# Week 12 Independent Project

## Defining The Question

### a) Specifying the Question.

Identify which individuals are most likely to click on her ads in Cryptography course website.

### b) Defining the Metrics for Success.

For this study, we will perform conclusive Exploratory Data Analysis to enable us identify individuals who are most likely to click on ads.

### 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. Using the data previously collected, she is looking to do a study to identify which individuals are most likely to click on her ads.

### d) Experimental Design.

The project was undertaken using the following design Datasets(<http://bit.ly/IPAdvertisingData>)

* Load dataset
* Data Cleaning
* Performing Exploratory Analysis
* Conclusion

### e) Data Relevance

Data is provided was collected in the past but from the same blog hence it is very suitable for this study.

Definition of Variables Daily Time Spent on Site

Age

Area

Income

Daily Internet Usage

Ad Topic Line

City

Male

Country

Timestamp

Clicked on Ad

## Data Preparation

### Importing the Libraries

# load libraries  
library(tidyverse)

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

## v ggplot2 3.3.3 v purrr 0.3.4  
## v tibble 3.1.0 v dplyr 1.0.5  
## v tidyr 1.1.3 v stringr 1.4.0  
## v readr 1.4.0 v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(data.table)

##   
## Attaching package: 'data.table'

## The following objects are masked from 'package:dplyr':  
##   
## between, first, last

## The following object is masked from 'package:purrr':  
##   
## transpose

options(warn = -1)

### Loading the data

#load data  
df <- fread('http://bit.ly/IPAdvertisingData')

#priview the first 6 rows  
head(df)

## Daily Time Spent on Site Age Area Income Daily Internet Usage  
## 1: 68.95 35 61833.90 256.09  
## 2: 80.23 31 68441.85 193.77  
## 3: 69.47 26 59785.94 236.50  
## 4: 74.15 29 54806.18 245.89  
## 5: 68.37 35 73889.99 225.58  
## 6: 59.99 23 59761.56 226.74  
## Ad Topic Line City Male Country  
## 1: Cloned 5thgeneration orchestration Wrightburgh 0 Tunisia  
## 2: Monitored national standardization West Jodi 1 Nauru  
## 3: Organic bottom-line service-desk Davidton 0 San Marino  
## 4: Triple-buffered reciprocal time-frame West Terrifurt 1 Italy  
## 5: Robust logistical utilization South Manuel 0 Iceland  
## 6: Sharable client-driven software Jamieberg 1 Norway  
## Timestamp Clicked on Ad  
## 1: 2016-03-27 00:53:11 0  
## 2: 2016-04-04 01:39:02 0  
## 3: 2016-03-13 20:35:42 0  
## 4: 2016-01-10 02:31:19 0  
## 5: 2016-06-03 03:36:18 0  
## 6: 2016-05-19 14:30:17 0

# Checking the class of our dataset  
class(df)

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

# Find the number of column and rows  
class(df); ncol(df); nrow(df)

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

## [1] 10

## [1] 1000

We have 10 columns and 1000 rows

#checking the first 10 rows in our dataset  
(print(head(df, n=10)))

## 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  
## 7: 88.91 33 53852.85 208.36  
## 8: 66.00 48 24593.33 131.76  
## 9: 74.53 30 68862.00 221.51  
## 10: 69.88 20 55642.32 183.82  
## 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  
## 7: Enhanced dedicated support Brandonstad 0 Myanmar  
## 8: Reactive local challenge Port Jefferybury 1 Australia  
## 9: Configurable coherent function West Colin 1 Grenada  
## 10: Mandatory homogeneous architecture Ramirezton 1 Ghana  
## 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  
## 7: 2016-01-28 20:59:32 0  
## 8: 2016-03-07 01:40:15 1  
## 9: 2016-04-18 09:33:42 0  
## 10: 2016-07-11 01:42:51 0

## 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  
## 7: 88.91 33 53852.85 208.36  
## 8: 66.00 48 24593.33 131.76  
## 9: 74.53 30 68862.00 221.51  
## 10: 69.88 20 55642.32 183.82  
## 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  
## 7: Enhanced dedicated support Brandonstad 0 Myanmar  
## 8: Reactive local challenge Port Jefferybury 1 Australia  
## 9: Configurable coherent function West Colin 1 Grenada  
## 10: Mandatory homogeneous architecture Ramirezton 1 Ghana  
## 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  
## 7: 2016-01-28 20:59:32 0  
## 8: 2016-03-07 01:40:15 1  
## 9: 2016-04-18 09:33:42 0  
## 10: 2016-07-11 01:42:51 0

#checking the last 10 rows in our dataset  
(print(tail(df, n=10)))

## Daily Time Spent on Site Age Area Income Daily Internet Usage  
## 1: 35.79 44 33813.08 165.62  
## 2: 38.96 38 36497.22 140.67  
## 3: 69.17 40 66193.81 123.62  
## 4: 64.20 27 66200.96 227.63  
## 5: 43.70 28 63126.96 173.01  
## 6: 72.97 30 71384.57 208.58  
## 7: 51.30 45 67782.17 134.42  
## 8: 51.63 51 42415.72 120.37  
## 9: 55.55 19 41920.79 187.95  
## 10: 45.01 26 29875.80 178.35  
## Ad Topic Line City Male  
## 1: Enterprise-wide tangible model North Katie 1  
## 2: Versatile mission-critical application Mauricefurt 1  
## 3: Extended leadingedge solution New Patrick 0  
## 4: Phased zero tolerance extranet Edwardsmouth 1  
## 5: Front-line bifurcated ability Nicholasland 0  
## 6: Fundamental modular algorithm Duffystad 1  
## 7: Grass-roots cohesive monitoring New Darlene 1  
## 8: Expanded intangible solution South Jessica 1  
## 9: Proactive bandwidth-monitored policy West Steven 0  
## 10: Virtual 5thgeneration emulation Ronniemouth 0  
## Country Timestamp Clicked on Ad  
## 1: Tonga 2016-04-20 13:36:42 1  
## 2: Comoros 2016-07-21 16:02:40 1  
## 3: Montenegro 2016-03-06 11:36:06 1  
## 4: Isle of Man 2016-02-11 23:45:01 0  
## 5: Mayotte 2016-04-04 03:57:48 1  
## 6: Lebanon 2016-02-11 21:49:00 1  
## 7: Bosnia and Herzegovina 2016-04-22 02:07:01 1  
## 8: Mongolia 2016-02-01 17:24:57 1  
## 9: Guatemala 2016-03-24 02:35:54 0  
## 10: Brazil 2016-06-03 21:43:21 1

## Daily Time Spent on Site Age Area Income Daily Internet Usage  
## 1: 35.79 44 33813.08 165.62  
## 2: 38.96 38 36497.22 140.67  
## 3: 69.17 40 66193.81 123.62  
## 4: 64.20 27 66200.96 227.63  
## 5: 43.70 28 63126.96 173.01  
## 6: 72.97 30 71384.57 208.58  
## 7: 51.30 45 67782.17 134.42  
## 8: 51.63 51 42415.72 120.37  
## 9: 55.55 19 41920.79 187.95  
## 10: 45.01 26 29875.80 178.35  
## Ad Topic Line City Male  
## 1: Enterprise-wide tangible model North Katie 1  
## 2: Versatile mission-critical application Mauricefurt 1  
## 3: Extended leadingedge solution New Patrick 0  
## 4: Phased zero tolerance extranet Edwardsmouth 1  
## 5: Front-line bifurcated ability Nicholasland 0  
## 6: Fundamental modular algorithm Duffystad 1  
## 7: Grass-roots cohesive monitoring New Darlene 1  
## 8: Expanded intangible solution South Jessica 1  
## 9: Proactive bandwidth-monitored policy West Steven 0  
## 10: Virtual 5thgeneration emulation Ronniemouth 0  
## Country Timestamp Clicked on Ad  
## 1: Tonga 2016-04-20 13:36:42 1  
## 2: Comoros 2016-07-21 16:02:40 1  
## 3: Montenegro 2016-03-06 11:36:06 1  
## 4: Isle of Man 2016-02-11 23:45:01 0  
## 5: Mayotte 2016-04-04 03:57:48 1  
## 6: Lebanon 2016-02-11 21:49:00 1  
## 7: Bosnia and Herzegovina 2016-04-22 02:07:01 1  
## 8: Mongolia 2016-02-01 17:24:57 1  
## 9: Guatemala 2016-03-24 02:35:54 0  
## 10: Brazil 2016-06-03 21:43:21 1

# Check the structure of the data   
print(str(df))

## 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" "Monitored national standardization" "Organic bottom-line service-desk" "Triple-buffered reciprocal time-frame" ...  
## $ 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 : POSIXct, format: "2016-03-27 00:53:11" "2016-04-04 01:39:02" ...  
## $ Clicked on Ad : int 0 0 0 0 0 0 0 1 0 0 ...  
## - attr(\*, ".internal.selfref")=<externalptr>   
## NULL

The advertising data is a dataframe/datatable containing 1000 observations and 10 variables. The variables has discrete and countinous values and also it has character values.

#distribution of dataset on gender  
table(df$gender)

## < table of extent 0 >

There are more female that male who use the site

#distribution of dataset on click on ad  
table(df$clicked\_on\_ad)

## < table of extent 0 >

The dataset is well balanced with a ratio of 1:1 of those who click ad and those who did not

## Data Cleaning

### Renaming of columns

# Changing column names to lower and replacing spaces with an underscore for readability  
colnames(df) = tolower(str\_replace\_all(colnames(df), c(' ' = '\_')))  
  
# Checking whether the column names have been renames appriopriately  
print(colnames(df))

## [1] "daily\_time\_spent\_on\_site" "age"   
## [3] "area\_income" "daily\_internet\_usage"   
## [5] "ad\_topic\_line" "city"   
## [7] "male" "country"   
## [9] "timestamp" "clicked\_on\_ad"

# The male column should be renamed to gender  
colnames(df)[colnames(df) == 'male'] = 'gender'  
head(df)

## daily\_time\_spent\_on\_site age area\_income daily\_internet\_usage  
## 1: 68.95 35 61833.90 256.09  
## 2: 80.23 31 68441.85 193.77  
## 3: 69.47 26 59785.94 236.50  
## 4: 74.15 29 54806.18 245.89  
## 5: 68.37 35 73889.99 225.58  
## 6: 59.99 23 59761.56 226.74  
## ad\_topic\_line city gender 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

### Finding missing values

#   
#check missing values in each column  
colSums(is.na(df))

## daily\_time\_spent\_on\_site age area\_income   
## 0 0 0   
## daily\_internet\_usage ad\_topic\_line city   
## 0 0 0   
## gender country timestamp   
## 0 0 0   
## clicked\_on\_ad   
## 0

### Finding Duplicates values

# find the duplicated rows   
#   
anyDuplicated(df)

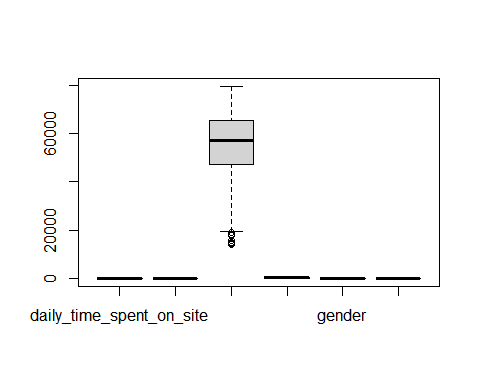
## [1] 0

### Finding Outliers

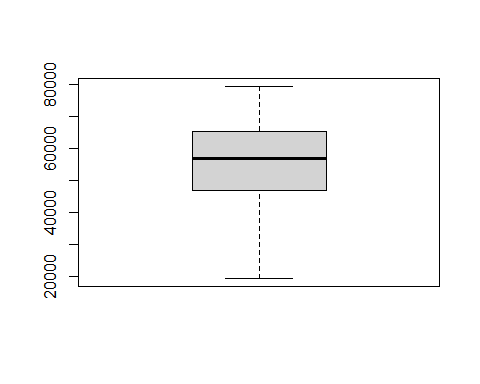
# get numerical columnns  
num <- df %>% select\_if(is.numeric)  
num

## daily\_time\_spent\_on\_site age area\_income daily\_internet\_usage gender  
## 1: 68.95 35 61833.90 256.09 0  
## 2: 80.23 31 68441.85 193.77 1  
## 3: 69.47 26 59785.94 236.50 0  
## 4: 74.15 29 54806.18 245.89 1  
## 5: 68.37 35 73889.99 225.58 0  
## ---   
## 996: 72.97 30 71384.57 208.58 1  
## 997: 51.30 45 67782.17 134.42 1  
## 998: 51.63 51 42415.72 120.37 1  
## 999: 55.55 19 41920.79 187.95 0  
## 1000: 45.01 26 29875.80 178.35 0  
## clicked\_on\_ad  
## 1: 0  
## 2: 0  
## 3: 0  
## 4: 0  
## 5: 0  
## ---   
## 996: 1  
## 997: 1  
## 998: 1  
## 999: 0  
## 1000: 1

#check for outliers in the numerical columns  
boxplot(num)

 From the results we can conclude that there are outliers in Area Income columns we will not delete them because we will get insights from them

non\_outliers <- 47032 - 1.5 \* IQR(df$area\_income)   
df$area\_income[df$area\_income < non\_outliers]<- non\_outliers  
  
boxplot(df$area\_income)



# Exploratory Data Analysis

## Univariate Analysis

# checking for the dataset statistical summary  
(summary(num))

## 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   
## gender clicked\_on\_ad  
## Min. :0.000 Min. :0.0   
## 1st Qu.:0.000 1st Qu.:0.0   
## Median :0.000 Median :0.5   
## Mean :0.481 Mean :0.5   
## 3rd Qu.:1.000 3rd Qu.:1.0   
## Max. :1.000 Max. :1.0

### Mean

print("Age")

## [1] "Age"

mean(replace(df$age, df$clicked\_on\_ad==0, NA), na.rm = TRUE)

## [1] 40.334

print("Area Income")

## [1] "Area Income"

mean(replace(df$area\_income, df$clicked\_on\_ad==0, NA), na.rm = TRUE)

## [1] 48665.06

print("Daily internet usage")

## [1] "Daily internet usage"

mean(replace(df$daily\_internet\_usage, df$clicked\_on\_ad==0, NA), na.rm = TRUE)

## [1] 145.4865

print("Daily time spent on site")

## [1] "Daily time spent on site"

mean(replace(df$daily\_time\_spent\_on\_site, df$clicked\_on\_ad==0, NA), na.rm = TRUE)

## [1] 53.14578

### range

print("Age")

## [1] "Age"

range(replace(df$age, df$clicked\_on\_ad==0, NA), na.rm = TRUE)

## [1] 19 61

print("Area Income")

## [1] "Area Income"

range(replace(df$area\_income, df$clicked\_on\_ad==0, NA), na.rm = TRUE)

## [1] 19373.75 78520.99

print("Daily internet usage")

## [1] "Daily internet usage"

range(replace(df$daily\_internet\_usage, df$clicked\_on\_ad==0, NA), na.rm = TRUE)

## [1] 104.78 269.96

print("Daily time spent on site")

## [1] "Daily time spent on site"

range(replace(df$daily\_time\_spent\_on\_site, df$clicked\_on\_ad==0, NA), na.rm = TRUE)

## [1] 32.60 91.37

### Interquartile range

# Finding the Quantiles for numerical columns  
# Quantiles of daily time on site  
print("Quantiles of daily time on site")

## [1] "Quantiles of daily time on site"

daily\_time\_quantiles <-quantile(df$daily\_time\_on\_site)  
daily\_time\_quantiles

## 0% 25% 50% 75% 100%   
## NA NA NA NA NA

# Quantiles of age  
print("Quantiles of age")

## [1] "Quantiles of age"

age\_quantiles <-quantile(df$age)  
age\_quantiles

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

# Quantiles of area income  
print("Quantiles of area income")

## [1] "Quantiles of area income"

area\_income\_quantiles <-quantile(df$area\_income)  
area\_income\_quantiles

## 0% 25% 50% 75% 100%   
## 19373.75 47031.80 57012.30 65470.64 79484.80

# Quantiles of daily internet usage  
print("Quantiles of daily internet usage")

## [1] "Quantiles of daily internet usage"

daily\_internet\_usage\_quantiles <-quantile(df$daily\_internet\_usage)  
daily\_internet\_usage\_quantiles

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

### Standard deviation

print("Age")

## [1] "Age"

sd(replace(df$age, df$clicked\_on\_ad==0, NA), na.rm = TRUE)

## [1] 8.856598

print("Area Income")

## [1] "Area Income"

sd(replace(df$area\_income, df$clicked\_on\_ad==0, NA), na.rm = TRUE)

## [1] 14003.2

print("Daily internet usage")

## [1] "Daily internet usage"

sd(replace(df$daily\_internet\_usage, df$clicked\_on\_ad==0, NA), na.rm = TRUE)

## [1] 30.02583

print("Daily time spent on site")

## [1] "Daily time spent on site"

sd(replace(df$daily\_time\_spent\_on\_site, df$clicked\_on\_ad==0, NA), na.rm = TRUE)

## [1] 12.82209

### Variance

print("Age")

## [1] "Age"

var(replace(df$age, df$clicked\_on\_ad==0, NA), na.rm = TRUE)

## [1] 78.43932

print("Area Income")

## [1] "Area Income"

var(replace(df$area\_income, df$clicked\_on\_ad==0, NA), na.rm = TRUE)

## [1] 196089721

print("Daily internet usage")

## [1] "Daily internet usage"

var(replace(df$daily\_internet\_usage, df$clicked\_on\_ad==0, NA), na.rm = TRUE)

## [1] 901.5502

print("Daily time spent on site")

## [1] "Daily time spent on site"

var(replace(df$daily\_time\_spent\_on\_site, df$clicked\_on\_ad==0, NA), na.rm = TRUE)

## [1] 164.406

### Skewness

#skewness of Area Income for those who click ad  
library(moments)  
print("Age")

## [1] "Age"

skewness(replace(df$age, df$clicked\_on\_ad==0, NA), na.rm = TRUE)

## [1] 0.02515906

print("Area Income")

## [1] "Area Income"

skewness(replace(df$area\_income, df$clicked\_on\_ad==0, NA), na.rm = TRUE)

## [1] -0.1307072

print("Daily internet usage")

## [1] "Daily internet usage"

skewness(replace(df$daily\_internet\_usage, df$clicked\_on\_ad==0, NA), na.rm = TRUE)

## [1] 1.236424

print("Daily time spent on site")

## [1] "Daily time spent on site"

skewness(replace(df$daily\_time\_spent\_on\_site, df$clicked\_on\_ad==0, NA), na.rm = TRUE)

## [1] 0.5337215

All the variables has positive value this implies that the distribution of the data is slightly skewed to the right or positive skewed. accept for area income which negative which means it is skewed to left

### Kurtosis

print("Age")

## [1] "Age"

kurtosis(replace(df$age, df$clicked\_on\_ad==0, NA), na.rm = TRUE)

## [1] 2.303378

print("Area Income")

## [1] "Area Income"

kurtosis(replace(df$area\_income, df$clicked\_on\_ad==0, NA), na.rm = TRUE)

## [1] 2.286888

print("Daily internet usage")

## [1] "Daily internet usage"

kurtosis(replace(df$daily\_internet\_usage, df$clicked\_on\_ad==0, NA), na.rm = TRUE)

## [1] 4.816968

print("Daily time spent on site")

## [1] "Daily time spent on site"

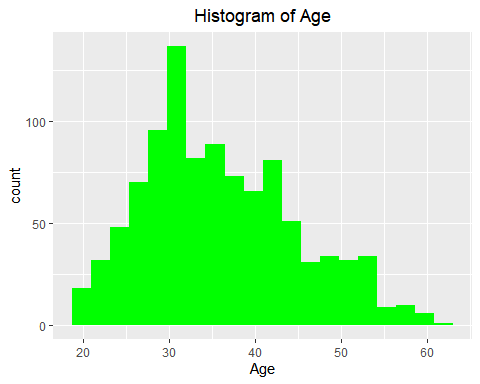
kurtosis(replace(df$daily\_time\_spent\_on\_site, df$clicked\_on\_ad==0, NA), na.rm = TRUE)

## [1] 2.561506

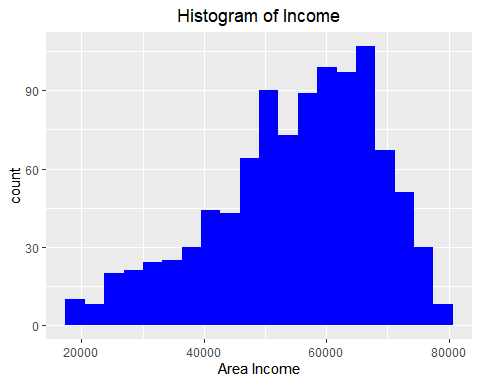
The distribution of the data is platykurtic, since the computed values for all the varibles are less than 3 accept variable daily internet usage.

### Histogram

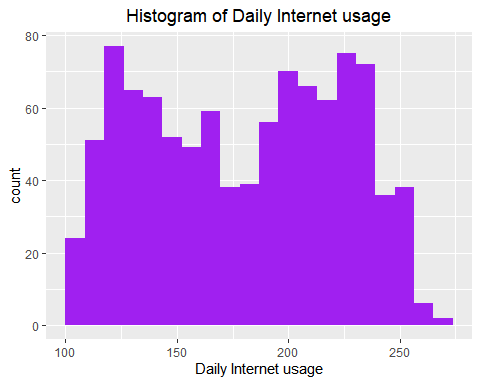
# Histogram of Age  
ggplot(data = df, mapping = aes(x = age)) +  
 geom\_histogram(bins = 20, fill = "green") +  
 labs(x = "Age") +  
 ggtitle("Histogram of Age") +  
 theme(plot.title = element\_text(hjust = 0.5))

 It is skewed to left. Age between 25 and 45 visit the site mostly

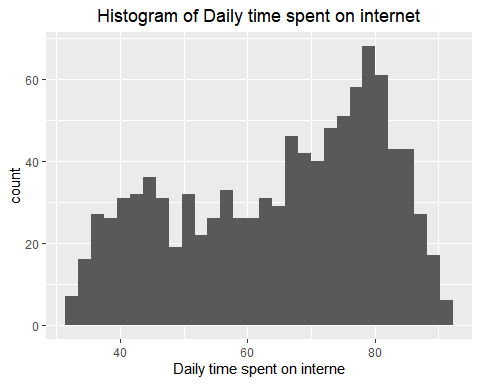
# Histogram of Income  
ggplot(data = df, mapping = aes(x = area\_income)) +  
 geom\_histogram(bins = 20, fill = "blue") +  
 labs(x = "Area Income") +  
 ggtitle("Histogram of Income") +  
 theme(plot.title = element\_text(hjust = 0.5))

 It is skewed to right, people whose income ranges between 45000 and 75000 mostly visit the site.

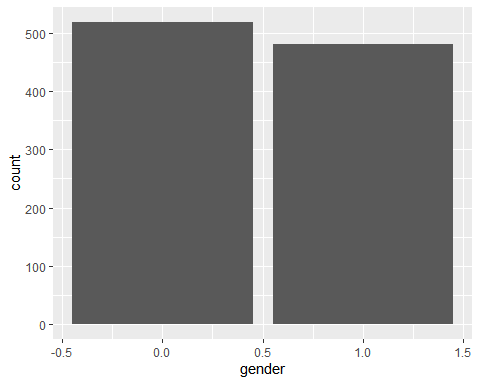
# Histogram of Daily Internet usage  
ggplot(data = df, mapping = aes(x = daily\_internet\_usage)) +  
 geom\_histogram(bins = 20,fill="purple") +  
 labs(x = "Daily Internet usage") +  
 ggtitle("Histogram of Daily Internet usage") +  
 theme(plot.title = element\_text(hjust = 0.5))



# Histogram of Daily time spent on internet  
ggplot(data = df, mapping = aes(x = daily\_time\_spent\_on\_site)) +  
 geom\_histogram(bins = 30) +  
 labs(x = "Daily time spent on interne") +  
 ggtitle("Histogram of Daily time spent on internet") +  
 theme(plot.title = element\_text(hjust = 0.5))

 Most of the people who are above 60 visits the site mostly

#Which gender is mainly active on the blog?  
ggplot(data = df) +  
 geom\_bar(mapping = aes(x = gender))



#Assuming that if male = 1 then we can conclude that more females  
# frequennt the blog more as compared to males

Female visit the site mostly than male

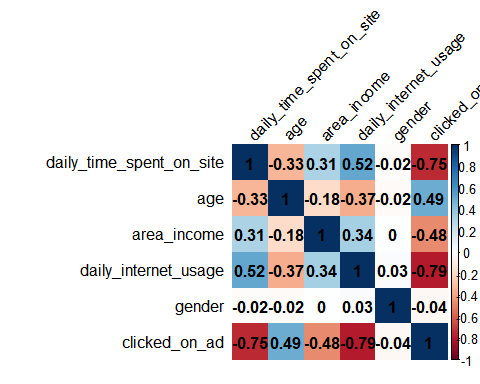
## Bivariate and Multivariate Analysis.

### Correlation

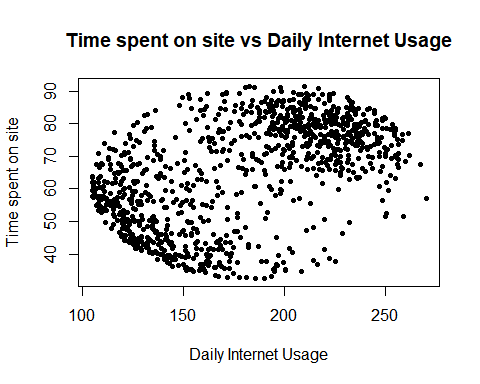
library(corrplot)

## corrplot 0.84 loaded

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

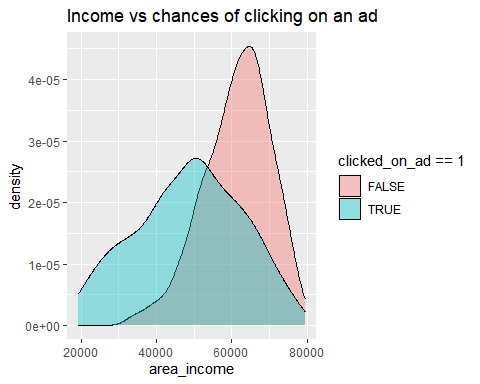
 There are a bit of correlation between Daily time spent on site and Daily internet usage

x <- df$daily\_internet\_usage  
y <- df$daily\_time\_spent\_on\_site  
# Plot with main and axis titles  
# Change point shape (pch = 19) and remove frame.  
plot(x, y, main = "Time spent on site vs Daily Internet Usage",  
 xlab = "Daily Internet Usage", ylab = "Time spent on site",  
 pch = 20)

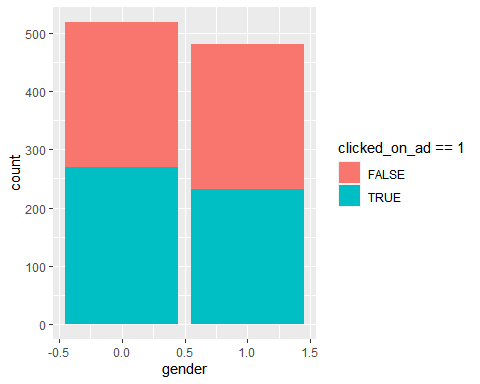


The plot shows that there is no correlation between the two columns. But we can see that people who spend less time on site use less internet. Also, most of the people who use alot of internet per day seem to spend most of there time on the site.

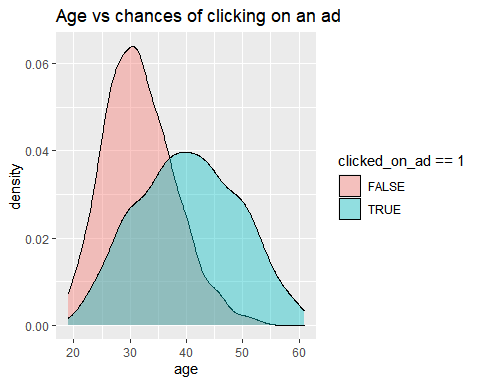
#Income class and it's relationship to clicking an ad  
ggplot(df,   
 aes(x = area\_income,   
 fill = clicked\_on\_ad==1)) +  
 geom\_density(alpha = 0.4) +  
 labs(title = "Income vs chances of clicking on an ad")

 People from all range of income click on ads as compared to those who do not click ad. should focus on people whose income range on 20000 to 50000 are the ones are likely click ad

#Who is likely to click on an ad, female or male?  
# stacked bar chart  
library(ggplot2)  
ggplot(df,   
 aes(x = gender,   
 fill = clicked\_on\_ad==1)) +   
 geom\_bar(position = "stack")

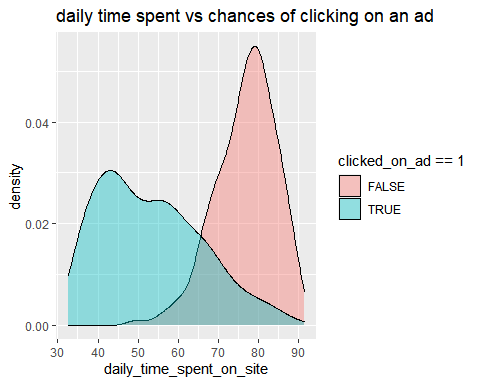
 There are more female who click on ad than male

#Age and it's relationship to clicking an ad  
ggplot(df,   
 aes(x = age,   
 fill = clicked\_on\_ad==1)) +  
 geom\_density(alpha = 0.4) +  
 labs(title = "Age vs chances of clicking on an ad")



People who click on ad are disriputed across the age. Those who are in the bracket of 30 to 50 are the ones who click on add mostly

#Time on and it's relationship to clicking an ad  
ggplot(df,   
 aes(x = daily\_time\_spent\_on\_site,   
 fill = clicked\_on\_ad==1)) +  
 geom\_density(alpha = 0.4) +  
 labs(title = "daily time spent vs chances of clicking on an ad")



People who spend less time on site click on ad unlike those who spend longer time on site

# EDA Conclusion

1. Most of the people who use alot of internet per day seem to spend most of there time on the site.She should target those who spend alot of internet.
2. Those who are in the age bracket of 30 to 50 are the ones who click on ad mostly so she should focus on people who are in age bracket named.
3. Famele who click on ad are more than male so she should forcus on them, without neglecting male because no big difference between them interms of clicking on ad
4. Those who click on ad is distributed across income but those who are in the range of 20000 to 50000 click on ad mostly, so she should consider them.
5. Those who spend less than 70mins on site mostly click on ad

# Implement the Solution

## Modelling

` ### KNN

set.seed(100)  
  
# Randomizing the rows, creates a uniform distribution of 150  
random <- runif(150)  
df\_random <- num[order(random),]  
  
# Selecting the first 6 rows from iris\_random  
head(df\_random)

## daily\_time\_spent\_on\_site age area\_income daily\_internet\_usage gender  
## 1: 47.53 30 33258.09 135.18 0  
## 2: 49.89 39 17709.98 160.03 1  
## 3: 49.78 46 71718.51 152.24 0  
## 4: 79.57 31 61227.59 230.93 0  
## 5: 35.61 46 51868.85 158.22 0  
## 6: 76.99 31 56729.78 244.34 1  
## clicked\_on\_ad  
## 1: 1  
## 2: 1  
## 3: 1  
## 4: 0  
## 5: 1  
## 6: 0

dfnum <- as.data.frame(scale(num))  
head(dfnum)

## daily\_time\_spent\_on\_site age area\_income daily\_internet\_usage  
## 1 0.2491419 -0.1148475 0.50943618 1.7331628  
## 2 0.9606516 -0.5701399 1.00202882 0.3136484  
## 3 0.2819420 -1.1392555 0.35677007 1.2869451  
## 4 0.5771428 -0.7977862 -0.01444841 1.5008289  
## 5 0.2125572 -0.1148475 1.40816290 1.0382112  
## 6 -0.3160289 -1.4807248 0.35495265 1.0646335  
## gender clicked\_on\_ad  
## 1 -0.9622138 -0.9994999  
## 2 1.0382307 -0.9994999  
## 3 -0.9622138 -0.9994999  
## 4 1.0382307 -0.9994999  
## 5 -0.9622138 -0.9994999  
## 6 1.0382307 -0.9994999

normalize <- function(x) {  
return ((x - min(x)) / (max(x) - min(x)))  
}  
df\_n <- as.data.frame(lapply(dfnum, normalize))

Train <- df\_n[1:200, ]  
Test <- df\_n[201:393, ]  
Train\_labels <- Train[1:200, 6]  
Test\_labels <- Test[201:393, 5]

Test

## daily\_time\_spent\_on\_site age area\_income daily\_internet\_usage gender  
## 201 0.57453680 0.50000000 0.97072836 0.66376075 0  
## 202 0.62468129 0.23809524 0.60216634 0.89593171 1  
## 203 0.05184430 0.50000000 0.26369978 0.41046131 1  
## 204 0.63640999 0.28571429 0.92816946 0.47868991 1  
## 205 0.63199048 0.23809524 0.86143036 0.69021673 0  
## 206 0.43549210 0.23809524 0.35653300 0.09759051 0  
## 207 0.75301717 0.21428571 0.79573512 0.65316624 0  
## 208 0.87506374 0.26190476 0.73814544 0.49993946 1  
## 209 0.45741968 0.92857143 0.38382642 0.21824676 0  
## 210 0.12833588 0.45238095 0.37791163 0.18222545 1  
## 211 0.75148734 0.21428571 0.78144829 0.68452597 1  
## 212 0.15791263 0.45238095 0.83256795 0.35476450 0  
## 213 0.75250722 0.19047619 0.62268023 0.79047100 1  
## 214 0.58949516 0.57142857 0.95055850 0.30887517 1  
## 215 0.83936767 0.50000000 0.78587198 0.75335997 0  
## 216 0.56790753 0.09523810 0.30739399 0.28556726 0  
## 217 0.49243583 0.80952381 0.32459966 0.12810268 1  
## 218 0.35186130 0.35714286 0.47179741 0.04334665 0  
## 219 0.03858576 0.50000000 0.69668200 0.57785446 0  
## 220 0.18697943 0.45238095 0.10475215 0.39780845 0  
## 221 0.76967534 0.42857143 0.63151601 0.90682891 0  
## 222 0.73482917 0.19047619 0.81723865 0.58003390 0  
## 223 0.29491756 0.47619048 0.83589359 0.19257779 0  
## 224 0.48172701 0.52380952 0.96043629 0.30385035 0  
## 225 0.96124426 0.54761905 0.70073708 0.40131977 0  
## 226 0.78361380 0.26190476 0.60232423 0.17683739 0  
## 227 0.42070372 0.23809524 0.41778165 0.09117327 0  
## 228 0.03841577 0.45238095 0.54889438 0.30251846 0  
## 229 0.64728880 0.28571429 0.92281492 0.57089236 0  
## 230 0.73873874 0.09523810 0.68160465 0.58275820 0  
## 231 0.57912630 0.33333333 0.89651694 0.74615571 1  
## 232 0.24086351 0.30952381 0.26106083 0.19142753 1  
## 233 0.50688424 0.45238095 0.92304549 0.23707471 0  
## 234 0.77715451 0.21428571 0.53922059 0.81571619 0  
## 235 0.08023117 0.73809524 0.65261489 0.57192154 1  
## 236 0.13292538 0.61904762 0.39987677 0.17629253 1  
## 237 0.75080741 0.40476190 0.67724937 0.11339145 0  
## 238 0.56178820 0.26190476 0.88891436 0.32249667 0  
## 239 0.70916199 0.33333333 0.73392667 0.14160310 0  
## 240 0.69131396 0.30952381 0.81109084 0.78683860 1  
## 241 0.80622132 0.59523810 0.15321149 0.27884732 0  
## 242 0.35832058 0.66666667 0.64415216 0.06344594 1  
## 243 0.90498045 0.30952381 0.73613806 0.53317593 1  
## 244 0.89121197 0.26190476 0.70815917 0.60382613 0  
## 245 0.64320925 0.11904762 0.78309652 0.44781451 1  
## 246 0.82644909 0.80952381 0.31017357 0.71624894 1  
## 247 0.12510624 0.61904762 0.69652610 0.25033297 1  
## 248 0.41560428 0.52380952 0.55402858 1.00000000 1  
## 249 0.16726160 0.88095238 0.44975377 0.38436857 0  
## 250 0.50314465 0.14285714 0.17425571 0.34132462 0  
## 251 0.74961754 0.40476190 0.87252211 0.71116358 0  
## 252 0.48648649 0.61904762 0.76177439 0.08626952 1  
## 253 0.88288288 0.33333333 0.77732923 0.60079913 1  
## 254 0.22913480 0.26190476 0.80607238 0.36233200 1  
## 255 0.40965494 0.69047619 0.74499338 0.11109093 0  
## 256 0.82321945 0.21428571 0.75938755 0.58342414 0  
## 257 0.82117967 0.30952381 0.72702651 0.76667877 0  
## 258 0.12680605 0.45238095 0.65333319 0.20523066 1  
## 259 0.86469488 0.47619048 0.77038601 0.73453203 0  
## 260 0.70100289 0.28571429 0.42826658 0.09849861 0  
## 261 0.71477138 0.21428571 0.81957479 0.65250030 0  
## 262 0.46999830 0.38095238 0.62422524 0.03020947 0  
## 263 0.45232024 0.38095238 0.90628662 0.24119143 1  
## 264 0.17712052 0.59523810 0.55282043 0.12374379 0  
## 265 0.87438382 0.45238095 0.56082033 0.84616782 0  
## 266 0.64694884 0.57142857 0.75516482 0.09789321 1  
## 267 0.64558899 0.16666667 0.65198852 0.19469669 0  
## 268 0.67720551 0.35714286 0.86352310 0.75892965 0  
## 269 0.12867585 0.16666667 0.50994529 0.40277273 1  
## 270 0.79126296 0.16666667 0.73777652 0.59601647 0  
## 271 0.20210777 0.80952381 0.74980768 0.38273399 1  
## 272 0.68740439 0.42857143 0.91147579 0.70837874 1  
## 273 0.74247833 0.33333333 0.59079210 0.90543649 0  
## 274 0.61669216 0.42857143 0.97915200 0.45283933 0  
## 275 0.68842427 0.21428571 0.65688589 0.83297009 0  
## 276 0.25599184 0.23809524 0.19988151 0.31335513 0  
## 277 0.92979772 0.38095238 0.67707789 0.67859305 0  
## 278 0.96447391 0.30952381 0.55306841 0.44194212 1  
## 279 0.82899881 0.16666667 0.59633461 0.31299189 1  
## 280 0.83409825 0.21428571 0.74717866 0.55684708 1  
## 281 0.23406425 0.78571429 0.28785340 0.23907253 0  
## 282 0.37871834 0.11904762 0.27501584 0.26534689 0  
## 283 0.13717491 0.38095238 0.53317264 0.17193365 0  
## 284 0.66564678 0.38095238 0.84426058 0.80282116 0  
## 285 0.25344212 0.76190476 0.60627730 0.15516406 1  
## 286 0.72327044 0.07142857 0.58047773 0.65437704 1  
## 287 0.39792623 0.16666667 0.49387066 0.13609396 0  
## 288 0.85449601 0.42857143 0.82369324 0.65734350 0  
## 289 0.21162672 0.54761905 0.79941684 0.22145538 0  
## 290 0.47475778 0.11904762 0.80743278 0.37801187 0  
## 291 0.30460649 0.28571429 0.88984414 0.40464947 0  
## 292 0.88577257 0.30952381 0.72705430 0.63839448 0  
## 293 0.38415774 0.47619048 0.95534760 0.33103281 1  
## 294 0.83307836 0.33333333 0.74320161 0.75057513 0  
## 295 0.66207717 0.40476190 0.94201499 0.35845744 1  
## 296 0.84650688 0.40476190 0.80545914 0.69130645 1  
## 297 0.70287268 0.38095238 0.82027355 0.81002543 0  
## 298 0.67091620 0.28571429 0.85074754 0.73659039 0  
## 299 0.81234064 0.28571429 0.79820350 0.66569803 0  
## 300 0.56433792 0.14285714 0.71534641 0.76710255 1  
## 301 0.63522013 0.21428571 0.62675470 0.87916213 0  
## 302 0.34030257 0.73809524 0.91415948 0.43431408 0  
## 303 0.11303757 0.47619048 0.73878158 0.28804940 0  
## 304 0.76423593 0.45238095 0.75342237 0.15770674 1  
## 305 0.01563828 0.57142857 0.43053660 0.36796222 0  
## 306 0.80248173 0.11904762 0.64436029 0.44841991 1  
## 307 0.88713242 0.33333333 0.72952573 0.66442669 0  
## 308 0.85160632 0.38095238 0.72359933 0.76455987 0  
## 309 0.88866225 0.30952381 0.76240061 0.49461194 0  
## 310 0.37939827 0.83333333 0.15237607 0.34132462 0  
## 311 0.74723780 0.35714286 0.85790301 0.70680470 1  
## 312 0.63131056 0.71428571 0.80210205 0.83902409 0  
## 313 0.73006969 0.07142857 0.42534132 0.39120959 0  
## 314 0.67295597 0.33333333 0.72194087 0.88128103 1  
## 315 0.87863335 0.52380952 0.71221699 0.77345926 0  
## 316 0.70185280 0.47619048 0.50641153 0.03571861 0  
## 317 0.73499915 0.04761905 0.52743284 0.49763894 0  
## 318 0.69318375 0.14285714 0.59647922 0.79555636 1  
## 319 0.81795003 0.28571429 0.83401890 0.49394600 0  
## 320 0.50076492 0.59523810 0.46081132 0.00133188 0  
## 321 0.32126466 0.35714286 0.81012425 0.18482867 0  
## 322 0.99218086 0.42857143 0.61879206 0.46004359 0  
## 323 0.92095869 0.26190476 0.62539293 0.39562901 1  
## 324 0.57079721 0.38095238 0.84717301 0.84047706 0  
## 325 0.87931328 0.52380952 0.62596571 0.82437341 0  
## 326 0.07258202 0.40476190 0.23518873 0.55170117 0  
## 327 0.03705592 0.69047619 0.44076453 0.62616540 1  
## 328 0.75199728 0.30952381 0.79230779 0.76770795 0  
## 329 0.58558559 0.14285714 0.78525309 0.70311176 0  
## 330 0.15077341 0.28571429 0.71703541 0.69627073 0  
## 331 0.81778004 0.16666667 0.67920636 0.57997336 0  
## 332 0.80724120 0.28571429 0.80381289 0.66170239 0  
## 333 0.40285569 0.71428571 0.81593903 0.18440489 1  
## 334 0.79483257 0.35714286 0.66063572 0.85361424 1  
## 335 0.91415944 0.50000000 0.79957519 0.50544860 1  
## 336 0.10776815 0.52380952 0.66562271 0.22938612 1  
## 337 0.92911780 0.38095238 0.74920329 0.47977963 0  
## 338 0.72616012 0.21428571 0.70244609 0.77987650 1  
## 339 0.71018188 0.50000000 0.91662343 0.69784478 1  
## 340 0.56603774 0.07142857 0.71087752 0.64541712 0  
## 341 0.06306306 0.66666667 0.67167418 0.38830367 0  
## 342 0.67363590 0.69047619 0.59155345 0.06399080 0  
## 343 0.94373619 0.45238095 0.49867305 0.76359123 0  
## 344 0.87319395 0.21428571 0.65645665 0.60975905 1  
## 345 0.48427673 0.16666667 0.63125291 0.16285265 1  
## 346 0.56382798 0.04761905 0.43009820 0.68912701 1  
## 347 0.83256842 0.14285714 0.39247667 0.57276910 0  
## 348 0.08958015 0.78571429 0.64741824 0.50720426 1  
## 349 0.74111848 0.42857143 0.93219476 0.44636154 0  
## 350 0.48121707 0.00000000 0.60030738 0.48528878 0  
## 351 0.71205167 0.21428571 0.68713984 0.80251846 0  
## 352 0.69879313 0.09523810 0.65062950 0.64535658 1  
## 353 0.77494476 0.26190476 0.74027712 0.75087783 0  
## 354 0.79789223 0.59523810 0.86268998 0.68349679 1  
## 355 0.71851096 0.78571429 0.45354712 0.13433830 0  
## 356 0.92622811 0.40476190 0.66792099 0.70952900 1  
## 357 0.08244093 0.66666667 0.26393279 0.38188643 0  
## 358 0.29304776 0.47619048 0.48564370 0.04122775 0  
## 359 0.31922489 0.95238095 0.43314592 0.32558421 0  
## 360 0.86350501 0.35714286 0.80464342 0.62410703 0  
## 361 0.10725820 0.33333333 0.64703527 0.27860516 1  
## 362 0.50212477 0.52380952 0.69327483 0.03723211 1  
## 363 0.80095190 0.21428571 0.79447184 0.53892723 1  
## 364 0.69182390 0.40476190 0.82785551 0.18731081 1  
## 365 0.62060173 0.54761905 0.91027695 0.76704201 1  
## 366 0.66802652 0.83333333 0.72123509 0.21413004 1  
## 367 0.67737549 0.23809524 0.89068444 0.54837147 1  
## 368 0.75590685 0.50000000 0.46669298 0.94587722 0  
## 369 0.71426143 0.40476190 0.90414089 0.68416273 0  
## 370 0.84089750 0.14285714 0.49971629 0.60903257 1  
## 371 0.44195139 0.73809524 0.47953604 0.05400170 0  
## 372 0.05915349 0.61904762 0.41871250 0.28266134 1  
## 373 0.79619242 0.16666667 0.71538046 0.61756871 0  
## 374 0.15519293 0.66666667 0.71487991 0.24173629 0  
## 375 0.68995411 0.14285714 0.76763911 0.59910401 1  
## 376 0.76491586 0.11904762 0.67424822 0.56029786 1  
## 377 0.95869454 0.42857143 0.58151884 0.71122412 1  
## 378 0.62213157 0.54761905 0.18365861 0.11514711 0  
## 379 0.59425463 0.28571429 0.73787043 0.12513622 0  
## 380 0.82457930 0.47619048 0.64469638 0.86820438 1  
## 381 0.80945096 0.26190476 0.73005117 0.72526940 0  
## 382 0.18748938 0.52380952 0.57515220 0.11181741 1  
## 383 0.76593575 0.23809524 0.81059426 0.38364209 0  
## 384 0.71443141 0.16666667 0.57993428 0.79434556 1  
## 385 0.29015808 0.19047619 0.21412359 0.29610122 0  
## 386 0.81574027 0.42857143 0.82072767 0.72315050 0  
## 387 0.86503485 0.33333333 0.80285333 0.52046253 1  
## 388 0.20159782 0.54761905 0.25181567 0.16878557 1  
## 389 0.60343362 0.50000000 0.93004262 0.74433951 1  
## 390 0.53170151 0.45238095 0.09154643 0.19415183 0  
## 391 0.78582356 0.40476190 0.79486152 0.78617266 1  
## 392 0.80520143 0.59523810 0.86202482 0.67332607 0  
## 393 0.81438042 0.21428571 0.76367916 0.57815716 1  
## clicked\_on\_ad  
## 201 0  
## 202 0  
## 203 1  
## 204 0  
## 205 0  
## 206 1  
## 207 0  
## 208 0  
## 209 1  
## 210 1  
## 211 0  
## 212 1  
## 213 0  
## 214 1  
## 215 0  
## 216 1  
## 217 1  
## 218 1  
## 219 1  
## 220 1  
## 221 0  
## 222 0  
## 223 1  
## 224 1  
## 225 0  
## 226 1  
## 227 1  
## 228 1  
## 229 0  
## 230 0  
## 231 0  
## 232 1  
## 233 1  
## 234 1  
## 235 1  
## 236 1  
## 237 1  
## 238 0  
## 239 1  
## 240 0  
## 241 1  
## 242 1  
## 243 0  
## 244 0  
## 245 0  
## 246 0  
## 247 1  
## 248 1  
## 249 1  
## 250 1  
## 251 0  
## 252 1  
## 253 0  
## 254 1  
## 255 1  
## 256 0  
## 257 0  
## 258 1  
## 259 0  
## 260 1  
## 261 0  
## 262 1  
## 263 1  
## 264 1  
## 265 0  
## 266 1  
## 267 1  
## 268 0  
## 269 1  
## 270 0  
## 271 1  
## 272 0  
## 273 0  
## 274 0  
## 275 0  
## 276 1  
## 277 0  
## 278 0  
## 279 0  
## 280 0  
## 281 1  
## 282 1  
## 283 1  
## 284 0  
## 285 1  
## 286 0  
## 287 1  
## 288 0  
## 289 1  
## 290 1  
## 291 1  
## 292 0  
## 293 1  
## 294 0  
## 295 0  
## 296 0  
## 297 0  
## 298 0  
## 299 0  
## 300 0  
## 301 0  
## 302 1  
## 303 1  
## 304 1  
## 305 1  
## 306 1  
## 307 0  
## 308 0  
## 309 0  
## 310 1  
## 311 0  
## 312 0  
## 313 1  
## 314 0  
## 315 0  
## 316 1  
## 317 0  
## 318 0  
## 319 0  
## 320 1  
## 321 1  
## 322 0  
## 323 0  
## 324 0  
## 325 0  
## 326 1  
## 327 1  
## 328 0  
## 329 0  
## 330 1  
## 331 0  
## 332 0  
## 333 1  
## 334 0  
## 335 0  
## 336 1  
## 337 0  
## 338 0  
## 339 0  
## 340 0  
## 341 1  
## 342 1  
## 343 0  
## 344 0  
## 345 1  
## 346 0  
## 347 0  
## 348 1  
## 349 0  
## 350 1  
## 351 0  
## 352 0  
## 353 0  
## 354 0  
## 355 1  
## 356 0  
## 357 1  
## 358 1  
## 359 1  
## 360 0  
## 361 1  
## 362 1  
## 363 0  
## 364 1  
## 365 0  
## 366 1  
## 367 0  
## 368 0  
## 369 0  
## 370 0  
## 371 1  
## 372 1  
## 373 0  
## 374 1  
## 375 0  
## 376 0  
## 377 0  
## 378 1  
## 379 1  
## 380 0  
## 381 0  
## 382 1  
## 383 0  
## 384 0  
## 385 1  
## 386 0  
## 387 0  
## 388 1  
## 389 0  
## 390 1  
## 391 0  
## 392 0  
## 393 0

test\_target <- as.factor(Test[,6])  
test\_target

## [1] 0 0 1 0 0 1 0 0 1 1 0 1 0 1 0 1 1 1 1 1 0 0 1 1 0 1 1 1 0 0 0 1 1 1 1 1 1  
## [38] 0 1 0 1 1 0 0 0 0 1 1 1 1 0 1 0 1 1 0 0 1 0 1 0 1 1 1 0 1 1 0 1 0 1 0 0 0  
## [75] 0 1 0 0 0 0 1 1 1 0 1 0 1 0 1 1 1 0 1 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0 1 0  
## [112] 0 1 0 0 1 0 0 0 1 1 0 0 0 0 1 1 0 0 1 0 0 1 0 0 1 0 0 0 0 1 1 0 0 1 0 0 1  
## [149] 0 1 0 0 0 0 1 0 1 1 1 0 1 1 0 1 0 1 0 0 0 0 1 1 0 1 0 0 0 1 1 0 0 1 0 0 1  
## [186] 0 0 1 0 1 0 0 0  
## Levels: 0 1

library(class)   
require(class)  
Test\_pred <- knn(train = Train, test = Test,  
 cl = Train\_labels, k = 21)  
Test\_pred

## [1] 0 0 1 0 0 1 0 0 1 1 0 1 0 1 0 1 1 1 1 1 0 0 1 1 0 1 1 1 0 0 0 1 1 1 1 1 1  
## [38] 0 1 0 1 1 0 0 0 0 1 1 1 1 0 1 0 1 1 0 0 1 0 1 0 1 1 1 0 1 1 0 1 0 1 0 0 0  
## [75] 0 1 0 0 0 0 1 1 1 0 1 0 1 0 1 1 1 0 1 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0 1 0  
## [112] 0 1 0 0 1 0 0 0 1 1 0 0 0 0 1 1 0 0 1 0 0 1 0 0 1 0 0 0 0 1 1 0 0 1 0 0 1  
## [149] 0 1 0 0 0 0 1 0 1 1 1 0 1 1 0 1 0 1 0 0 0 0 1 1 0 1 0 0 0 1 1 0 0 1 0 0 1  
## [186] 0 0 1 0 1 0 0 0  
## Levels: 0 1

#Model Evaluation  
t<-table(factor(Test\_pred))  
t

##   
## 0 1   
## 105 88

t1<-table(test\_target,Test\_pred)  
t1

## Test\_pred  
## test\_target 0 1  
## 0 105 0  
## 1 0 88

# Checking the accuracy  
accuracy <- function(x){sum(diag(x)/(sum(rowSums(x)))) \* 100}  
accuracy(t1)

## [1] 100

# Challenge the Solution

I didnt find any need to challenge to question because the accuracy of the model is 100%.