Advert Analysis

Predict whether a user will click an Ad

Introduction

Defining the Question

Carry out an analysis on whether a user will click an Ad. We are to perform Exploratory Data Analysis on the data : Univariate,Bivariate and Multivariate analysis.

Context

A Kenyan entrepreneur has created an online cryptography course and would want to advertise it on her blog. She currently targets audiences originating from various countries.In the past, she ran ads to advertise a related course on the same blog and collected data in the process.She would now like to employ your services as a Data Science Consultant to create a solution that would allow her to determine whether ads targeted to audiences of certain characteristics i.e. city, male country, ad topic, etc. would click on her ads.

Experimental Design

Installing packages and loading libraries

Loading the data

Exploratory Data Analysis

Data Cleaning

Univariate,Bivariate,Multivariate analysis

Conclusion

Appropriateness of the Data

The columns in the data set include:

Daily Time Spent on Site

Age

Area Income

Daily Internet Usage

Ad Topic Line

City

Male

Country

Timestamp

Clicked on Ad

The data set has 1000 observations and 10 variables.

Loading our Data

```
# Loading the dataset
advert = read.csv("http://bit.ly/IPAdvertisingData")
```

Exploratory Data Analysis

Checking the data*

```
# Checking the top of our dataset
head(advert)
```

```
##   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
```

```
# Checking the bottom of our dataset
tail(advert)
```

```
##      Daily.Time.Spent.on.Site Age Area.Income Daily.Internet.Usage
## 995                     43.70  28    63126.96               173.01
## 996                     72.97  30    71384.57               208.58
## 997                     51.30  45    67782.17               134.42
## 998                     51.63  51    42415.72               120.37
## 999                     55.55  19    41920.79               187.95
```

```
## 1000                    45.01  26    29875.80              178.35
##                             Ad.Topic.Line          City Male
## 995         Front-line bifurcated ability  Nicholasland    0
## 996         Fundamental modular algorithm     Duffystad    1
## 997       Grass-roots cohesive monitoring   New Darlene    1
## 998          Expanded intangible solution South Jessica    1
## 999  Proactive bandwidth-monitored policy   West Steven    0
## 1000      Virtual 5thgeneration emulation   Ronniemouth    0
##                      Country           Timestamp Clicked.on.Ad
## 995                  Mayotte 2016-04-04 03:57:48             1
## 996                  Lebanon 2016-02-11 21:49:00             1
## 997  Bosnia and Herzegovina 2016-04-22 02:07:01             1
## 998                 Mongolia 2016-02-01 17:24:57             1
## 999                Guatemala 2016-03-24 02:35:54             0
## 1000                  Brazil 2016-06-03 21:43:21             1
```

```
# checking the structure of our dataset
# checking the data types of our variables
str(advert)
```

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

```
# Checking the number of rows and columns
dim(advert)
```

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

Our dataset has 1000 observations(rows) and 10 variables(columns)

Data cleaning

```
# checking for the sum of missing values in each column
colSums(is.na(advert))
```

```
## 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 missing values within our dataset.

```
# checking for duplicates
duplicated_rows <- colSums(advert[duplicated(advert),])
duplicated_rows
```

```
## 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 duplicates in our dataset.

```
# checking our column names
names(advert)
```

```
##  [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"
```

```
# lower case of the column names
names(advert) <- tolower(names(advert))
names(advert)
```

```
##  [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"
```

```
# checking dataframe to see if column names case has been lowered
head(advert)
```

```
##   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
```

```
# checking for outliers
# detect outliers by use ofsome descriptive statistics,
# and in particular with the minimum and maximum.
summary(advert)
```

```
##  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
```

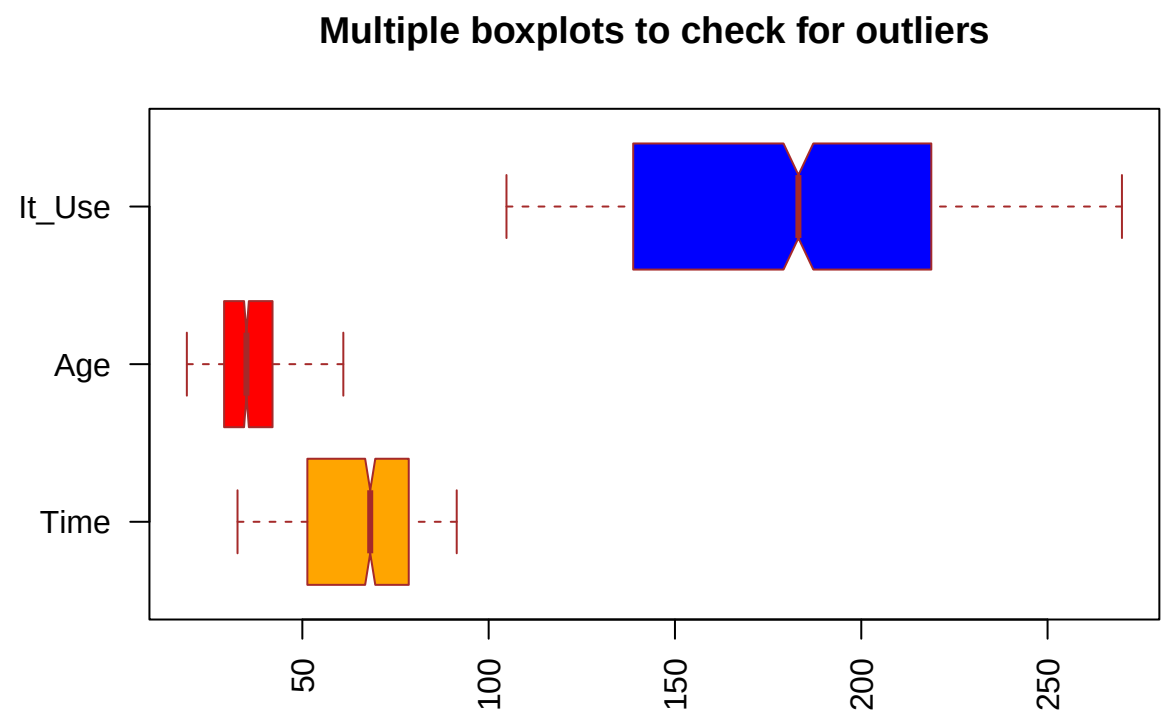
There appear to be no outliers based on the summary statistics. However, we will continue to investigate in order to evaluate and confirm.

```
# Using a boxplot to check for observations far away from other data points.
# We will Use all three double type columns: specifying each


Daily_Time_Spent_on_Site <- advert$ daily.time.spent.on.site
Age <- advert$age
Daily_Internet_Usage <- advert$daily.internet.usage
Area_Income <- advert$area.income


boxplot(Daily_Time_Spent_on_Site,Age, Daily_Internet_Usage,
main = "Multiple boxplots to check for outliers",
at = c(1,2,3),
```

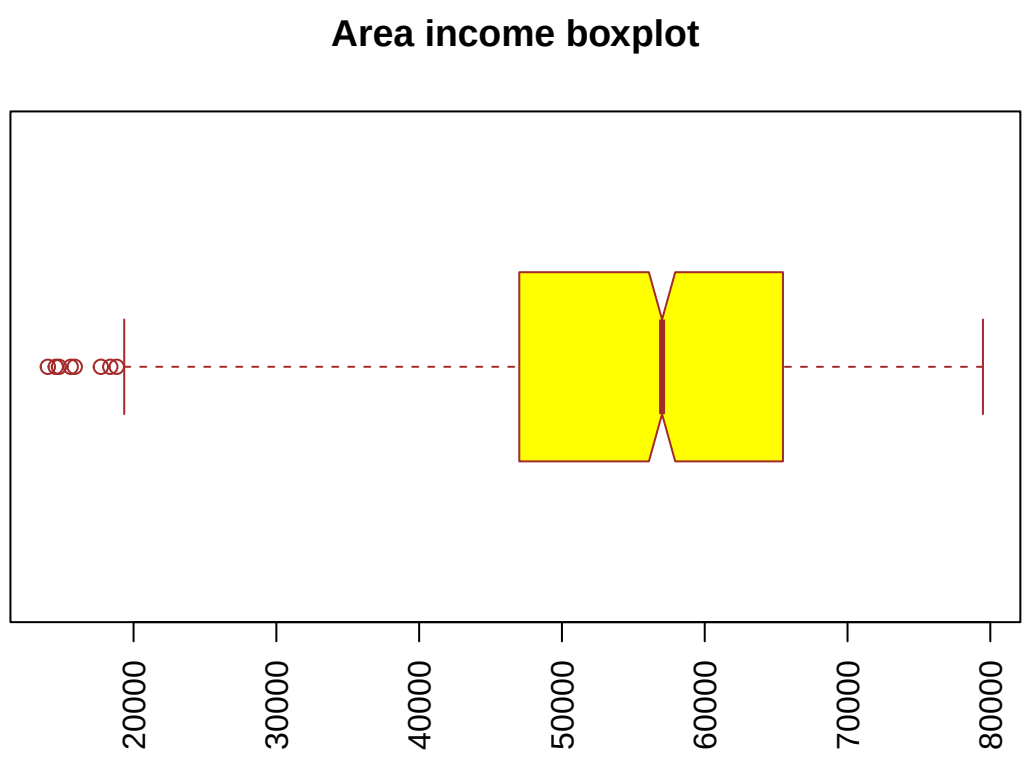
```
names = c("Time", "Age","It_Use"),
las = 2,
col = c("orange","red","blue"),
border = "brown",
horizontal = TRUE,
notch = TRUE
)
```




```
# We will remove the area income and plot it on a different code so as to see the other plots more clea
```

The Daily_Time_Spent_on_Site,Age, Daily_Internet_Usage varaibles do not seem to have any outliers.

```
# plotting area income
boxplot(Area_Income,
main = "Area income boxplot",
at = c(1),
names = c("Income"),
las = 2,
col = c("yellow"),
border = "brown",
horizontal = TRUE,
notch = TRUE
)
```



Area Income has outliers on the first quartile as shown above.

```
#Let us check out the outliers in the Area Income column

boxplot.stats(advert$area.income)$out
```

```
## [1] 17709.98 18819.34 15598.29 15879.10 14548.06 13996.50 14775.50 18368.57
```

We've come to the conclusion that the outliers appear to be within our maximum and minimum entries and appear to be viable, thus they'll be kept.

Univariate analysis*

```


# Calculating mean of our variables
# Lets check the numerical columns of our dataset
sapply(advert, class)
```

Measures of Central Tendency

```
## daily.time.spent.on.site                      age             area.income
##                "numeric"                "integer"                "numeric"
##     daily.internet.usage           ad.topic.line                     city
```

```
##                "numeric"              "character"              "character"
##                     male                  country                timestamp
##                "integer"              "character"              "character"
##            clicked.on.ad
##                "integer"
```

```
# our numerical columns are:
  # 1. daily.time.spent.on.site
  # 2. area.income
  # 3. daily.internet.usage
  # 4. age
  # 5. male
  # 6. clicked.on.ad

# because the male and clicked.on.ad columns are encoded, we'll check for the mean,mode and median of t
```

```
# checking the mean

dailytime.mean = mean(advert$daily.time.spent.on.site)
age.mean = mean(advert$age)
areaincome.mean = mean(advert$area.income)
dailyinternet.mean = mean(advert$daily.internet.usage)

print("The mean of the Daily Time Spent on the site:",quote=FALSE)
```

```
## [1] The mean of the Daily Time Spent on the site:
```

```
dailytime.mean
```

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

```
print("The mean of the Age:",quote=FALSE)
```

```
## [1] The mean of the Age:
```

```
age.mean
```

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

```
print("The mean of the Area Income:",quote=FALSE)
```

```
## [1] The mean of the Area Income:
```

```
areaincome.mean
```

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

```r
print("The mean of the Daily Internet Usage:",quote=FALSE)
```

```
## [1] The mean of the Daily Internet Usage:
```

```r
dailyinternet.mean
```

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

```r
# checking median

dailytime.median = median(advert$daily.time.spent.on.site)
age.median = median(advert$age)
areaincome.median = median(advert$area.income)
dailyinternet.median = median(advert$daily.internet.usage)

print("The median of the Daily Time Spent on the site:",quote=FALSE)
```

```
## [1] The median of the Daily Time Spent on the site:
```

```r
dailytime.median
```

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

```r
print("The median of the Age:",quote=FALSE)
```

```
## [1] The median of the Age:
```

```r
age.median
```

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

```r
print("The median of the Area Income:",quote=FALSE)
```

```
## [1] The median of the Area Income:
```

```r
areaincome.median
```

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

```r
print("The median of the Daily Internet Usage:",quote=FALSE)
```

```
## [1] The median of the Daily Internet Usage:
```

```r
dailyinternet.median
```

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

```r
# checking the mode

#Set the function to get the mode

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

dailytime.mode = getmode(advert$daily.time.spent.on.site)
age.mode = getmode(advert$age)
areaincome.mode = getmode(advert$area.income)
dailyinternet.mode = getmode(advert$daily.internet.usage)

print("The mode of the Daily Time Spent on the Site:",quote=FALSE)
```

```
## [1] The mode of the Daily Time Spent on the Site:
```

```r
dailytime.mode
```

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

```r
print("The mode of the Age:",quote=FALSE)
```

```
## [1] The mode of the Age:
```

```r
age.mode
```

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

```r
print("The mode of the Area Income:",quote=FALSE)
```

```
## [1] The mode of the Area Income:
```

```r
areaincome.mode
```

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

```r
print("The mode of the Daily Internet Usage:",quote=FALSE)
```

```
## [1] The mode of the Daily Internet Usage:
```

```r
dailyinternet.mode
```

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

```
# Finding the minimum values of our columns
dailytime.min = min(advert$daily.time.spent.on.site)
age.min = min(advert$age)
areaincome.min = min(advert$area.income)
dailyinternet.min = min(advert$daily.internet.usage)

print("The minimum value of the Daily Time Spent on the site:",quote=FALSE)
```

Measures of dispersion

```
## [1] The minimum value of the Daily Time Spent on the site:
```

```
dailytime.min
```

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

```
print("The minimum value of the Age:",quote=FALSE)
```

```
## [1] The minimum value of the Age:
```

```
age.min
```

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

```
print("The minimum value of the Area Income:",quote=FALSE)
```

```
## [1] The minimum value of the Area Income:
```

```
areaincome.min
```

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

```
print("The minimum value of the Daily Internet Usage:",quote=FALSE)
```

```
## [1] The minimum value of the Daily Internet Usage:
```

```
dailyinternet.min
```

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

```
# Finding the maximum values of our columns
dailytime.max = max(advert$daily.time.spent.on.site)
age.max = max(advert$age)
areaincome.max = max(advert$area.income)
dailyinternet.max = max(advert$daily.internet.usage)

print("The maximum value of the Daily Time Spent on the site:",quote=FALSE)
```

```
## [1] The maximum value of the Daily Time Spent on the site:
```

```
dailytime.max
```

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

```
print("The maximum value of the Age:",quote=FALSE)
```

```
## [1] The maximum value of the Age:
```

```
age.max
```

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

```
print("The maximum value of the Area Income:",quote=FALSE)
```

```
## [1] The maximum value of the Area Income:
```

```
areaincome.max
```

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

```
print("The maximum value of the Daily Internet Usage:",quote=FALSE)
```

```
## [1] The maximum value of the Daily Internet Usage:
```

```
dailyinternet.max
```

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

```
# Finding the range of values of our columns
dailytime.range = range(advert$daily.time.spent.on.site)
age.range = range(advert$age)
areaincome.range = range(advert$area.income)
dailyinternet.range = range(advert$daily.internet.usage)

print("The range value of the Daily Time Spent on the site:",quote=FALSE)
```

```
## [1] The range value of the Daily Time Spent on the site:
```

```
dailytime.range
```

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

```
print("The range value of the Age:",quote=FALSE)
```

```
## [1] The range value of the Age:
```

```
age.range
```

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

```
print("The range value of the Area Income:",quote=FALSE)
```

```
## [1] The range value of the Area Income:
```

```
areaincome.range
```

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

```
print("The range value of the Daily Internet Usage:",quote=FALSE)
```

```
## [1] The range value of the Daily Internet Usage:
```

```
dailyinternet.range
```

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

```
#Find the quantile in the numerical columns in the dataset

dailytime.quantile = quantile(advert$daily.time.spent.on.site)
age.quantile = quantile(advert$age)
areaincome.quantile = quantile(advert$area.income)
dailyinternet.quantile = quantile(advert$daily.internet.usage)

print("The quantiles of the Daily Time Spent on the site:",quote=FALSE)
```

```
## [1] The quantiles of the Daily Time Spent on the site:
```

```
dailytime.quantile
```

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

```
print("The quantiles of the Age:",quote=FALSE)
```

```
## [1] The quantiles of the Age:
```

```
age.quantile
```

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

```
print("The quantiles of the Area Income:",quote=FALSE)
```

```
## [1] The quantiles of the Area Income:
```

```
areaincome.quantile
```

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

```
print("The quantiles of the Daily Internet Usage:",quote=FALSE)
```

```
## [1] The quantiles of the Daily Internet Usage:
```

```
dailyinternet.quantile
```

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

```
#Find the variance in ouer columns in the dataset

dailytime.variance = var(advert$daily.time.spent.on.site)
age.variance = var(advert$age)
areaincome.variance = var(advert$area.income)
dailyinternet.variance = var(advert$daily.internet.usage)

print("The variance of the Daily Time Spent on the Site:",quote=FALSE)
```

```
## [1] The variance of the Daily Time Spent on the Site:
```

```
dailytime.variance
```

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

```
print("The variance of the Age:",quote=FALSE)
```

```
## [1] The variance of the Age:
```

```
age.variance
```

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

```
print("The variance of the Area Income:",quote=FALSE)
```

```
## [1] The variance of the Area Income:
```

```
areaincome.variance
```

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

```
print("The variance of the Daily Internet Usage:",quote=FALSE)
```

```
## [1] The variance of the Daily Internet Usage:
```

```
dailyinternet.variance
```

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

```
#Find the standard deviation in our columns in the dataset

dailytime.std = sd(advert$daily.time.spent.on.site)
age.std = sd(advert$age)
areaincome.std = sd(advert$area.income)
dailyinternet.std = sd(advert$daily.internet.usage)

print("The standard deviation of the Daily Time Usage:",quote=FALSE)
```

```
## [1] The standard deviation of the Daily Time Usage:
```

```
dailytime.std
```

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

```
print("The standard deviation of the Age:",quote=FALSE)
```

```
## [1] The standard deviation of the Age:
```

```
age.std
```

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

```
print("The standard deviation of the Area Income:",quote=FALSE)
```

```
## [1] The standard deviation of the Area Income:
```

```
areaincome.std
```

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

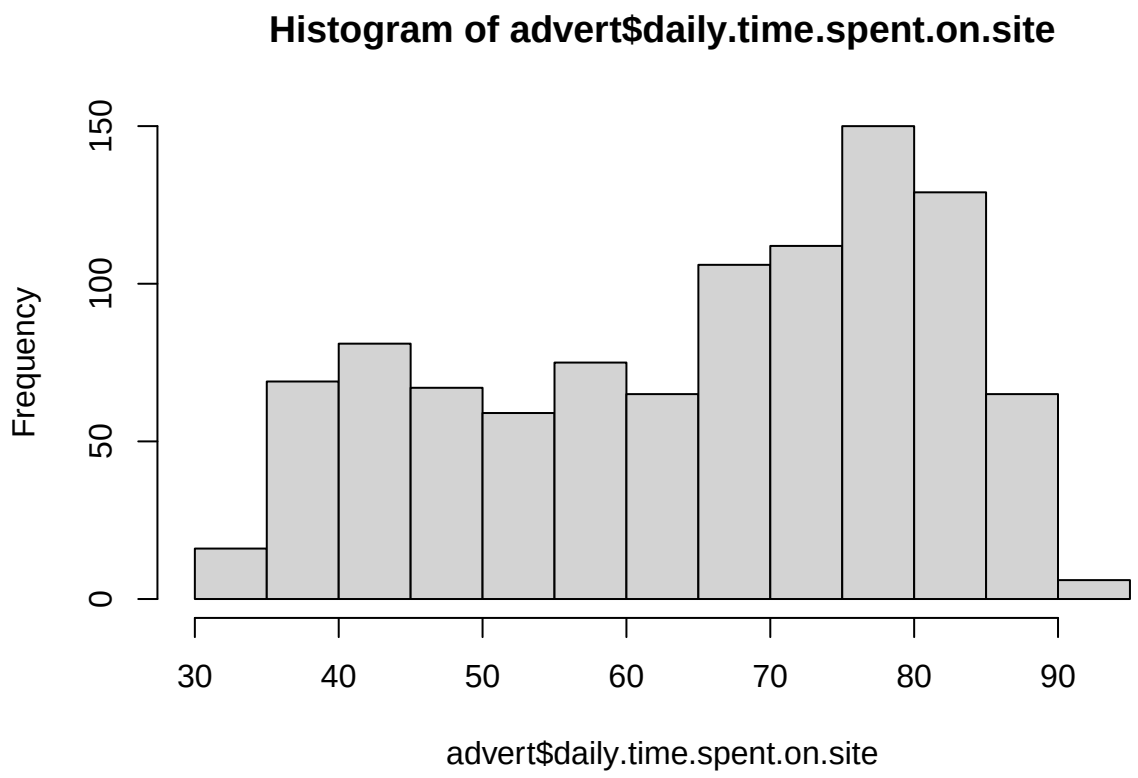
```
print("The standard deviation of the Daily Internet Usage:",quote=FALSE)
```

```
## [1] The standard deviation of the Daily Internet Usage:
```

```
dailyinternet.std
```

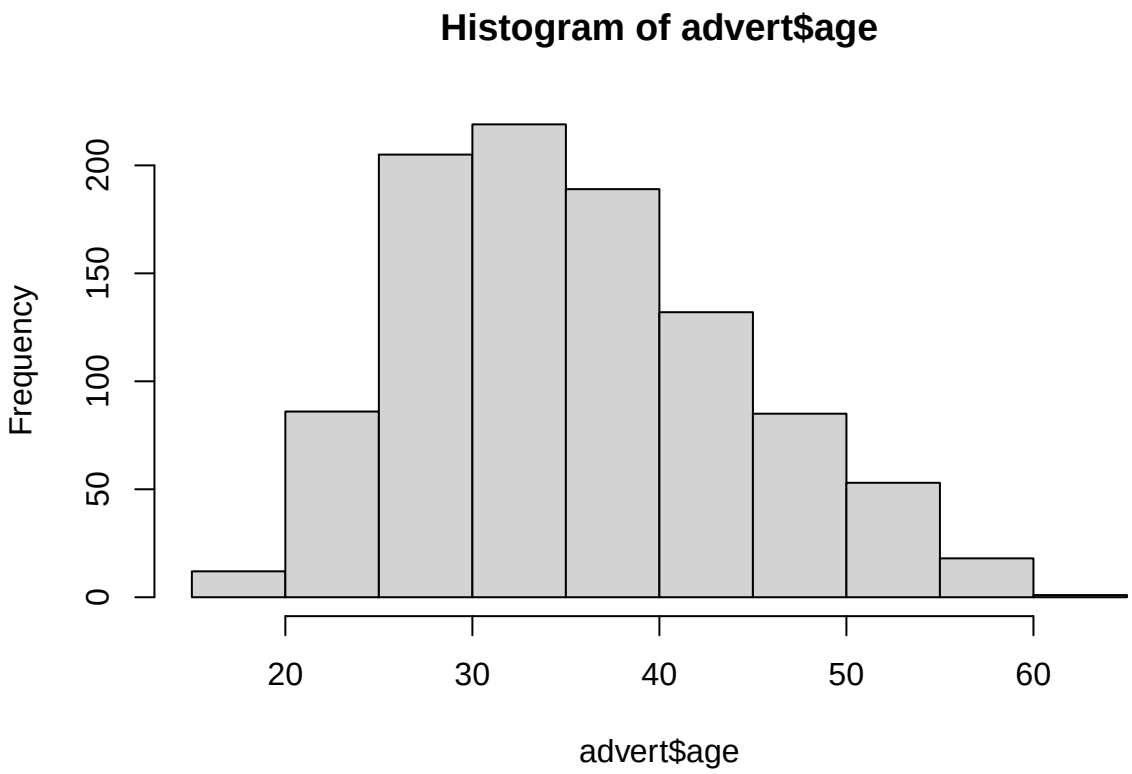
```
## [1] 43.90234
```

```
# Creating a histogram for daily time spent
hist(advert$daily.time.spent.on.site)
```



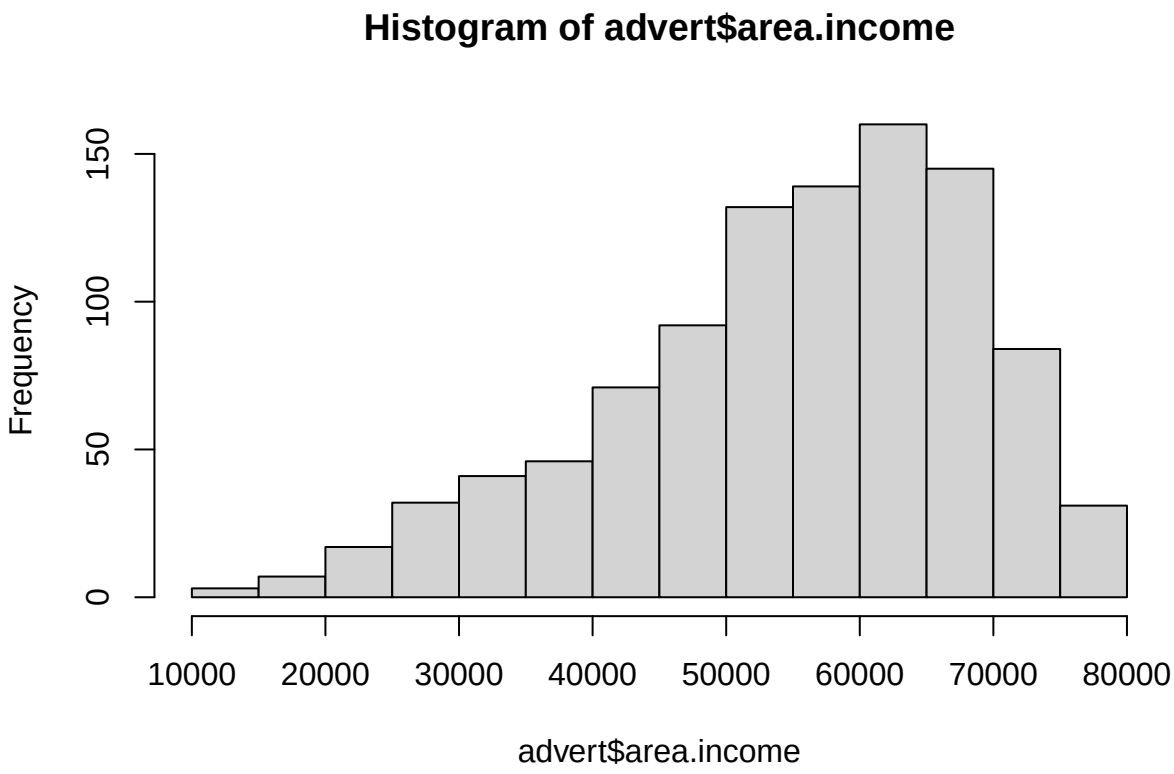
Most of the users spend 75 minutes on the site.

```
# Creating a histogram for age
hist(advert$age,)
```



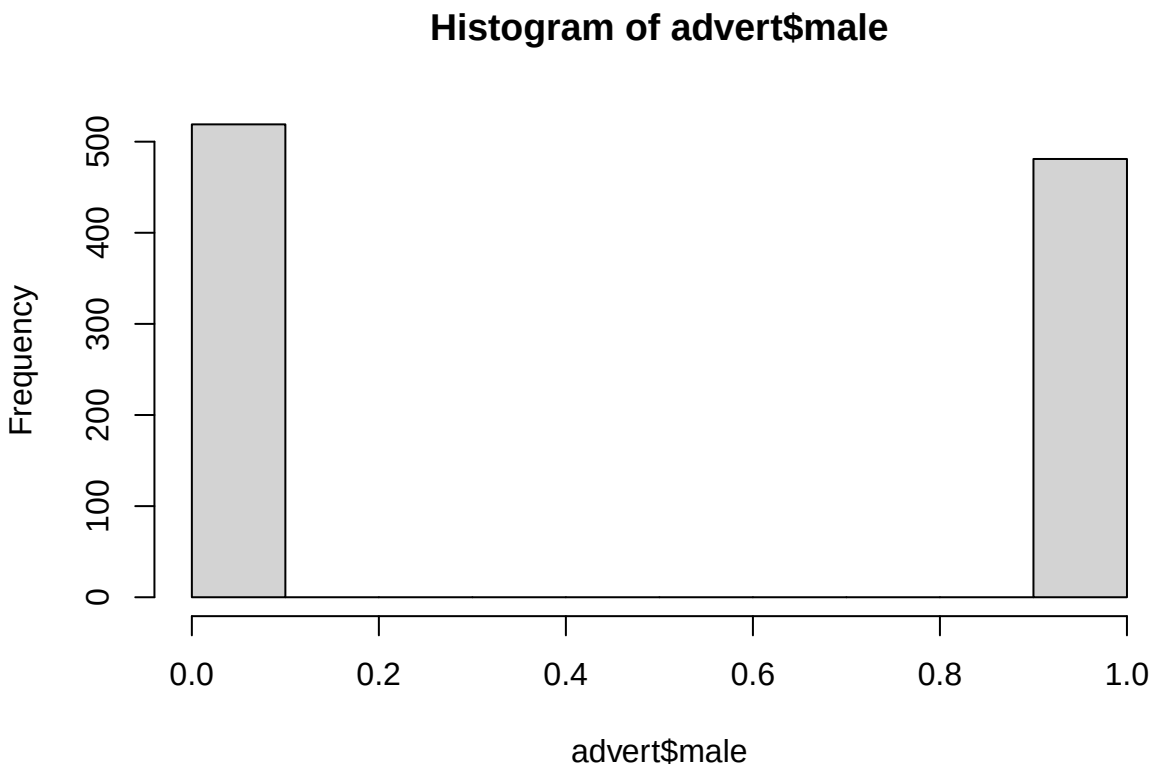
Majority of the users are between the age 25 to 35.

```
# Creating a histogram for area income
hist(advert$area.income)
```



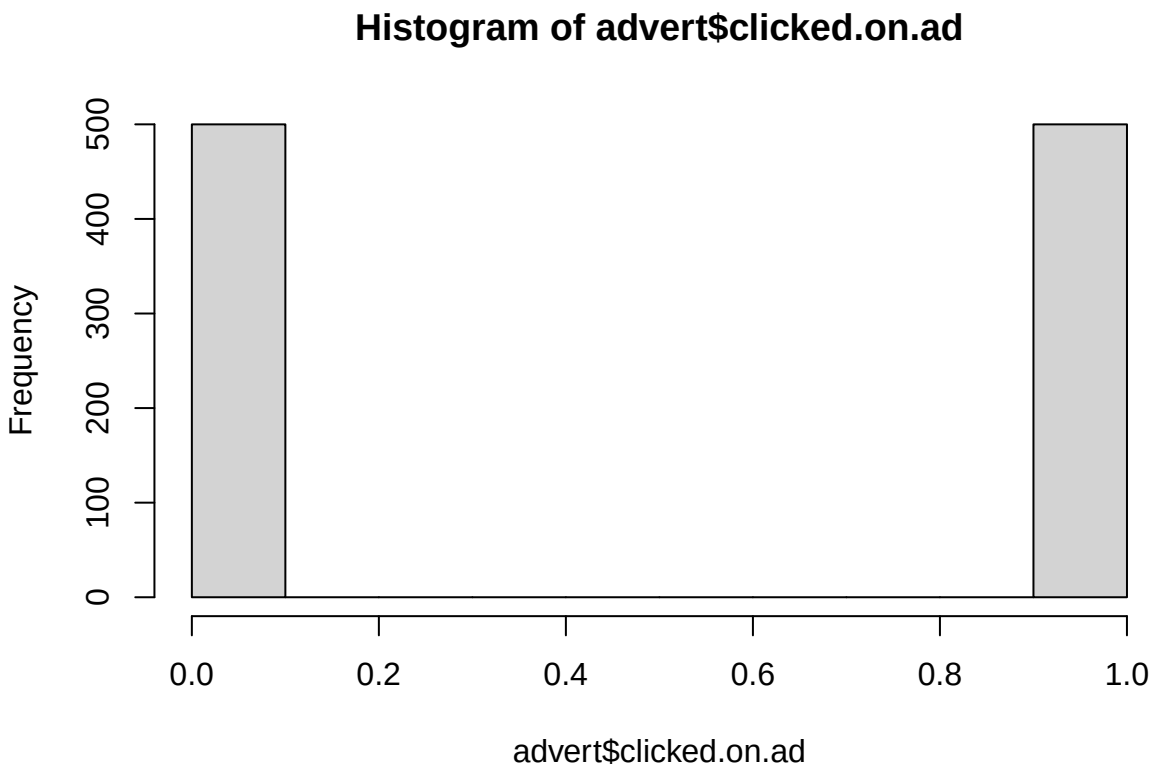
Majority of the users have an income of 60000

```
# Creating a histogram for male column
# male = 1 female = 0
hist(advert$male)
```



Majority of the users were female but the male ratio was still considerably high.

```
# Creating a histogram for clicked on ad
# clicked = 1 no click = 0
hist(advert$clicked.on.ad)
```



There was an equal ratio of those who clicked and those who did not click on an ad.

Bivariate Analysis #Covariance of age and click on ad

```
# Covariance of age and click on ad
cov(advert$age, advert$clicked.on.ad)
```

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

The covariance is positive hence there is a positive relation between age and clicking on an ad.

```
# Covariance of Daily.Time.Spent.on.Site and click on ad
cov(advert$daily.time.spent.on.site, advert$clicked.on.ad)
```

```
## [1] -5.933143
```

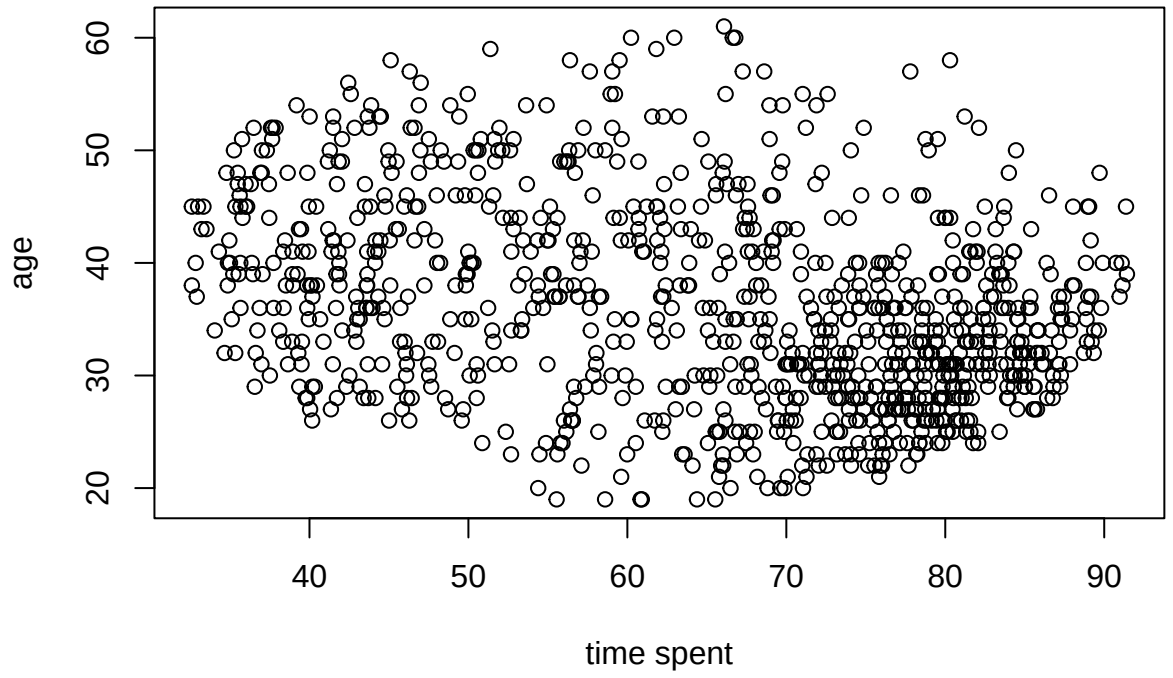
There is a negative covariance implying a negative relation to user clicking on an ad.

```
# Covariance of area income and click on ad
cov(advert$area.income, advert$clicked.on.ad)
```

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

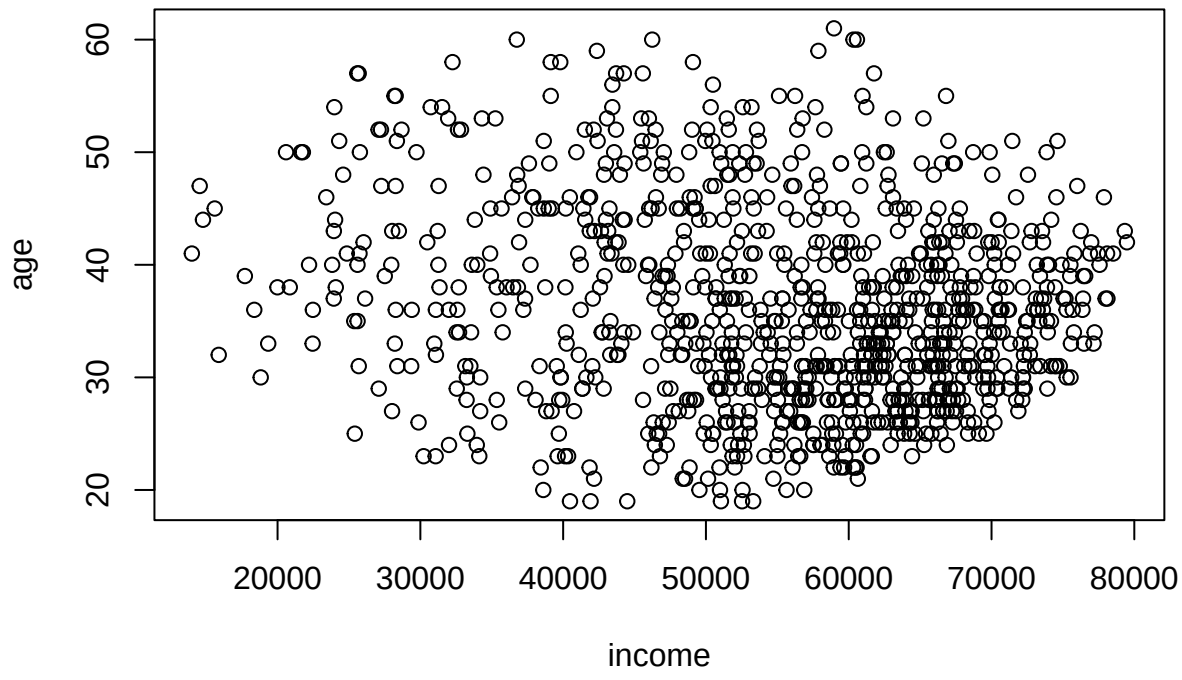
There is a negative covariance between income and a user clicking on an ad.

```
# plotting scatter plots between age and time spent
plot(advert$daily.time.spent.on.site, advert$age, xlab="time spent", ylab="age")
```

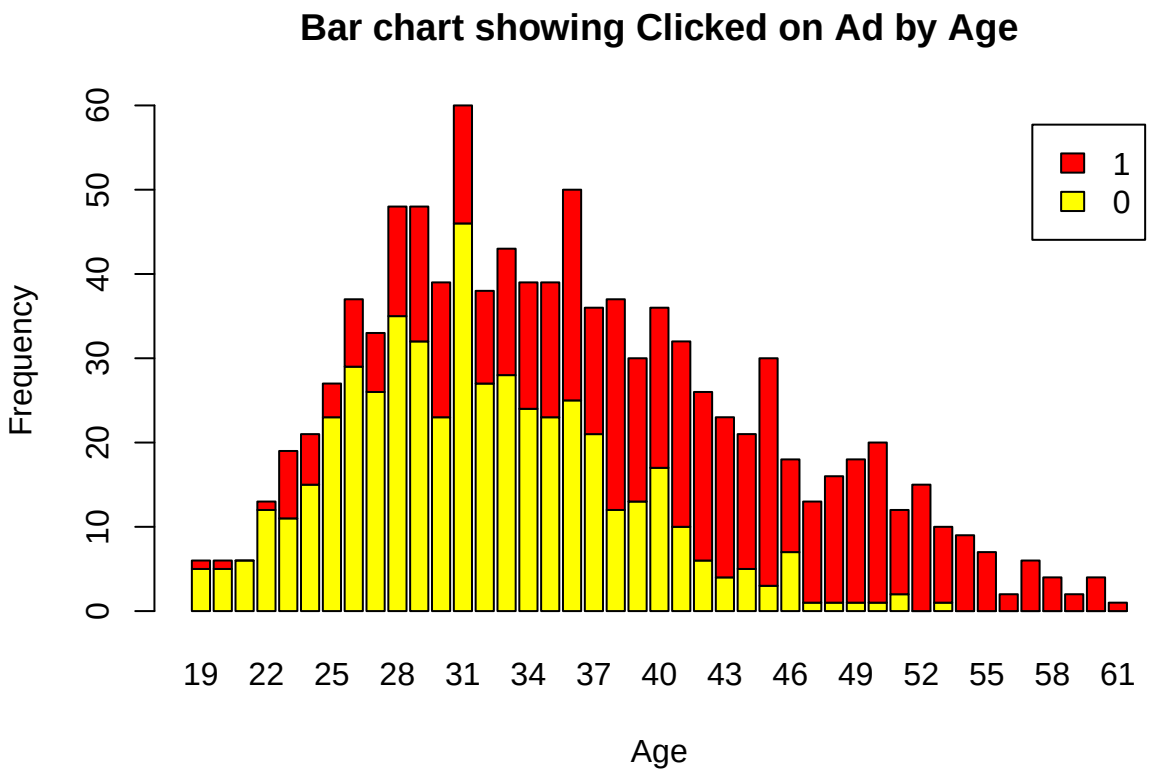


Majority of those who are young spent more time on the site.

```
# scatter plot of age and area income
plot(advert$area.income, advert$age, xlab="income", ylab="age")
```

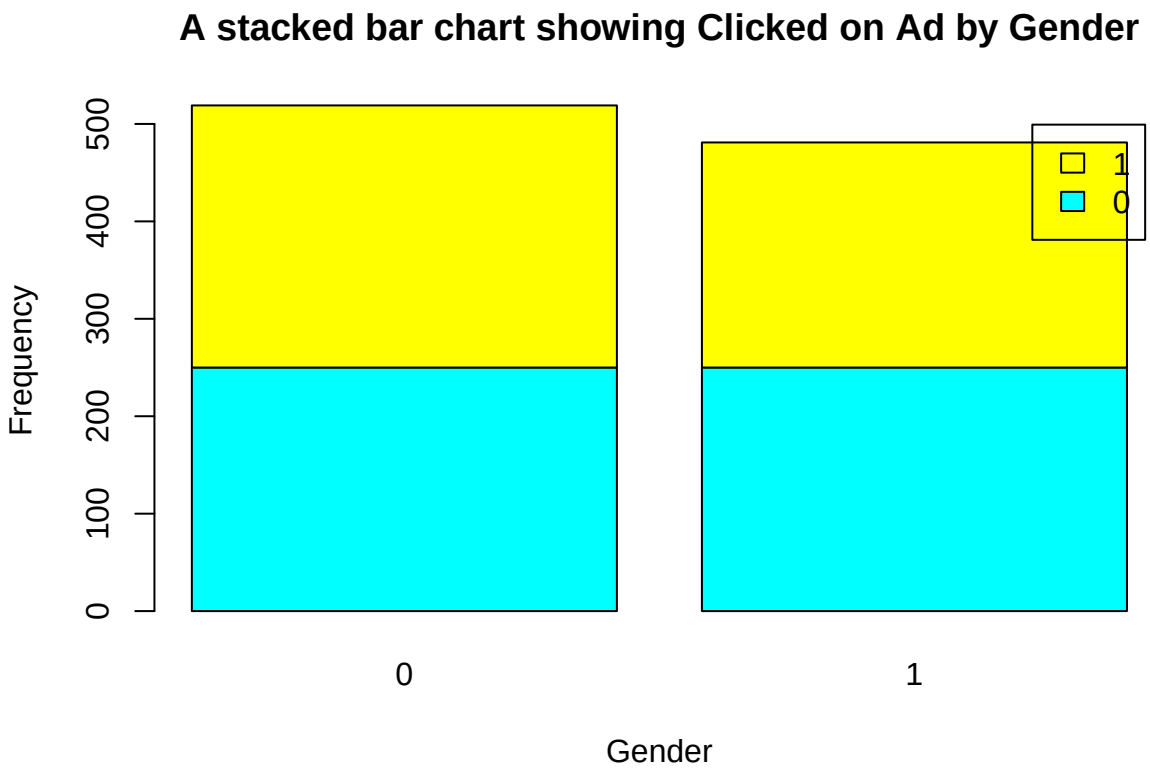



```
# Creating a Stacked bar chart
counts <- table(advert$clicked.on.ad, advert$age)
barplot(counts,
  main="Bar chart showing Clicked on Ad by Age",
  xlab="Age",
  ylab = "Frequency",
  col=c("yellow","red"),
  legend = rownames(counts))
```



- The stacked bar chart shows the distribution of the number of people who clicked on an Ad by age. - The highest age of the participants was 61 and lowest was 19. - The people who clicked most on Ads were between age 28 to 43.

```
# Stacked bar chart
counts <- table(advert$clicked.on.ad, advert$male)
barplot(counts,
  main="A stacked bar chart showing Clicked on Ad by Gender",
  xlab="Gender",
  ylab = "Frequency",
  col=c("cyan","yellow"),
  legend = rownames(counts))
```



- More females clicked on Ad compared to males - There are more female users compares to male users

```
# create a correlation heat map
# Heat map

#numeric_tbl <- advert %>%
  #select_if(is.numeric) %>%
  #select(daily.time.spent.on.site, age, area.income,daily.internet.usage)

# Calculate the correlations
#corr <- cor(numeric_tbl, use = "complete.obs")

#ggcorrplot(round(corr, 2),
           #type = "full", lab = T)
```

- There is a moderate relationship between daily time spent on the site and and daily internet usage.
- The other variables have weak relationships.

```


# Data Cleaning
# There were no duplicates nor missing values in our dataset
# Area Income has outliers on the first quartile as shown above.

# Measures of central tendency
```

```
#The mode of the Daily Time Spent on the Site: 62.26
#The mode of the Age: 31
#The mode of the Area Income:61833.9
#The mode of the Daily Internet Usage: 167.22

#The median of the Daily Time Spent on the site:68.215
#The median of the Age:35
#The median of the Area Income:57012.3
#The median of the Daily Internet Usage:183.13

#The mean of the Daily Time Spent on the site:65.0002
#The mean of the Age:36.009
#The mean of the Area Income:55000.00008
#The mean of the Daily Internet Usage:180.0001

# Measures of dispersion
#The minimum value of the Daily Time Spent on the site:32.6
#The minimum value of the Age:19
#The minimum value of the Area Income:13996.5
#The minimum value of the Daily Internet Usage:104.78

#The maximum value of the Daily Time Spent on the site:91.43
#The maximum value of the Age:61
#The maximum value of the Area Income:79484.8
#The maximum value of the Daily Internet Usage:269.96

#The range value of the Daily Time Spent on the site:32.6 91.43
#The range value of the Age:19 61
#The range value of the Area Income:13996.5 79484.8
#The range value of the Daily Internet Usage:104.78 269.96

#The variance of the Daily Time Spent on the Site:251.337094854855
#The variance of the Age:77.1861051051051
#The variance of the Area Income:179952405.951775
#The variance of the Daily Internet Usage:1927.41539618619

#The standard deviation of the Daily Time Usage:15.8536145675002
#The standard deviation of the Age:8.78556231012592
#The standard deviation of the Area Income:13414.6340222824
#The standard deviation of the Daily Internet Usage:43.9023393019801
```

Conclusion

- Majority of the users have an income of 60000
- Majority of the users were female but the male ratio was still considerably high.
- There was an equal ratio of those who clicked and those who did not click on an ad.
- The covariance is positive hence there is a positive relation between age and clicking on an ad.
- There is a negative covariance between daily time spent implying a negative relation to user clicking on an ad.
- There is a negative covariance between income and a user clicking on an ad.
- Majority of those who are young spent more time on the site.
- The highest age of the participants was 61 and lowest was 19.
- The people who clicked most on Ads were between age 28 to 43.

- More females clicked on Ad compared to males.
- There are more female users compares to male users.

Modeling

```
library("dplyr")
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

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

```
# Drop columns
advert <-advert %>% select(-c(timestamp, city, `ad.topic.line`, country,city))
```

Decision Trees

```
advert[,'clicked.on.ad']<-factor(advert[,'clicked.on.ad'])
```

```
library(rpart)
library(rpart.plot)
library(caret)
```

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

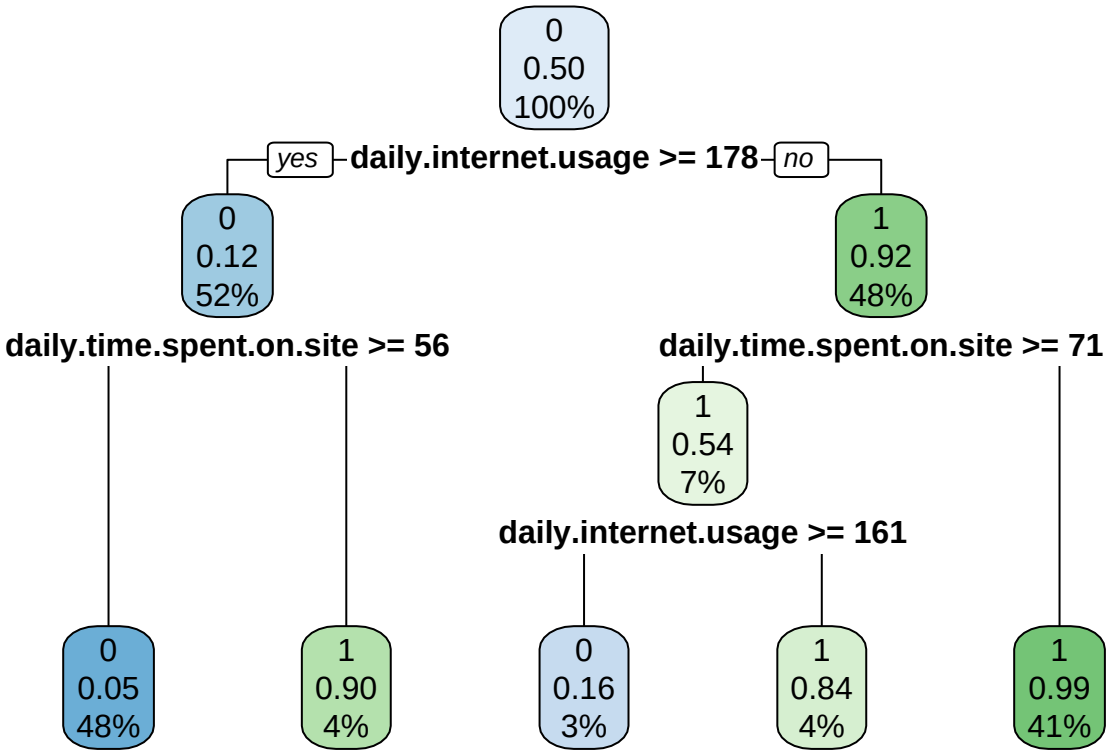
```
## Loading required package: ggplot2
```

```
# Using decision tree
# Fitting the model
# Specifying the target and predictor variables

m <- rpart(clicked.on.ad ~ . ,
    data = advert,
    method = "class")
```

```
# Plotting the decision tree model

rpart.plot(m)
```




```
# Making predictions
# Printing the confusion matrix

p <- predict(m, advert, type ="class")
table(p, advert$clicked.on.ad)
```

```
##
## p     0   1
##   0 485  28
##   1  15 472
```

```
# Printing the Accuracy

mean(advert$clicked.on.ad == p)
```

```
## [1] 0.957
```

- The accuracy of the model is 95.7%.
- This is a useful model for predictions.
- We'll assess this model or put it to the test against another model.

Random Forest

```
# Training the model
# Setting seed for randomness

set.seed(12)
model <- train(clicked.on.ad ~. ,
               data = advert,
               method = "ranger")
```

```
# print model
model
```

```
## Random Forest
##
## 1000 samples
##    5 predictor
##    2 classes: '0', '1'
##
## No pre-processing
## Resampling: Bootstrapped (25 reps)
## Summary of sample sizes: 1000, 1000, 1000, 1000, 1000, 1000, ...
## Resampling results across tuning parameters:
##
##   mtry  splitrule   Accuracy   Kappa
##   2     gini        0.9651495  0.9302174
##   2     extratrees  0.9652495  0.9304122
##   3     gini        0.9628152  0.9255473
##   3     extratrees  0.9637168  0.9273410
##   5     gini        0.9551554  0.9101945
##   5     extratrees  0.9629524  0.9258088
##
## Tuning parameter 'min.node.size' was held constant at a value of 1
## Accuracy was used to select the optimal model using the largest value.
## The final values used for the model were mtry = 2, splitrule = extratrees
##  and min.node.size = 1.
```

- The Random Forest model had a 96.5% accuracy rate.When compared to Decision Tree, this is a better model.

Conclusion

Random Forest is a better model to use for our predictions as it has the best accuracy. In comparison to the decision tree classifier, which only uses one tree, the model employs the bagging method and employs a large number of trees.