IP - Week 13 mod 3

Eltonjohn Oketch

2022-06-05

Defining the Question

In this week's project I'll be working as a Data Science Consultant to a Kenyan entrepreneur who 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.

Metrics of Success

1. Find and deal with outliers, anomalies, and missing data within the dataset.
2. Perform univariate and bivariate analysis.
3. Provide a conclusion and recommendation

Understanding the Context

Every mainstream media website, news site, top blog, YouTube, and social media site uses advertising, and it's because advertising is a proven moneymaker. If you're a blogger with an audience that companies want to reach, you have the potential to make money by selling ads. Therefore, it's important that you know what type of advertising is going to sell best to your specific audience. As such, learning stats about your audience can help you to learn the basic demographics of your audience. The more you know about your audience, the better you'll be able to sell advertising to them.

Recording the experimental design.

The following steps will be followed in conducting this study:

1. Define the question, the metric for success, the context, experimental design taken.
2. Data Sourcing
3. Check the Data
4. Perform Data Cleaning
5. Perform Exploratory Data Analysis (Univariate, Bivariate & Multivariate) 6. Implement the Solution
7. Challenge the Solution
8. Follow up Questions

Data Relevance

Our data to be used in these research is http://bit.ly/IPAdvertisingData, there are 10 columns with the names:

1. Daily.Time.Spent.on.Site
2. Age

3. Area.Income
4. Daily.Internet.Usage
5. Ad.Topic.Line
6. City
7. Male
8. Country
9. Timestamp
10. Clicked.on.Ad

Data sourcing

Loading the dataset and libraries.

```
advert_data <- read.csv("http://bit.ly/IPAdvertisingData") 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
```

We see the first six entries for each column.

Checking the data

finding the summary of the data

```
# finding the data summary summary(advert_data)
```

```
## 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             Male            Country
## Length:1000        Length:1000        Min.   :0.000   Length:1000
## Class :character   Class :character   1st Qu.:0.000   Class :character
## Mode  :character   Mode  :character   Median :0.000   Mode  :character
##                                       Mean   :0.481
##                                       3rd Qu.:1.000
##                                       Max.   :1.000
##    Timestamp          Clicked.on.Ad
## Length:1000        Min.   :0.0
## Class :character   1st Qu.:0.0
## Mode  :character   Median :0.5
##                    Mean   :0.5
##                    3rd Qu.:1.0
##                    Max.   :1.0
```

There are 10 columns, 6 are in numeric form while 4 are in character form. For numeric columns we can find the minimum, 1st quantile, median, mean, 3rd quantile and maximum value, these is because mathematical equations can only be formed on numeric data.

```
# dropping the ad topic line and time stamp column
```

```
advert_data = advert_data[,!(names(advert_data) %in% c("Ad.Topic.Line","Timestamp", "City"))]
```
Droping irrelevant columns

Data Cleaning

Finding the missing data

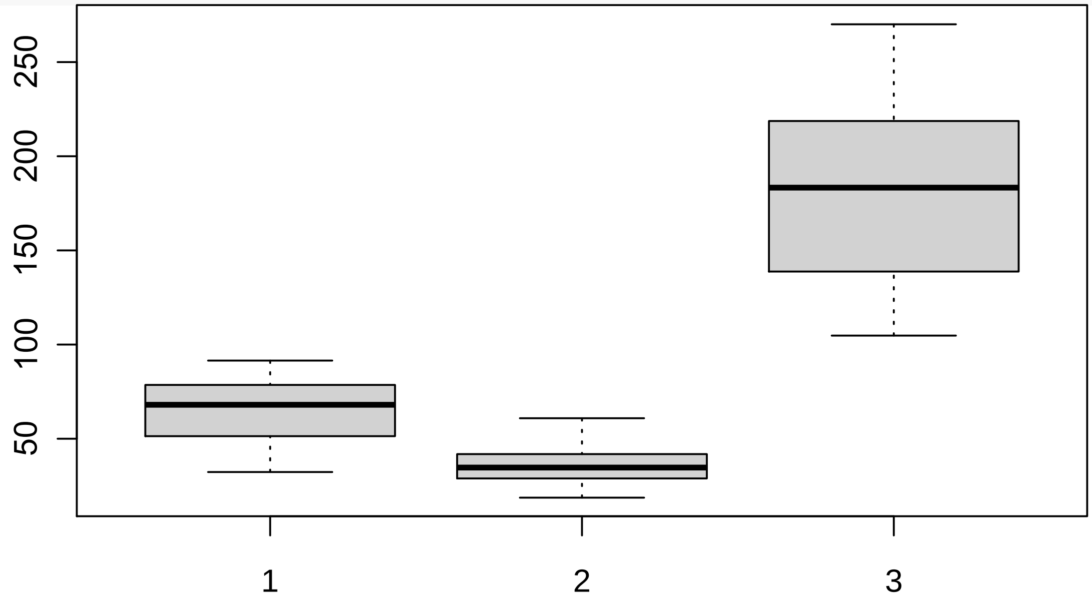
```
# Lets Identify missing data in your dataset
# # colSums(is.na(advert_data))
```

```
## Daily.Time.Spent.on.Site                 Age         Area.Income
##                        0                   0                   0
##       Daily.Internet.Usage                Male             Country
##                        0                   0                   0
##            Clicked.on.Ad
##                        0
```

There are no null values in the dataset, hence we can say all the data entries were covered.

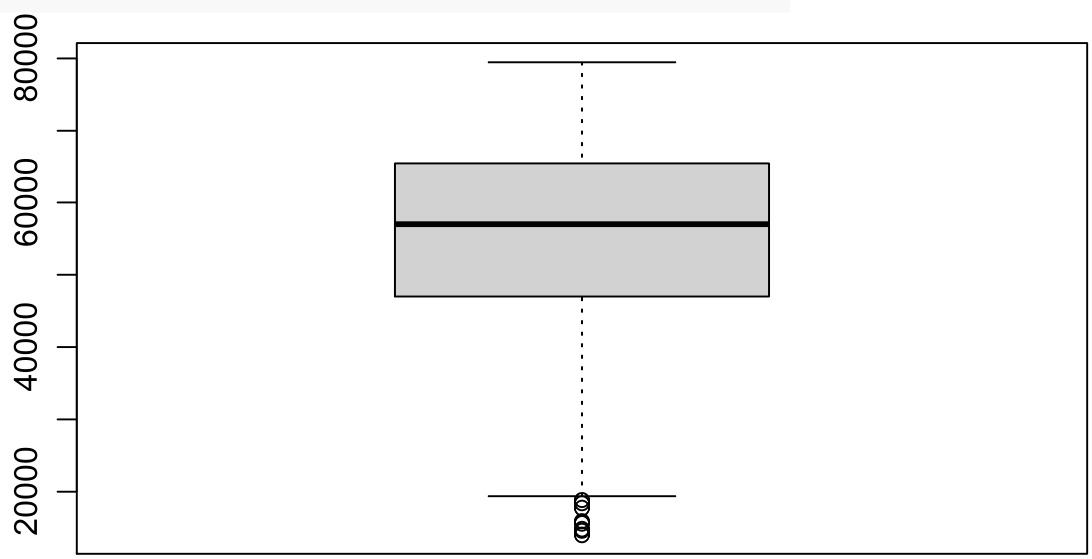
Checking for the outliers

```
# we shall check for the outliers in the dataset using the boxplot boxplot(advert_data$Daily.Time.Spent.on.Site, advert_data$Age,
advert_data$Daily.Internet.Usage)
```



> There are no outliers in daily time spent on site and daily internet usage.

```
# checking for outliers in area income column.

boxplot(advert_data$Area.Income)
```



```
# listing the outliers in the vectors
# ---
# boxplot.stats(advert_data$Area.Income)$out
```

[1] 17709.98 18819.34 15598.29 15879.10 14548.06 13996.50 14775.50 18368.57

There are seven outliers in the dataset, between 17000 and 18400 area income, we cannot drop the outliers since these is a true income, since their people earning below 17,000.

Checking for duplicates

```
# checking for duplicated data duplicated_rows <-
advert_data[duplicated(advert_data),]
# printing the duplicated_rows duplicated_rows
```

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

There are no duplicated rows in the dataset.

Exploratory Data Analysis

Univariate Data Analysis

Checking for the mean of the dataset

```
# Checking for mean of time spent on site advert.Daily.Time.Spent.on.Site.mean
<- mean(advert_data$Daily.Time.Spent.on.Site)
# Printing out
# --advert.Daily.Time.Spent.on.Site.mean
```

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

```
#----
# Checking for mean of age advert.Age.mean <-
mean(advert_data$Age)
# Printing out # ---
advert.Age.mean
```

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

```
#----
# Checking for the mean of area income
# --advert.Area.Income.mean <- mean(advert_data$Area.Income)
# Printing out # ---
advert.Area.Income.mean
```

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

```
# Checking for mean of age advert.Daily.Internet.Usage.mean <-
mean(advert_data$Daily.Internet.Usage)
# Printing out
```

```
# --advert.Daily.Internet.Usage.mean
```

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

The average time spent on site is 65, the average mean of age is 36 for person using the internet, mean income for the persons using the internet is 55000, while the mean daily internet usage is 180.

Checking for the median of the dataset

```
# Checking for median of time spent on site
advert.Daily.Time.Spent.on.Site.median <-
median(advert_data$Daily.Time.Spent.on.Site)
# Printing out median of time spent on site
# --advert.Daily.Time.Spent.on.Site.median
```

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

```
#----
# Checking for median of age advert.Age.median <-
median(advert_data$Age)
# Printing out the median of age
# --advert.Age.median
```

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

```
#----
# Checking for the median of area income
# --advert.Area.Income.median <- median(advert_data$Area.Income)
# Printing out median of area income
# --advert.Area.Income.median
```

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

```
# Checking for median of internet usage advert.Daily.Internet.Usage.median <-
median(advert_data$Daily.Internet.Usage)
# Printing out median of internet usage
# --advert.Daily.Internet.Usage.median
```

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

The median time spent on site is 68 minutes, while the median age of people using the internet is 35 years, for the median income is 57012, while the daily data usage is 183mbs.

Checking for the mode of the dataset

```
# Using a function for finding mode
getmode <- function(v) {
uniqv <- unique(v) uniqv[which.max(tabulate(match(v, uniqv)))]
}
# finding the mode advert_data.Male.mode <-
getmode(advert_data$Male) # Then printing out
advert_data.Male.mode advert_data.Male.mode
```

[1] 0

```
# finding the mode of clicks on the add.
advert_data.Clicked.on.Ad.mode <- getmode(advert_data$Clicked.on.Ad)
# printing the mode advert_data.Clicked.on.Ad.mode
```

[1] 0

The most common value for male and clicked ad is 0.

Finding the minimum values in the dataset

```
# Checking for minimum of time spent on site
advert.Daily.Time.Spent.on.Site.min <-
min(advert_data$Daily.Time.Spent.on.Site)
# Printing out minimum time spent on site
# --advert.Daily.Time.Spent.on.Site.min
```

[1] 32.6

```
#----
# Checking for minimum of age advert.Age.min <-
min(advert_data$Age)
# Printing out the minimum of age
# --advert.Age.min
```

[1] 19

```
#----
# Checking for the minimum of area income
# --advert.Area.Income.min <- min(advert_data$Area.Income)
# Printing out the minimum
# --advert.Area.Income.min
```

[1] 13996.5

```
# Checking for minimum of age advert.Daily.Internet.Usage.min <-
min(advert_data$Daily.Internet.Usage)
# Printing out the minimum age
# --advert.Daily.Internet.Usage.min
```

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

For the given columns with a range of values we have minimum values, the minimum time spent on site is 32 minutes, the youngest person using the internet is 19 years of age, the minimum income is 13996, while the minimum daily internet usage is 104 mbs for all internet users.

Finding the maximum values

```
# Checking for maximum of time spent on site
advert.Daily.Time.Spent.on.Site.max <-
max(advert_data$Daily.Time.Spent.on.Site)
# Printing out maximum time spent on site
# --advert.Daily.Time.Spent.on.Site.max
```

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

```
#----
# Checking for maximum of age advert.Age.max <-
max(advert_data$Age)
# Printing out the maximum of age
# --advert.Age.max
```

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

```
#----
# Checking for the maximum of area income
# --advert.Area.Income.max <- max(advert_data$Area.Income)
# Printing out the maximum
# --advert.Area.Income.max
```

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

```
# Checking for maximum of internet usage advert.Daily.Internet.Usage.max <-
max(advert_data$Daily.Internet.Usage)
# Printing out the maximum internet usage
# --advert.Daily.Internet.Usage.max
```

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

The maximum time ever spent on site was 91 seconds, the oldest person using the internet and visiting the site was 61 years, the highest income was 79484, being the highest salary, while the largest amount of internet used on site is 269.

Finding the range of the dataset

```
# Checking for range of time spent on site advert.Daily.Time.Spent.on.Site.range
<- range(advert_data$Daily.Time.Spent.on.Site)
```

```
# Printing out range time spent on site
# --advert.Daily.Time.Spent.on.Site.range
```

[1] 32.60 91.43

```
#----
# Checking for range of age advert.Age.range <-
range(advert_data$Age)
# Printing out the range of age
# --advert.Age.range
```

[1] 19 61

```
#----
# Checking for the range of area income
# --advert.Area.Income.range <- range(advert_data$Area.Income)
# Printing out the range
# --advert.Area.Income.range
```

[1] 13996.5 79484.8

```
# Checking for range of internet usage advert.Daily.Internet.Usage.range <-
range(advert_data$Daily.Internet.Usage)
# Printing out the range of internet usage
# --advert.Daily.Internet.Usage.range
```

[1] 104.78 269.96

```
#----
```

The range of the time spent on site is between 32.6 to 91.43, the range of age of persons who visited the site is 19 to 61 year of age, the range of income of the site users range between 13996 to 79484, while the internet usage of all the site users was between 104 to 269 mbs.

Finding the quantifies of the dataset

```
# Checking for quantile of time spent on site
advert.Daily.Time.Spent.on.Site.quantile <-
quantile(advert_data$Daily.Time.Spent.on.Site)
# Printing out quantile time spent on site
# --advert.Daily.Time.Spent.on.Site.quantile
```

```
##      0%     25%     50%     75%    100%
## 32.6000 51.3600 68.2150 78.5475 91.4300
```

```
#----
# Checking for quantile of age advert.Age.quantile <-
quantile(advert_data$Age)
# Printing out the quantile of age
# --advert.Age.quantile
```

```
##     0% 25% 50% 75% 100%
##    19   29   35   42   61
```

```
#----
# Checking for the quantile of area income
# --advert.Area.Income.quantile <- quantile(advert_data$Area.Income)
# Printing out the quantile
# --advert.Area.Income.quantile
```

```
##       0%      25%      50%      75%     100%
## 13996.50 47031.80 57012.30 65470.64 79484.80
```

```
# Checking for quantile of internet usage advert.Daily.Internet.Usage.quantile <-
quantile(advert_data$Daily.Internet.Usage)
# Printing out the quantile of internet usage
# --advert.Daily.Internet.Usage.quantile
```

```
##       0%      25%      50%      75%     100%
## 104.7800 138.8300 183.1300 218.7925 269.9600
#----
```

The quantile divides the data into 0% quantile which is the minimum value, the 25% quantile which is the first quater, the 50% is the medium value, the 75% is the third quantile, while the 100% is the maximum value of the data in the column.

Finding the variance of the dataset

```
# Checking for variance of time spent on site
advert.Daily.Time.Spent.on.Site.variance <-
var(advert_data$Daily.Time.Spent.on.Site)
# Printing out variance time spent on site
# --advert.Daily.Time.Spent.on.Site.variance
```

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

```
#----
# Checking for variance of age advert.Age.variance <-
var(advert_data$Age)
# Printing out the variance of age
# --advert.Age.variance ## [1] 77.18611
```

```
#----
# Checking for the variance of area income
# --advert.Area.Income.variance <- var(advert_data$Area.Income)
# Printing out the variance
# --advert.Area.Income.variance
```

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

```
# Checking for variance of internet usage advert.Daily.Internet.Usage.variance
<- var(advert_data$Daily.Internet.Usage)
# Printing out the quantile of internet usage
# --advert.Daily.Internet.Usage.variance
```

[1] 1927.415

```
#----
```

variance is how spread the data is, the income data has the highest data spread out by 179952406, while age has the lowest data spread apart by 77

Finding the standard deviation

```
# Checking for standard deviation of time spent on site
advert.Daily.Time.Spent.on.Site.sd <- sd(advert_data$Daily.Time.Spent.on.Site)
# Printing out standard deviation time spent on site
# --advert.Daily.Time.Spent.on.Site.sd
```

[1] 15.85361

```
#----
# Checking for standard deviation of age advert.Age.sd <-
sd(advert_data$Age)
# Printing out the standard deviation of age
# --advert.Age.sd
```

[1] 8.785562

```
#----
# Checking for the standard deviation of area income
# --advert.Area.Income.sd <- sd(advert_data$Area.Income) # Printing out
the standard deviation of area income
# --advert.Area.Income.sd ## [1] 13414.63
```

```
# Checking for standard deviation of internet usage
advert.Daily.Internet.Usage.sd <- sd(advert_data$Daily.Internet.Usage)
# Printing out the standard deviation of internet usage
# --advert.Daily.Internet.Usage.sd
```

[1] 43.90234

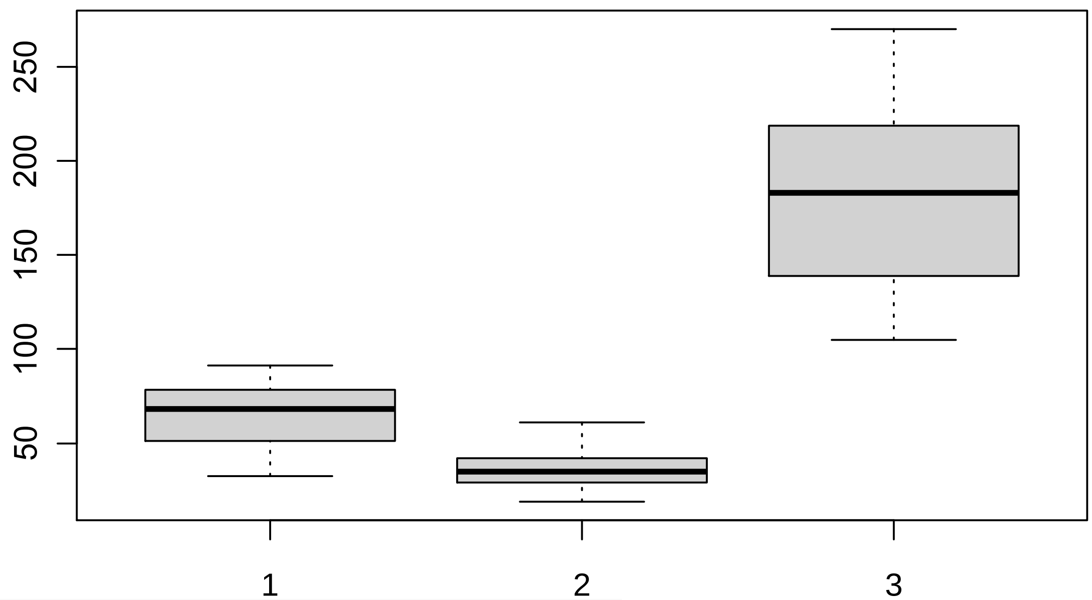
```
#----
```

Standard deviation measures how far the data is spread away from the mean value. for the area income there is a great deviation from the mean at 13414, while age having the least measure of deviation from the mean at 8.

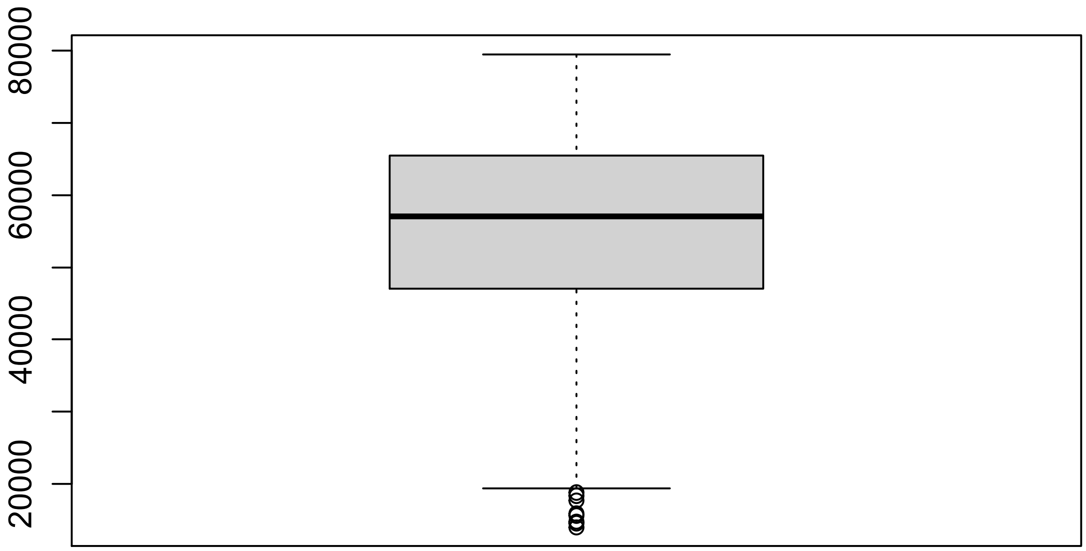
Plotting different graphs

Boxplot

```
# ploting box plot for time spent, age and internet usage boxplot(advert_data$Daily.Time.Spent.on.Site, advert_data$Age, advert_data$Daily.Internet.Usage)
```



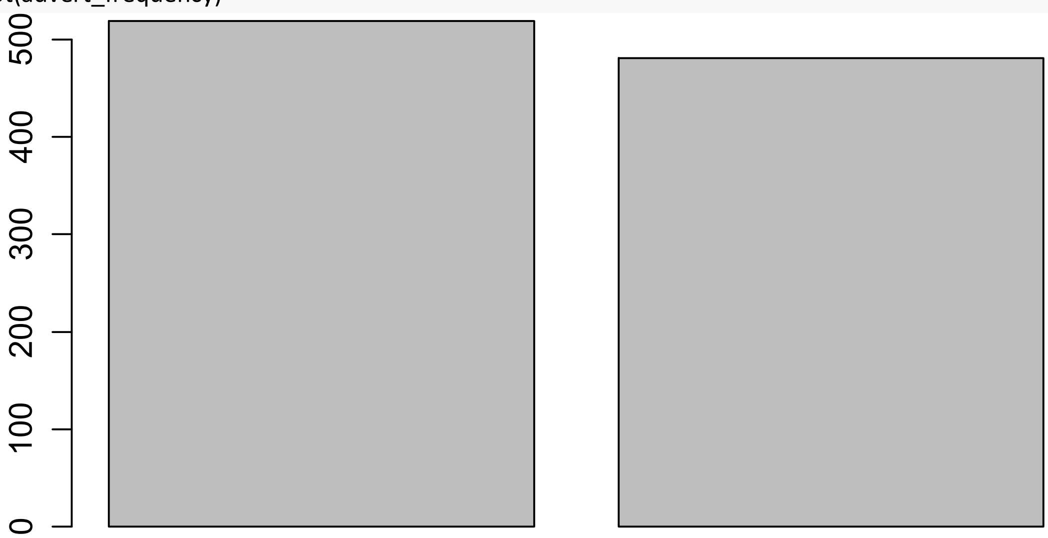

```
# plotting boxplot for area income

boxplot(advert_data$Area.Income)
```



> For the box plot there are no outliers in age, time spent on time and daily data internet, while for area income there is outliers for those receiving lowest income.

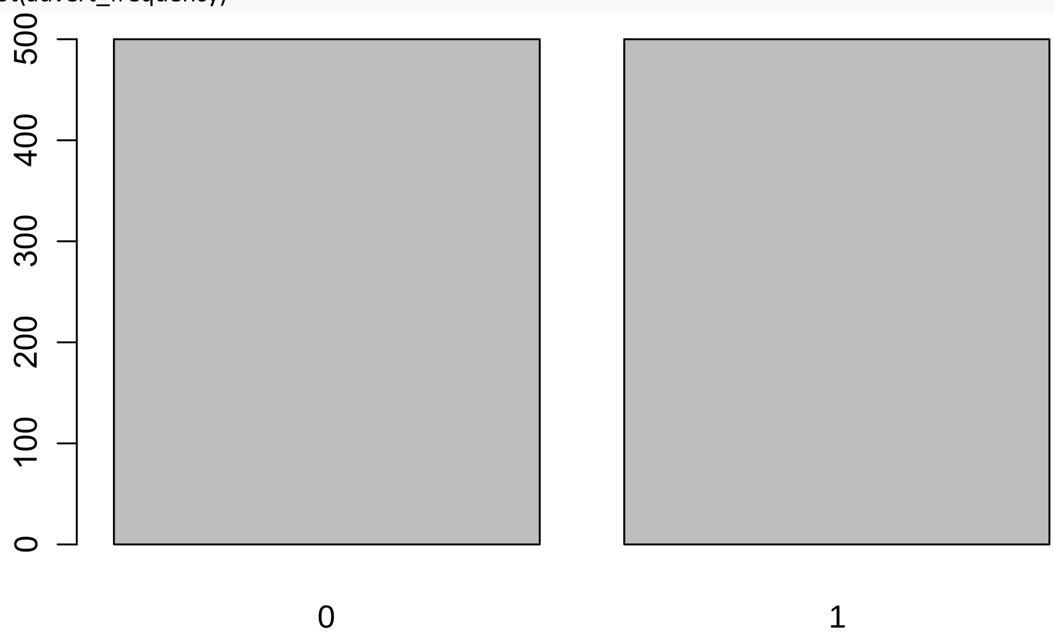
Bar Graph

```
# plotting bar graph on Male advert <-
advert_data$Male
# ---
# Applying table advert_frequency <-
table(advert)
# Then applying the barplot function to produce its bar graph
# ---
# barplot(advert_frequency)
```



0 1

```
#----
# ploting bar on clicked on ad data advert1 <-
advert_data$Clicked.on.Ad
# Applying table advert_frequency <-
table(advert1)
# Then applying the barplot function to produce its bar graph
# ---
# barplot(advert_frequency)
```

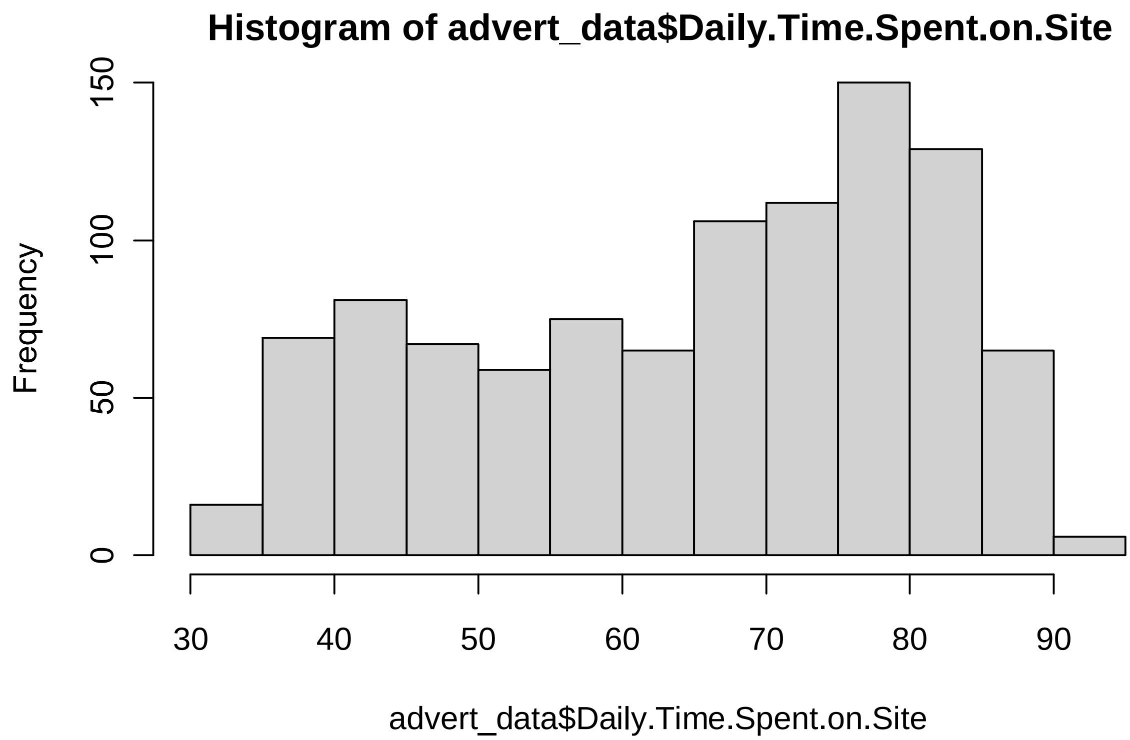


> For the bar graph, for those who visited the site, there is a slight difference between male persons who visited the site than the female who visited the site.

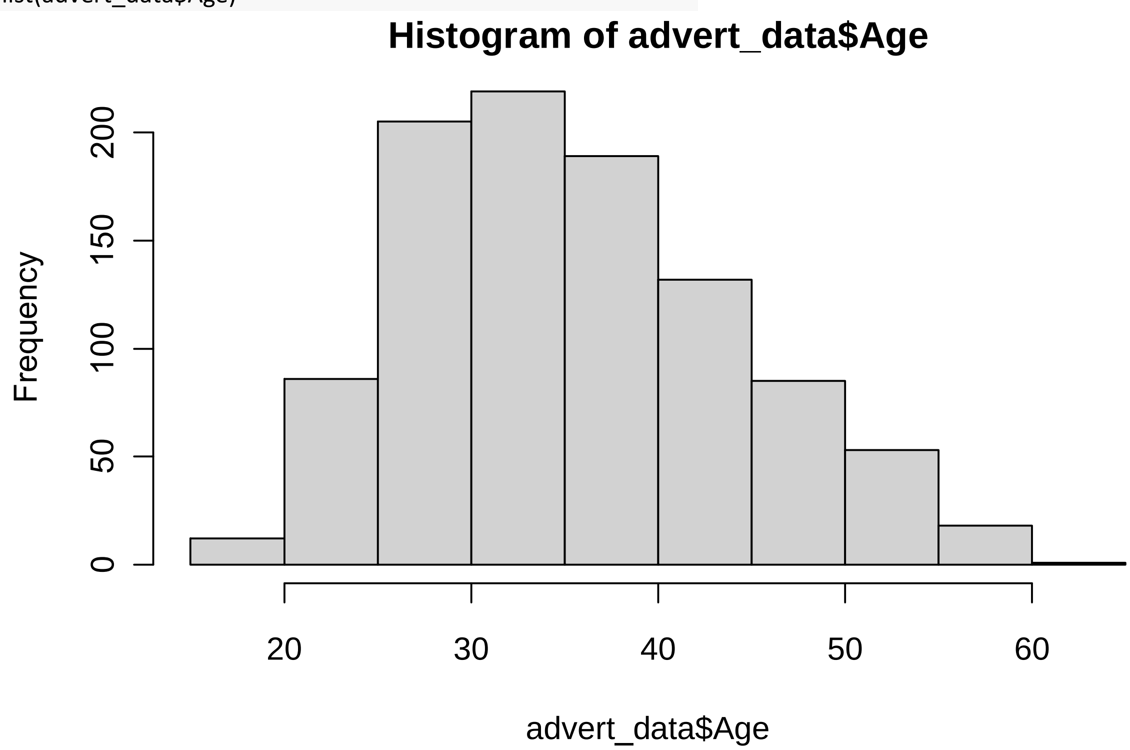
There is an equal representation of those who clicked on the ad and those who didn't click on the ad.

Plotting histogram

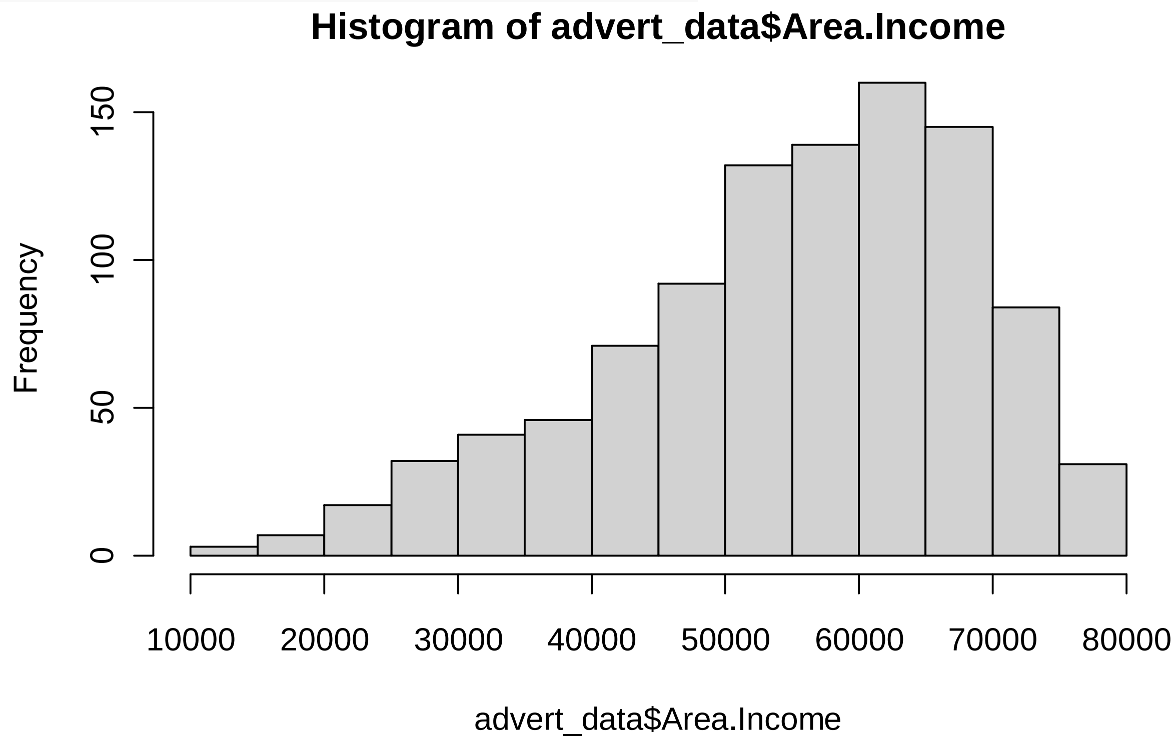
```
# histogram for time spent on site
# hist(advert_data$Daily.Time.Spent.on.Site)
```



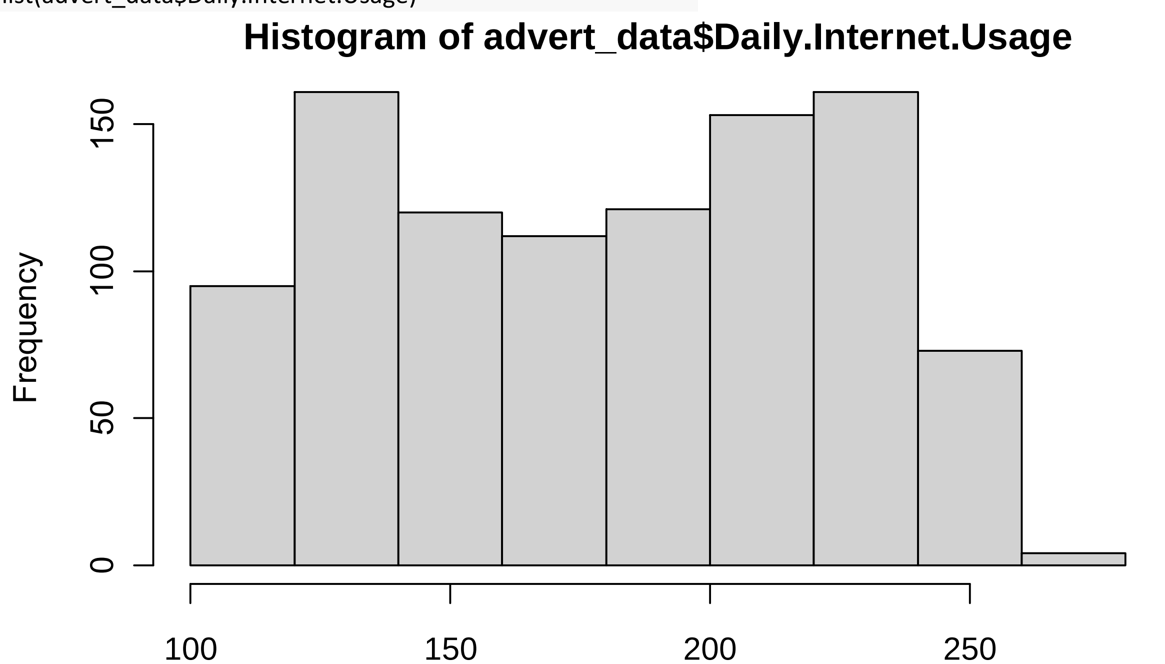

```
# histogram for age distribution

hist(advert_data$Age)
```



```
# histogram for area income distribution

hist(advert_data$Area.Income)
```




```
# histogram for internet usage distribution

hist(advert_data$Daily.Internet.Usage)
```



advert_data$Daily.Internet.Usage

> Most of the time spent on the site was between 75 to 80 seconds with a frequency of around 150, while the smallest time spent on site is between 90 to 95 second with a less frequency of 25.

Most of the persons visiting the site was between 30 to 35 years of age with a frequency of over 200 while the least age visiting the site is above 60 years followed by age below 20 years with a frequency below 25.

For the area income, the dataset is right skewed with majority of the persons visiing the site receive and income if between 60000 to 65000 with a frequency of over 150, while the least persons visiting the site have an income of between 10000 to 15000 with a frequency below 25.

There is a quite average internet usage of data, with those with most data usage falling between, 125 and 140 and 200 and 240 with a frequency of about 150.

Bivariate Analysis

Covariance

```
# assigning the daily time spent column to variable time spent time_spent <-
advert_data$Daily.Time.Spent.on.Site # assigning the age column to variable
age age <- advert_data$Age # finding the covariance cov(time_spent, age)
```

[1] -46.17415

```
# assigning the daily time spent column to variable time spent time_spent <-
advert_data$Daily.Time.Spent.on.Site # assigning the area income column to
variable income income <- advert_data$Area.Income
# finding the covariance cov(time_spent, income)
```

[1] 66130.81

```
# assigning the daily time spent column to variable time spent time_spent <-
advert_data$Daily.Time.Spent.on.Site
# assigning the daily time spent on site column to variable internet internet <-
advert_data$Daily.Internet.Usage
# finding the covariance cov(time_spent, internet)
```

[1] 360.9919

```
# assigning the age column to variable age age <-
advert_data$Age
# assigning the area income column to variable income income <-
advert_data$Area.Income
# finding the covariance cov(income, age)
```

[1] -21520.93

```
# assigning the age column to variable age age <-
advert_data$Age
# assigning the daily time spent on site column to variable internet internet <-
advert_data$Daily.Internet.Usage
# finding the covariance cov(internet, age)
```

[1] -141.6348

```
# assigning the daily time spent on site column to variable internet internet <-
advert_data$Daily.Internet.Usage
# assigning the area income column to variable income income <-
advert_data$Area.Income
# finding the covariance cov(income,
internet)
```

[1] 198762.5

Finding the correlation of the dataset *# assigning the daily time spent column to variable time spent* time_spent <- advert_data$Daily.Time.Spent.on.Site *# assigning the age column to variable age* age <- advert_data$Age *# finding the correlation* cor(time_spent, age)

[1] -0.3315133

```
# assigning the daily time spent column to variable time spent time_spent <-
advert_data$Daily.Time.Spent.on.Site # assigning the area income column to
variable income income <- advert_data$Area.Income # finding the correlation
cor(time_spent, income)
```

[1] 0.3109544

```
# assigning the daily time spent column to variable time spent time_spent <-
advert_data$Daily.Time.Spent.on.Site
# assigning the daily time spent on site column to variable internet internet <-
advert_data$Daily.Internet.Usage
# finding the correlation cor(time_spent, internet)
```

[1] 0.5186585

```
# assigning the age column to variable age age <-
advert_data$Age
# assigning the area income column to variable income income <-
advert_data$Area.Income # finding the correlation cor(income, age)
```

[1] -0.182605

```
# assigning the age column to variable age age <-
advert_data$Age
# assigning the daily time spent on site column to variable internet internet <-
advert_data$Daily.Internet.Usage
# finding the correlation cor(internet, age)
```

[1] -0.3672086

```
# assigning the daily time spent on site column to variable internet internet <-
advert_data$Daily.Internet.Usage
# assigning the area income column to variable income income <-
advert_data$Area.Income # finding the correlation cor(income,
internet)
```

[1] 0.3374955

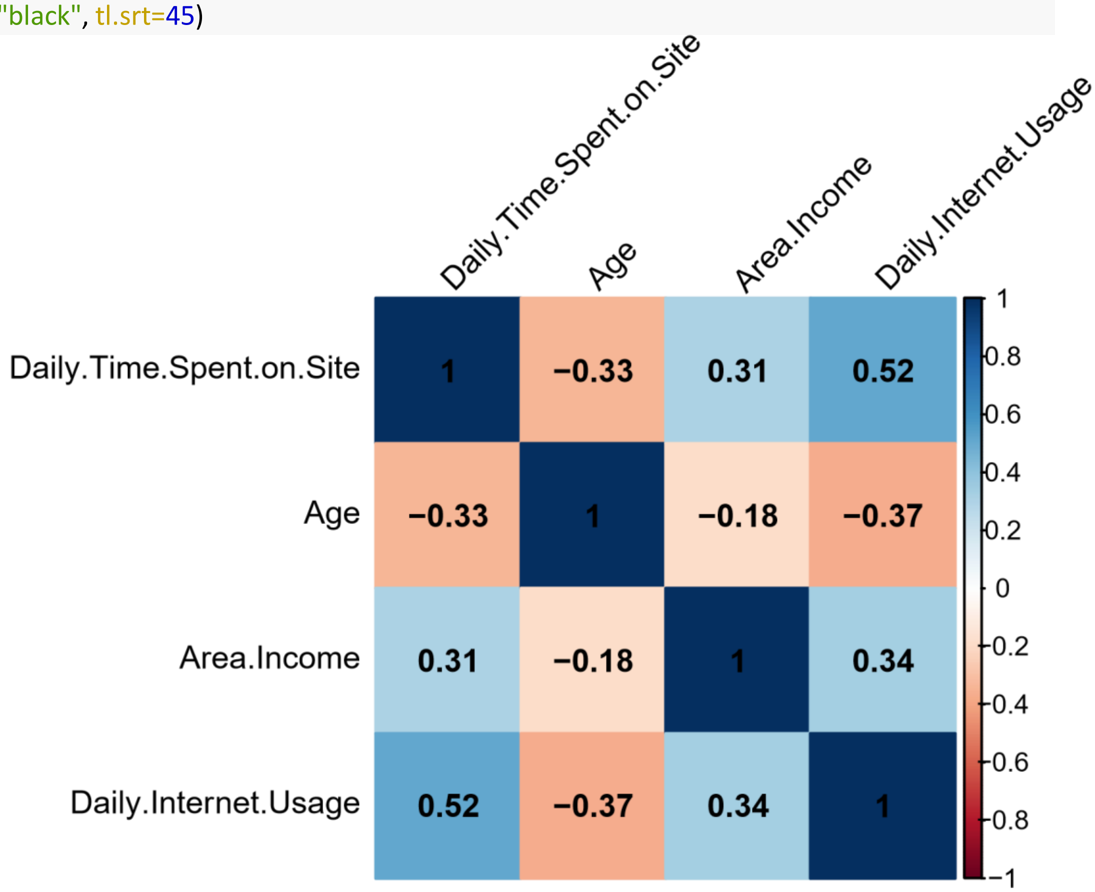
There is quite a slight positive and negative correlation between features time spent and age, time spent and income, income and age, internet and age, internet and income while a moderate positive correlation between time spent and internet of 0.51866.

Correlation plot

```
library(corrplot)
```

```
## corrplot 0.90 loaded
```

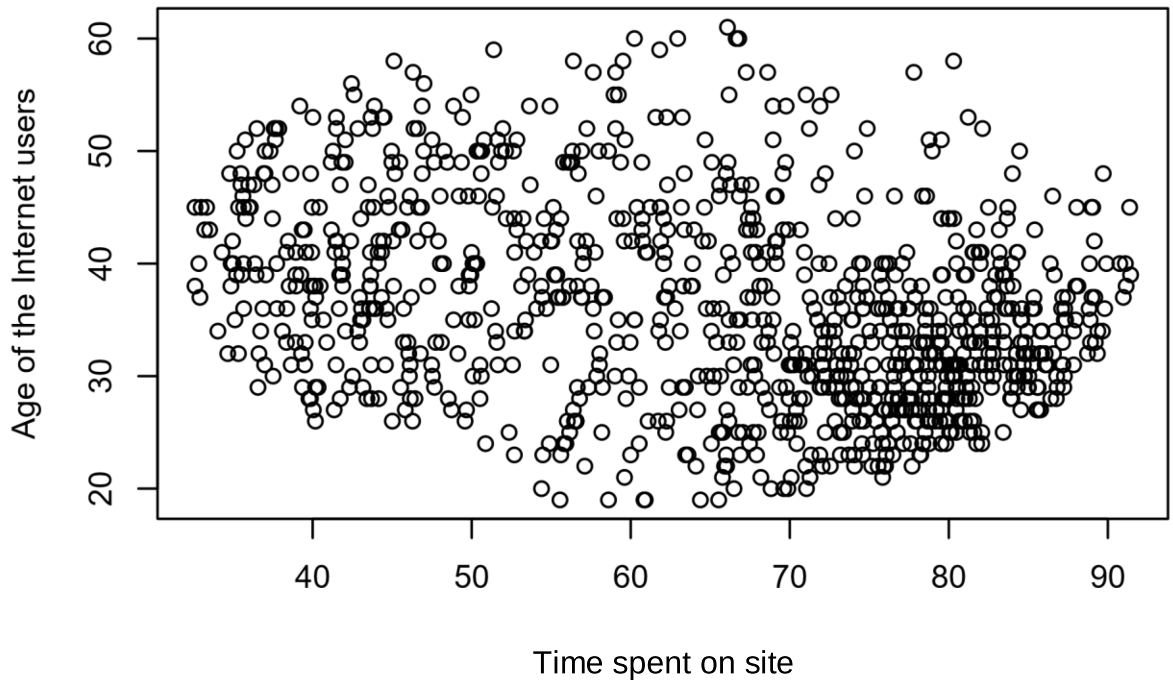
```
#Get the correlation matrix res =
cor(num)
#Plotting a correlation plot
corrplot(res, method="color",addCoef.col = "black",
         tl.col="black", tl.srt=45)
```



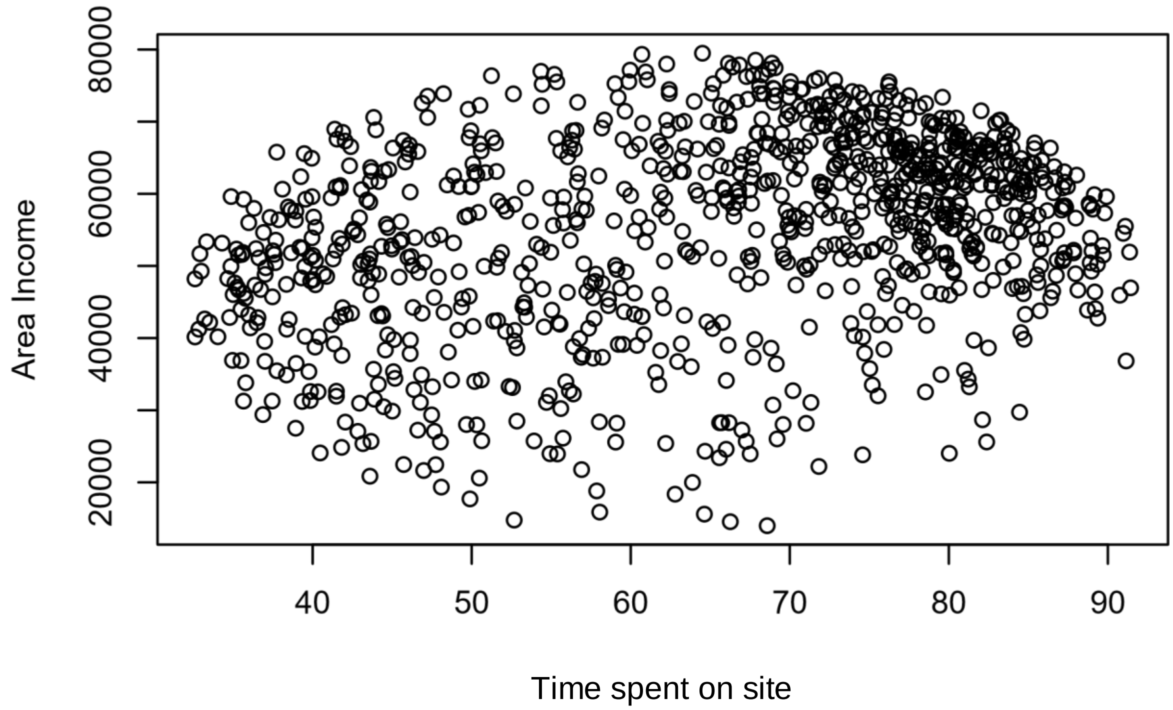
There is a fare correlation between amount spent on site and the Daily internet usage.

Scatterplots

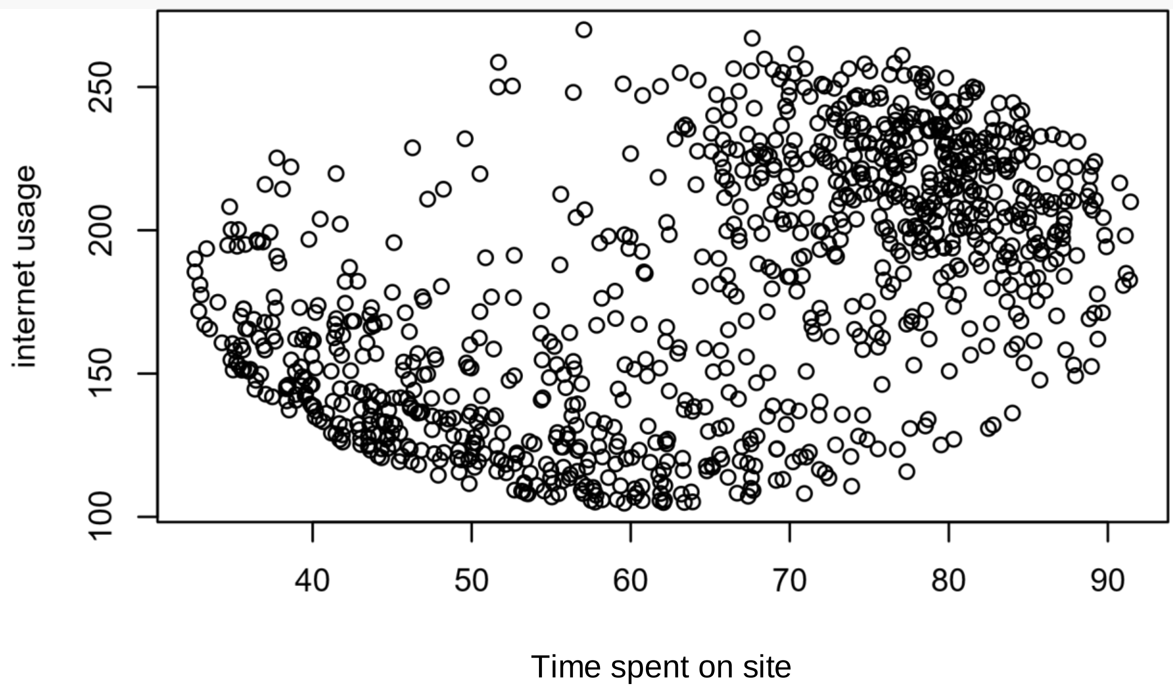
```
# assigning the daily time spent column to variable time spent time_spent <-
advert_data$Daily.Time.Spent.on.Site # assigning the age column to variable age age <- advert_data$Age #
plotting scatter plot plot(time_spent, age, xlab = "Time spent on site", ylab = "Age of the Internet users")
```



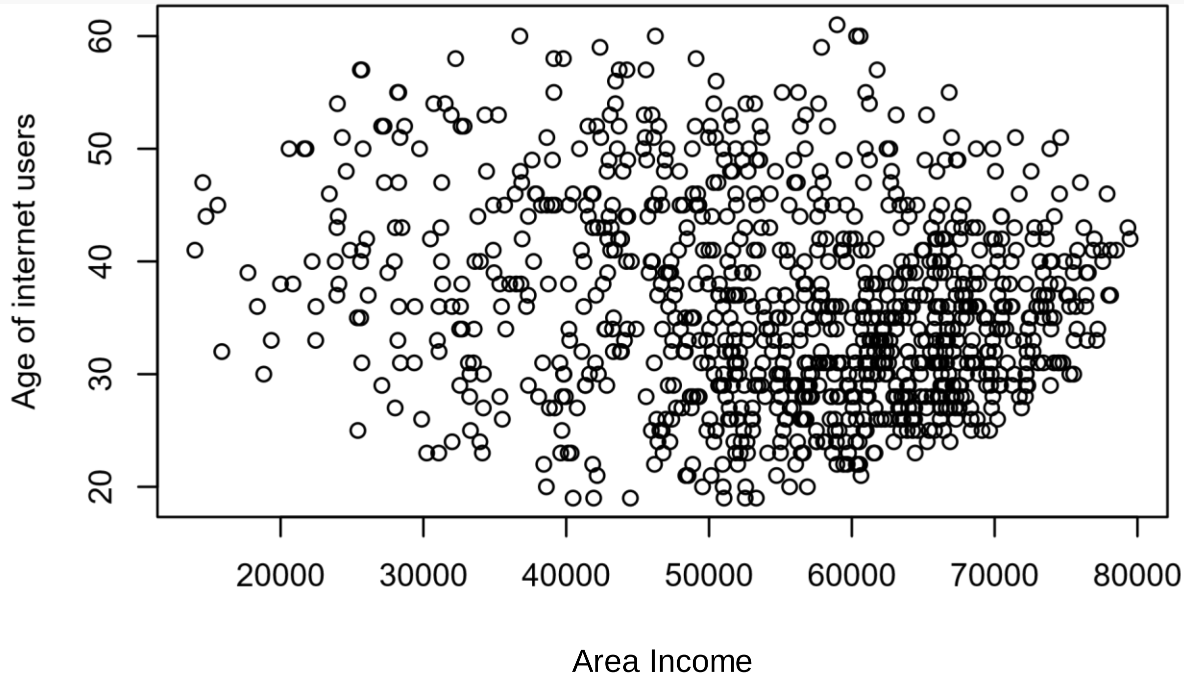

```
# assigning the daily time spent column to variable time spent time_spent <-
advert_data$Daily.Time.Spent.on.Site # assigning the area income column to
variable income income <- advert_data$Area.Income
# plotting scatter plot plot(time_spent, income, xlab = "Time spent on site",
ylab =
                                                                        "Area Income")
```



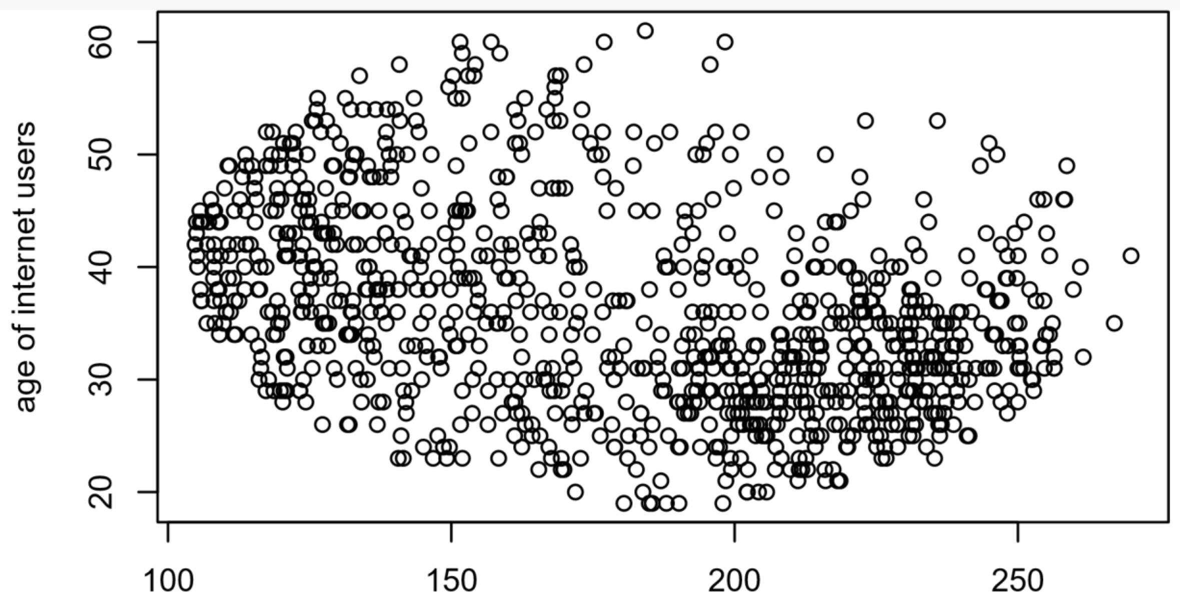

```
# assigning the daily time spent column to variable time spent time_spent <-
advert_data$Daily.Time.Spent.on.Site
# assigning the daily time spent on site column to variable internet internet <-
advert_data$Daily.Internet.Usage
# plotting scatter plot plot(time_spent, internet, xlab = "Time spent on site", ylab = "internet usage")
```



```
# assigning the age column to variable age age <-
advert_data$Age
# assigning the area income column to variable income income <-
advert_data$Area.Income
# plotting scatter plot plot(income, age, xlab = "Area Income", ylab = "Age of internet
users")
```

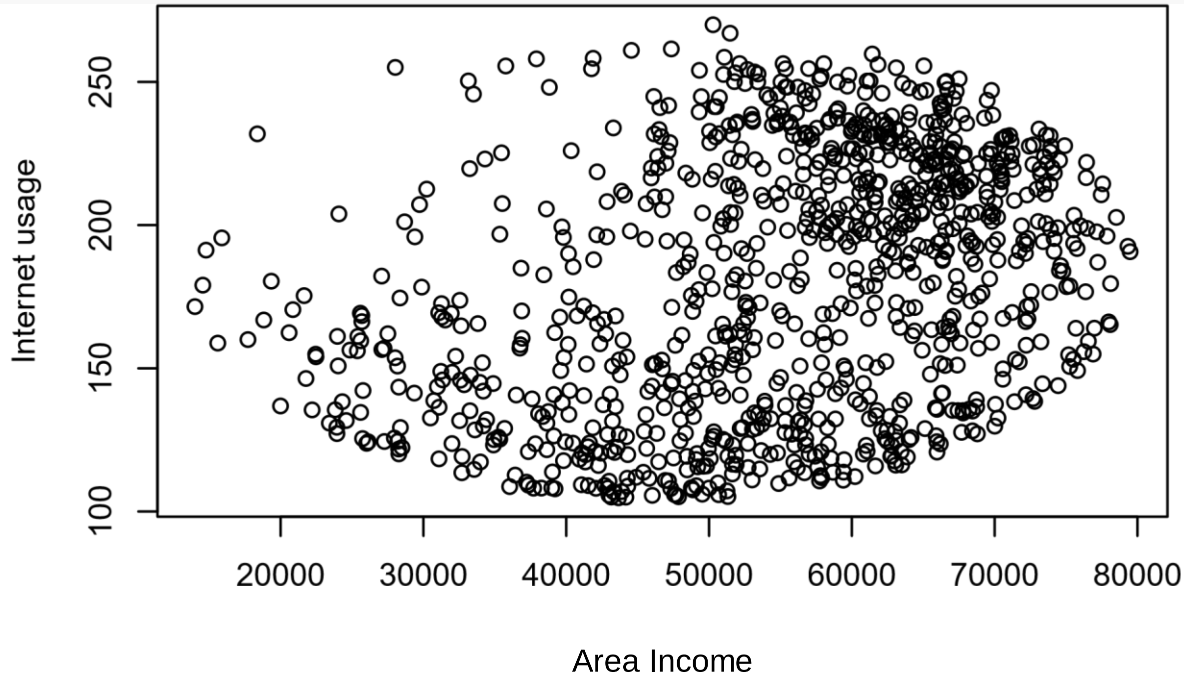



```
# assigning the age column to variable age age <-
advert_data$Age
# assigning the daily time spent on site column to variable internet internet <-
advert_data$Daily.Internet.Usage
# plotting scatter plot plot(internet, age, xlab = "Internet usage", ylab = "age of internet users")
```



Internet usage

```
# assigning the daily time spent on site column to variable internet internet <-
advert_data$Daily.Internet.Usage
# assigning the area income column to variable income income <-
advert_data$Area.Income
# plotting scatter plot plot(income, internet, xlab = "Area Income", ylab = "Internet
usage")
```



> In a scatter plot between age of the internet users and time spent on site, the highest time spent on site of between 70 to 90 seconds is spent by those of the age between 25 and 40, these is probably because the target product is more interesting to the age bracket.

In the second plot we find that people who earn an income of between 60000 to 75000 spent more time on the site than those receiving lower income of between 10000 and 20000.

In the third plot, those with lower internet usage of 150 mbs, spent less time on the site below 60 seconds, while those spending over 200 mbs spend more time on the site with over 70 seconds to 90 seconds.

In the fourth plot, most of the people of age between 30 and 40 years of age, with an income between 50000 and 70000 spent more time on the site,

There is a fair distribution of age in terms of internet usage, majority of the persons of the age between 35 and 50 spent between 100 and 150, while most those who spent internet between 200 and 250 mbs are of age between 20 and 35.

Majority of the of the persons who use internet earn an income of above 50000, interms of data usage in relation to income there is a fair distribution.

Implementing the solution Using K-NN

```
# Fitting a summary of our dataset summary(advert_data)
```

```
## 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
##      Male            Country            Clicked.on.Ad
## Min.   :0.000   Length:1000        Min.   :0.0
## 1st Qu.:0.000   Class :character   1st Qu.:0.0
## Median :0.000   Mode  :character   Median :0.5
## Mean   :0.481                      Mean   :0.5
## 3rd Qu.:1.000                      3rd Qu.:1.0
## Max.   :1.000                      Max.   :1.0
```

```
# Label encoding Country column advert_data$Country<-
as.integer(as.factor(advert_data$Country)) # Label encoding traffic data
advert_data$Clicked.on.Ad<-as.factor(as.factor(advert_data$Clicked.on.Ad))
summary(advert_data)
```

```
## 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
##      Male            Country       Clicked.on.Ad
## Min.   :0.000   Min.   : 1.0    0:500
```

```
## 1st Qu.:0.000        1st Qu.: 55.0     1:500
## Median :0.000        Median :114.5
## Mean   :0.481    Mean   :116.4
## 3rd Qu.:1.000        3rd Qu.:178.0
## Max.   :1.000    Max.   :237.0
```

Randomizing the data

```
set.seed(123456) # Randomizing the rows, creates a
uniform distribution of 1000
random <- runif(1000)
advert_random <- advert_data[order(random),] #
Selecting the first 4 rows from iris_random
head(advert_random)
```

```
##          Daily.Time.Spent.on.Site Age Area.Income Daily.Internet.Usage Male Country
## 621                     81.75 24    52656.13               190.08    1      39
## 400                     77.29 27    66265.34               201.24    1     220
## 82                      73.46 28    65653.47               222.75    1     154
## 798                     77.05 34    65756.36               236.08    0     147
## 914                     87.46 37    61009.10               211.56    1      92
## 67                      63.89 40    51317.33               105.22    0     176
##     Clicked.on.Ad
## 621             0
## 400             0
## 82              0
## 798             0
## 914             0
## 67              1
```

Normalizing the data

```
# Normalizing the numerical variables of the data set. Normalizing the numerical values is really effect # as it provide
0 to 1 which corresponds to min value to the max value of the data colum # We define a normal function which will no
values according to its minimum value and
normal <- function(x) (
  return( ((x - min(x)) /(max(x)-min(x))) )
) normal(1:7)
```

```
## [1] 0.0000000 0.1666667 0.3333333 0.5000000 0.6666667 0.8333333 1.0000000
```

```
advert_new <- as.data.frame(lapply(advert_random[,-7], normal)) summary(advert_new)
```

```
## Daily.Time.Spent.on.Site      Age              Area.Income
## Min.   :0.0000          Min.   :0.0000    Min.   :0.0000
## 1st Qu.:0.3189          1st Qu.:0.2381    1st Qu.:0.5044
## Median :0.6054          Median :0.3810    Median :0.6568
## Mean   :0.5507          Mean   :0.4050    Mean   :0.6261
## 3rd Qu.:0.7810          3rd Qu.:0.5476    3rd Qu.:0.7860
```

```
## Max.          :1.0000                     Max.       :1.0000       Max.         :1.0000
## Daily.Internet.Usage                      Male                Country
## Min.          :0.0000             Min.       :0.000       Min.        :0.0000
## 1st Qu.:0.2061                    1st Qu.:0.000         1st Qu.:0.2288
## Median :0.4743                    Median :0.000         Median :0.4809
## Mean          :0.4554             Mean       :0.481       Mean        :0.4890
## 3rd Qu.:0.6902                    3rd Qu.:1.000         3rd Qu.:0.7500
## Max.          :1.0000             Max.       :1.000       Max.        :1.0000
```

Training our data into training and testing

```
# Lets now create test and train data sets
train <- advert_new[1:750,] test <-
advert_new[751:1000,] train_sp <-
advert_random[1:750,5] test_sp <-
advert_random[751:1000,5]
```

Using knn to make classification

```
# Now we can use the K-NN algorithm. Lets call the "class" package which contains the K-NN algorithm. # We then ha
'k' value which is no of nearest neighbours(NN) to look for # in order to classify the test data point.
# Lets build a model on it; cl is the class of the training data set and k is the no of neighbours to lo # in order to classif
library(class) require(class) model <- knn(train= train,test=test, ,cl=
train_sp,k=13) table(factor(model))
```

```
##
##    0    1
## 133 117
```

Creating aconfusion matrix and checking the accuracy of the model

```
# Creating a confusin matrix confm <-
table(test_sp,model) # Checking the
accuracy
accuracy <- function(x){sum(diag(x)/(sum(rowSums(x)))) * 100} accuracy(confm)
```

```
## [1] 100
```

Decision Tree

Loading the libraries

```
# Load the party package. It will automatically load other
# dependent packages.
library(party)
```

```
## Loading required package: grid

## Loading required package: mvtnorm

## Loading required package: modeltools

## Loading required package: stats4
```

```
## Loading required package: strucchange

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

## Loading required package: sandwich
```

```
# Print some records from data setting the Clicked on ad as our target variable. head(advert_data)
```

```
##   Daily.Time.Spent.on.Site Age Area.Income Daily.Internet.Usage Male Country
## 1                    68.95 35    61833.90               256.09    0     216
## 2                    80.23 31    68441.85               193.77    1     148
## 3                    69.47 26    59785.94               236.50    0     185
## 4                    74.15 29    54806.18               245.89    1     104
## 5                    68.37 35    73889.99               225.58    0      97
## 6                    59.99 23    59761.56               226.74    1     159
##   Clicked.on.Ad
## 1             0
## 2             0
## 3             0
## 4             0
## 5             0
## 6             0
```

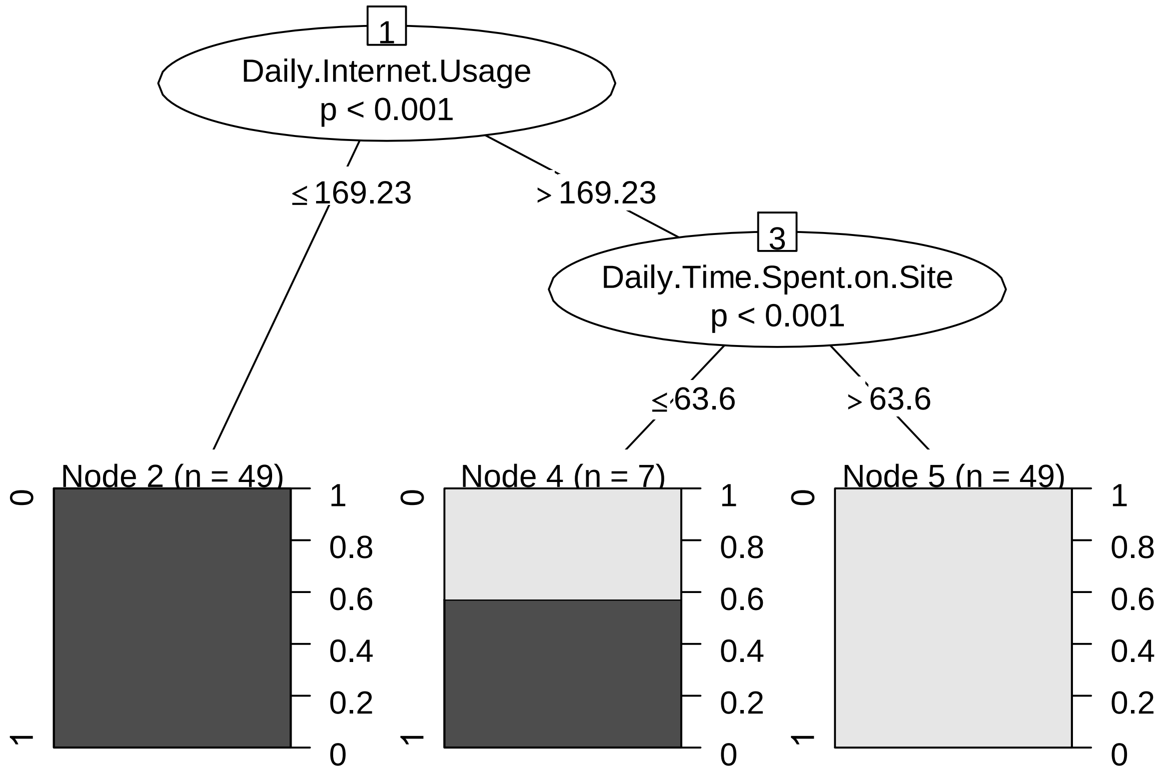
Creating a tree

```
# Creating the input data frame. input.dat <-
advert_data[c(1:105),] # Creating the tree.
  output.tree <- ctree(
  Clicked.on.Ad ~ Daily.Time.Spent.on.Site + Age + Area.Income + Daily.Internet.Usage + data = input.dat)
```

Male +
Country,

Plotting the tree

```
# Plotting the tree. plot(output.tree)
```



> For daily internet usage we have those spent <-169.23 being node 2 with n=49 and those who spent >= 169.23 with a probability of p < 0.001, of these who spent over 169.23 daily bundles we have time spent on site being <= 63.6 being node 4 with n=7 or >= 63.6 being node 5 with n=49

Challenging our solution

Challeng our solution by using different value of k

Using knn to make classification

```
# Now we can use the K-NN algorithm. Lets call the "class" package which contains the K-NN algorithm. # We then h
'k' value which is no of nearest neighbours(NN) to look for # in order to classify the test data point.
# Lets build a model on it; cl is the class of the training data set and k is the no of neighbours to lo # in order to classi
model <- knn(train= train,test=test, ,cl= train_sp,k=50) table(factor(model))
```

```
##
##   0   1
## 133 117
```

Creating aconfusion matrix and checking the accuracy of the model

```
# Creating a confusin matrix confm <-
table(test_sp,model) # Checking the
accuracy
accuracy <- function(x){sum(diag(x)/(sum(rowSums(x)))) * 100} accuracy(confm)
```

```
## [1] 100
```