Week 12 Core IP_Advertising Dataset

Martha Irungu

9/3/2020

Ia) Specifying the question

The objective of this study is to use the advertising dataset provided to support a Kenyan entrepreneur identify which individuals are most likely to click on her online advertisement.

b)Defining the Metrics for success

To meet the objective of the study we will need to do the following:

i) Find and deal with outliers, anomalies, and missing data within the dataset.

ii) Perform univariate and bivariate analysis.

iii) From the analysis done, share insights and provide a conclusion and recommendation.

iv) Create a model using supervised learning algorithims to establish which customer is likely to click or not.

c) Understanding the context

A Kenyan entrepreneur has created an online cryptography course and would want to advertise it on her blog. She currently targets audiences originating from various countries. In the past, she ran ads to advertise a related course on the same blog and collected data in the process.

Online marketing is the practice of leveraging web-based channels to spread a message about a company's brand, products, or services to its potential customers. The methods and techniques used for online marketing include email, social media, display advertising, search engine optimization, Google AdWords and more. The objective of marketing is to reach potential customers through the channels where they spend their time reading, searching, shopping, and socializing online.

Widespread adoption of the internet for business and personal use in Kenya and the world at large has generated new channels for advertising and marketing engagement, including those mentioned above. There are also many benefits and challenges inherent to online marketing, which uses primarily digital mediums to attract, engage, and convert virtual visitors to customers.

As entreprenuers adopt these options, it is important for one to establish the impact of the advertisements and one way to do this is to establish chances of potential customers

clicking on the ads and what are charateristics of such customers in order to maximize return on investment. This is exatly what this entreprenuer would like advise on.

d) Recording the experimental design

The following steps were implemented

1.) Business Understanding.

2.) Reading the data.

3.) Data Exploration and cleaning to prepare the data for analysis

4.) Univariate,Bivariate analysis

5.) Modelling using supervised learning algorithims

6.) Conclusion of the findings and recommendation.

e)Data Relevance

The data provided for this study consists of columns with factors likely to influence an individual to either click on Ad or not.Since the data was collected when the entreprenuer was running another Ad on the same blog, it is relevant to help us establish what kind of audience are likely to click on an Ad or not.

2. Reading and cheking the data

##Reading and previewing that dataset

```
library("data.table")
advertising <- fread("/Users/marthairungu/desktop/R/advertising.csv")
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
```

##Checking the summary of the dataset

```
summary(advertising)
```

```
##  Daily Time Spent on Site      Age         Area Income    Daily Internet U
sage
##  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
```

#This shows the summary of numeric variables as tabulated.

#Checking the number of rows and columns

```
dim(advertising)
```

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

#We observe that the dataset has 1,000 observations and 10 variables

#Checking the structure of the dataset

```
str(advertising)
```

```
## Classes 'data.table' and 'data.frame':   1000 obs. of  10 variables:
##  $ Daily Time Spent on Site: num  69 80.2 69.5 74.2 68.4 ...
##  $ Age                     : int  35 31 26 29 35 23 33 48 30 20 ...
##  $ Area Income             : num  61834 68442 59786 54806 73890 ...
##  $ Daily Internet Usage    : num  256 194 236 246 226 ...
##  $ Ad Topic Line           : chr  "Cloned 5thgeneration orchestration" "Mo
nitored national standardization" "Organic bottom-line service-desk" "Triple-
```

```
buffered reciprocal time-frame" ...
##  $ City                    : chr  "Wrightburgh" "West Jodi" "Davidton" "We
st 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" "2016-01-10 02:31:19" ...
##  $ Clicked on Ad           : int  0 0 0 0 0 0 0 1 0 0 ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

#We observe that out dataset has columns as listed, the datatypes are in numbers, integers and character/string. We will change the columns appropriately.We note that Ad Topic Line, City,Male, country and Clicked on Ad are categorical data whose data type is integer but should be factors.Factors are variables in R which take on a limited number of different values; such variables are often referred to as categorical variables.We will change this to the right datatype format.

##Details of the columns

#Daily Time Spent on Site: the daily time spent in minutes and seconds

#Age: Age of the individuals

#Area Income:Income earned in that area

#Daily Internet Usage:Daily usgae of internet

#Ad Topic Line:Topic of the Ad

#City:City the individual comes from

#Male columns: This represents 0: for female and 1: male

#Country: Name of country

#Time stamp:Time in year,month,date,hour,minutes and seconds

#Clicked on Ad:Chances of clicking on the Ad or not. 0: Not click, 1:click on the Ad

#Checking the class of the dataset

```
class(advertising)
```

```
## [1] "data.table" "data.frame"
```

#We will change the datatypes in integer format to factor. We will leave the ones in character datatype as we will still be able to get the information

```
advertising$Male <- as.factor(advertising$Male)
advertising$Clicked_on_Ad <- as.factor(advertising$Clicked_on_Ad)
```

#Spliting the time stamp column into year, Month, day, hour and minute for ease of determining which year,month,day,hour, minute individuals are likely to click on the Ad or not

```
advertising$year <- format(as.POSIXct(advertising$Timestamp, format="%Y-%m-%d %H:%M:%S"), "%Y")
advertising$month <- format(as.POSIXct(advertising$Timestamp, format="%Y-%m-%d %H:%M:%S"), "%m")
advertising$day <- format(as.POSIXct(advertising$Timestamp, format="%Y-%m-%d %H:%M:%S"), "%d")
advertising$hour <- format(as.POSIXct(advertising$Timestamp, format="%Y-%m-%d %H:%M:%S"), "%H")
advertising$minute <- format(as.POSIXct(advertising$Timestamp, format="%Y-%m-%d %H:%M:%S"), "%M")
```

#Printing the head to confirm this has been effected head(advertising)

#Check the data structure to estblish the data types of date

str(advertising)

#We note that year,month,day, hour and minute are in character datatype. We will change this to factor

```
advertising$year <- as.factor(advertising$year)
advertising$year <- as.factor(advertising$month)
advertising$year <- as.factor(advertising$day)
advertising$year <- as.factor(advertising$hour)
advertising$year <- as.factor(advertising$minute)
```

#Checking for missing values in the dataset

```
colSums(is.na(advertising))
```

```
## 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             Clicked_on_Ad                      year
##                        0                      1000                         0
##                    month                       day                      hour
##                        0                         0                         0
##                   minute
##                        0
```

#We note that our dataset does not have missing values. So we will not need to omit or replace them.

#Checking for duplicates in our dataset

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

```
## Empty data.table (0 rows and 16 cols): Daily Time Spent on Site,Age,Area I
ncome,Daily Internet Usage,Ad Topic Line,City...
```

#We observe that our dataset does not have duplicates

3. #**Univariate Graphical Exploratory Data Analysis**

#a). Measures of Central Tendency #Checking the mean of numerical Variables

```
Age.mean <- mean(advertising$Age)
Age.mean
```

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

#The mean age of individuals is 36 years

```
Age.median <-median(advertising$Age)
Age.median
```

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

#The median age of individuals is 35 years

#Calculating the mode using the getmode() function for age variable

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

mode.age <- getmode(advertising$Age)
print(mode.age)
```

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

#Mode of age is 31

#b). Measures of Dispersion

#Checking the minimum age

```
Age.min <-min(advertising$Age)
Age.min
```

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

#The minimum age of individual is 19 years

#Checking the maximum age

```
Age.max <-max(advertising$Age)
Age.max
```

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

#Maximum age is 61 years

#Checking the range(Difference between highest age and lowest age)

```
Age.range <-range(advertising$Age)
Age.range

## [1] 19 61
```

#Getting the first and third quantile and range using the quantile function

```
Age.quantile <-quantile(advertising$Age)
Age.quantile

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

#The age of 25%centile is 29, 50%centile is 35 and 3rd quantile is 42

#Finding the variance of age. This is a numerical measure of how the data values is dispersed around the mean.

```
Age.variance<-var(advertising$Age)
Age.variance

## [1] 77.18611
```

#Finding the standard deviation of age.

```
Age.sd<-sd(advertising$Age)
Age.sd

## [1] 8.785562
```

#Standard deviation of age is 8.78, this is a measure of spread around the mean.

#To get measures of central tendancy and dispersion for the other numerical varibles, we can use the summary function.

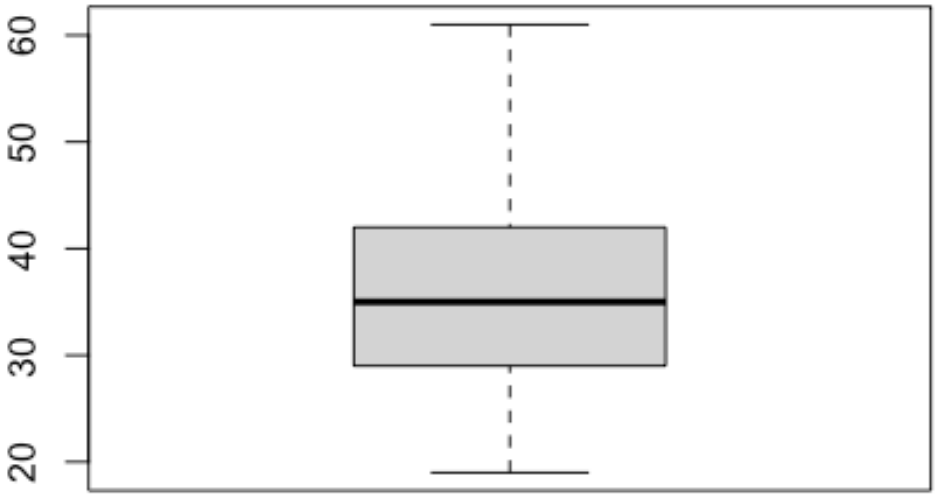
```
summary(advertising)

##  Daily Time Spent on Site      Age         Area Income    Daily Internet U
sage
##  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        0:519   Length:1000
```

```
##  Class :character   Class :character   1:481   Class :character
##  Mode  :character   Mode  :character           Mode  :character
##
##
##
##
##   Timestamp         Clicked on Ad Clicked_on_Ad      year
##  Length:1000        Min.   :0.0   NA's:1000     02     : 26
##  Class :character   1st Qu.:0.0                 07     : 24
##  Mode  :character   Median :0.5                 13     : 24
##                     Mean   :0.5                 10     : 22
##                     3rd Qu.:1.0                 21     : 21
##                     Max.   :1.0                 33     : 21
##                                                 (Other):862
##     month              day                hour              minute
##  Length:1000        Length:1000        Length:1000        Length:1000
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
```

#c). Univariate analysis

#Plotting boxplot for Age variable

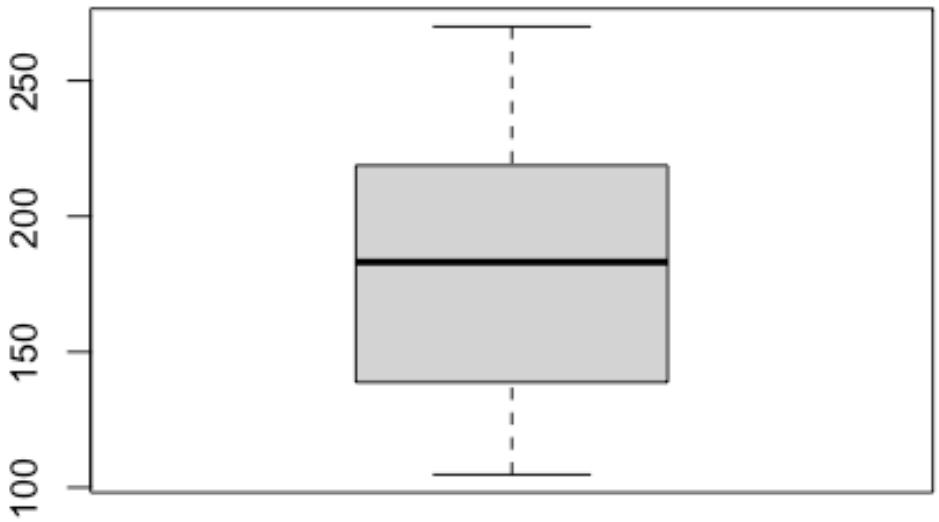
```
boxplot(advertising$Age)
```



#We observe that age has no outliers

#Plotting boxplot for Daily Internet Usage variable

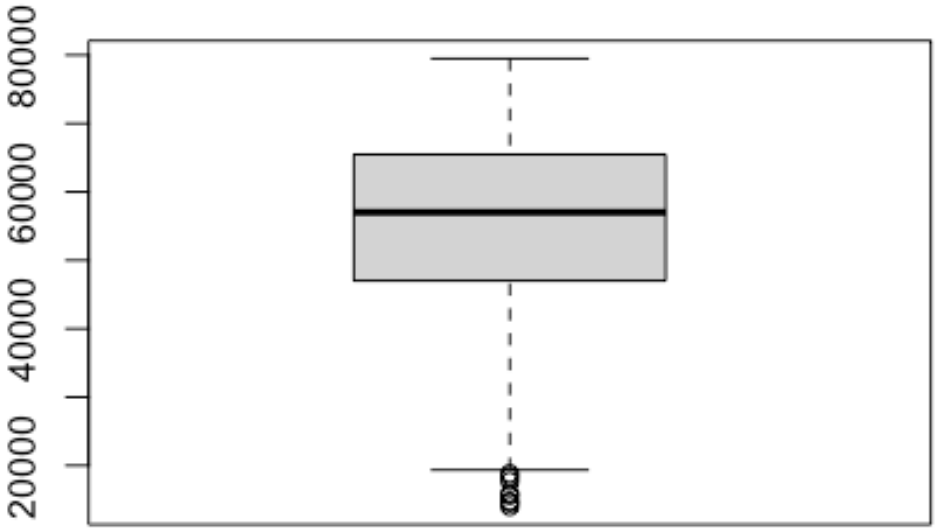
```
boxplot(advertising$'Daily Internet Usage')
```



#Daily Internet Usage has no outliers

#Plotting boxplot for Area Income variable

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



#Area Income has some outliers

#Checking the number of male and female individuals represented

```
Male_table <- table(advertising$Male)
Male_table

##
##   0   1
## 519 481
```

#We observe that we have 519 female and 481 Male in the dataset

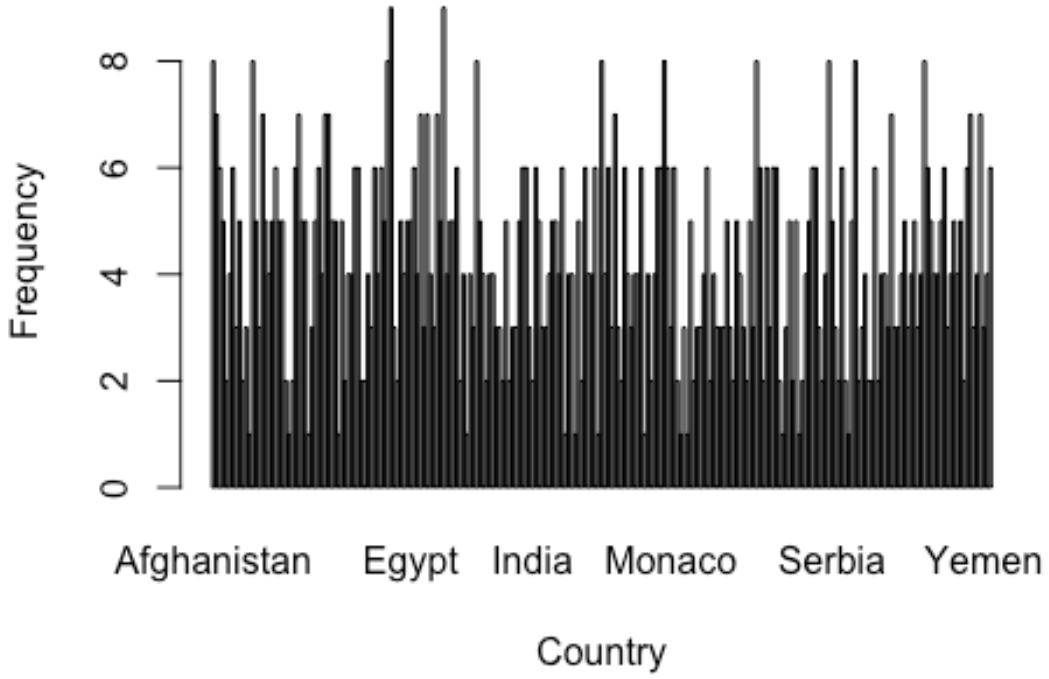
#Checking the number of those who clicked on the Ad and those individuals who did not click

```
Clicked_on_Ad_table <- table(advertising$`Clicked on Ad`)
Clicked_on_Ad_table

##
##   0   1
## 500 500
```

#We observe that individuals who clicked on Ad 500, same as those who did not click on the Ad

#Plotting barplot frequency by country

```
Country <-advertising$Country
Country_frequency <-table(Country)
barplot(Country_frequency,xlab='Country',ylab='Frequency')
```



#For better visualization we can check the top 6 Countries that occur freequently

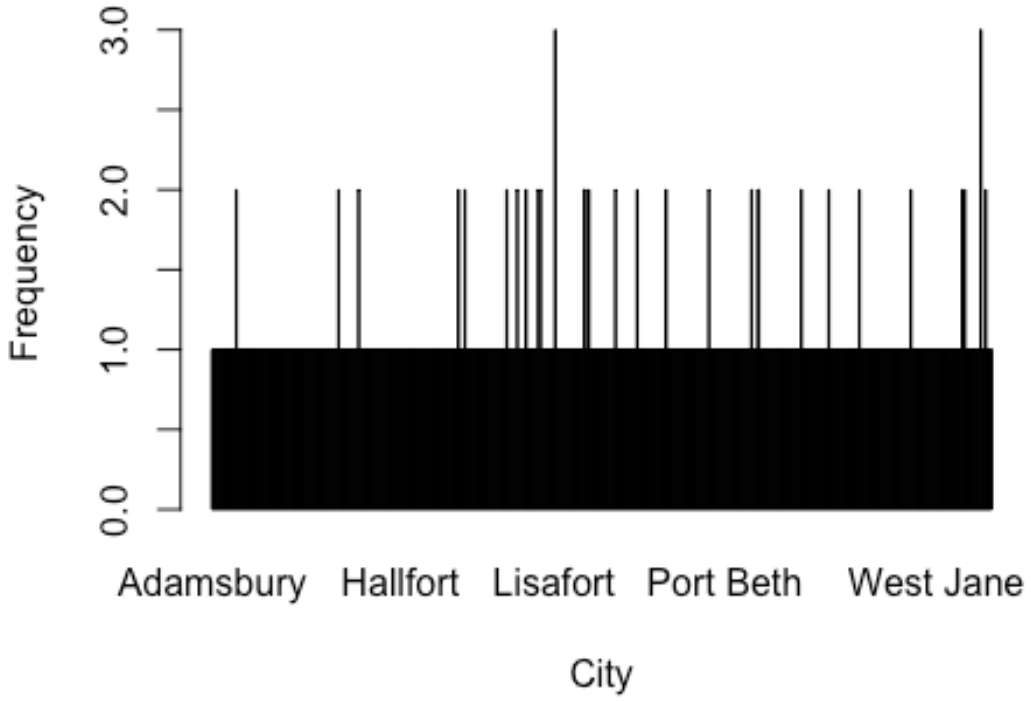
```
    library(plyr)
    frequency_Country<- count(advertising$Country)
    frequency_Country_head <- head(arrange(frequency_Country, desc(freq)))
  frequency_Country_head
```

```
##                x freq
## 1 Czech Republic    9
## 2         France    9
## 3    Afghanistan    8
## 4      Australia    8
## 5         Cyprus    8
## 6         Greece    8
```

#Czech Republic and France have the highest frequency

#Plotting frequency by city

```
City <-advertising$City
City_frequency <-table(City)
barplot(City_frequency,xlab='City',ylab='Frequency')
```



#For better visualization we can check the top 6 Cities that occur frequently

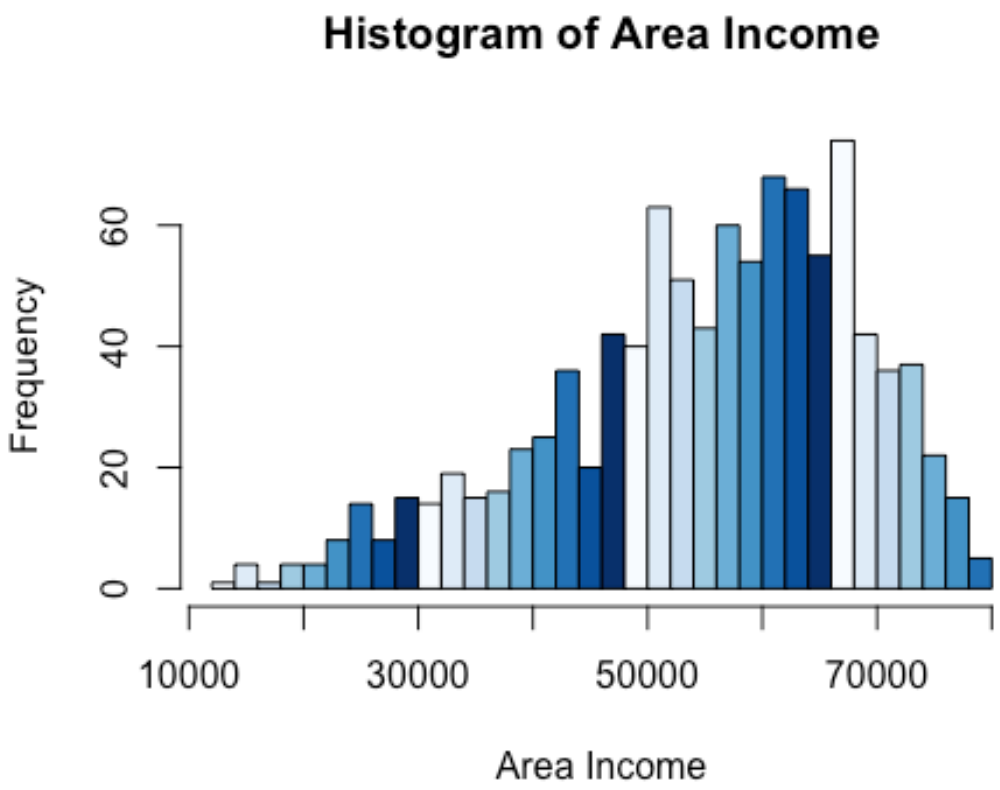
```
frequency_City<- count(advertising$City)
    frequency_City_head <- head(arrange(frequency_City, desc(freq)))
  frequency_City_head
```

```
##                 x freq
## 1       Lisamouth    3
## 2    Williamsport    3
## 3 Benjaminchester    2
## 4       East John    2
## 5    East Timothy    2
## 6        Johnstad    2
```

Lisamoth and Williamsport are top 2 cities

#Histogram for Area Income

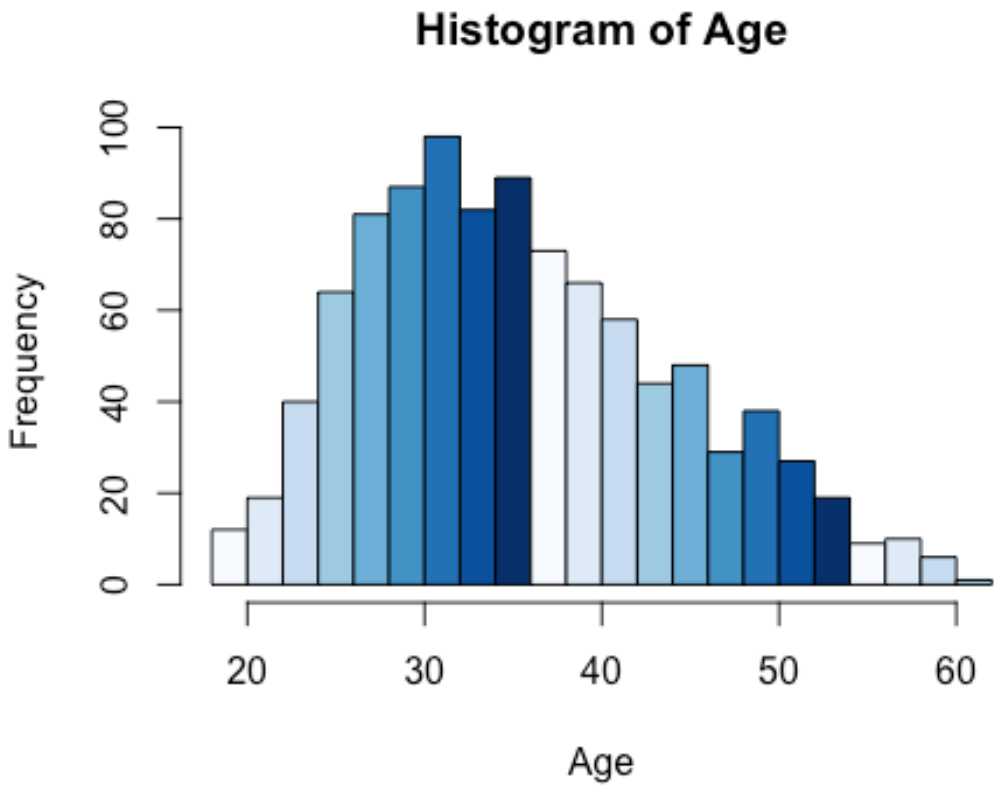
```
hist(advertising$`Area Income`, col=blues9,breaks=25,xlab="Area Income",main=
"Histogram of Area Income")
```



#Area income is negatively skewed and most of the area income is about 6000.

#Histogram- Age of individuals

```
hist(advertising$`Age`, col=blues9,breaks=25,xlab="Age",main="Histogram of Age")
```

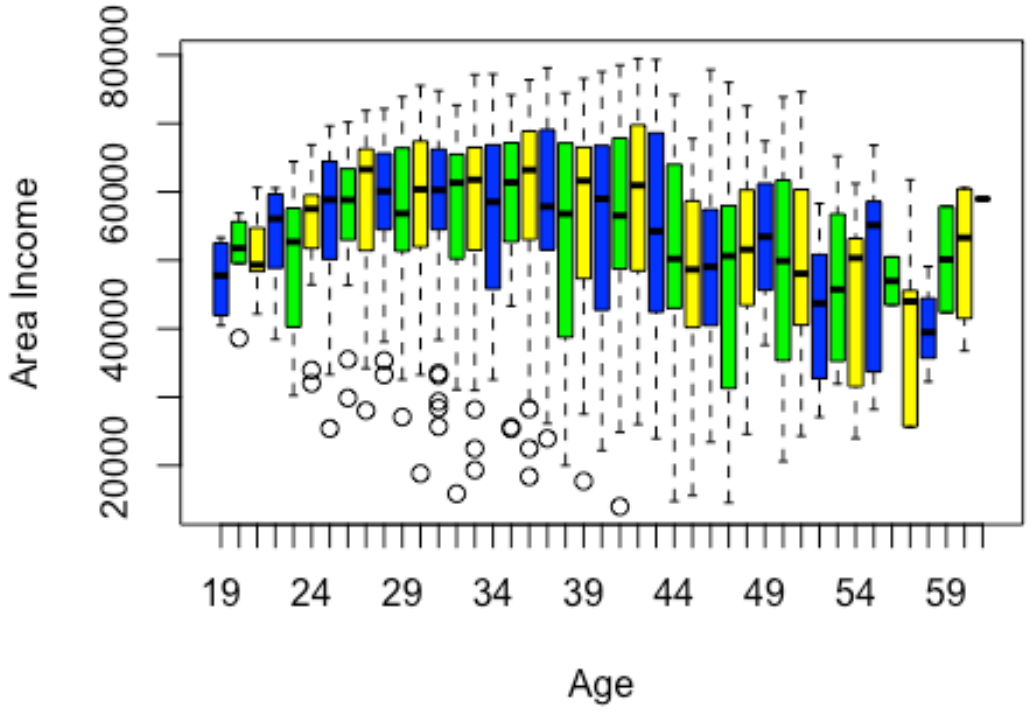


#Age distribution is positively skewed with median age being 35years and mean age 36years.

#c). Bivariate analysis

#Plotting boxplot for Area Income and Age variables

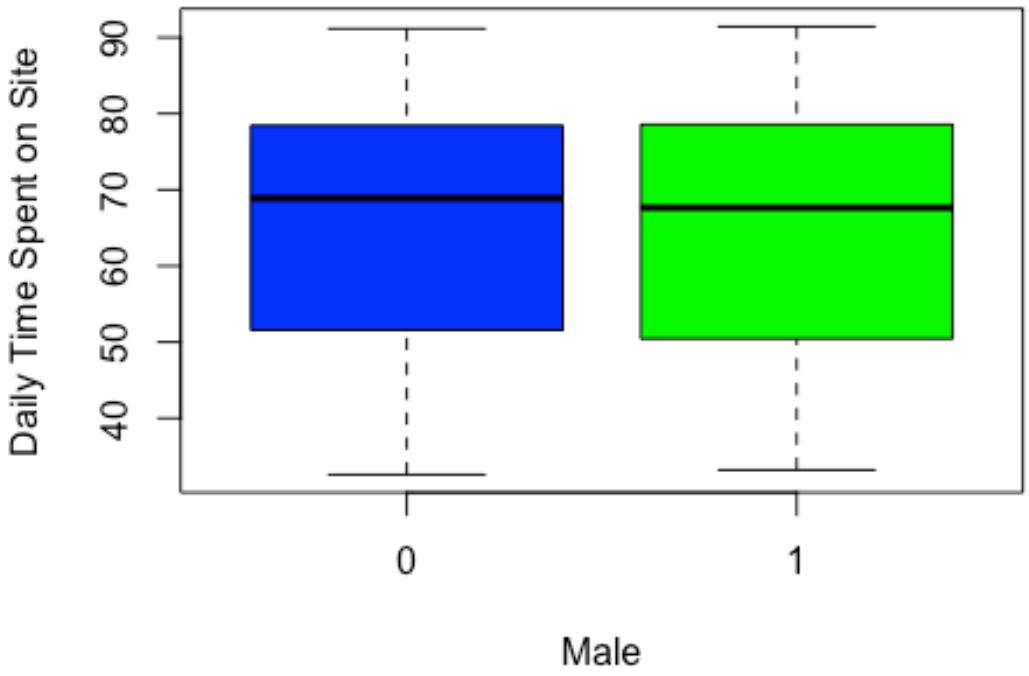
```
boxplot(advertising$'Area Income'~advertising$Age,xlab="Age", ylab="Area Income", notch=FALSE, col=c("blue","green","yellow"))
```



#We note that Area income increases with increase in age and begins to decline from age 43. We observe outliers in area income below the age of 40.

#Plotting boxplot for Daily Time Spent on Site and Male variables

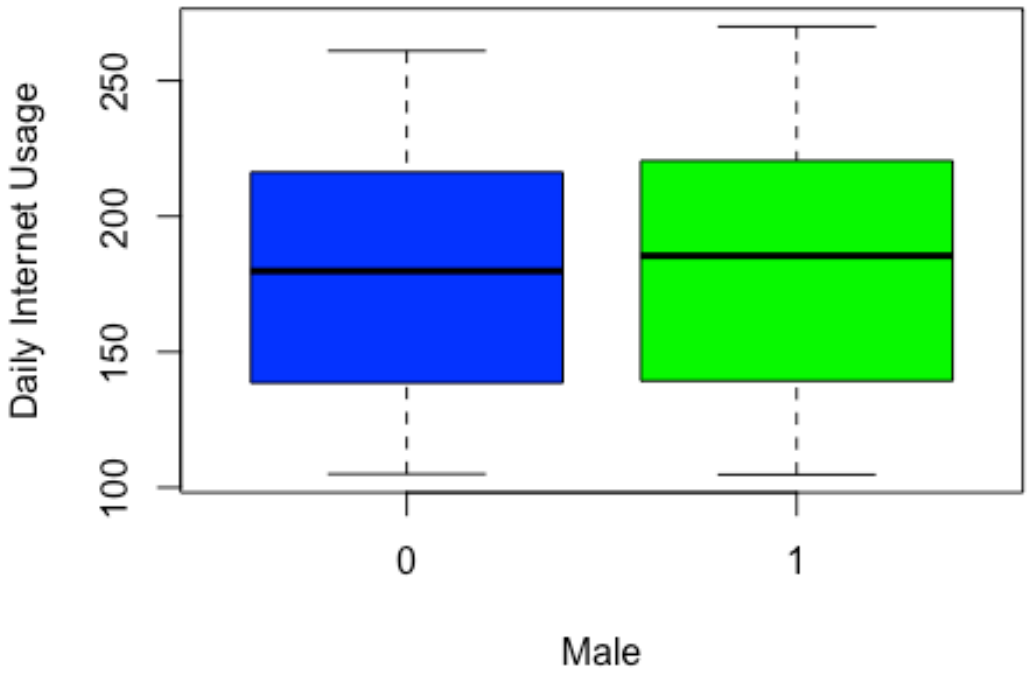
```
boxplot(advertising$'Daily Time Spent on Site'~advertising$Male,xlab="Male",
ylab="Daily Time Spent on Site",notch=FALSE, col=c("blue","green"))
```



#Time spent on the internet for both male and female is more or less the same

#Plotting boxplot for Daily Time Spent on Site and Male variables

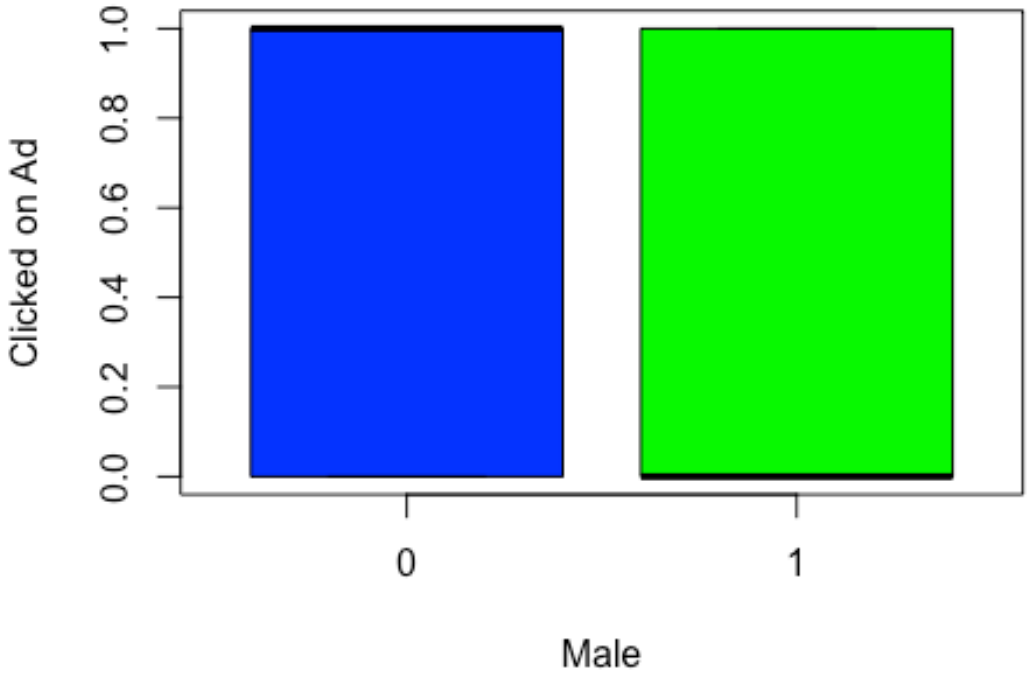
```
boxplot(advertising$'Daily Internet Usage'~advertising$Male,xlab="Male", ylab
="Daily Internet Usage",notch=FALSE, col=c("blue","green"))
```



#Daily internet usage for male is slightly higher than that of female with no outliers.

#Plotting relationship of those who clicked on the Ad by Gender using boxplot

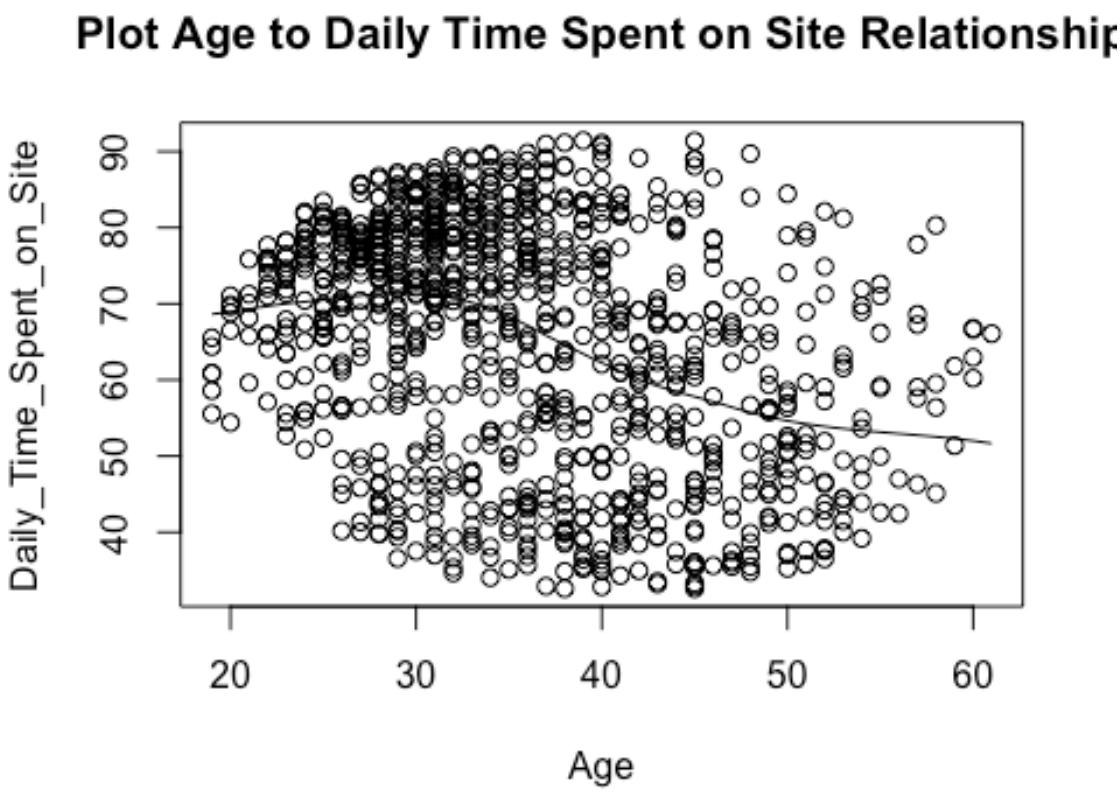
```
boxplot(advertising$'Clicked on Ad'~advertising$Male,xlab="Male", ylab="Click
ed on Ad",notch=FALSE, col=c("blue","green"))
```



#From the above we observe that the number of male and female who clicked on the Ad are more or less the same.Take a 50/50 kind of representation, hence either gender has chances of clicking on the Ad.

#The correlation between age and time spent on the site

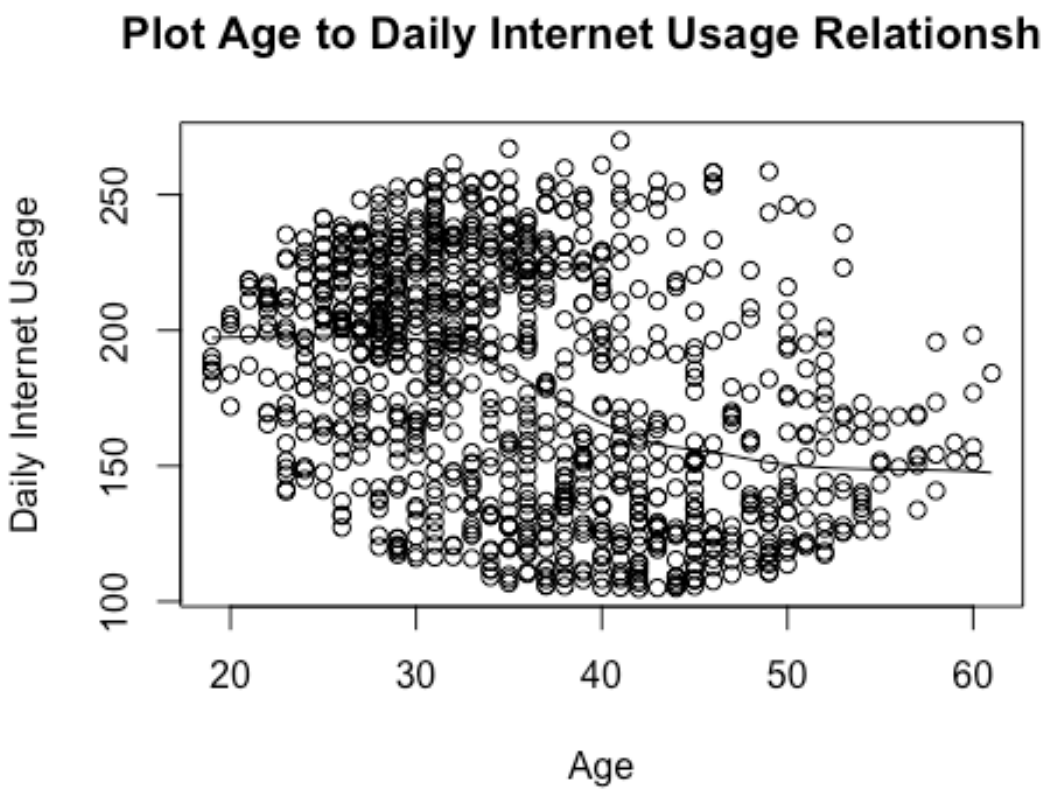
```
scatter.smooth(advertising$Age,advertising$`Daily Time Spent on Site`,main="P
lot Age to Daily Time Spent on Site Relationship",xlab = 'Age',ylab = 'Daily_
Time_Spent_on_Site')
```



#We observe that daily time spent site is higher with younger individuals, this time declines with age. We observe a peak at age 30years.

#Plotting relationship between Age and Daily Internet Usage

```
scatter.smooth(advertising$Age,advertising$`Daily Internet Usage`,main="Plot
Age to Daily Internet Usage Relationship",xlab = 'Age',ylab = 'Daily Internet
Usage')
```



#We observe that Daily Internet Usage declines with increase in age.

#Checking the relationship gender and clicking on the ad

```
scatter.smooth(advertising$Male,advertising$`Clicked on Ad`,main="Plot Male t
o Clicked on Ad Relationship",xlab = 'Male',ylab = 'Clicked on Ad')
```

```
## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = FALSE,
:
## pseudoinverse used at 0.995
```

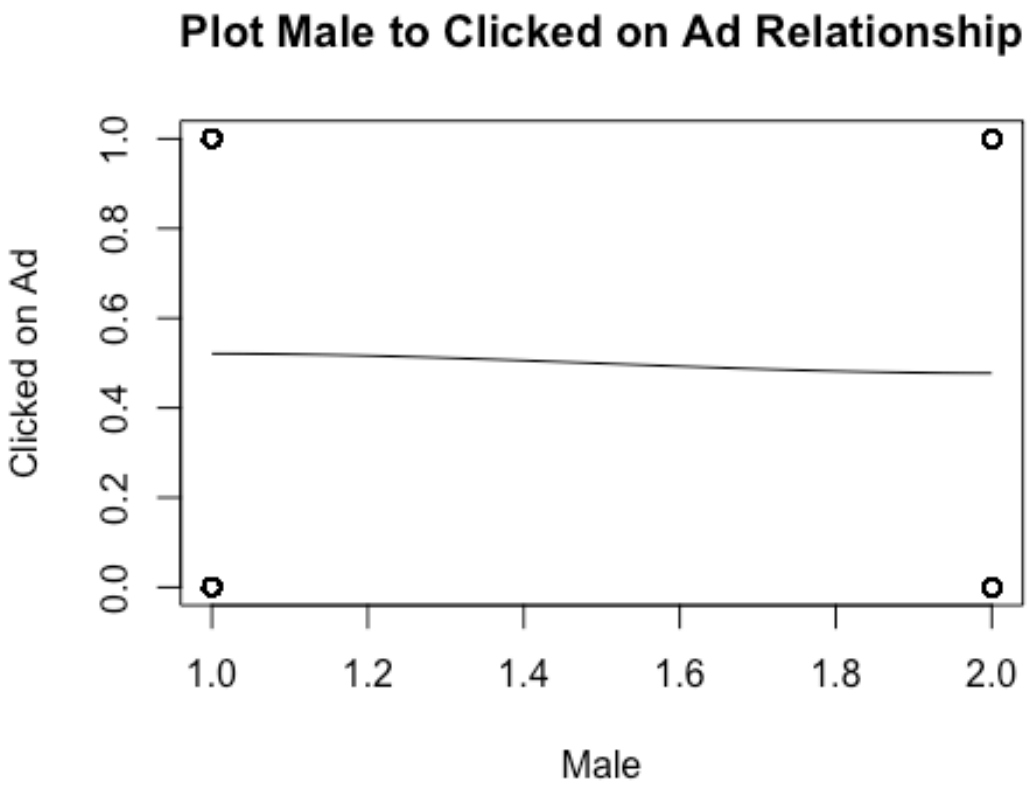
```
## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = FALSE,
:
## neighborhood radius 1.005
```

```
## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = FALSE,
:
## reciprocal condition number 5.4485e-15
```

```
## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = FALSE,
:
## There are other near singularities as well. 1.01
```

```
## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = FALSE,
:
## pseudoinverse used at 0.995

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = FALSE,
:
## neighborhood radius 1.005

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = FALSE,
:
## reciprocal condition number 4.5392e-15

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = FALSE,
:
## There are other near singularities as well. 1.01

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = FALSE,
:
## pseudoinverse used at 0.995

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = FALSE,
:
## neighborhood radius 1.005

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = FALSE,
:
## reciprocal condition number 1.46e-15

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = FALSE,
:
## There are other near singularities as well. 1.01

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = FALSE,
:
## pseudoinverse used at 0.995

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = FALSE,
:
## neighborhood radius 1.005

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = FALSE,
:
## reciprocal condition number 1.9741e-15

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = FALSE,
:
## There are other near singularities as well. 1.01

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = FALSE,
:
## pseudoinverse used at 0.995
```

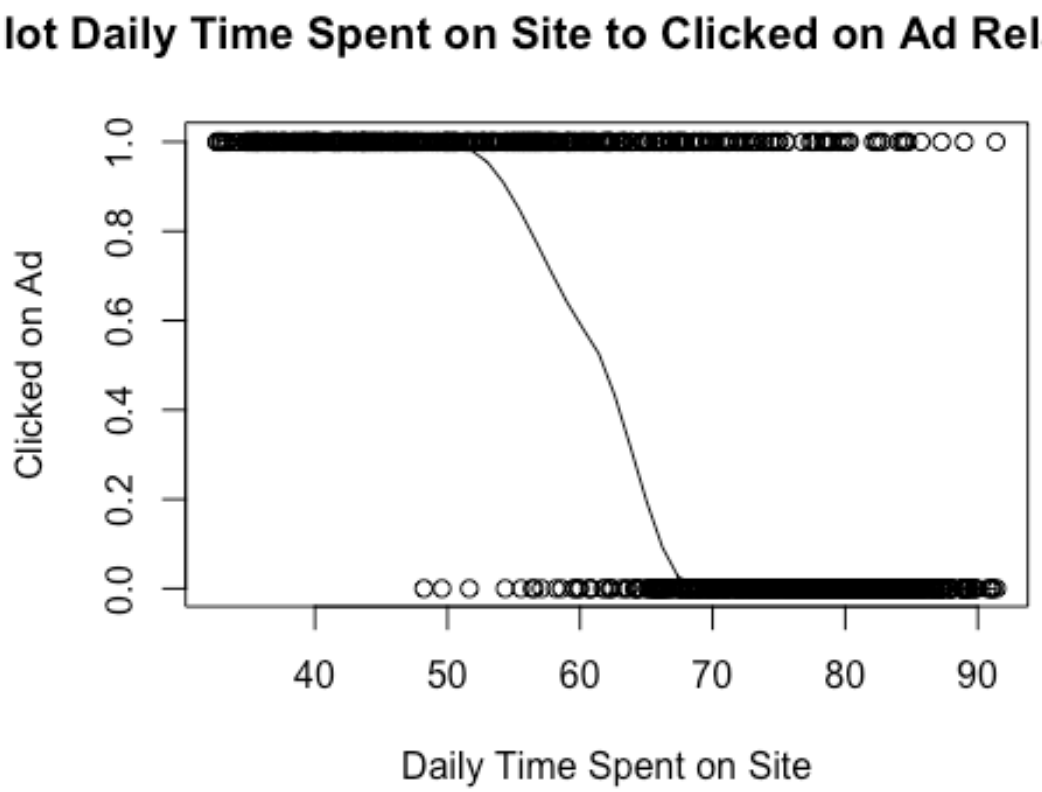
```
## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = FALSE,
:
## neighborhood radius 1.005

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = FALSE,
:
## reciprocal condition number 5.4485e-15

## Warning in simpleLoess(y, x, w, span, degree = degree, parametric = FALSE,
:
## There are other near singularities as well. 1.01
```



#Gender is not a key determinant when it comes to clicking on the Ad both male and female have equal chances of clicking on the Ad.

#Checking clicking on Ad by Daily Time Spent on Site

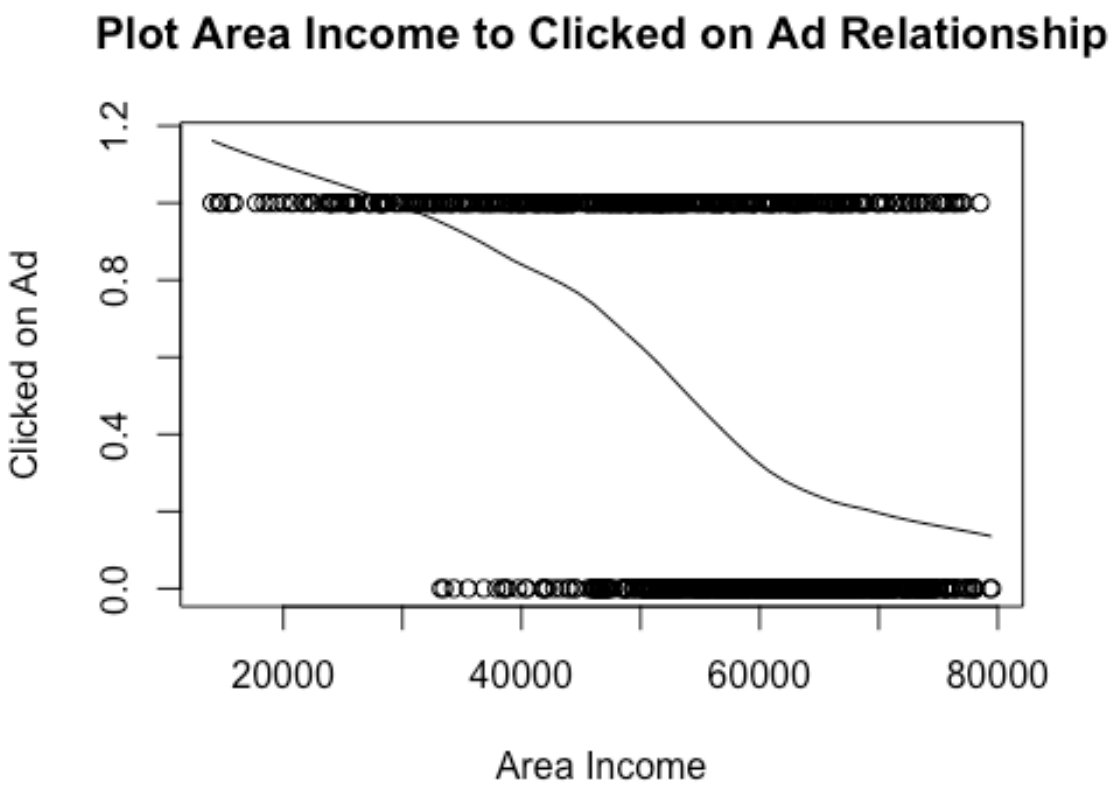
```
scatter.smooth(advertising$`Daily Time Spent on Site`,advertising$`Clicked on
Ad`,main="Plot Daily Time Spent on Site to Clicked on Ad Relationship",xlab =
'Daily Time Spent on Site',ylab = 'Clicked on Ad')
```



#The lower the Daily time spent on Site the higher the chances of clicking on Ad.

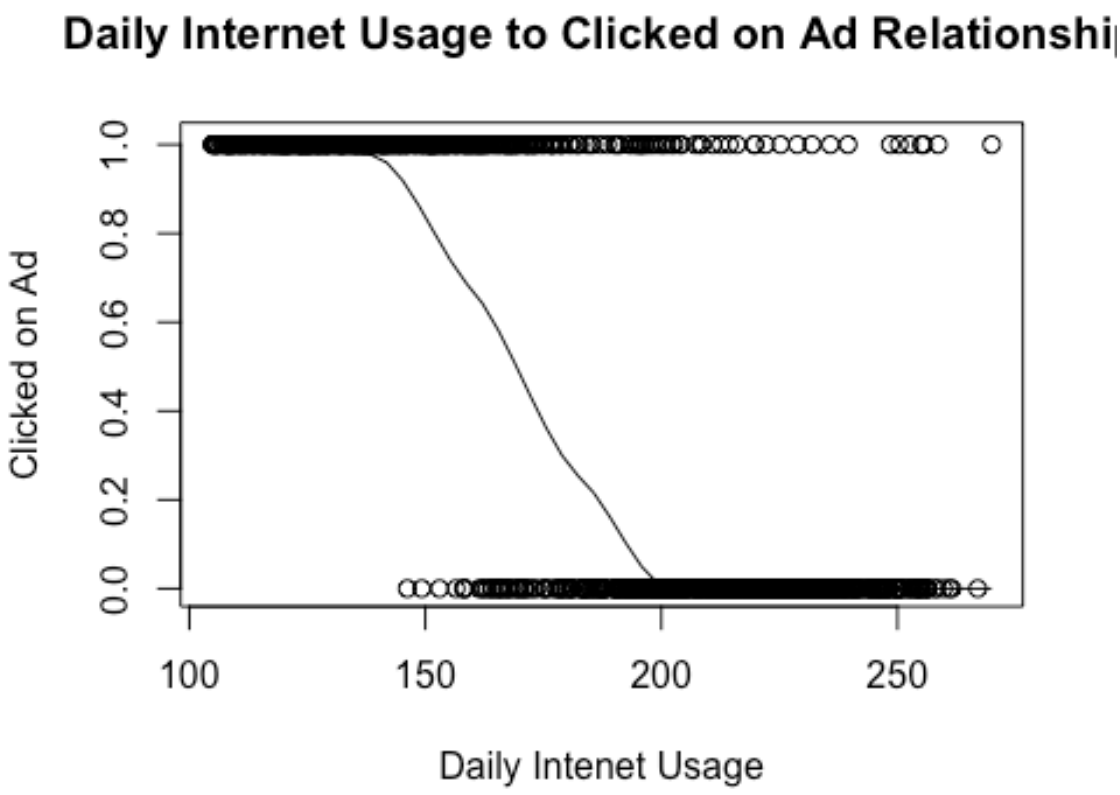
#Checking relationship clicking on Ad to Area Income

```
scatter.smooth(advertising$`Area Income`,advertising$`Clicked on Ad`,main="Pl
ot Area Income to Clicked on Ad Relationship",xlab = 'Area Income',ylab = 'Cl
icked on Ad')
```



#The lower the Area Income the higher the chances of clicking on Ad.As income increases the probability of clicking on Ad declines.

```
scatter.smooth(advertising$`Daily Internet Usage`,advertising$`Clicked on Ad`
,main="Daily Internet Usage to Clicked on Ad Relationship",xlab = 'Daily Inte
net Usage',ylab = 'Clicked on Ad')
```



#We observe that the lower the daily internet Usage the higher the chances of clicking on the Ad.

```
input <- cor(advertising[,c("Daily Internet Usage","Area Income","Daily Time
Spent on Site","Age","Clicked on Ad")])
round(input,2)
```

```
##                          Daily Internet Usage Area Income
## Daily Internet Usage                     1.00        0.34
## Area Income                              0.34        1.00
## Daily Time Spent on Site                 0.52        0.31
## Age                                     -0.37       -0.18
## Clicked on Ad                           -0.79       -0.48
##                          Daily Time Spent on Site   Age Clicked on Ad
## Daily Internet Usage                         0.52 -0.37         -0.79
## Area Income                                  0.31 -0.18         -0.48
## Daily Time Spent on Site                     1.00 -0.33         -0.75
## Age                                         -0.33  1.00          0.49
## Clicked on Ad                               -0.75  0.49          1.00
```

#Checking the individuals who clicked the Ad by year

```
year.table <- table(advertising$'Clicked on Ad', advertising$year)
names(dimnames(year.table)) <- c("Clicked on Ad", "year")
year.table
```

```
##              year
## Clicked on Ad 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19
20 21
##             0 11  9 12  1  7  9  8 10  9  3 13  4  5 13  6 11  8  9 12  8
7 13
##             1  9  9 14  5  7  4  8 14 11  7  9 11  9 11 12  4  7 11  8  7
5  8
##              year
## Clicked on Ad 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41
42 43
##             0  7 11  4  7 11 10  7  7 15 10 10 12  9 14  9  9 10  8  6 11
7  4
##             1  9  8  9  8  5  7  6  2  4  8  8  9 10  5 12 11  7 12  7  9
10 10
##              year
## Clicked on Ad 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59
##             0  3  8  7  8 10  7  8  8  6  5  7  9  6  9  9  4
##             1  8  8  7  5  8  8  9 10  9  9 12  5 10  8  8 10
```

#We observe that year 2002 and 2007 achieved the highest click on the Ad with 14 clicks

#Checking the those who clicked the Ad by month

```
month.table <- table(advertising$'Clicked on Ad', advertising$month)
names(dimnames(year.table)) <- c("Clicked on Ad", "month")
month.table
```

```
##
##     01 02 03 04 05 06 07
##   0 78 77 82 73 68 71 51
##   1 69 83 74 74 79 71 50
```

#We observe that most the highest number of click on the Ad were achieved in the month of February with 83 clicks followed by May with 79 clicks.

```
hour.table <- table(advertising$'Clicked on Ad', advertising$hour)
names(dimnames(hour.table)) <- c("Clicked on Ad", "hour")
hour.table
```

```
##              hour
## Clicked on Ad 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19
20 21
##             0 19 16 19 19 21 23 16 28 22 21 17 16 22 21 22 16 23 18 16 20
26 29
##             1 26 16 17 23 21 21 23 26 21 28 14 24 16 21 21 19 16 23 25 19
24 19
##              hour
## Clicked on Ad 22 23
##             0 24 26
##             1 19 18
```

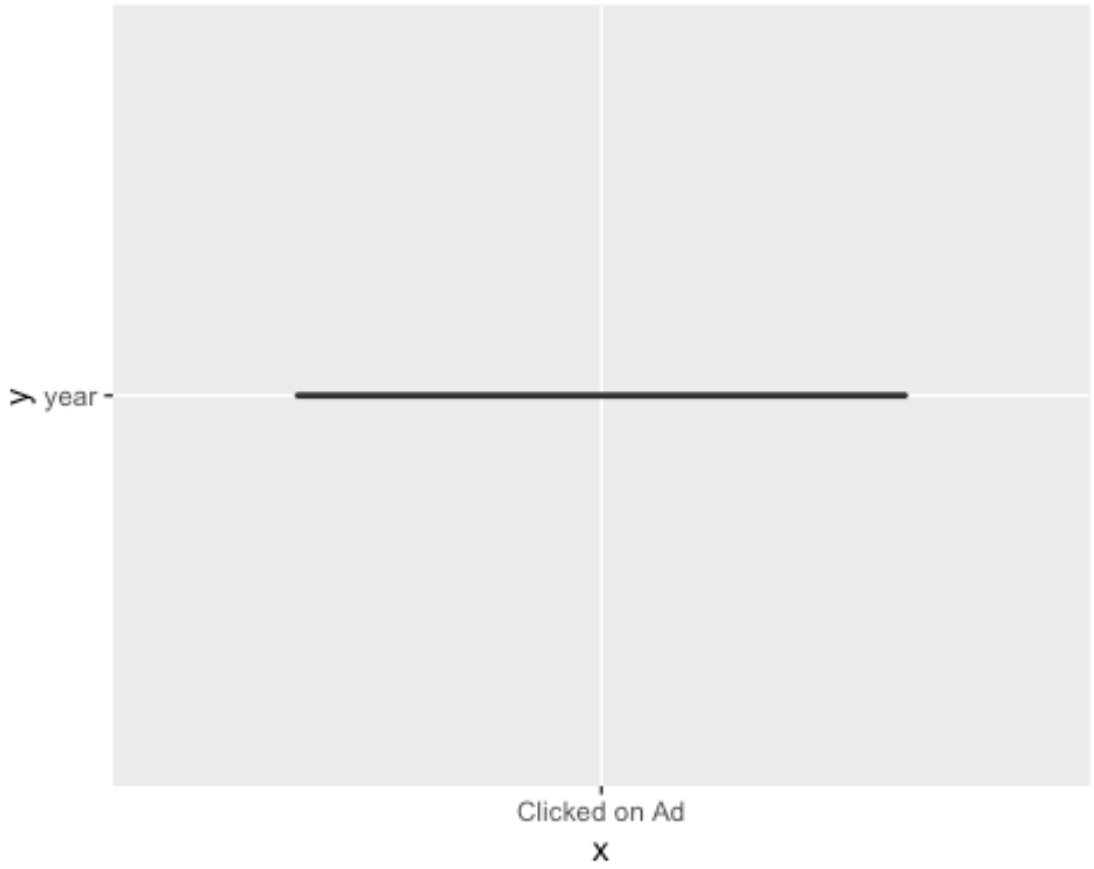
#We observe that the 9th hour achieved most clicks with 29clicks followed but the the 1st hour.

```
minute.table <- table(advertising$'Clicked on Ad', advertising$minute)
names(dimnames(minute.table)) <- c("Clicked on Ad", "minute")
minute.table
```

```
##              minute
## Clicked on Ad 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19
20 21
##             0 11  9 12  1  7  9  8 10  9  3 13  4  5 13  6 11  8  9 12  8
7 13
##             1  9  9 14  5  7  4  8 14 11  7  9 11  9 11 12  4  7 11  8  7
5  8
##              minute
## Clicked on Ad 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41
42 43
##             0  7 11  4  7 11 10  7  7 15 10 10 12  9 14  9  9 10  8  6 11
7  4
##             1  9  8  9  8  5  7  6  2  4  8  8  9 10  5 12 11  7 12  7  9
10 10
##              minute
## Clicked on Ad 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59
##             0  3  8  7  8 10  7  8  8  6  5  7  9  6  9  9  4
##             1  8  8  7  5  8  8  9 10  9  9 12  5 10  8  8 10
```

#the 7th minute achieved the highest number of clicks on the Ad.

```
library(ggplot2)
ggplot(advertising, aes(x='Clicked on Ad', y='year')) + geom_boxplot()
```



4)

Implementing supervised learning with r

a)SVM in r

```
str(advertising)
```

```
## Classes 'data.table' and 'data.frame':   1000 obs. of  16 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" "Mo
nitored national standardization" "Organic bottom-line service-desk" "Triple-
buffered reciprocal time-frame" ...
##  $ City                    : chr  "Wrightburgh" "West Jodi" "Davidton" "We
st Terrifurt" ...
##  $ Male                    : Factor w/ 2 levels "0","1": 1 2 1 2 1 2 1 2 2
2 ...
##  $ 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" "2016-01-10 02:31:19" ...
##  $ Clicked on Ad           : int  0 0 0 0 0 0 0 1 0 0 ...
##  $ Clicked_on_Ad           : Factor w/ 0 levels: NA NA NA NA NA NA NA NA N
```

```
A NA ...
##  $ year                    : Factor w/ 60 levels "00","01","02",..: 54 40
36 32 37 31 60 41 34 43 ...
##  $ month                   : chr  "03" "04" "03" "01" ...
##  $ day                     : chr  "27" "04" "13" "10" ...
##  $ hour                    : chr  "00" "01" "20" "02" ...
##  $ minute                  : chr  "53" "39" "35" "31" ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

#Change the class label column to factor

```
advertising$`Clicked on Ad` = factor(advertising$`Clicked on Ad`)
```

#Checking if this has been effected

```
str(advertising)
```

```
## Classes 'data.table' and 'data.frame':   1000 obs. of  16 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" "Mo
nitored national standardization" "Organic bottom-line service-desk" "Triple-
buffered reciprocal time-frame" ...
##  $ City                    : chr  "Wrightburgh" "West Jodi" "Davidton" "We
st Terrifurt" ...
##  $ Male                    : Factor w/ 2 levels "0","1": 1 2 1 2 1 2 1 2 2
2 ...
##  $ 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" "2016-01-10 02:31:19" ...
##  $ Clicked on Ad           : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 2 1
1 ...
##  $ Clicked_on_Ad           : Factor w/ 0 levels: NA NA NA NA NA NA NA NA N
A NA ...
##  $ year                    : Factor w/ 60 levels "00","01","02",..: 54 40
36 32 37 31 60 41 34 43 ...
##  $ month                   : chr  "03" "04" "03" "01" ...
##  $ day                     : chr  "27" "04" "13" "10" ...
##  $ hour                    : chr  "00" "01" "20" "02" ...
##  $ minute                  : chr  "53" "39" "35" "31" ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

#check column names

```
colnames(advertising)
```

```
##  [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"
## [11] "Clicked_on_Ad"             "year"
## [13] "month"                     "day"
## [15] "hour"                      "minute"

names(advertising)[10]<-'click'
```

#Spliting the time stamp column into year, Month, day, hour and minute for ease of determining which year,month,day,hour, minute individuals are likely to click on the Ad or not

```
advertising$year <- format(as.POSIXct(advertising$Timestamp, format="%Y-%m-%d %H:%M:%S"), "%Y")
advertising$month <- format(as.POSIXct(advertising$Timestamp, format="%Y-%m-%d %H:%M:%S"), "%m")
advertising$day <- format(as.POSIXct(advertising$Timestamp, format="%Y-%m-%d %H:%M:%S"), "%d")
advertising$hour <- format(as.POSIXct(advertising$Timestamp, format="%Y-%m-%d %H:%M:%S"), "%H")
advertising$minute <- format(as.POSIXct(advertising$Timestamp, format="%Y-%m-%d %H:%M:%S"), "%M")
```

#Printing the head to confirm this has been effected head(advertising)

#Check the data structure to establish the data types of date str(advertising)

#We note that year,month,day, hour and minute are in character datatype. We will change this to factor

```
advertising$year <- as.factor(advertising$year)
advertising$year <- as.factor(advertising$month)
advertising$year <- as.factor(advertising$day)
advertising$year <- as.factor(advertising$hour)
advertising$year <- as.factor(advertising$minute)
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 click Clicked_on_Ad year month day hour minute
## 1: 2016-03-27 00:53:11     0          <NA>   53    03  27   00     53
## 2: 2016-04-04 01:39:02     0          <NA>   39    04  04   01     39
## 3: 2016-03-13 20:35:42     0          <NA>   35    03  13   20     35
## 4: 2016-01-10 02:31:19     0          <NA>   31    01  10   02     31
## 5: 2016-06-03 03:36:18     0          <NA>   36    06  03   03     36
## 6: 2016-05-19 14:30:17     0          <NA>   30    05  19   14     30
```

#Change the datatype of some variables to numeric and drop the caterical columns

```
advertising$'Male' <-as.numeric(advertising$'Male')
advertising$'Country' <-as.numeric(advertising$'Country')

## Warning: NAs introduced by coercion

advertising$'year' <-as.numeric(advertising$'year')
advertising$'month' <-as.numeric(advertising$'month')
advertising$'day' <-as.numeric(advertising$'day')
advertising$'hour' <-as.numeric(advertising$'hour')
advertising$'minute' <-as.numeric(advertising$'minute')
advertising$Timestamp <- NULL#remove the column as we no longer need it
advertising$'Ad Topic Line' <- NULL
advertising$City <- NULL
advertising$Country <- NULL
str(advertising)

## Classes 'data.table' and 'data.frame':   1000 obs. of  12 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 ...
##  $ Male                    : num  1 2 1 2 1 2 1 2 2 2 ...
##  $ click                   : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 2 1
1 ...
##  $ Clicked_on_Ad           : Factor w/ 0 levels: NA NA NA NA NA NA NA NA N
A NA ...
##  $ year                    : num  54 40 36 32 37 31 60 41 34 43 ...
##  $ month                   : num  3 4 3 1 6 5 1 3 4 7 ...
##  $ day                     : num  27 4 13 10 3 19 28 7 18 11 ...
##  $ hour                    : num  0 1 20 2 3 14 20 1 9 1 ...
##  $ minute                  : num  53 39 35 31 36 30 59 40 33 42 ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

#assigning click on Ad column to last column

```
advert<-advertising[ , c(1,2,3,4,5,7,8,9,10,11,6)]
head(advert)

##    Daily Time Spent on Site Age Area Income Daily Internet Usage Male
## 1:                    68.95  35    61833.90                256.09    1
## 2:                    80.23  31    68441.85                193.77    2
## 3:                    69.47  26    59785.94                236.50    1
```

```
## 4:                    74.15  29    54806.18                245.89    2
## 5:                    68.37  35    73889.99                225.58    1
## 6:                    59.99  23    59761.56                226.74    2
##    Clicked_on_Ad year month day hour click
## 1:          <NA>   54     3  27    0     0
## 2:          <NA>   40     4   4    1     0
## 3:          <NA>   36     3  13   20     0
## 4:          <NA>   32     1  10    2     0
## 5:          <NA>   37     6   3    3     0
## 6:          <NA>   31     5  19   14     0
```

```
library(caret)
```

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

```
intrain <- createDataPartition(y = advert$`click`, p= 0.7, list = FALSE)
training <- advert[intrain,]
testing <- advert[-intrain,]
head(training)
```

```
##    Daily Time Spent on Site Age Area Income Daily Internet Usage Male
## 1:                    80.23  31    68441.85                193.77    2
## 2:                    69.47  26    59785.94                236.50    1
## 3:                    74.15  29    54806.18                245.89    2
## 4:                    68.37  35    73889.99                225.58    1
## 5:                    88.91  33    53852.85                208.36    1
## 6:                    66.00  48    24593.33                131.76    2
##    Clicked_on_Ad year month day hour click
## 1:          <NA>   40     4   4    1     0
## 2:          <NA>   36     3  13   20     0
## 3:          <NA>   32     1  10    2     0
## 4:          <NA>   37     6   3    3     0
## 5:          <NA>   60     1  28   20     0
## 6:          <NA>   41     3   7    1     1
```

#We check the dimensions of out training dataframe and testing dataframe

```
dim(training);
```

```
## [1] 700  11
```

```
dim(testing);
```

```
## [1] 300  11
```

#We then clean the data using the anyNA() method that checks for any null values.

```
anyNA(advertising)
```

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

#Then check the summary of our data by using the summary() function

```
summary(advert)
```

```
##  Daily Time Spent on Site      Age         Area Income    Daily Internet U
sage
##  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       Clicked_on_Ad      year           month            day
##  Min.   :1.000   NA's:1000     Min.   : 1.00   Min.   :1.000   Min.   : 1.
00
##  1st Qu.:1.000                 1st Qu.:15.00   1st Qu.:2.000   1st Qu.: 8.
00
##  Median :1.000                 Median :31.00   Median :4.000   Median :15.
00
##  Mean   :1.481                 Mean   :30.05   Mean   :3.817   Mean   :15.
48
##  3rd Qu.:2.000                 3rd Qu.:44.00   3rd Qu.:5.000   3rd Qu.:23.
00
##  Max.   :2.000                 Max.   :60.00   Max.   :7.000   Max.   :31.
00
##       hour       click
##  Min.   : 0.00   0:500
##  1st Qu.: 6.00   1:500
##  Median :12.00
##  Mean   :11.66
##  3rd Qu.:18.00
##  Max.   :23.00
```

#From our output above, we can see that the values of the various variables are not standardized.

#Training SVM model

#Before we train our model we will need to control all the computational overheads. #We will implement this through the trainControl() method. #This will allow us to use the train() function provided by the caret package. #The trainControl method will take three parameters: #a)The “method” parameter defines the resampling method, #in this demo we’ll be using the repeatedcv or the repeated cross-validation method. #b)The next parameter is the “number”, this basically holds the number of resampling iterations. #c)The “repeats” parameter contains the sets to compute for our repeated cross-validation. # We are using setting number =10 and repeats =3

```
#trctrl <- trainControl(method = "repeatedcv", number = 10, repeats = 3)
#svm_Linear <- train(click ~., data = training, method = "svmLinear",
#trControl=trctrl,
#preProcess = c("center", "scale"),
#tuneLength = 10)
```

```
#svm_Linear
```

#We can use the predict() method for predicting results as shown below. #We pass 2 arguements, our trained model and our testing data frame.

```
#test_pred <- predict(svm_Linear, newdata = testing)
#test_pred

#confusionMatrix(table(test_pred, testing$click))
```

#Our SVM model achieved an accuracy level of 96.7% which is good, with most of the clicks classified right save for 10 clicks that were misclassified.

b)Decision Trees in r

install.packages("rpart") install.packages("rpart.plot")

```
library(rpart)
```

#splitting our data into training and testing sets #we will split it 90:10

```
library(rpart)
data_intraining <- createDataPartition(y = advert$click, p = 0.9, list = FALS
E)
training <- advert[data_intraining,]
testing <- advert[-data_intraining,]
```

#fitting and training the model using the decision tree classifier

```
#library(rpart)
#library(rpart.plot)
#fit <- rpart(click ~ ., data = training, method = 'class')
#rpart.plot(fit, extra = 106)

#In [90]:
# making predictions
#prediction <- predict(fit, testing, type = 'class')
#In [91]:
# comparing predicted values to actual results
#table_dec <- table(testing$click, prediction)
#table_dec
```

#Predictions

```
#prediction <- predict(fit, testing, type = 'class')
#prediction
```

install.packages("gmodels") library(gmodels)

#comparing predicted values to actual results

```
#table_dec <- table(testing$click, prediction)
#table_dec
```

#From the confusion matrix the model had performe well with most of the data points classified right except 7 that were misclassified.

#Checking the performance of the model using accuracy metric

```
#model_accuracy <- sum(diag(table_dec)) / sum(table_dec)
#print(paste('Accuracy:', model_accuracy))
```

#We observe that the decision tree model accuracy is 96.5% this is good performance, its slightly lower than the performance that SVM achieved.

c)Naive Bayes with r

#printing the head of the dataset

```
head(advert)
```

```
##    Daily Time Spent on Site Age Area Income Daily Internet Usage Male
## 1:                    68.95  35    61833.90                256.09    1
## 2:                    80.23  31    68441.85                193.77    2
## 3:                    69.47  26    59785.94                236.50    1
## 4:                    74.15  29    54806.18                245.89    2
## 5:                    68.37  35    73889.99                225.58    1
## 6:                    59.99  23    59761.56                226.74    2
##    Clicked_on_Ad year month day hour click
## 1:          <NA>   54     3  27    0     0
## 2:          <NA>   40     4   4    1     0
## 3:          <NA>   36     3  13   20     0
## 4:          <NA>   32     1  10    2     0
## 5:          <NA>   37     6   3    3     0
## 6:          <NA>   31     5  19   14     0
```

#Change class label to factor

```
advert$click<-as.factor(advert$click)
str(advert)
```

```
## Classes 'data.table' and 'data.frame':   1000 obs. of  11 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 ...
##  $ Male                    : num  1 2 1 2 1 2 1 2 2 2 ...
##  $ Clicked_on_Ad           : Factor w/ 0 levels: NA NA NA NA NA NA NA NA N
A NA ...
##  $ year                    : num  54 40 36 32 37 31 60 41 34 43 ...
##  $ month                   : num  3 4 3 1 6 5 1 3 4 7 ...
##  $ day                     : num  27 4 13 10 3 19 28 7 18 11 ...
##  $ hour                    : num  0 1 20 2 3 14 20 1 9 1 ...
##  $ click                   : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 2 1
1 ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

```r
library(psych)
```

```
##
## Attaching package: 'psych'
```

```
## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

```r
data(advert)
```

```
## Warning in data(advert): data set 'advert' not found
```

```r
describe(advert)
```

```
## Warning in FUN(newX[, i], ...): no non-missing arguments to min; returning
Inf
```

```
## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning
-Inf
```

```
##                           vars    n     mean       sd   median  trimmed
mad
## Daily Time Spent on Site    1 1000    65.00    15.85    68.22    65.74
17.92
## Age                         2 1000    36.01     8.79    35.00    35.51
8.90
## Area Income                 3 1000 55000.00 13414.63 57012.30 56038.94 133
16.62
## Daily Internet Usage        4 1000   180.00    43.90   183.13   179.99
58.61
## Male                        5 1000     1.48     0.50     1.00     1.48
0.00
## Clicked_on_Ad*              6    0      NaN       NA       NA      NaN
NA
## year                        7 1000    30.05    17.25    31.00    29.99
22.24
## month                       8 1000     3.82     1.93     4.00     3.77
2.97
## day                         9 1000    15.48     8.73    15.00    15.42
10.38
## hour                       10 1000    11.66     6.96    12.00    11.69
8.90
## click*                     11 1000     1.50     0.50     1.50     1.50
0.74
##                              min      max    range  skew kurtosis     se
## Daily Time Spent on Site    32.60    91.43    58.83 -0.37    -1.10   0.50
## Age                         19.00    61.00    42.00  0.48    -0.41   0.28
## Area Income              13996.50 79484.80 65488.30 -0.65    -0.11 424.21
## Daily Internet Usage       104.78   269.96   165.18 -0.03    -1.28   1.39
## Male                         1.00     2.00     1.00  0.08    -2.00   0.02
## Clicked_on_Ad*                Inf     -Inf     -Inf    NA       NA     NA
```

```
## year                          1.00    60.00    59.00  0.02    -1.18   0.55
## month                         1.00     7.00     6.00  0.09    -1.19   0.06
## day                           1.00    31.00    30.00  0.04    -1.17   0.28
## hour                          0.00    23.00    23.00  0.00    -1.23   0.22
## click*                        1.00     2.00     1.00  0.00    -2.00   0.02
```

install.packages("caret") library(caret)

install.packages('tidyverse') library(tidyverse)

install.packages('ggplot2') library(ggplot2)

install.packages('caret') library(caret)

install.packages('caretEnsemble') library(caretEnsemble)

install.packages('psych') library(psych)

install.packages('Amelia') library(Amelia)

install.packages('mice') library(mice)

install.packages('GGally') library(GGally)

install.packages('rpart') library(rpart)

install.packages('randomForest') library(randomForest)

install.packages('lattice') library(lattice)

install.packages('Rcpp') library(Rcpp)

install.packages("numDeriv") library(numDeriv)

install.packages("caret") library(caret)

library(lattice) library(ggplot2)

#Splitting data into training and test data sets

```
set.seed(1234)
ind <- sample(2,nrow(advert),replace = T, prob = c(0.8,0.2))
train <-advert[ind == 1,]
test <- advert[ind ==2,]
```

#building Naive Bayes Model

```
library(e1071)
model <-naiveBayes(click~., data=train)
model

##
## Naive Bayes Classifier for Discrete Predictors
##
```

```
## Call:
## naiveBayes.default(x = X, y = Y, laplace = laplace)
##
## A-priori probabilities:
## Y
##         0         1
## 0.4911616 0.5088384
##
## Conditional probabilities:
##    Daily Time Spent on Site
## Y       [,1]      [,2]
##   0 76.94105  7.393651
##   1 53.68089 12.984956
##
##    Age
## Y       [,1]     [,2]
##   0 31.63496 6.205286
##   1 40.56824 8.828222
##
##    Area Income
## Y       [,1]      [,2]
##   0 61288.89  8872.077
##   1 48238.82 14357.075
##
##    Daily Internet Usage
## Y       [,1]     [,2]
##   0 214.8682 23.72015
##   1 145.2700 30.01940
##
##    Male
## Y       [,1]      [,2]
##   0 1.503856 0.5006290
##   1 1.464020 0.4993236
##
##    Clicked_on_Ad
## Y
##   0
##   1
##
##    year
## Y       [,1]     [,2]
##   0 30.10540 16.50325
##   1 30.83871 18.04696
##
##    month
## Y       [,1]     [,2]
##   0 3.845758 1.982361
##   1 3.816377 1.925485
##
##    day
```

```
## Y       [,1]     [,2]
##   0 15.39332 8.680590
##   1 15.01489 8.722635
##
##    hour
## Y       [,1]     [,2]
##   0 12.07198 7.139982
##   1 11.19603 6.808963
```

#We observe that 49% of train data did not click on the Ad and 50% clicked #Then we have mean and standard deviation of each variable for customers who clicked and those that did not click

`

#Model Evalution #Predicting our testing set

d)implementing regression model

#Previewing the head

```
head(advert)
```

```
##    Daily Time Spent on Site Age Area Income Daily Internet Usage Male
## 1:                    68.95  35    61833.90                256.09    1
## 2:                    80.23  31    68441.85                193.77    2
## 3:                    69.47  26    59785.94                236.50    1
## 4:                    74.15  29    54806.18                245.89    2
## 5:                    68.37  35    73889.99                225.58    1
## 6:                    59.99  23    59761.56                226.74    2
##    Clicked_on_Ad year month day hour click
## 1:          <NA>   54     3  27    0     0
## 2:          <NA>   40     4   4    1     0
## 3:          <NA>   36     3  13   20     0
## 4:          <NA>   32     1  10    2     0
## 5:          <NA>   37     6   3    3     0
## 6:          <NA>   31     5  19   14     0
```

#Change the label to numeric

```
advert$click<-as.numeric(advert$click)

# Applying the lm() function.
#multiple_lm <- lm(click ~ ., advert)

# Generating the anova table
#anova(multiple_lm)
```

#The table tabulates the analysis of degree of freedom,sum of squared mean, mean sq and the p value of the variables

```
# Then performing our prediction
#prediction <- predict(multiple_lm, advert)
```

#Printing out our result

```
#prediction
```

e) KNNsupervised learning with r

```
head(advert)
```

```
##    Daily Time Spent on Site Age Area Income Daily Internet Usage Male
## 1:                    68.95  35    61833.90                256.09    1
## 2:                    80.23  31    68441.85                193.77    2
## 3:                    69.47  26    59785.94                236.50    1
## 4:                    74.15  29    54806.18                245.89    2
## 5:                    68.37  35    73889.99                225.58    1
## 6:                    59.99  23    59761.56                226.74    2
##    Clicked_on_Ad year month day hour click
## 1:          <NA>   54     3  27    0     1
## 2:          <NA>   40     4   4    1     1
## 3:          <NA>   36     3  13   20     1
## 4:          <NA>   32     1  10    2     1
## 5:          <NA>   37     6   3    3     1
## 6:          <NA>   31     5  19   14     1
```

#Randomizing the rows, creates a uniform distribution of 1000

```
set.seed(1234)
random <- runif(1000)
advert_random <- advert[order(random),]
```

#Selecting the first 6 rows from advert_random

```
head(advert_random)
```

```
##    Daily Time Spent on Site Age Area Income Daily Internet Usage Male
## 1:                    80.46  29    56909.30                230.78    1
## 2:                    78.37  24    55015.08                207.27    1
## 3:                    57.99  50    62466.10                124.58    1
## 4:                    72.97  30    71384.57                208.58    2
## 5:                    77.66  29    67080.94                168.15    1
## 6:                    38.91  33    56369.74                150.80    2
##    Clicked_on_Ad year month day hour click
## 1:          <NA>   14     6   4    9     1
## 2:          <NA>   48     1  23    4     1
## 3:          <NA>   47     2  12    8     2
## 4:          <NA>   50     2  11   21     2
## 5:          <NA>    9     6  19   22     1
## 6:          <NA>   42     7  13    7     2
```

```
# Normalizing the numerical variables of the data set. Normalizing the numeri
cal values is really effective for algorithms,
# as it provides a measure from 0 to 1 which corresponds to min value to the
max value of the data column.
# We define a normal function which will normalize the set of values accordin
g to its minimum value and maximum value.
#normal <- function(x) (
#  return( ((x - min(x)) /(max(x)-min(x))) )
#)
#normal(1:9)
#advert_new <- as.data.frame(lapply(advert_random[,-9], normal))
#summary(advert_new)

# Create test and train data sets

#train <- advert_new[1:800,]
#test <- advert_new[801:1000,]
#train_label <- advert_random[1:800,9]
#test_label<- advert_random[801:1000,9]

# Now we can use the K-NN algorithm. Lets call the "class" package which cont
ains the K-NN algorithm.
# We then have to provide '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 look for
# in order to classify it accordingly.

#library(class)
#require(class)
#model <- knn(train= train,test=test,cl= train_label,k=10)
#table(factor(model))
#table(test_label,model)
```

#this function divides the correct predictions by total number of predictions that tell us how accurate the model is.

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

5.Conclusion and Recommendation

Based on the findings from our analysis we conclude the following:

Gender is not really a key factor to influence clicking on the Ad or not, we established the number of female and those of male that clicked the Ad was more or less the same. So the Ad can target both genders.

We noted that Daily time spent on the site and Daily internet Usage is higher with younger individuals. These 2 variables decrease as age increases. Therefore age is a key factor to consider for success of the Ad. Therefore since we established that those who spend less time are likely to click on the Ad, we conclude that the Ad should target older individuals.

Income is key factor to consider. From our findings, those with lower are income have higher chances of clicking on the Ad than those with higher area income levels. The Ad should therefore target those with lower are income.

Those with lower Daily internet Usage and lower Daily time spent on site have higher chances of clicking on the Ad. The probability of clicking decreases with increase in the two variables. Therefore the individuals who spend less time and use lower internet should be targeted.

Year 2002 and 2007 achieved the highest clicks. This can be investigated further establish what was unique with these 2 years that can be applied into the future.

The month of February achieved moset click followed by the month of May, This could be contributed by May being a school holiday month and February could be as a result of it is not a very busy month, so individuals can afford the time.

The 9th hour and the 1st achieved most clicks, the entrepreneur can target scheduling these hours when placing the Ad.

Zcheck republic and France are the top 2 countries that appreared more frequently these can and say 8 more can be targeted with the Ad. If the entrepreneur considers these factors, they will achieve better performance with getting more individuals clicking on the Ad.

conclusion on modelling

SVM and Decision tree algorithims performed best with accuracy score of 96.7 and 96.5% respectively.Naive bayes model more offered the probabilities if an individual is likely to click on the Ad or not. Linear regression didn't do well since this is more a classification problem than a regression problem. I encountered errors when implementing the knn model, its performance will be compared with the others when successful.

Followup questions

Did we have the right data? Yes

Did we achieve the objective of the study? Yes, we did as were able to come up with characteristics of customers who are likely to click on the ad or not and we believe that this entreprenuer is well advised now.