "228.81" "230.36" "234.75" "235.28"
"236.96"
## [33] "247.05" "256.4"
```

```r
#each of the values printed below appear thrice in the dataset

#distribution
hist(Daily.Internet.Usage_x, col = c('brown'))
```



The mean data usage is 180 units.

The median data usage is 183.13 units .

Ad.Topic.Line

```
Ad_topic_line <- advert_df$Ad.Topic.Line
#all the values are unique in this column thus we would drop it when
modelling since it
#does not provide any additional meaningful information

#levels(unique(Ad_topic_line))

#factor(unique(Ad_topic_line))
```

City

City where the user is located

```
#city where the user is located
# measures of central tendency

length(levels(advert_df$City))

## [1] 969

#there are 969 unique cities in the dataset

# mode
City_x <- advert_df$City

#sort(City_x) #this code gives an ordered list of all the elements in the
cities column

#The modal cities in the dataset
names(table(City_x))[table(City_x)==max(table(City_x))]

## [1] "Lisamouth"    "Williamsport"

#the most popular cities in the dataset are: Lisamouth and williamsport
```

Male

```
#gender of the user
#1 indicates that the user is male while indicates that they are female
# measures of central tendency

#levels(advert_df$Male) #this code does not work
#obtaining the unique levels in the gender(Male column)

unique(factor(advert_df$Male))

## [1] 0 1
## Levels: 0 1
```

```r
Male_x <- table(advert_df$Male)
#distribution
barplot(Male_x, main="Gender Distribution",col=c("darkgreen"),xlab="Gender")
```

Country

```r
#country where the user belongs
# measures of central tendency

# mode
Country_x <- advert_df$Country

#levels(Country_x) #this code gives the names of the countries

#There are 237 unique countries represented in the dataset
length(levels(Country_x))
```

```
## [1] 237
```

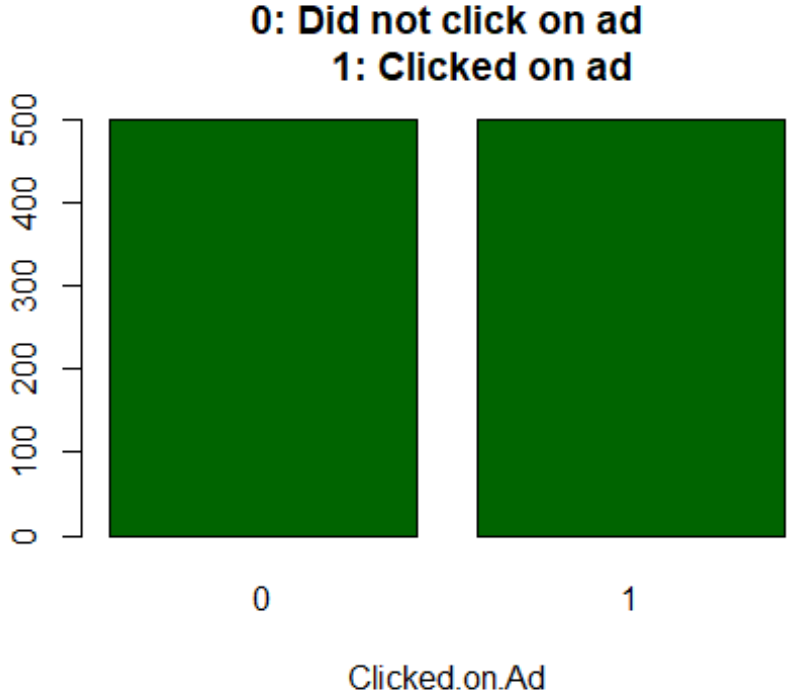
```r
#the modal countries in the dataset
names(table(Country_x))[table(Country_x)==max(table(Country_x))]
```

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

```r
#the most popular countries are:Czech Republic and France
```

Clicked.on.Ad

```
#zero indicates that a user did not click on an add while 1 indicates that a
user clicked on an add
# measures of central tendency

#levels(advert_df$Clicked.on.Ad) #this code does not work

unique(factor(advert_df$Clicked.on.Ad))

## [1] 0 1
## Levels: 0 1

#there are two unique factors in the clicked on ad column
# mode
Clicked.on.Ad_x <- table(advert_df$Clicked.on.Ad)
#sort(Daily.Internet.Usage_x)
names(table(Clicked.on.Ad_x))[table(Clicked.on.Ad_x)==max(table(Clicked.on.Ad
_x))]

## [1] "500"

#

#distribution
barplot(Clicked.on.Ad_x, main="0: Did not click on ad
        1: Clicked on ad " , col=c("darkgreen"),xlab="Clicked.on.Ad")
```



```
#the distribution is equal. 500 0's and 500 1's
```

Bivariate Analysis and Multivariate Graphical Data Analysis

```
advert_df2 <- subset(advert_df, select = c(Daily.Time.Spent.on.Site,
Age,Area.Income,Daily.Internet.Usage,Male,Clicked.on.Ad ))

head(advert_df2)
```

```
##   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
```

Correlation

```
#The default method is Pearson, but we can also compute Spearman or Kendall
coefficients.
mydata = cor(advert_df2, method = c("spearman"))
mydata1= cor(advert_df2, method = c("kendall"))
mydata2= cor(advert_df2, method = c("pearson"))

mydata #spearman
```

```
##                           Daily.Time.Spent.on.Site         Age Area.Income
## Daily.Time.Spent.on.Site               1.00000000 -0.31686155  0.28313439
## Age                                   -0.31686155  1.00000000 -0.13595396
## Area.Income                            0.28313439 -0.13595396  1.00000000
## Daily.Internet.Usage                   0.51410805 -0.37086395  0.33916021
## Male                                  -0.01592213 -0.02315468 -0.01436909
## Clicked.on.Ad                         -0.74487253  0.48633733 -0.46722440
##                           Daily.Internet.Usage        Male Clicked.on.Ad
## Daily.Time.Spent.on.Site           0.51410805 -0.01592213   -0.74487253
## Age                               -0.37086395 -0.02315468    0.48633733
## Area.Income                        0.33916021 -0.01436909   -0.46722440
## Daily.Internet.Usage               1.00000000  0.02820432   -0.77660702
## Male                               0.02820432  1.00000000   -0.03802747
## Clicked.on.Ad                     -0.77660702 -0.03802747    1.00000000
```

```
mydata1 #kendall
```

```
##                           Daily.Time.Spent.on.Site         Age Area.Income
## Daily.Time.Spent.on.Site               1.00000000 -0.19668659  0.16578119
## Age                                   -0.19668659  1.00000000 -0.08005810
## Area.Income                            0.16578119 -0.08005810  1.00000000
## Daily.Internet.Usage                   0.29323600 -0.23244607  0.20837546
## Male                                  -0.01300823 -0.01921715 -0.01173817
## Clicked.on.Ad                         -0.60855366  0.40363397 -0.38167782
##                           Daily.Internet.Usage        Male Clicked.on.Ad
## Daily.Time.Spent.on.Site           0.29323600 -0.01300823   -0.60855366
## Age                               -0.23244607 -0.01921715    0.40363397
## Area.Income                        0.20837546 -0.01173817   -0.38167782
## Daily.Internet.Usage               1.00000000  0.02304102   -0.63443547
## Male                               0.02304102  1.00000000   -0.03802747
## Clicked.on.Ad                     -0.63443547 -0.03802747    1.00000000
```

```
mydata2 #pearson
```

```
##                           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
```

Using the 3 correlation coefficients to get the correlation between the features, we can see that the correlation is very low and negative in most cases.

This means that most of the variables are NOT dependent of each other

Significance levels (p-values) can also be generated using the rcorr function which is

found in the Hmisc package.

First install the required package and load the library.

```
#install_version("latticeExtra")
#install.packages("Hmisc", dependencies = T)
library("Hmisc")
```

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

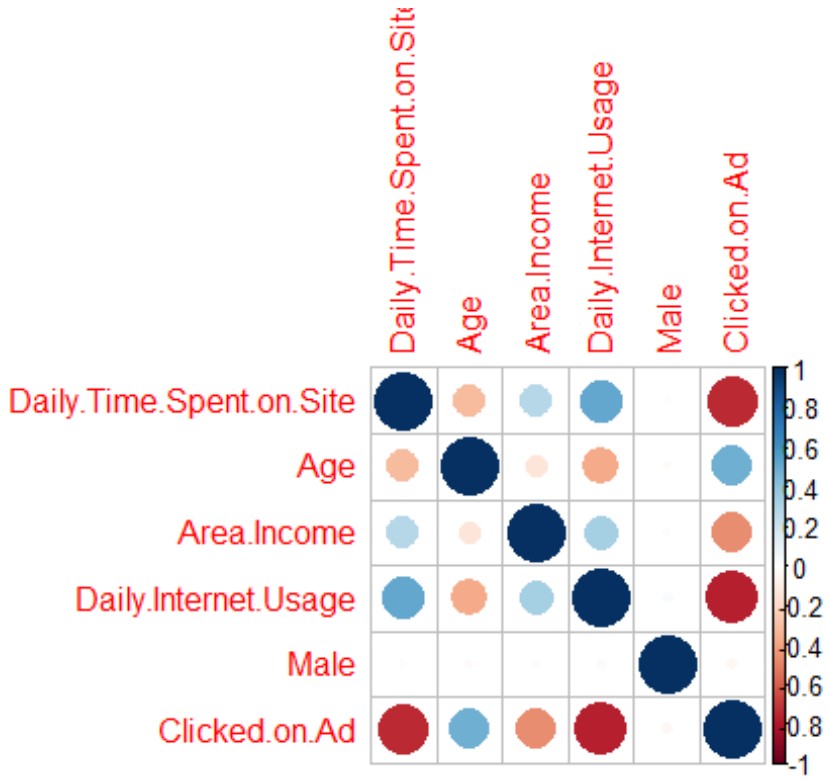
```
## Loading required package: survival
```

```
## Loading required package: Formula
```

```
## Loading required package: ggplot2

##
## Attaching package: 'Hmisc'

## The following objects are masked from 'package:base':
##
##     format.pval, units

mydata.rcorr = rcorr(as.matrix(mydata)) #feed the data as a matrix
mydata.rcorr

##                          Daily.Time.Spent.on.Site   Age Area.Income
## Daily.Time.Spent.on.Site                     1.00 -0.79        0.65
## Age                                         -0.79  1.00       -0.61
## Area.Income                                  0.65 -0.61        1.00
## Daily.Internet.Usage                         0.88 -0.83        0.70
## Male                                        -0.08 -0.15       -0.15
## Clicked.on.Ad                               -0.95  0.85       -0.77
##                          Daily.Internet.Usage  Male Clicked.on.Ad
## Daily.Time.Spent.on.Site                 0.88 -0.08         -0.95
## Age                                     -0.83 -0.15          0.85
## Area.Income                              0.70 -0.15         -0.77
## Daily.Internet.Usage                     1.00 -0.03         -0.97
## Male                                    -0.03  1.00          0.00
## Clicked.on.Ad                           -0.97  0.00          1.00
##
## n= 6
##
##
## P
##                          Daily.Time.Spent.on.Site Age    Area.Income
## Daily.Time.Spent.on.Site                          0.0626 0.1620
## Age                      0.0626                          0.1966
## Area.Income              0.1620                  0.1966
## Daily.Internet.Usage     0.0213                  0.0422 0.1252
## Male                     0.8853                  0.7736 0.7717
## Clicked.on.Ad            0.0034                  0.0335 0.0742
##                          Daily.Internet.Usage Male   Clicked.on.Ad
## Daily.Time.Spent.on.Site 0.0213               0.8853 0.0034
## Age                      0.0422               0.7736 0.0335
## Area.Income              0.1252               0.7717 0.0742
## Daily.Internet.Usage                          0.9623 0.0015
## Male                     0.9623                      0.9936
## Clicked.on.Ad            0.0015               0.9936
```

This generates one table of correlation coefficients (the correlation matrix) and another table of the p-values. By default, the correlations and p-values are stored in an object of class type rcorr.

```
#mydata.coeff = mydata.rcorr$r
#mydata.p = mydata.rcorr$P
library(corrplot)

## corrplot 0.84 loaded

corrplot(mydata)
```



A default correlation matrix plot (called a Correlogram) is generated. Positive correlations are displayed in a blue scale while negative correlations are displayed in a red scale

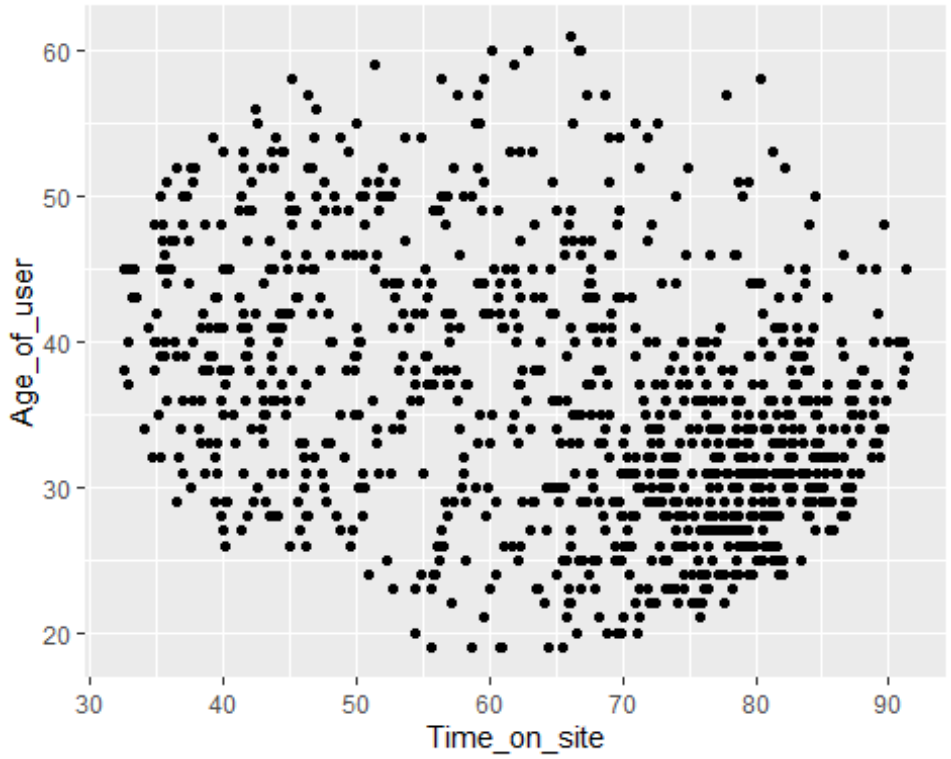
There is very minimal positive correlation between the variables in the data

The Plots below are scatterplots of a few pairs of variables

Time spent on the site vs age of the user

```
#Time spent on the site vs age of the user
# Libraries
library(ggplot2)

# create data
Time_on_site <- advert_df$Daily.Time.Spent.on.Site
Age_of_user <- advert_df$Age
data <- data.frame(Time_on_site,Age_of_user)
```

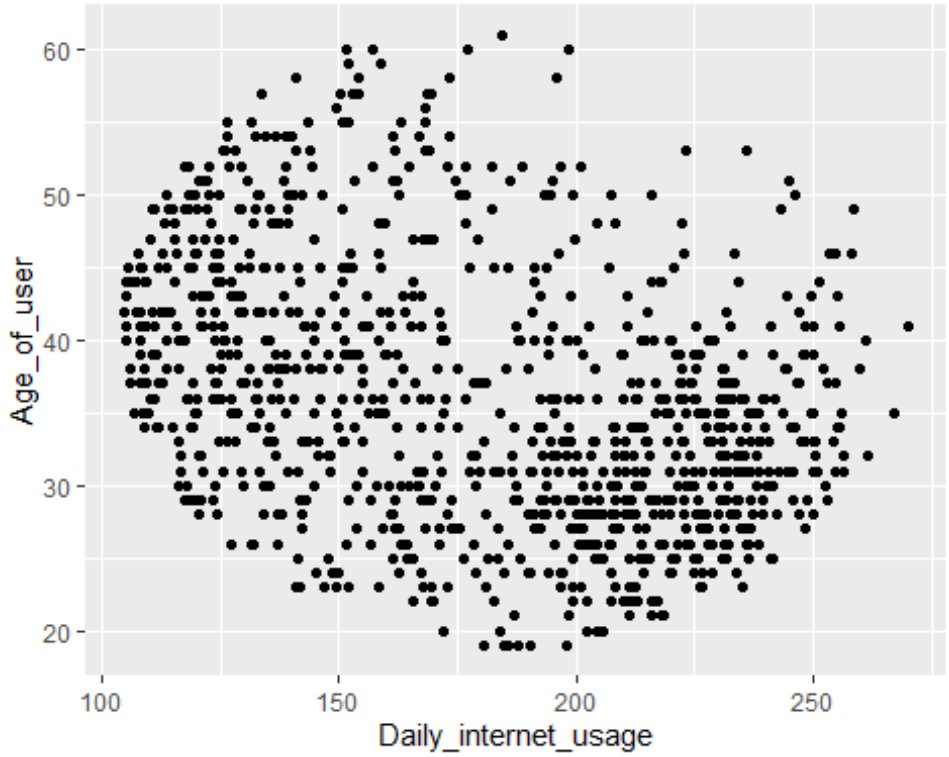
```r
# Plot
ggplot(data, aes(x=Time_on_site, y=Age_of_user)) + geom_point()
```




```r
#positive non-linear correlation
```

#Age of the user vs daily internet usage

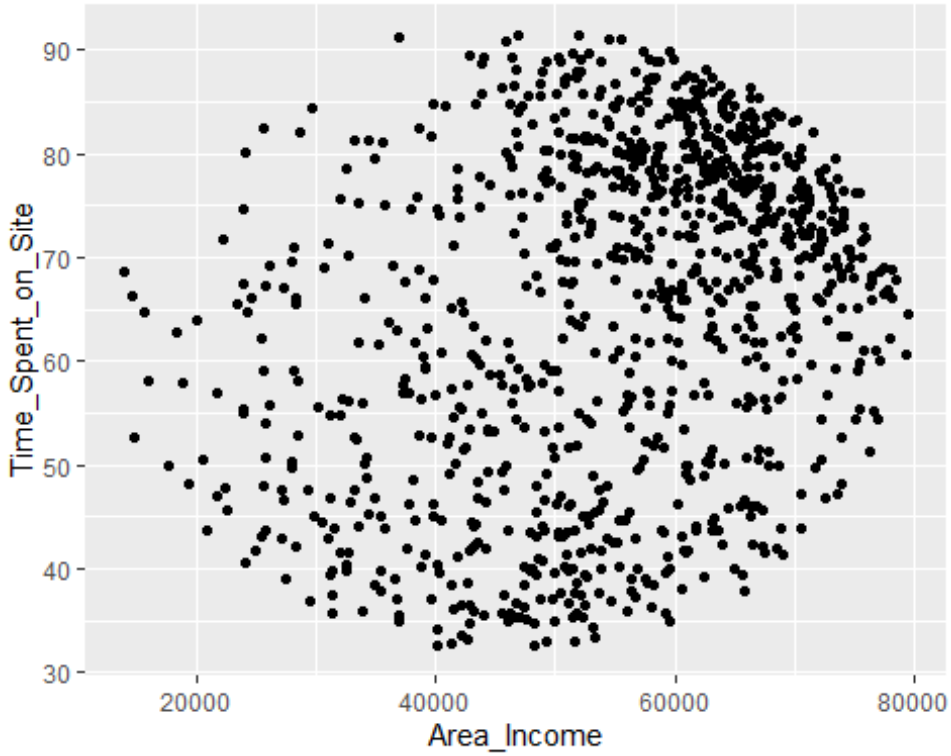
```r
Daily_internet_usage <- advert_df$Daily.Internet.Usage
Age_of_user <- advert_df$Age
data1 <- data.frame(Daily_internet_usage,Age_of_user)

# Plot
ggplot(data1, aes(x=Daily_internet_usage, y=Age_of_user)) + geom_point()
```




```
#the plot shows that there is positive non-linear correlation
```

#time spent on the site vs area.income

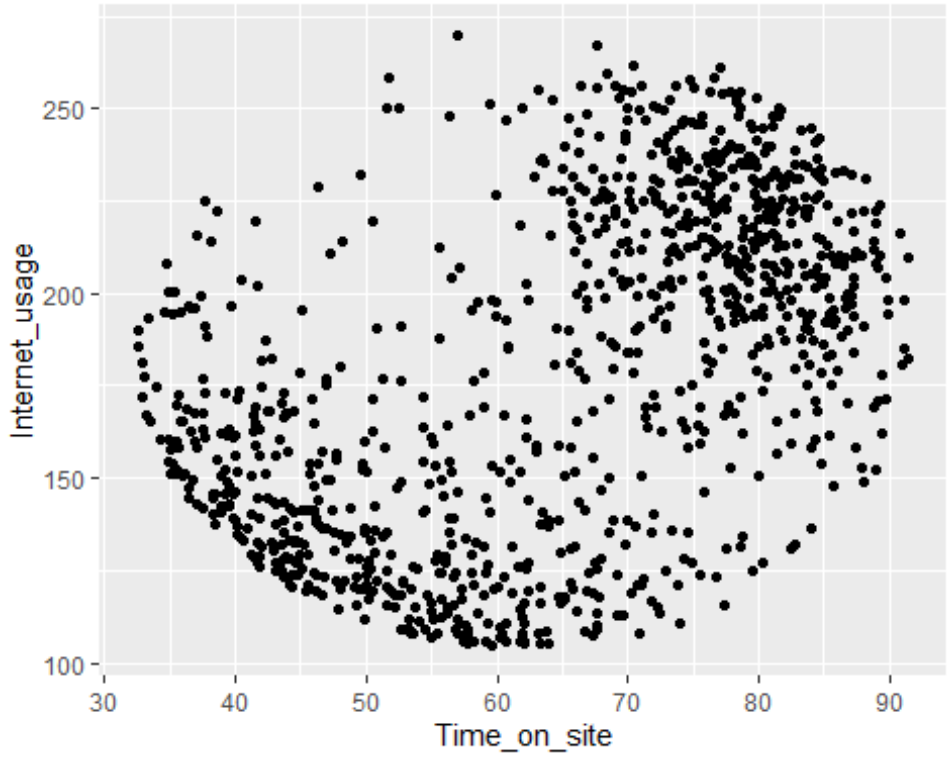
```
Area_Income <- advert_df$Area.Income
Time_Spent_on_Site <- advert_df$Daily.Time.Spent.on.Site
data2 <- data.frame(Area_Income,Time_Spent_on_Site)

# Plot
ggplot(data2, aes(x=Area_Income, y=Time_Spent_on_Site)) + geom_point()
```




```
#positive non-linear correlation
```

#time spent on the site vs daily iternet usage

```
Time_on_site <- advert_df$Daily.Time.Spent.on.Site
Internet_usage <- advert_df$Daily.Internet.Usage
data3 <- data.frame(Time_on_site,Internet_usage)

# Plot
ggplot(data3, aes(x=Time_on_site, y=Internet_usage)) + geom_point()
```



Seperating the data Clicked and Gender columns

```
#creating a new column with null values
advert_df2["Female"] <- NA
dim(advert_df2)

## [1] 1000    7

#populating the column with false values from the male column
advert_df2$Female <- advert_df2$Male == 0
dim(advert_df2)

## [1] 1000    7

#converting the column to nu,eric
dim(advert_df2 <- apply(advert_df2, 2, as.numeric))

## [1] 1000    7
```

Gender VS Clicked on Add

```
library(tidyverse)

## -- Attaching packages ---------------------------------------------------
tidyverse 1.3.0 --

## v tibble  2.1.3     v dplyr   0.8.4
## v tidyr   1.0.2     v stringr 1.4.0
## v purrr   0.3.3     v forcats 0.5.0
```

```
## -- Conflicts ------------------------------------------------------
tidyverse_conflicts() --
## x dplyr::between()   masks data.table::between()
## x dplyr::filter()    masks stats::filter()
## x dplyr::first()     masks data.table::first()
## x dplyr::lag()       masks stats::lag()
## x dplyr::last()      masks data.table::last()
## x dplyr::src()       masks Hmisc::src()
## x dplyr::summarize() masks Hmisc::summarize()
## x purrr::transpose() masks data.table::transpose()
```

```
#Male respondents who clicked on an add
dim(advert_df%>% filter(Male == 1 , Clicked.on.Ad == 1))
```

```
## [1] 231  10
```

```
#231
```

```
#Male respondents did not click on an add
dim(advert_df%>% filter(Male == 1, Clicked.on.Ad == 0))
```

```
## [1] 250  10
```

```
#250
```

```
#Female respondents who clicked on an add
dim(advert_df%>% filter(Male == 0 , Clicked.on.Ad == 1))
```

```
## [1] 269  10
```

```
# 269
```

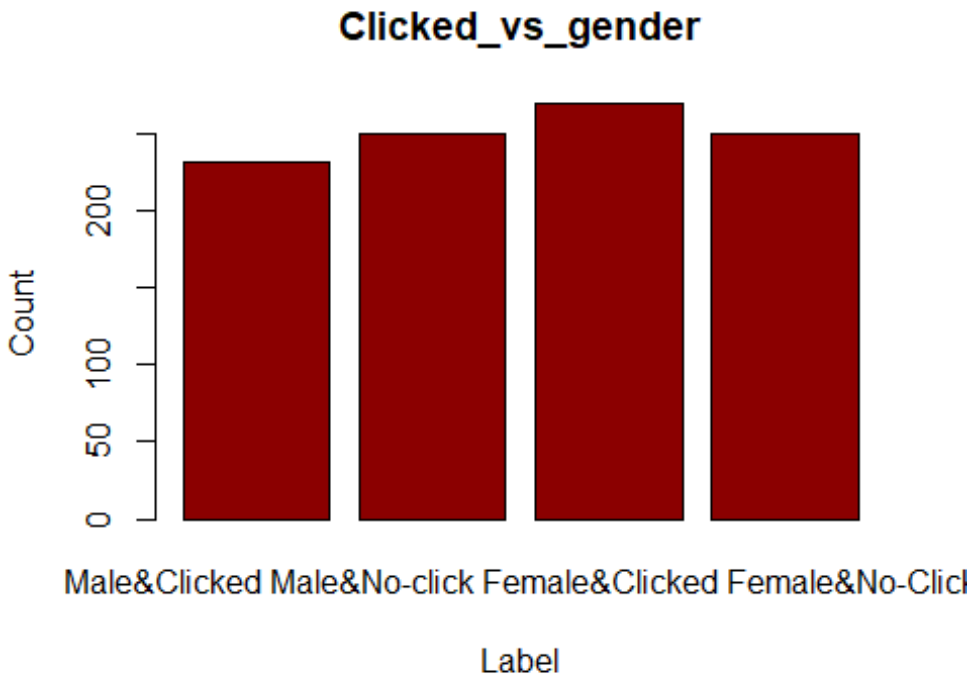
```
#Female respondents who clicked did not on an add
dim(advert_df%>% filter(Male == 0, Clicked.on.Ad == 0))
```

```
## [1] 250  10
```

```
# 250
```

```
Clicked_vs_gender <- c( 231 , 250 , 269 , 250 )
```

```
# barchart with added parameters
barplot(Clicked_vs_gender, main = " Clicked_vs_gender " , xlab = " Label ",
ylab = " Count ",
names.arg = c("Male&Clicked Male&No-click Female&Clicked Female&No-Click"),
col = "darkred",
horiz = FALSE)
```



Summary

1. The time a user spends on the site does not influence the possibility of clicking on an add.

2. The gender of respondents who clicked on an add, and those who did not click on an add does not vary much. This means that the Gender of the respondent should be considered in equal measure.

3. Most of the site users who are likely to click on an add earn between 55,000 to 70,000 per month. There are low income earners who click on ads but the majority earn the amount stated above