Moringa_Dsc14_Core_Week12_IP_Rprogramming_Jonah_Okiru_05_2022

Jonah okiru

2022-05-27

##1. Define question.

#a).Specifying the question.

To help an online cryptography entrepreneur to identify the individuals who are most likely to click on her advert.

#b). The metric of success.

To identify individual who are most likely to click on an advert with an accuracy score of 90%

#c). The context

With the ever growing in the technology globally and internally, it has make it possible for entrepreneurs to market their products on global scale while at the comfort of their homes. These has been successful, thanks to platforms such as blogs.These platforms attracts viewers and readers of the blogs content from all over the world and as they do so , adverts of various types pops up after certain time intervals, Some of the visitors of these platforms click on these adverts that pops up on the platform as they consume the content of the platform. The owners of these platforms always would like to know the type of individuals who are visitors to the platforms and are likely to click on the pop up adverts so that they could promote the products which target those individuals at their platforms.

#d). recording Exprimental design

Hypothesis: To identify individual who visit the blog platform and the individual is most likely to click on the pop up adverts.

X-Axis: Age, gender, income area, city and country.

Y-Axis: Click on advert.

Experimental set up: I identify the individual who visits the blog platform as either most likely to click on the advert or not.

Design of the experiment: Analyze the 1000 records from the previous visitors of the blog platform.

Sample size: 1000 records from the previous visitors of the blog.

```
#Load the data
library("data.table")
advertising <- fread("http://bit.ly/IPAdvertisingData")
#Print the first six records of the data
head(advertising)
```

```
##    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
```

```
#Drop unnecessary columns.
#select the variables with names of columns to be retained
age <- advertising$Age
gender <- advertising$Male
Click_on_advert <- advertising$`Clicked on Ad`
area_income <- advertising$`Area Income`
city <- advertising$City
country <- advertising$Country

#Create the data table from the selected variables and assigned it to df
df <- data.table(age, gender, area_income, city, country, Click_on_advert)

#Print the first six records of the data
head(df)
```

```
##    age gender area_income           city    country Click_on_advert
## 1:  35      0    61833.90    Wrightburgh    Tunisia               0
## 2:  31      1    68441.85      West Jodi      Nauru               0
## 3:  26      0    59785.94       Davidton San Marino               0
## 4:  29      1    54806.18 West Terrifurt      Italy               0
## 5:  35      0    73889.99   South Manuel    Iceland               0
## 6:  23      1    59761.56      Jamieberg     Norway               0
```

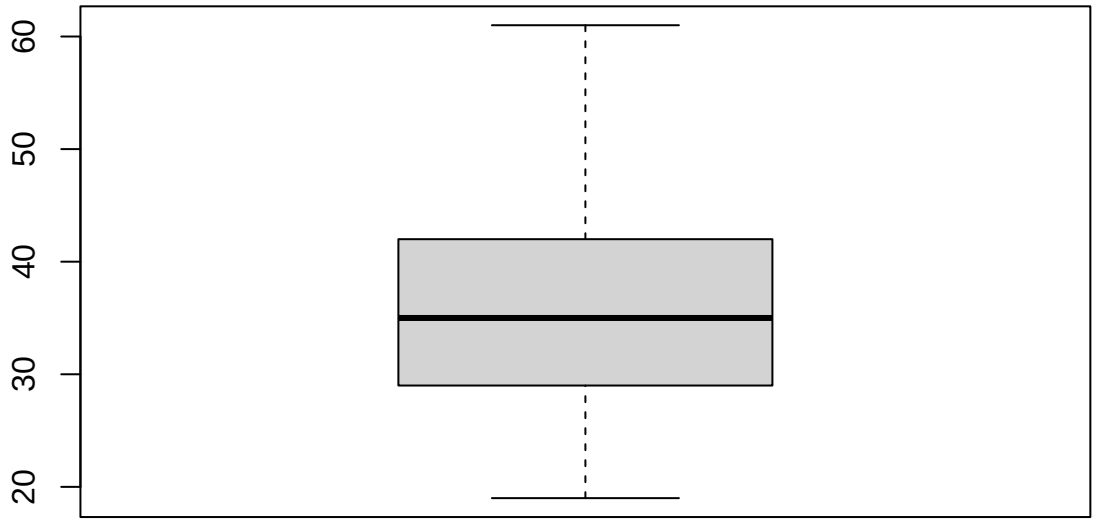
The other columns were dropped in the cell above since they're of no importance in the further analysis of the data hence had to be dropped.

```
#Check for the missing value in the
#Find the total missing value in each column of the df dataset.
colSums(is.na(df))
```

```
##             age          gender     area_income            city         country
##               0               0               0               0               0
## Click_on_advert
##               0
```

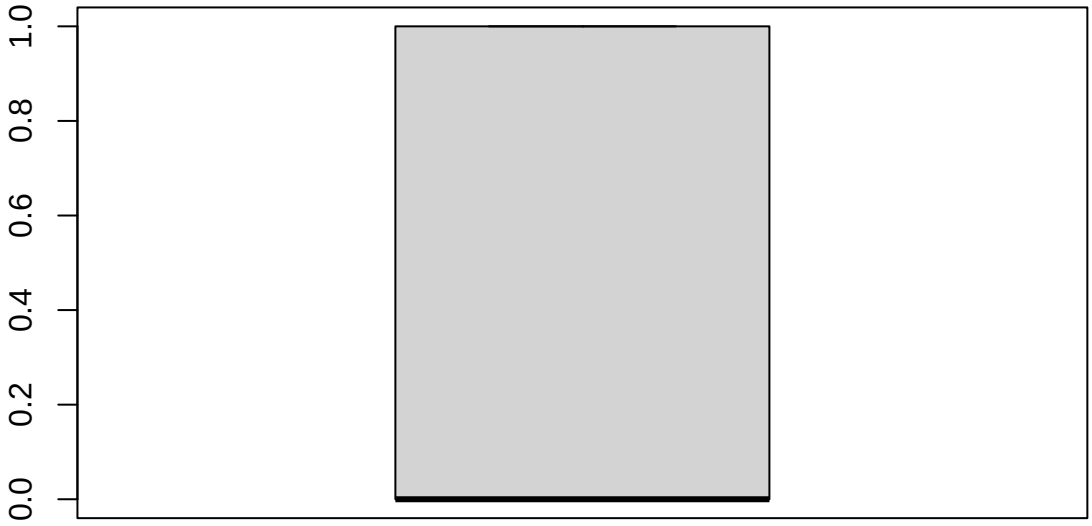
All the column of the dataset has no missing values.

```
#Check for the outliers within the dataset numeric columns using boxplot.
#Check for the existence of outliers in the age column
age <- df$age
#The box plot
boxplot(age)
```



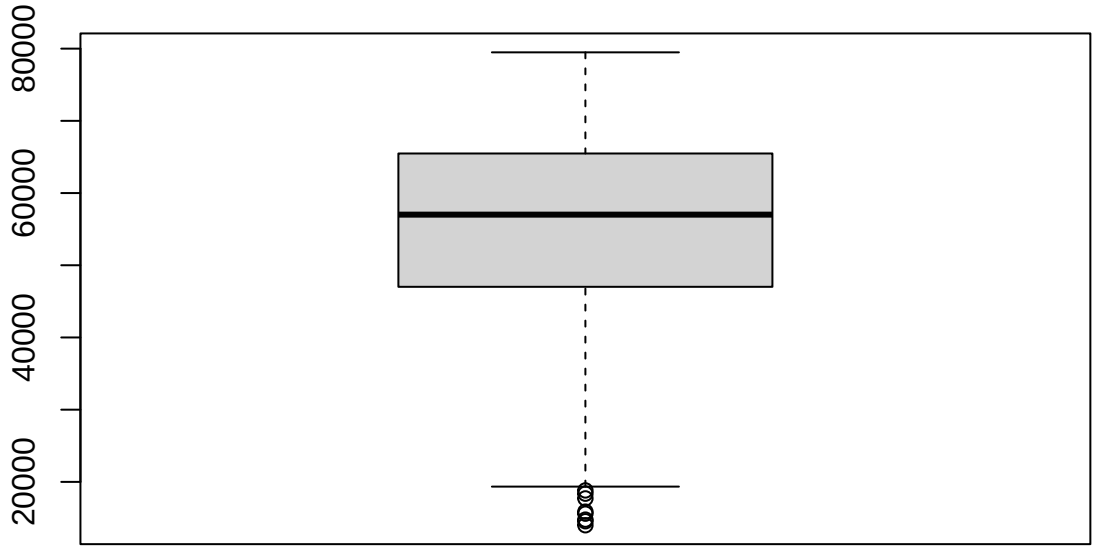
From the above age column boxplot, we note that the age column data has no outliers.

```
#Check for the existence of outliers in the gender column
Gender <- df$gender
#Boxplot of gender
boxplot(Gender)
```



From the boxplot above , we conclude that there is no outliers in the gender column. This is due to values in the gender columns being categorical so they are either class '0' or class one '1'

```
#Check for the existence of outliers in the income column
area_income <- df$area_income
#Box plot
boxplot(area_income)
```



The existence of outliers in the income column is due disparities of income level among countries and cities.

```
#Check for duplicates in the dataset
duplicate_rows <- df[duplicated(df),]
#print the duplicate rows
duplicate_rows
```

```
## Empty data.table (0 rows and 6 cols): age,gender,area_income,city,country,Click_on_advert
```

There were no duplicate records in the dataset.

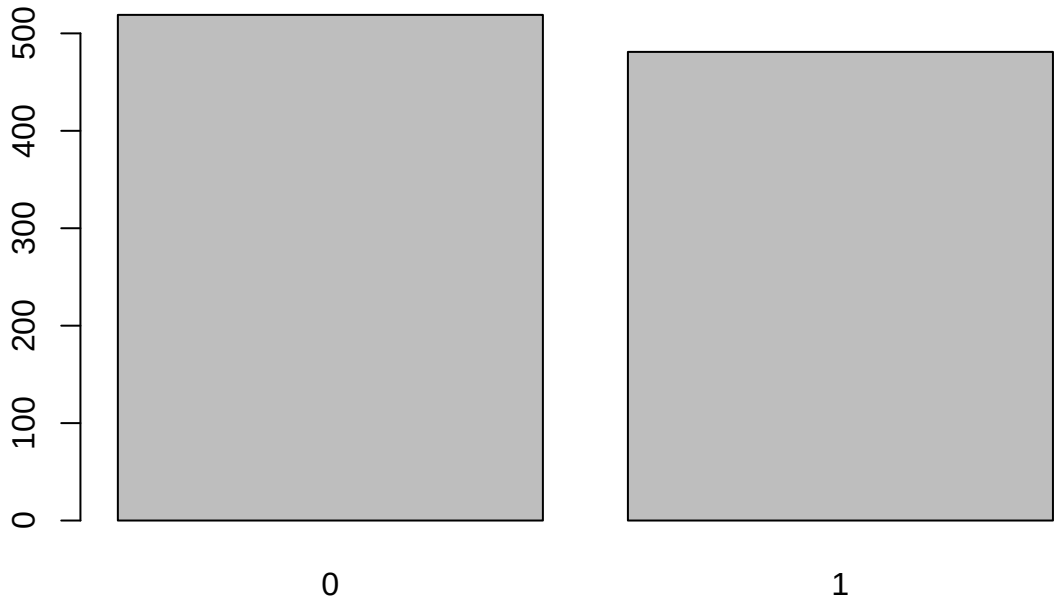
```
#Univariate analysis
#Check for the descriptive statistics of the data
library(descr)
descr(df)
```

```
##       age             gender       area_income        city
##  Min.   :19.00   Min.   :0.000   Min.   :13996   Length:1000
##  1st Qu.:29.00   1st Qu.:0.000   1st Qu.:47032   Class :character
##  Median :35.00   Median :0.000   Median :57012   Mode  :character
##  Mean   :36.01   Mean   :0.481   Mean   :55000
##  3rd Qu.:42.00   3rd Qu.:1.000   3rd Qu.:65471
##  Max.   :61.00   Max.   :1.000   Max.   :79485
##    country          Click_on_advert
##  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
```

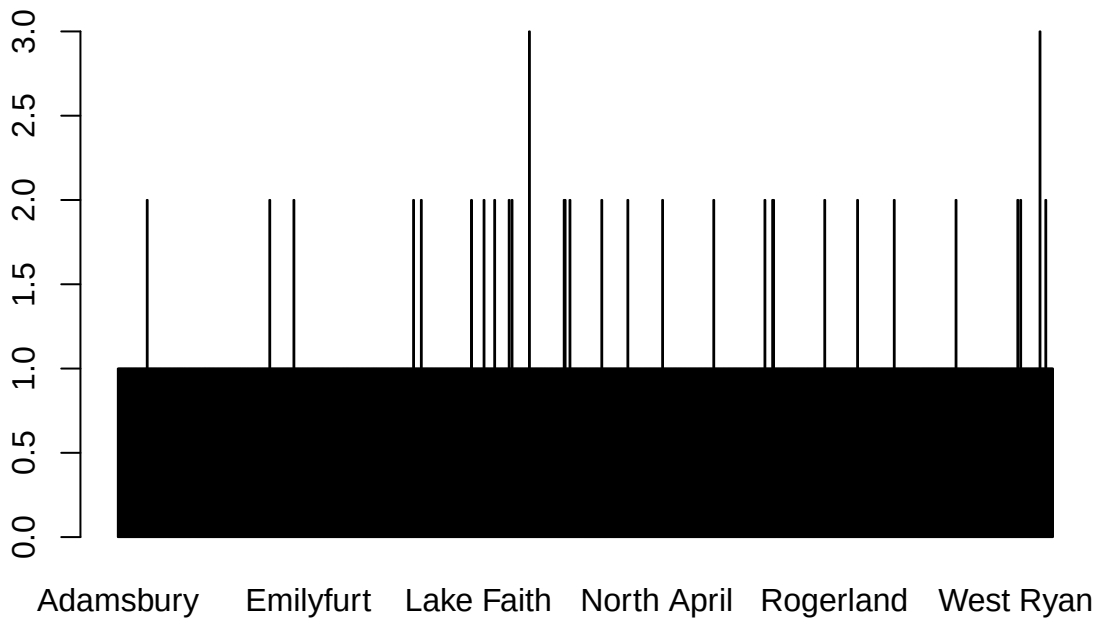
From the above descriptive statistics of the columns, we observe that the minmum age of was 19 years, maximum age was 61 years, the mean age is 36 years and the median age is 35 years. Also we note that the minimum area income is 13996, the mean area income is 55000, the median is 57012 and the maximum is 79485

```
#Barplots of categorical class columns
#Bar plot of gender
#Frequency table of gender column
gender_freq_table <- table(gender)
#Barplot of gender
barplot(gender_freq_table)
```

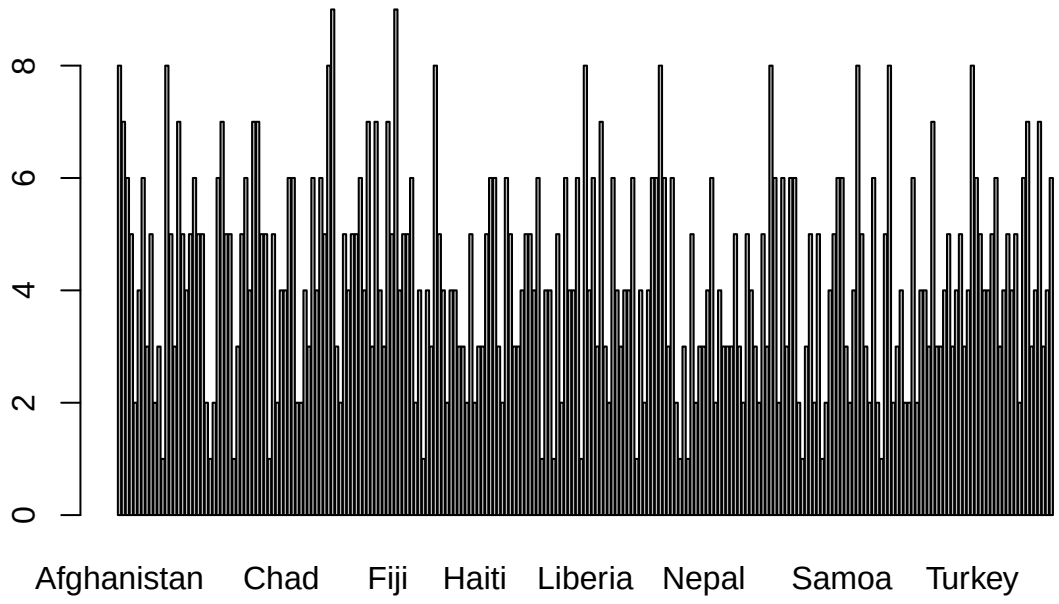


From the bar chart we note the number of female gender who click on advert is slightly higher compared to the male gender

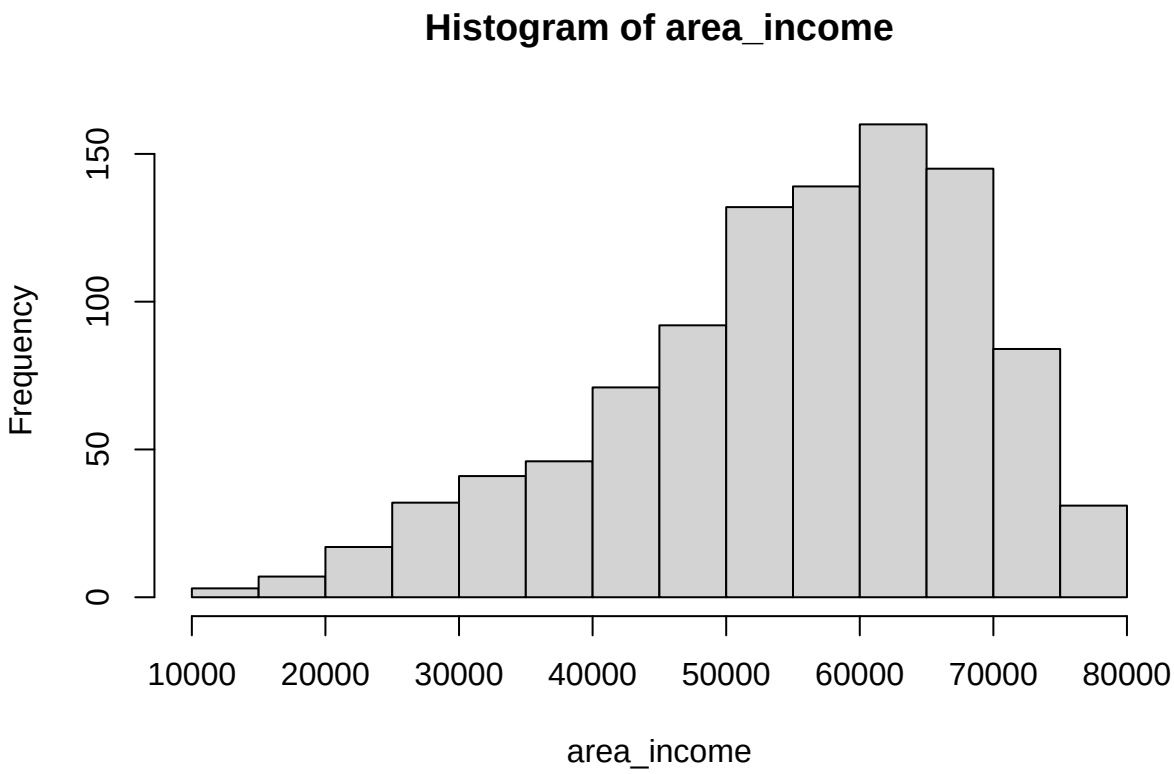
```
#Barplot of city
#Frequency table of the city
city_freq_table <- table(city)
#Barplot of the city
barplot(city_freq_table)
```




```r
#Bar plot of the country
#Country frequency table
country_frequencytable <- table(country)
#Barplot of country column
barplot(country_frequencytable)
```

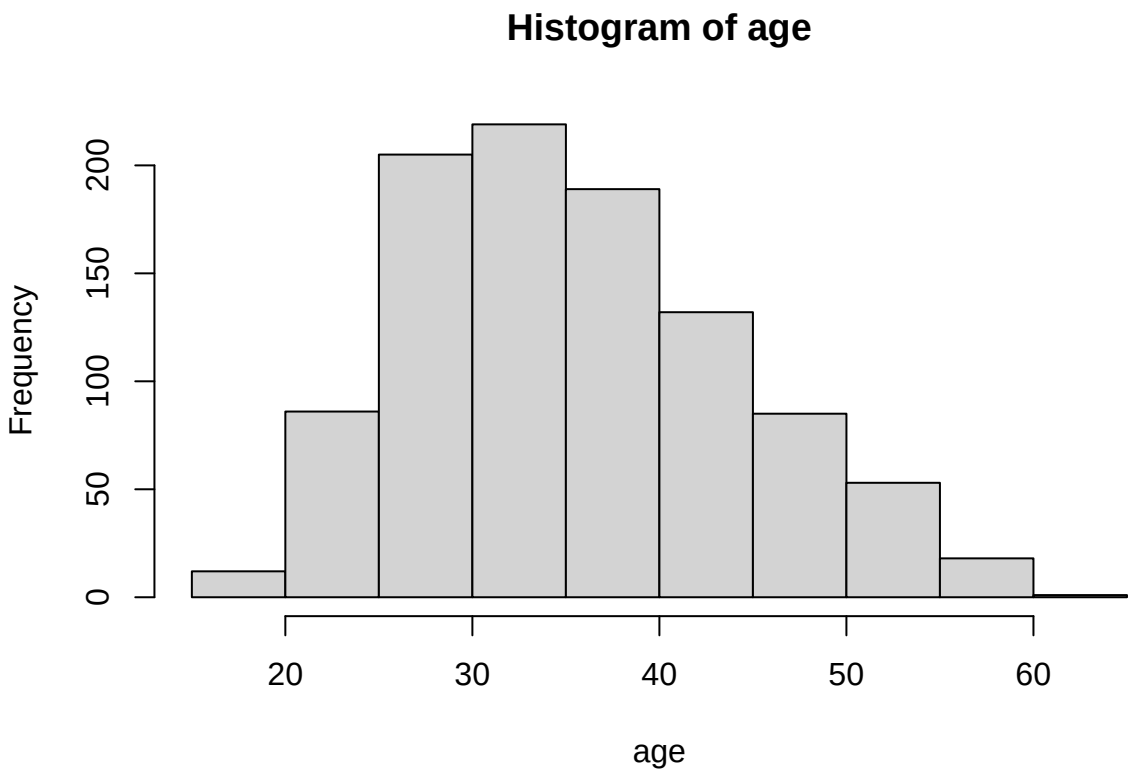



```
#Plot the Histograms
#The histogram of area income
area_income <- df$area_income
hist(area_income)
```



From the histogram above, it's bit skewed the left these implies that majority of the people have area income less than the mean area income.

```
#Histogram of age column
age <- df$age
hist(age)
```



From the histogram it's noted that age histogram is skewed to the right these implies that the age of majority of the participants in the datasets was above the mean age.

```
#Bivariate and multivariate analysis
#Covariance between age and click on advert
#Age variable
age <- df$age
#click on the advert variable
Click_on_advert <- df$Click_on_advert
#covariance between age and click on advert
cov(age, Click_on_advert)
```

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

The above covariance of 2.164665 implies that the relationship between the age and click on advert is a positive relationship. These means that a when the age increases the the click on advert also increases and vice versa.

```
#Covariance between gender and click on advert
gender <- df$gender
#Covariance btwn gender and click on advert
cov(gender, Click_on_advert)
```

```
## [1] -0.00950951
```

The covariance of -0.00950951 above indicates there is very slight negative relationship between the gender and the click on advert. These implies there is no a noticeable effect when the genders changes on the click on advert.

```
#Covariance between area income and click on advert
area_income <- df$area_income
cov(area_income, Click_on_advert)
```

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

The covariance above of -3195.989 implies that the relationship between area income and click on advert is a negative relationship. These implies that a slight decrease in the income will lead to a huge increase on the the click on advert.

```
#Correlation between gender and click on advert
gender <- df$gender
#Covariance btw gender and click on advert
cor(gender, Click_on_advert)
```

```
## [1] -0.03802747
```

The correlation above of -0.03802747 , indicates that the correlation between the gender and click on advert is very weak negative correlation.

```
#Correlation between gender and click on advert
age <- df$age
#Covariance btwn gender and click on advert
cor(age, Click_on_advert)
```

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

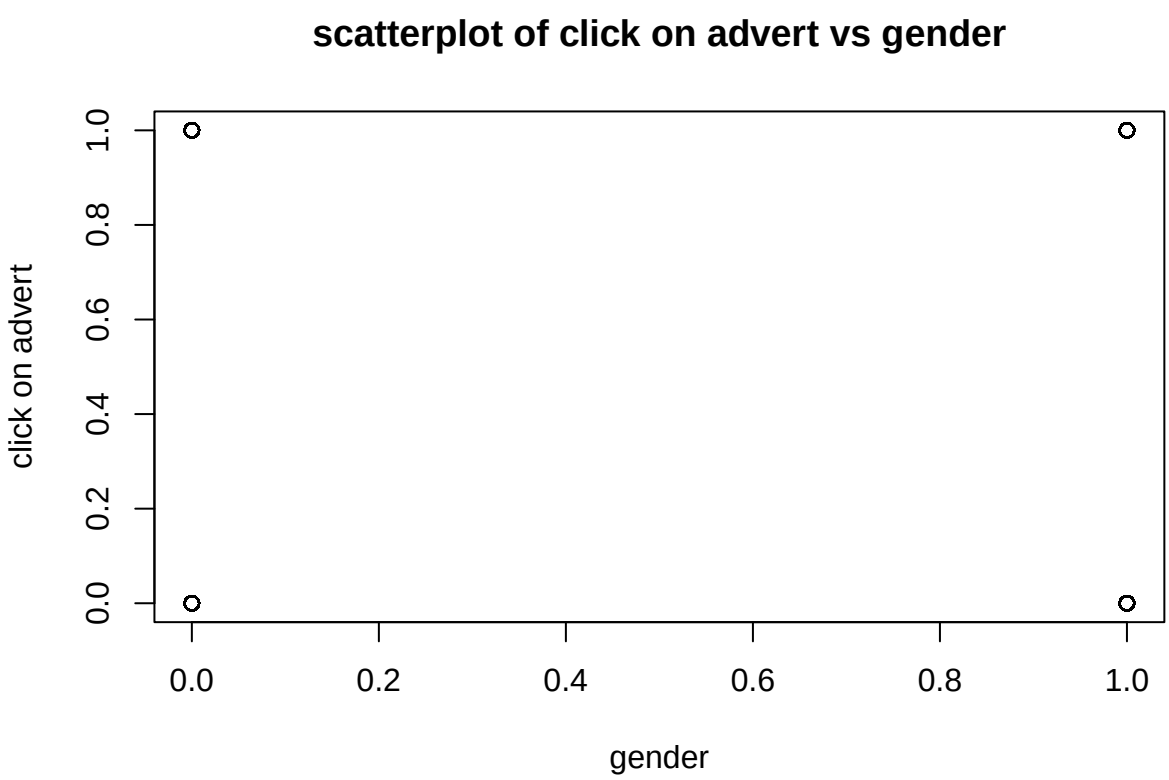
The correlation above 0.4925313, implies that the correlation between age and click on advert is a moderate positive correlation. These implies an increase in age leads to an increase on the click on advert at the same proportion.

```
#Correlation between area income and click on advert
area_income <- df$area_income
cor(area_income, Click_on_advert)
```

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

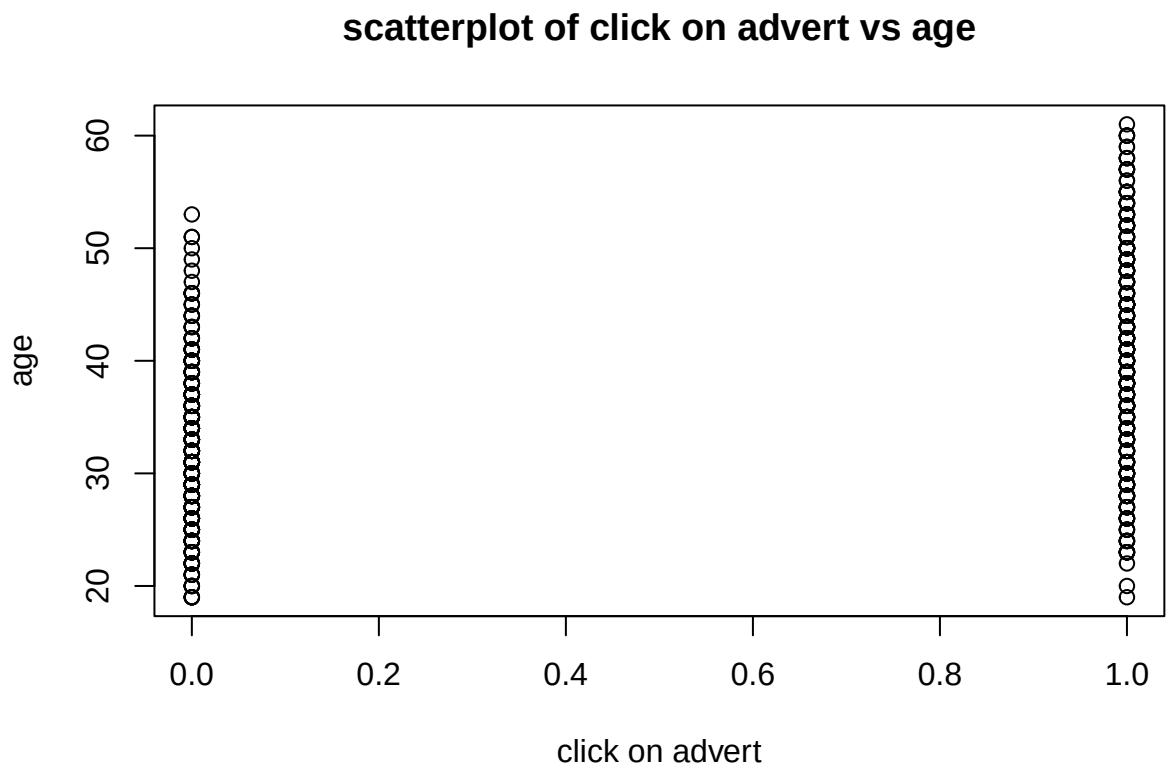
The correlation above of -0.4762546 is a moderate negative correlation. These implies that a decrease in area income leads to proportionate increase in the click on the advert.

```
#Scatter plot between gender and click on advert
plot(Click_on_advert, gender, xlab=("gender"), ylab =("click on advert"), main = "scatterplot of click
```



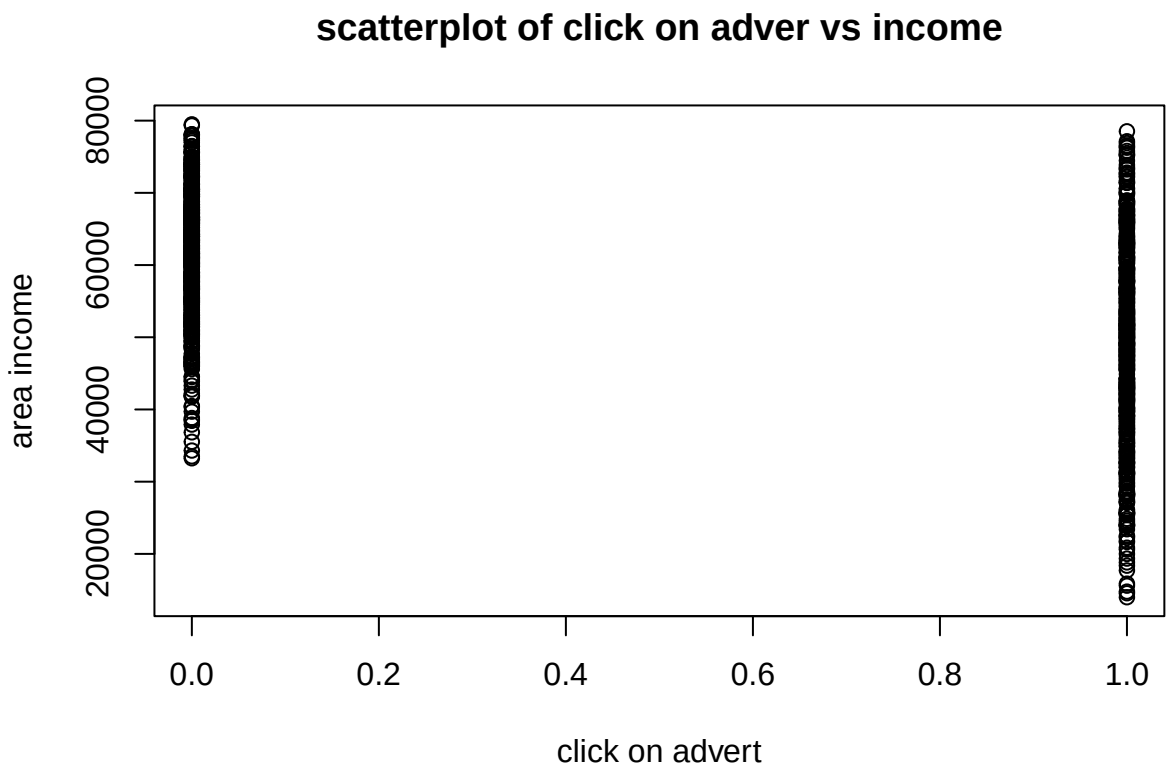
The scatter plot above between the gender and the click on advert implies that there's no correlation between the gender and click on the advert. A change in gender will not lead to change on the click on advert and vice versa.

```
#Scatter plot btwn age and click on advert
plot(Click_on_advert, age, xlab = "click on advert", ylab = "age", main = "scatterplot of click on adve
```



The scatter plot above between age and click on advert shows there's a moderate positive relationship between gender and click on advert. These implies a positive change in age will also leads to a proportionate increase on the clicks on advert.

```
#Scatter plot between area income and click on advert
plot(Click_on_advert, area_income, xlab="click on advert", ylab = "area income", main= "scatterplot of
```



The scatter plot above between the area income and click on advert indicates the relationship between the area income and click on advert is a moderate negative relationship. These implies that positive change in area income will lead to a negative change in the click on the advert.

The Modelling(Unsupervised learning)

```
#Preview the original data
head(advertising)
```

```
##    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
```

```
#Convert timestamp column from s3: POSIXct to as.POSIXlt
library(lubridate)
```

```
##
## Attaching package: 'lubridate'
```

```
## The following objects are masked from 'package:data.table':
##
##     hour, isoweek, mday, minute, month, quarter, second, wday, week,
##     yday, year
```

```
## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
Timestamp1 <- as.POSIXlt(advertising$Timestamp)
```

The Time stamp is convert from POSIXCT to as.POSIXlt, so that we could able to split it into date and time of the day.

```
#Rename the columns
daily_time_spent_on_site <- advertising$`Daily Time Spent on Site`
age <- advertising$Age
gender <- advertising$Male
Click_on_advert <- advertising$`Clicked on Ad`
area_income <- advertising$`Area Income`
city <- advertising$City
country <- advertising$Country
daily_internet_usage <- advertising$`Daily Internet Usage`
ad_topic_line <- advertising$`Ad Topic Line`
head(ad_topic_line)
```

```
## [1] "Cloned 5thgeneration orchestration"
## [2] "Monitored national standardization"
## [3] "Organic bottom-line service-desk"
## [4] "Triple-buffered reciprocal time-frame"
## [5] "Robust logistical utilization"
## [6] "Sharable client-driven software"
```

The column are named so that to replace the space in the column that contains more than one word.

```
#Split the timestamp column into sec, min, hour mday, month, year zone
sec <- Timestamp1$sec
min <- Timestamp1$min
hour <- Timestamp1$hour
month_day <- Timestamp1$mday
month <- Timestamp1$mon
```

The time stamp column is broken down into the columns of sec, min, hour, month day, month and year

```
#Combine the rename columns and broken down columns into one dataframe
df_modelling <- data.frame(daily_time_spent_on_site, age, gender, area_income, city, country, daily_inte
                           sec, min, hour, month_day, month, Click_on_advert)
#preview the first six rows of the dataset
head(df_modelling)
```

```
##   daily_time_spent_on_site age gender area_income           city    country
## 1                    68.95  35      0    61833.90    Wrightburgh    Tunisia
## 2                    80.23  31      1    68441.85      West Jodi      Nauru
## 3                    69.47  26      0    59785.94       Davidton San Marino
## 4                    74.15  29      1    54806.18 West Terrifurt      Italy
## 5                    68.37  35      0    73889.99   South Manuel    Iceland
## 6                    59.99  23      1    59761.56      Jamieberg     Norway
##   daily_internet_usage                          ad_topic_line sec min hour
## 1               256.09    Cloned 5thgeneration orchestration  11  53    0
## 2               193.77    Monitored national standardization   2  39    1
## 3               236.50      Organic bottom-line service-desk  42  35   20
## 4               245.89 Triple-buffered reciprocal time-frame  19  31    2
## 5               225.58         Robust logistical utilization  18  36    3
## 6               226.74       Sharable client-driven software  17  30   14
##   month_day month Click_on_advert
## 1        27     2               0
## 2         4     3               0
## 3        13     2               0
## 4        10     0               0
## 5         3     5               0
## 6        19     4               0
```

The columns are combined together to for a data frame which helps for the easy review of the dataset.

```
#Label encode the categorical variables
#Load the library
library(superml)
```

```
## Loading required package: R6
```

```
label <- LabelEncoder$new()
#Convert the string categorical variables to numerics
df_modelling$city <- label$fit_transform(df_modelling$city)
df_modelling$country <- label$fit_transform(df_modelling$country)
df_modelling$ad_topic_line <- label$fit_transform(df_modelling$ad_topic_line)
#Preview the first six rows of the dataset again
head(df_modelling)
```

```
##   daily_time_spent_on_site age gender area_income city country
## 1                    68.95  35      0    61833.90    0       0
## 2                    80.23  31      1    68441.85    1       1
## 3                    69.47  26      0    59785.94    2       2
## 4                    74.15  29      1    54806.18    3       3
## 5                    68.37  35      0    73889.99    4       4
```

```
## 6                    59.99  23      1    59761.56    5       5
##   daily_internet_usage ad_topic_line sec min hour month_day month
## 1               256.09             0  11  53    0        27     2
## 2               193.77             1   2  39    1         4     3
## 3               236.50             2  42  35   20        13     2
## 4               245.89             3  19  31    2        10     0
## 5               225.58             4  18  36    3         3     5
## 6               226.74             5  17  30   14        19     4
##   Click_on_advert
## 1               0
## 2               0
## 3               0
## 4               0
## 5               0
## 6               0
```

All the categorical string columns are converted to numeric, since machine learning model only recognize the numerical characters.

Build the Knn model

```
#Build the KNN supervised model
#Check for the missing value in the dataset
colSums(is.na(df_modelling))
```

```
## daily_time_spent_on_site                      age                   gender
##                        0                        0                        0
##              area_income                     city                  country
##                        0                        0                        0
##     daily_internet_usage           ad_topic_line                      sec
##                        0                        0                        0
##                      min                     hour                month_day
##                        0                        0                        0
##                    month          Click_on_advert
##                        0                        0
```

There's no any missing values in the dataset.

```
#Summary of the data
summary(df_modelling)
```

```
##  daily_time_spent_on_site      age            gender       area_income
##  Min.   :32.60            Min.   :19.00   Min.   :0.000   Min.   :13996
##  1st Qu.:51.36            1st Qu.:29.00   1st Qu.:0.000   1st Qu.:47032
##  Median :68.22            Median :35.00   Median :0.000   Median :57012
##  Mean   :65.00            Mean   :36.01   Mean   :0.481   Mean   :55000
##  3rd Qu.:78.55            3rd Qu.:42.00   3rd Qu.:1.000   3rd Qu.:65471
##  Max.   :91.43            Max.   :61.00   Max.   :1.000   Max.   :79485
##       city          country      daily_internet_usage ad_topic_line
##  Min.   :  0.0   Min.   :  0.0   Min.   :104.8        Min.   :  0.0
##  1st Qu.:234.8   1st Qu.: 52.0   1st Qu.:138.8        1st Qu.:249.8
##  Median :473.5   Median :107.0   Median :183.1        Median :499.5
##  Mean   :477.9   Mean   :108.9   Mean   :180.0        Mean   :499.5
```

```
##  3rd Qu.:721.2   3rd Qu.:162.0   3rd Qu.:218.8        3rd Qu.:749.2
##  Max.   :968.0   Max.   :236.0   Max.   :270.0        Max.   :999.0
##       sec             min             hour         month_day         month
##  Min.   : 0.0   Min.   : 0.00   Min.   : 0.00   Min.   : 1.00   Min.   :0.000
##  1st Qu.:15.0   1st Qu.:14.00   1st Qu.: 6.00   1st Qu.: 8.00   1st Qu.:1.000
##  Median :30.0   Median :30.00   Median :12.00   Median :15.00   Median :3.000
##  Mean   :29.8   Mean   :29.05   Mean   :11.66   Mean   :15.48   Mean   :2.817
##  3rd Qu.:44.0   3rd Qu.:43.00   3rd Qu.:18.00   3rd Qu.:23.00   3rd Qu.:4.000
##  Max.   :59.0   Max.   :59.00   Max.   :23.00   Max.   :31.00   Max.   :6.000
##  Click_on_advert
##  Min.   :0.0
##  1st Qu.:0.0
##  Median :0.5
##  Mean   :0.5
##  3rd Qu.:1.0
##  Max.   :1.0
```

The summary of data gives us the overview of the data by displaying the mean, median, quartiles, maximum and minimum of each numerical column.

```
#Normalizing the dataset
norm <-function(x) { (x -min(x))/(max(x)-min(x))   }
norm
```

```
## function(x) { (x -min(x))/(max(x)-min(x))   }
```

Built the normalization function, the function will be used for the normalization of the dataset

```
#Normalization of the dataset
df_modelling_norm <- as.data.frame(lapply(df_modelling[, -15], norm))
head(df_modelling_norm)
```

```
##   daily_time_spent_on_site       age gender area_income        city     country
## 1                0.6178820 0.3809524      0   0.7304725 0.000000000 0.000000000
## 2                0.8096209 0.2857143      1   0.8313752 0.001033058 0.004237288
## 3                0.6267211 0.1666667      0   0.6992003 0.002066116 0.008474576
## 4                0.7062723 0.2380952      1   0.6231599 0.003099174 0.012711864
## 5                0.6080231 0.3809524      0   0.9145678 0.004132231 0.016949153
## 6                0.4655788 0.0952381      1   0.6988280 0.005165289 0.021186441
##   daily_internet_usage ad_topic_line        sec       min       hour  month_day
## 1            0.9160310   0.000000000 0.18644068 0.8983051 0.00000000 0.86666667
## 2            0.5387456   0.001001001 0.03389831 0.6610169 0.04347826 0.10000000
## 3            0.7974331   0.002002002 0.71186441 0.5932203 0.86956522 0.40000000
## 4            0.8542802   0.003003003 0.32203390 0.5254237 0.08695652 0.30000000
## 5            0.7313234   0.004004004 0.30508475 0.6101695 0.13043478 0.06666667
## 6            0.7383460   0.005005005 0.28813559 0.5084746 0.60869565 0.60000000
##       month Click_on_advert
## 1 0.3333333               0
## 2 0.5000000               0
## 3 0.3333333               0
## 4 0.0000000               0
## 5 0.8333333               0
## 6 0.6666667               0
```

The normalization helps in scaling the data, this will improve the accuracy and performance of machine learning model.

```
#Create train and test dataset
train = df_modelling_norm[1:800, ]
test = df_modelling_norm[801:1000, ]
#Extract the 14th column of the train and test dataset
train_target = df_modelling_norm[1:800, 14]
test_target = df_modelling_norm[801:1000, 14]
```

The train _target will be used as "cl" argument in knn model and the test target will be used in calculations of confusion matrix and accuracy.

```
#Apply the Knn algorithm, firstly we call the class package
library(class)
require(class)
model <- knn(train= train, test=test, cl= train_target, k=13)
table(factor(model))
```

```
##
##   0   1
##  88 112
```

```
#Confusion matrix
confusion_matrix <- table(test_target,model)
confusion_matrix
```

```
##            model
## test_target   0   1
##           0  88   0
##           1   0 112
```

From the confusion matrix above of the 200 records 88 were classied as ("0" not click on advert) and 112 were classified as ("1" click on advert).

```
#Accuracy score
accuracy <- function(x){sum(diag(x)/(sum(rowSums(x)))) * 100}
accuracy(confusion_matrix)
```

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

The model had an accuracy of 100%. these implies the model was able to classiffy all the records from the test target correctly.

CONCLUSION

1. There's no any relationship between the gender of an individual who visits the blog and the chances of he or she clicking the adverts on the blog site

2.There's a moderate positive relationship between the age of individuals who visit the blog site and the chances of the visitor to click on the advert on the blog site

3. There's a moderate negative relationship between individual area income whom visits the blog site and the chances of the individual to vlick on the advert at the blog site.

Recommendation

The individual who is most likely to click on the advert on blog site, is person of gender either male or female whose area national income is less than 55000 and the age is greater than the 36 years.

Challenging the solution

The set metric of success of the model was 90%, but the build machine learning model was able to classify the class labels with 100% level of accuracy. This is a bit challenging, since it's not clear whether the machine would able to maintain the same level of accuracy if tested with a different data.

Follow up question

a). Do we have the right data?

Yes, the data was appropriate with no missing values.

b). Do we need another data?

No, the data was appropriate. The only thing needed is further analysis.

c).Do we have the right question?

Yes, the question is clear and straight forward.