Week 12 Core IP

##1. Defining the Question ## a) Specifying the Data Analytic Question

Identify which individuals are most likely to click on ads from a cryptography course website

b) Defining the Metric for Success

For this sttudy, 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) 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 ### 1.4 Drafting the Experimental Design 1. Define the question, set the metric for success, outline the context, drafting the experimental design, and determining the appropriateness of the data. 2. Load the dataset and previewing it. 3. Check for missing and duplicated values and deal with them where necessary. 4. Check for outliers and other anomalies and deal with them where necessary. 5. Perform univariate and bivariate analysis. 6. Create a baseline model and assess its accuracy score. 7. Challenge the solution. 8. Conclude and provide insights on how this project can be improved.

2. Data Preparation and Cleaning

```
# importing and previewing the dataset
df<-read.csv('http://bit.ly/IPAdvertisingData')
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
```

```
#Data Dimensions
paste("The dimensions of the data frame are ", paste (dim(df), collapse = ','))
```

```
## [1] "The dimensions of the data frame are  1000,10"
```

```
#Datatypes
sapply(df, class)
```

```
## Daily.Time.Spent.on.Site                      Age              Area.Income
##                "numeric"                "integer"                "numeric"
##     Daily.Internet.Usage            Ad.Topic.Line                     City
##                "numeric"              "character"              "character"
##                     Male                  Country                Timestamp
##                "integer"              "character"              "character"
##            Clicked.on.Ad
##                "integer"
```

#We have a mix of datatypes from numeric, integer and character

```
#Summary
summary(df)
```

```
##  Daily.Time.Spent.on.Site      Age         Area.Income    Daily.Internet.Usage
##  Min.   :32.60            Min.   :19.00   Min.   :13996   Min.   :104.8
##  1st Qu.:51.36            1st Qu.:29.00   1st Qu.:47032   1st Qu.:138.8
##  Median :68.22            Median :35.00   Median :57012   Median :183.1
##  Mean   :65.00            Mean   :36.01   Mean   :55000   Mean   :180.0
##  3rd Qu.:78.55            3rd Qu.:42.00   3rd Qu.:65471   3rd Qu.:218.8
##  Max.   :91.43            Max.   :61.00   Max.   :79485   Max.   :270.0
##  Ad.Topic.Line          City                 Male          Country
```

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

```r
#Checking for unique characters
sapply(df, function(x) length(unique(x)))
```

```
## Daily.Time.Spent.on.Site                      Age              Area.Income
##                      900                       43                     1000
##     Daily.Internet.Usage            Ad.Topic.Line                     City
##                      966                     1000                      969
##                     Male                  Country                Timestamp
##                        2                      237                     1000
##            Clicked.on.Ad
##                        2
```

Data Cleaning

```r
# checking for duplicates
anyDuplicated(df)
```

```
## [1] 0
```

There are no duplicated records so there is no need to remove any of them.

```r
# looking for missing values
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
##                     Male                  Country                Timestamp
##                        0                        0                        0
##            Clicked.on.Ad
##                        0
```

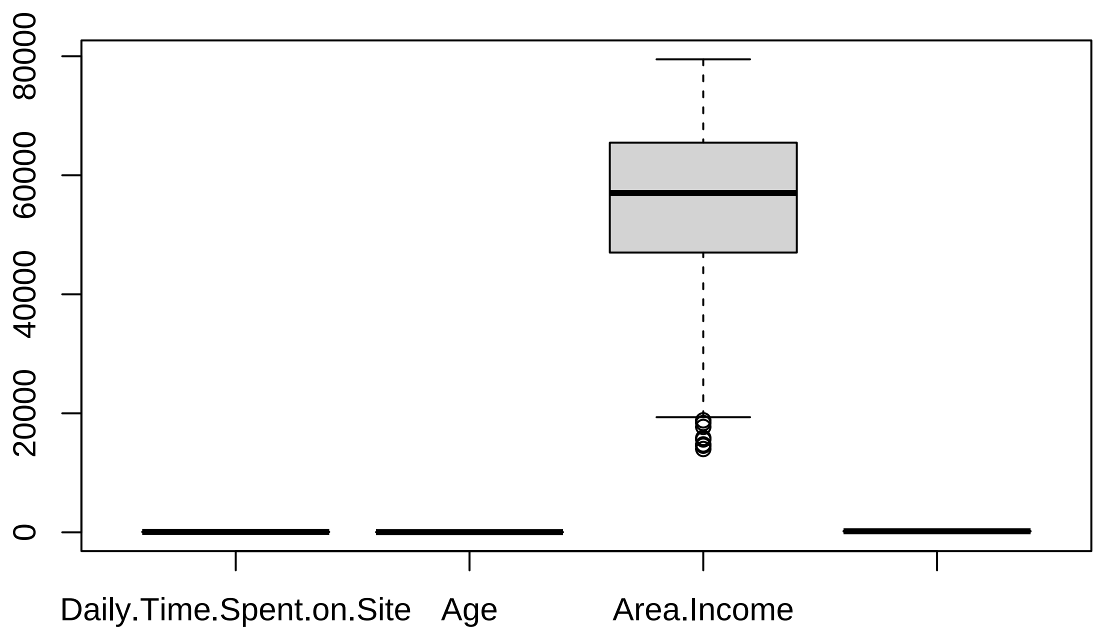
There are no missing values in each column so we don't need to carry out imputation or replacement.

#Checking for outliers #First we select numeric columns excluding male and clicked.on.ad since they are binary column

```
df1 <- subset(df, select = -c(Ad.Topic.Line,City,   Male,   Country,    Timestamp,  Clicked.on.Ad))
head(df1)
```

```
##   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
```

```
#Plotting boxplots to check for outliers
boxplot(df1
        )
```




```
boxplot.stats(df1$Area.Income)$out
```

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

We won't remove the above figures because it concerns income and people earn different amounts of money.

```
#Change datattypes
df$Male <- as.factor(df$Male)
```

```r
df$Clicked.on.Ad <- as.factor(df$Clicked.on.Ad)
#Checking datatypes
sapply(df, class)
```

```
## Daily.Time.Spent.on.Site                      Age              Area.Income
##                "numeric"                "integer"                "numeric"
##     Daily.Internet.Usage            Ad.Topic.Line                     City
##                "numeric"              "character"              "character"
##                     Male                  Country                Timestamp
##                 "factor"              "character"              "character"
##            Clicked.on.Ad
##                 "factor"
```

```r
# split timestamp column into year, month, day, and hour
# NB: minute and second are irrelevant to our analysis
df$year <- format(as.POSIXct(df$Timestamp, format="%Y-%m-%d %H:%M:%S"), "%Y")
df$month <- format(as.POSIXct(df$Timestamp, format="%Y-%m-%d %H:%M:%S"), "%m")
df$day <- format(as.POSIXct(df$Timestamp, format="%Y-%m-%d %H:%M:%S"), "%d")
df$hour <- format(as.POSIXct(df$Timestamp, format="%Y-%m-%d %H:%M:%S"), "%H")
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 year month day hour
## 1 2016-03-27 00:53:11             0 2016    03  27   00
## 2 2016-04-04 01:39:02             0 2016    04  04   01
## 3 2016-03-13 20:35:42             0 2016    03  13   20
## 4 2016-01-10 02:31:19             0 2016    01  10   02
## 5 2016-06-03 03:36:18             0 2016    06  03   03
## 6 2016-05-19 14:30:17             0 2016    05  19   14
```

```r
#Dropping the column Timestamp and Ad.Topic.Line
df_clean = subset(df, select = -c(Timestamp,Ad.Topic.Line))
head(df_clean)
```

```
##   Daily.Time.Spent.on.Site Age Area.Income Daily.Internet.Usage           City
## 1                    68.95  35    61833.90               256.09    Wrightburgh
## 2                    80.23  31    68441.85               193.77      West Jodi
## 3                    69.47  26    59785.94               236.50       Davidton
## 4                    74.15  29    54806.18               245.89 West Terrifurt
```

```
## 5                    68.37  35    73889.99               225.58   South Manuel
## 6                    59.99  23    59761.56               226.74      Jamieberg
##   Male    Country Clicked.on.Ad year month day hour
## 1    0    Tunisia             0 2016    03  27   00
## 2    1      Nauru             0 2016    04  04   01
## 3    0 San Marino             0 2016    03  13   20
## 4    1      Italy             0 2016    01  10   02
## 5    0    Iceland             0 2016    06  03   03
## 6    1     Norway             0 2016    05  19   14
```

```
#Datatypes
sapply(df_clean, class)
```

```
## Daily.Time.Spent.on.Site                      Age              Area.Income
##                "numeric"                "integer"                "numeric"
##     Daily.Internet.Usage                     City                     Male
##                "numeric"              "character"                 "factor"
##                  Country            Clicked.on.Ad                     year
##              "character"                 "factor"              "character"
##                    month                      day                     hour
##              "character"              "character"              "character"
```

```
# set the new columns to be of data type Factor
df_clean$year <- as.factor(df_clean$year)
df_clean$month <- as.factor(df_clean$month)
df_clean$day <- as.factor(df_clean$day)
df_clean$hour <- as.factor(df_clean$hour)
```

```
#Datatypes
sapply(df_clean, class)
```

```
## Daily.Time.Spent.on.Site                      Age              Area.Income
##                "numeric"                "integer"                "numeric"
##     Daily.Internet.Usage                     City                     Male
##                "numeric"              "character"                 "factor"
##                  Country            Clicked.on.Ad                     year
##              "character"                 "factor"                 "factor"
##                    month                      day                     hour
##                 "factor"                 "factor"                 "factor"
```

Exploratory Data Analysis

Univariate Analysis

```
colnames(df_clean)
```

```
##  [1] "Daily.Time.Spent.on.Site" "Age"
##  [3] "Area.Income"              "Daily.Internet.Usage"
##  [5] "City"                     "Male"
##  [7] "Country"                  "Clicked.on.Ad"
##  [9] "year"                     "month"
## [11] "day"                      "hour"
```

```r
#Selecting the numeric columns
num <- subset(df_clean, select = -c(City,   Male,   Country,    Clicked.on.Ad, month,day,hour,year))
#Getting the measures of central tendency
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
```

Variance and Standard deviation

```r
var(df_clean$Age)
```

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

```r
sd(df_clean$Age)
```

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

```r
var(df_clean$Area.Income)
```

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

```r
sd(df_clean$Area.Income)
```

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

```r
var(df_clean$Daily.Internet.Usage)
```

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

```r
sd(df_clean$Daily.Internet.Usage)
```

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

```r
var(df_clean$Daily.Time.Spent.on.Site)
```

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

```
sd(df_clean$Daily.Time.Spent.on.Site)
```

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

Conclusions

1. The minimum amount of time spent on the blog is 32.60 and maximum is 91.43 with a mean at 65 and median at 68
2. The mean age of people visiting the site is 36, max age is 61 and min age is 19 which makes sense since the range between 61 and 19 are the people most active online. 3.From data, the maximum income of individuals is 79485 and a min income of 13996 4.The mean daily internet usage on the website is 180 and a median level at 183.1

```
library(moments)
```

```
#Checking for skewness
paste("Daily Time_Spent_Skewness: ", paste (skewness(df_clean$Daily.Time.Spent.on.Site), collapse = ','
```

```
## [1] "Daily Time_Spent_Skewness:  -0.371202614867441"
```

```
paste("Income_Skewness: ", paste (skewness(df_clean$Area.Income), collapse = ','))
```

```
## [1] "Income_Skewness:  -0.649396701694076"
```

```
paste("Age_Skewness: ", paste (skewness(df_clean$Age), collapse = ','))
```

```
## [1] "Age_Skewness:  0.478422676206608"
```

```
paste("Daily_Internet_Usage_Skewness: ", paste (skewness(df_clean$Daily.Internet.Usage), collapse = ','
```

```
## [1] "Daily_Internet_Usage_Skewness:  -0.0334870316434409"
```

```
#Checking for kurtosis
paste("Daily Time_Spent_Kurtosis: ", paste (kurtosis(df_clean$Daily.Time.Spent.on.Site), collapse = ','
```

```
## [1] "Daily Time_Spent_Kurtosis:  1.90394215401081"
```

```
paste("Income_Kurtosis: ", paste (kurtosis(df_clean$Area.Income), collapse = ','))
```

```
## [1] "Income_Kurtosis:  2.89469406161926"
```

```
paste("Age_Kurtosis: ", paste (kurtosis(df_clean$Age), collapse = ','))
```

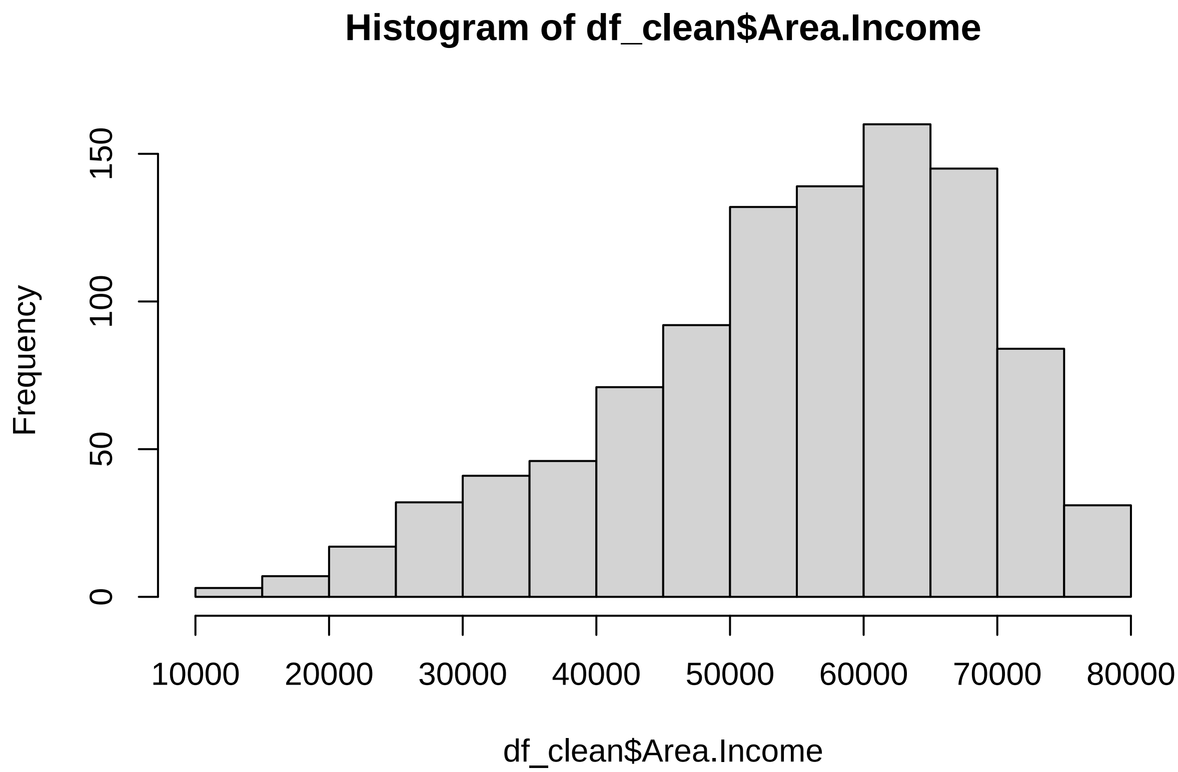
```
## [1] "Age_Kurtosis:  2.59548176807726"
```

```
paste("Daily_Internet_Usage_Kurtosis: ", paste (kurtosis(df_clean$Daily.Internet.Usage), collapse = ','
```

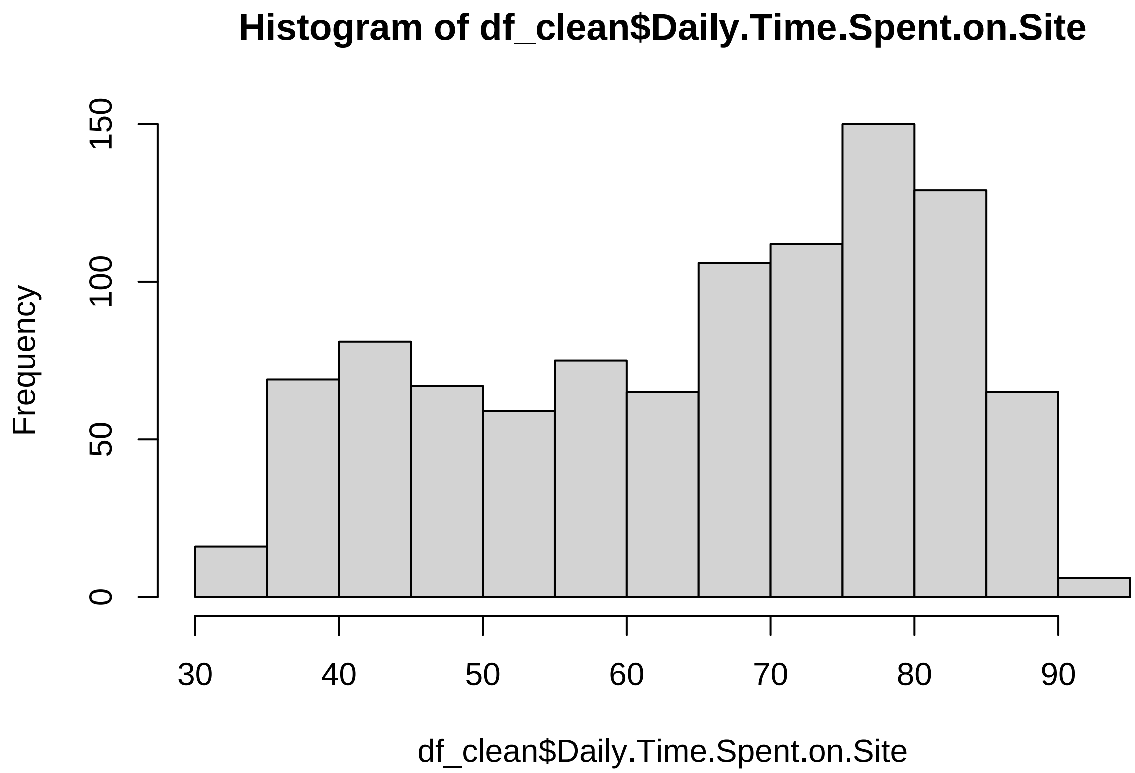
```
## [1] "Daily_Internet_Usage_Kurtosis:  1.72770118094819"
```

```
hist(df_clean$Age)
```

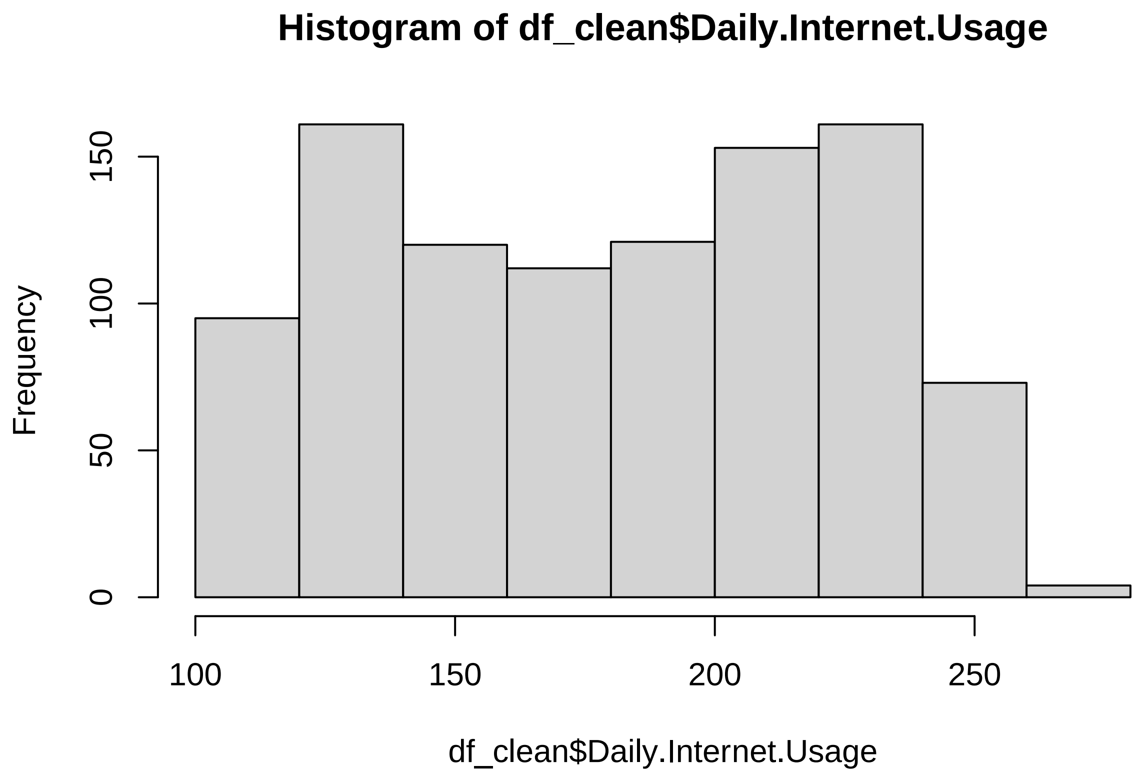

```
hist(df_clean$Area.Income)
```




```
hist(df_clean$Daily.Time.Spent.on.Site)
```




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



Observation -Age: Most people who visit the blog are between 25 and 40 years, data is skewed to the right of the mean. Graph doesn't show a sharp peak. The skewness value implies that the distribution is almost fairly symmetrical, so our initial assumption based on just looking at the visualization of the distribution is slightly wrong.

-Income: Data on income is mostly skewed to the right of the 55,00 mean. A kurtosis value of 2.89 indicates that the distribution is platykurtic although it is getting very close to being mesokurtic.The distribution is negatively skewed.

-Daily internet usage: The distribution is platykurtic. The distribution appears to be relatively uniform and bimodal.

-Time spent on site: There are lots of variations on how much time people spend on the site. A good number does spend between 65 and 85 time on the site.

```
library(plyr)
```

City

```
# displaying the first 6 frequently occurring cities
count_city <- count(df_clean$City)
count_city_head <- head(arrange(count_city, desc(freq)))
count_city_head
```

```
##                  x freq
## 1       Lisamouth    3
```

```
## 2    Williamsport    3
## 3 Benjaminchester    2
## 4       East John    2
## 5    East Timothy    2
## 6        Johnstad    2
```

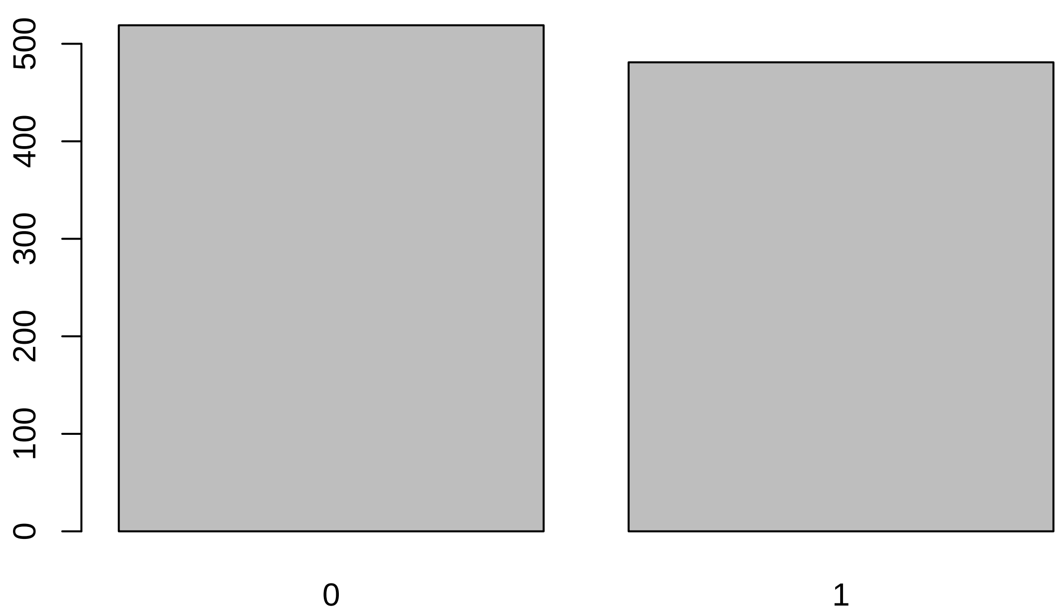
male

```r
male_table <- table(df_clean$Male)
male_table
```

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

We see here that 591 are not male while 481 are. To easily visualize this:

```r
barplot(male_table)
```



country

```r
# displaying the first 10 frequently occuring countries
count_country <- count(df_clean$Country)
count_country_head <- head(arrange(count_country, desc(freq)), 10)
count_country_head
```

```
##                  x freq
## 1  Czech Republic    9
## 2          France    9
## 3     Afghanistan    8
## 4       Australia    8
## 5          Cyprus    8
## 6          Greece    8
## 7         Liberia    8
## 8      Micronesia    8
## 9            Peru    8
## 10        Senegal    8
```

month

```
# displaying the months in order of most frequently occurring to least frequently occurring
count_months <- count(df_clean$month)
arrange(count_months, desc(freq))
```

```
##    x freq
## 1 02  160
## 2 03  156
## 3 01  147
## 4 04  147
## 5 05  147
## 6 06  142
## 7 07  101
```

We see here that February is the most frequently occurring month with July being the least frequently occurring month.

day

```
# displaying top 5 frequently occurring days
count_days <- count(df_clean$day)
head(arrange(count_days, desc(freq)), 5)
```

```
##    x freq
## 1 03   46
## 2 17   42
## 3 15   41
## 4 10   37
## 5 04   36
```

The 3rd day is the most frequently occurring day overall. However, to get a more accurate picture of this, we will look at which day occurs most frequently in which month. We will do this in bivariate analysis.

```
tail(arrange(count_days, desc(freq)),5)
```

```
##     x freq
## 27 02   25
## 28 06   25
## 29 22   24
## 30 25   23
## 31 31   18
```

The 31st day seems to be the least occurring day.

hour

```r
# displaying the top 5 hours
count_hours <- count(df_clean$hour)
head(arrange(count_hours, desc(freq)), 5)
```

```
##    x freq
## 1 07   54
## 2 20   50
## 3 09   49
## 4 21   48
## 5 00   45
```

Most frequently occurring time appears to be around 7 AM.

```r
tail(arrange(count_hours, desc(freq)), 5)
```

```
##     x freq
## 20 12   38
## 21 02   36
## 22 15   35
## 23 01   32
## 24 10   31
```

Least frequently occurring time appears to be around 10 AM. This is probably because more people get engrossed in the day's work.

clicked on ad

```r
ad_table <- table(df_clean$Clicked.on.Ad)
print(ad_table)
```

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

Looks like the number of people who both clicked on the ad and didn't click on the ad is the same (500 each).

Bivariate Analysis

We will start by looking at the relationship between our target variable (clicked_on_ad) and the other variables.

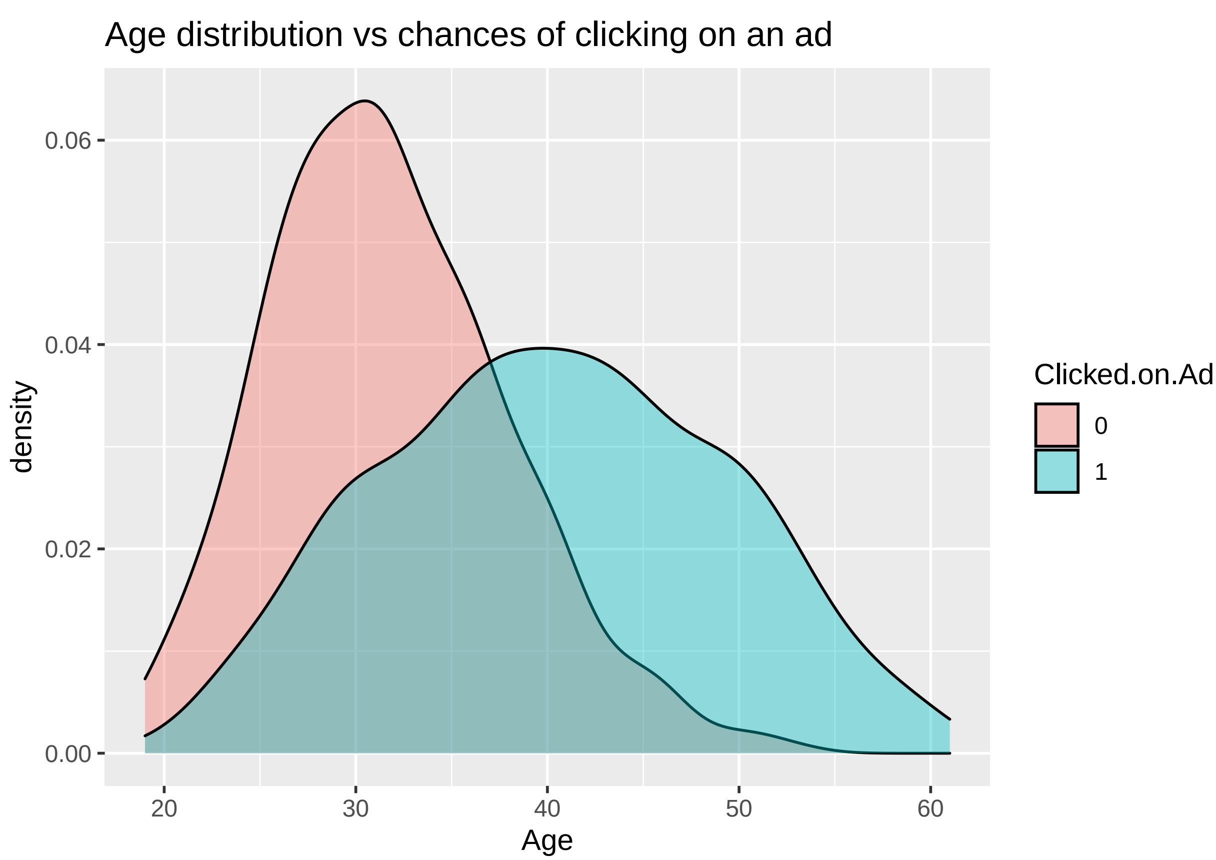
```
# how many males clicked on ads
ad_male.table <- table(df_clean$Clicked.on.Ad, df_clean$Male)
names(dimnames(ad_male.table)) <- c("Clicked on Ad?", "Male?")
ad_male.table
```

```
##                Male?
## Clicked on Ad?   0   1
##              0 250 250
##              1 269 231
```

From this we see that of those who clicked on the ad, 269 were female while 231 were male. There was no difference in gender of those who did not click on the ad.

```
library(ggplot2)
```

```
#Age and it's relationship to clicking an ad
ggplot(df_clean,
       aes(x = Age,
           fill = Clicked.on.Ad)) +
  geom_density(alpha = 0.4) +
  labs(title = "Age distribution vs chances of clicking on an ad")
```



People from all age groups click on ads on the site. People above 40 are more likely to click on an ad as per the graph above. while younger people dont click as often

```
# ad clicked per month
ad_month.table <- table(df_clean$month, df_clean$Clicked.on.Ad)
names(dimnames(ad_month.table)) <- c("Month", "Clicked on Ad?")
ad_month.table
```

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

Looking at this table, we see that February reports the highest number of ads clicked and July the least.

```
# ad clicked per day
ad_day.table <- table(df_clean$day, df_clean$Clicked.on.Ad)
names(dimnames(ad_day.table)) <- c("Day", "Clicked on Ad?")
ad_day.table
```

```
##     Clicked on Ad?
## Day   0  1
##   01 14 19
##   02 15 10
##   03 20 26
##   04 22 14
##   05 17 18
##   06 11 14
##   07 18 14
##   08 20 15
##   09 14 20
##   10 18 19
##   11 17 15
##   12  9 20
##   13 13 17
##   14 12 21
##   15 21 20
##   16 21 14
##   17 24 18
##   18 18 17
##   19 17 12
##   20 22 11
##   21 17 15
##   22 14 10
##   23 13 22
##   24 15 18
##   25  8 15
##   26 21 15
```

```
##   27 19 16
##   28 13 17
##   29 14 15
##   30 14 14
##   31  9  9
```

Day 03 has the highest number of ads clicked. Day 31 has the least.

```r
# ad clicked per hour
ad_hour.table <- table(df_clean$hour, df_clean$Clicked.on.Ad)
names(dimnames(ad_hour.table)) <- c("Hour", "Clicked on Ad?")
ad_hour.table
```

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

Hour 09 (9 AM) returned the highest number of ads clicked, 28, whereas Hour 10 (10 AM) returned the lowest, 14.

```r
# ad clicked per city
ad_city.table <- table(df_clean$City, df_clean$Clicked.on.Ad)
names(dimnames(ad_city.table)) <- c("City", "Clicked on Ad?")
ad_city.table
```

```
##                           Clicked on Ad?
## City                        0 1
##   Adamsbury                 0 1
##   Adamside                  0 1
##   Adamsstad                 1 0
```

```
##    Alanview                 1 0
##    Alexanderfurt            0 1
##    Alexanderview            0 1
##    Alexandrafort            1 0
##    Alexisland               1 0
##    Aliciatown               0 1
##    Alvaradoport             0 1
##    Alvarezland              0 1
##    Amandafort               0 1
##    Amandahaven              0 1
##    Amandaland               1 0
##    Amyfurt                  1 0
##    Amyhaven                 1 0
##    Andersonchester          0 1
##    Andersonfurt             0 1
##    Andersonton              1 0
##    Andrewborough            0 1
##    Andrewmouth              1 0
##    Angelhaven               1 0
##    Anthonyfurt              1 0
##    Ashleychester            1 0
##    Ashleymouth              1 0
##    Austinborough            1 0
##    Austinland               1 0
##    Bakerhaven               1 0
##    Barbershire              1 0
##    Beckton                  1 0
##    Benjaminchester          2 0
##    Bernardton               0 1
##    Bethburgh                0 1
##    Birdshire                1 0
##    Blairborough             0 1
##    Blairville               1 0
##    Blevinstown              0 1
##    Bowenview                1 0
##    Boyerberg                0 1
##    Bradleyborough           1 0
##    Bradleyburgh             0 1
##    Bradleyside              0 1
##    Bradshawborough          1 0
##    Bradyfurt                0 1
##    Brandiland               0 1
##    Brandonbury              0 1
##    Brandonstad              1 0
##    Brandymouth              0 1
##    Brendaburgh              1 0
##    Brendachester            0 1
##    Brianabury               1 0
##    Brianfurt                0 1
##    Brianland                0 1
##    Brittanyborough          0 1
##    Brownbury                1 0
##    Brownport                0 1
##    Brownton                 0 1
```

```
##    Browntown               0 1
##    Brownview               1 0
##    Bruceburgh              1 0
##    Burgessside             0 1
##    Butlerfort              0 1
##    Calebberg               1 0
##    Cameronberg             0 1
##    Campbellstad            1 0
##    Cannonbury              1 0
##    Carsonshire             1 0
##    Carterburgh             1 0
##    Carterland              0 1
##    Carterport              1 0
##    Carterton               1 0
##    Cassandratown           1 0
##    Catherinefort           0 1
##    Cervantesshire          0 1
##    Chapmanland             1 0
##    Chapmanmouth            0 1
##    Charlenetown            0 1
##    Charlesbury             1 0
##    Charlesport             0 1
##    Charlottefort           0 1
##    Chaseshire              0 1
##    Chrismouth              0 1
##    Christinehaven          0 1
##    Christinetown           0 1
##    Christopherchester      1 0
##    Christopherport         0 1
##    Christopherville        1 0
##    Clarkborough            0 1
##    Claytonside             1 0
##    Clineshire              1 0
##    Codyburgh               0 1
##    Coffeytown              1 0
##    Colebury                0 1
##    Colemanshire            1 0
##    Collinsburgh            1 0
##    Combsstad               0 1
##    Contrerasshire          1 0
##    Costaburgh              0 1
##    Courtneyfort            0 1
##    Coxhaven                1 0
##    Cranemouth              1 0
##    Crawfordfurt            0 1
##    Cunninghamhaven         0 1
##    Curtisport              0 1
##    Curtisview              1 0
##    Cynthiaside             1 0
##    Daisymouth              1 0
##    Danielview              0 1
##    Davidmouth              0 1
##    Davidside               0 1
##    Davidstad               0 1
```

```
##    Davidton                 1 0
##    Davidview                0 1
##    Daviesborough            1 0
##    Davieshaven              1 0
##    Davilachester            0 1
##    Davisfurt                0 1
##    Dayton                   1 0
##    Deannaville              1 0
##    Debraburgh               0 1
##    Derrickhaven             1 0
##    Destinyfurt              0 1
##    Dianashire               1 0
##    Dianaville               0 1
##    Donaldshire              1 0
##    Douglasview              1 0
##    Duffystad                0 1
##    Dustinborough            1 0
##    Dustinchester            1 0
##    Dustinmouth              0 1
##    East Aaron               1 0
##    East Anthony             0 1
##    East Barbara             0 1
##    East Benjaminville       1 0
##    East Breannafurt         0 1
##    East Brettton            0 1
##    East Brianberg           1 0
##    East Brittanyville       0 1
##    East Carlos              1 0
##    East Christopher         1 0
##    East Christopherbury     1 0
##    East Connie              1 0
##    East Dana                0 1
##    East Deborahhaven        1 0
##    East Debraborough        1 0
##    East Donna               0 1
##    East Donnatown           1 0
##    East Eric                0 1
##    East Ericport            0 1
##    East Georgeside          0 1
##    East Graceland           1 0
##    East Heatherside         0 1
##    East Heidi               0 1
##    East Henry               1 0
##    East Jason               0 1
##    East Jennifer            1 0
##    East Jessefort           0 1
##    East John                1 1
##    East Johnport            1 0
##    East Kevinbury           0 1
##    East Lindsey             0 1
##    East Maureen             0 1
##    East Michaelland         1 0
##    East Michaelmouth        0 1
##    East Michaeltown         1 0
```

```
##    East Michele            0 1
##    East Michelleberg       0 1
##    East Mike               0 1
##    East Paul               1 0
##    East Rachaelfurt        0 1
##    East Rachelview         0 1
##    East Ronald             0 1
##    East Samanthashire      0 1
##    East Sharon             0 1
##    East Shawn              0 1
##    East Shawnchester       1 0
##    East Sheriville         1 0
##    East Stephen            0 1
##    East Susanland          1 0
##    East Tammie             0 1
##    East Theresashire       1 0
##    East Tiffanyport        1 0
##    East Timothy            2 0
##    East Timothyport        1 0
##    East Toddfort           1 0
##    East Troyhaven          1 0
##    East Tylershire         0 1
##    East Valerie            1 0
##    East Vincentstad        0 1
##    East Yvonnechester      0 1
##    Edwardmouth             1 0
##    Edwardsmouth            1 0
##    Edwardsport             0 1
##    Elizabethbury           0 1
##    Elizabethmouth          1 0
##    Elizabethport           0 1
##    Elizabethstad           0 1
##    Emilyfurt               1 0
##    Ericksonmouth           0 1
##    Erikville               1 0
##    Erinmouth               1 0
##    Erinton                 0 1
##    Estesfurt               0 1
##    Estradafurt             1 0
##    Estradashire            0 1
##    Evansfurt               1 0
##    Evansville              0 1
##    Faithview               1 0
##    Florestown              0 1
##    Fosterside              0 1
##    Frankbury               0 1
##    Frankchester            1 0
##    Frankport               0 1
##    Fraziershire            0 1
##    Garciamouth             0 1
##    Garciaside              0 1
##    Garciatown              1 0
##    Garciaview              0 1
##    Garnerberg              1 0
```

```
##    Garrettborough         1 0
##    Garychester            1 0
##    Gilbertville           1 0
##    Gomezport              1 0
##    Gonzalezburgh          1 0
##    Grahamberg             0 1
##    Gravesport             1 0
##    Greenechester          1 0
##    Greentown              1 0
##    Greerport              0 1
##    Greerton               1 0
##    Greghaven              1 0
##    Guzmanland             0 1
##    Haleberg               1 0
##    Haleview               1 0
##    Hallfort               1 0
##    Hamiltonfort           0 1
##    Hammondport            1 0
##    Hannahside             1 0
##    Hannaport              0 1
##    Hansenland             0 1
##    Hansenmouth            0 1
##    Harmonhaven            1 0
##    Harperborough          0 1
##    Harrishaven            1 0
##    Harrisonmouth          1 0
##    Hartmanchester         0 1
##    Hartport               1 0
##    Harveyport             0 1
##    Hatfieldshire          1 0
##    Hawkinsbury            0 1
##    Hayesmouth             1 0
##    Heatherberg            0 1
##    Helenborough           0 1
##    Hendrixmouth           0 1
##    Henryfort              1 0
##    Henryland              0 1
##    Hernandezchester       1 0
##    Hernandezfort          1 0
##    Hernandezside          0 1
##    Hernandezville         0 1
##    Hessstad               1 0
##    Hintonport             0 1
##    Hobbsbury              0 1
##    Holderville            0 1
##    Hollandberg            1 0
##    Hollyfurt              1 0
##    Hubbardmouth           0 1
##    Huffmanchester         0 1
##    Hughesport             0 1
##    Hurleyborough          1 0
##    Ianmouth               1 0
##    Ingramberg             1 0
##    Isaacborough           0 1
```

```
##    Jacksonburgh              0 1
##    Jacksonmouth              1 0
##    Jacksonstad               0 1
##    Jacobstad                 0 1
##    Jacquelineshire           0 1
##    Jamesberg                 1 0
##    Jamesfurt                 0 1
##    Jamesmouth                0 1
##    Jamesville                1 0
##    Jamieberg                 1 0
##    Jamiefort                 1 0
##    Janiceview                1 0
##    Jasminefort               1 0
##    Jayville                  1 0
##    Jeffreyburgh              0 1
##    Jeffreymouth              0 1
##    Jeffreyshire              1 0
##    Jenniferhaven             0 1
##    Jenniferstad              1 0
##    Jensenborough             0 1
##    Jensenton                 0 1
##    Jeremybury                0 1
##    Jeremyshire               1 0
##    Jessicahaven              0 1
##    Jessicashire              0 1
##    Jessicastad               0 1
##    Joanntown                 1 0
##    Joechester                0 1
##    Johnport                  1 0
##    Johnsonfort               1 0
##    Johnsontown               0 1
##    Johnsonview               0 1
##    Johnsport                 1 0
##    Johnstad                  2 0
##    Johnstonmouth             0 1
##    Johnstonshire             1 0
##    Jonathanland              0 1
##    Jonathantown              0 1
##    Jonesland                 1 0
##    Jonesmouth                1 0
##    Jonesshire                0 1
##    Joneston                  1 1
##    Jordanmouth               1 0
##    Jordanshire               0 1
##    Jordantown                0 1
##    Josephberg                0 1
##    Josephmouth               0 1
##    Josephstad                0 1
##    Joshuaburgh               1 0
##    Joshuamouth               1 0
##    Juanport                  1 0
##    Juliaport                 1 0
##    Julietown                 0 1
##    Karenmouth                1 0
```

```
##    Karenton                 1 0
##    Katieport                0 1
##    Kaylashire               1 0
##    Keithtown                0 1
##    Kellytown                1 0
##    Kennedyfurt              1 0
##    Kennethview              1 0
##    Kentmouth                0 1
##    Kevinberg                0 1
##    Kevinchester             1 0
##    Kimberlyhaven            1 0
##    Kimberlymouth            0 1
##    Kimberlytown             1 0
##    Kingchester              0 1
##    Kingshire                1 0
##    Klineside                0 1
##    Knappburgh               1 0
##    Kristineberg             1 0
##    Kristinfurt              0 1
##    Kristintown              0 1
##    Kyleborough              0 1
##    Kylieview                1 0
##    Lake Adrian              1 0
##    Lake Allenville          0 1
##    Lake Amanda              0 1
##    Lake Amy                 1 0
##    Lake Angela              1 0
##    Lake Annashire           1 0
##    Lake Beckyburgh          0 1
##    Lake Brandonview         0 1
##    Lake Brian               1 0
##    Lake Cassandraport       0 1
##    Lake Charlottestad       0 1
##    Lake Christopherfurt     0 1
##    Lake Conniefurt          0 1
##    Lake Courtney            1 0
##    Lake Craigview           0 1
##    Lake Cynthia             1 0
##    Lake Danielle            1 0
##    Lake David               0 2
##    Lake Deannaborough       1 0
##    Lake Deborahburgh        1 0
##    Lake Dustin              0 1
##    Lake Edward              0 1
##    Lake Elizabethside       1 0
##    Lake Evantown            0 1
##    Lake Faith               0 1
##    Lake Gerald              0 1
##    Lake Hailey              1 0
##    Lake Ian                 0 1
##    Lake Jacob               1 0
##    Lake Jacqueline          1 0
##    Lake James               0 2
##    Lake Jasonchester        1 0
```

```
##    Lake Jennifer           0 1
##    Lake Jenniferton        1 0
##    Lake Jessica            0 1
##    Lake Jessicaville       0 1
##    Lake Jesus              0 1
##    Lake Jillville          1 0
##    Lake John               0 1
##    Lake Johnbury           0 1
##    Lake Jonathanview       1 0
##    Lake Jose               1 1
##    Lake Joseph             1 0
##    Lake Josetown           1 0
##    Lake Joshuafurt         0 1
##    Lake Kevin              1 0
##    Lake Kurtmouth          1 0
##    Lake Lisa               1 0
##    Lake Matthew            0 1
##    Lake Matthewland        1 0
##    Lake Melindamouth       1 0
##    Lake Michael            1 0
##    Lake Michaelport        1 0
##    Lake Michelle           0 1
##    Lake Michellebury       0 1
##    Lake Nicole             1 0
##    Lake Patrick            2 0
##    Lake Rhondaburgh        0 1
##    Lake Stephenborough     0 1
##    Lake Susan              1 1
##    Lake Timothy            1 0
##    Lake Tracy              0 1
##    Lake Vanessa            0 1
##    Lake Zacharyfurt        1 0
##    Lauraburgh              1 0
##    Laurieside              1 0
##    Lawrenceborough         1 0
##    Lawsonshire             0 1
##    Leahside                0 1
##    Leonchester             1 0
##    Lesliebury              0 1
##    Lesliefort              1 0
##    Lewismouth              0 1
##    Lindaside               1 0
##    Lindsaymouth            1 0
##    Lisaberg                1 0
##    Lisafort                1 0
##    Lisamouth               1 2
##    Lopezberg               0 1
##    Lopezmouth              1 0
##    Loriville               0 1
##    Lovemouth               0 1
##    Luischester             1 0
##    Luisfurt                1 0
##    Lukeport                1 0
##    Mackenziemouth          1 0
```

```
##    Marcushaven             1 0
##    Mariahview              0 1
##    Mariebury               1 0
##    Mariemouth              1 0
##    Markhaven               0 1
##    Masonhaven              1 0
##    Masseyshire             0 1
##    Mataberg                1 0
##    Matthewtown             0 1
##    Mauricefurt             0 1
##    Mauriceshire            1 0
##    Mcdonaldfort            1 0
##    Mclaughlinbury          1 0
##    Meaganfort              1 0
##    Meghanchester           0 1
##    Melanieton              0 1
##    Melissachester          0 1
##    Melissafurt             1 0
##    Melissastad             1 0
##    Meyerchester            1 0
##    Meyersstad              0 1
##    Mezaton                 0 1
##    Michaelland             1 0
##    Michaelmouth            1 0
##    Michaelshire            0 1
##    Micheletown             0 1
##    Michellefort            0 1
##    Michelleside            0 2
##    Millerbury              0 2
##    Millerchester           0 1
##    Millerfort              1 0
##    Millerland              1 0
##    Millerside              0 1
##    Millertown              1 1
##    Millerview              1 0
##    Mollyport               1 0
##    Monicaview              0 1
##    Morganfort              1 0
##    Morganport              0 1
##    Morrismouth             0 1
##    Mosleyburgh             1 0
##    Mullenside              1 0
##    Munozberg               1 0
##    Murphymouth             1 0
##    Nelsonfurt              0 1
##    New Amanda              0 1
##    New Angelview           0 1
##    New Brandy              1 0
##    New Brendafurt          0 1
##    New Charleschester      0 1
##    New Christinatown       0 1
##    New Cynthia             1 0
##    New Daniellefort        0 1
##    New Darlene             0 1
```

```
##    New Dawnland            1 0
##    New Debbiestad          0 1
##    New Denisebury          0 1
##    New Frankshire          1 0
##    New Gabriel             1 0
##    New Henry               0 1
##    New Hollyberg           0 1
##    New James               0 1
##    New Jamestown           1 0
##    New Jasmine             1 0
##    New Jay                 0 1
##    New Jeffreychester      1 0
##    New Jessicaport         2 0
##    New Johnberg            1 0
##    New Joshuaport          0 1
##    New Juan                1 0
##    New Julianberg          0 1
##    New Julie               1 0
##    New Karenberg           0 1
##    New Kayla               1 0
##    New Keithburgh          0 1
##    New Lindaberg           0 1
##    New Lucasburgh          0 1
##    New Marcusbury          0 1
##    New Maria               1 0
##    New Matthew             0 1
##    New Michael             0 1
##    New Michaeltown         1 0
##    New Nancy               0 1
##    New Nathan              1 0
##    New Patriciashire       1 0
##    New Patrick             0 1
##    New Paul                1 0
##    New Rachel              0 1
##    New Rebecca             0 1
##    New Sabrina             0 1
##    New Sean                1 0
##    New Shane               1 0
##    New Sharon              1 0
##    New Sheila              2 0
##    New Sonialand           1 0
##    New Steve               1 0
##    New Tammy               0 1
##    New Taylorburgh         1 0
##    New Teresa              0 1
##    New Theresa             0 1
##    New Thomas              0 1
##    New Timothy             0 1
##    New Tina                0 1
##    New Tinamouth           1 0
##    New Traceystad          1 0
##    New Travis              1 0
##    New Travistown          0 1
##    New Tyler               1 0
```

```
## 	New Wanda	1 0
## 	New Williammouth	0 1
## 	New Williamville	0 1
## 	Newmanberg	1 0
## 	Nicholasland	0 1
## 	Nicholasport	1 0
## 	North Aaronburgh	0 1
## 	North Aaronchester	0 1
## 	North Alexandra	1 0
## 	North Anaport	1 0
## 	North Andrew	0 1
## 	North Andrewstad	0 1
## 	North Angelastad	0 1
## 	North Angelatown	0 1
## 	North Anna	1 0
## 	North April	0 1
## 	North Brandon	1 0
## 	North Brittanyburgh	0 1
## 	North Cassie	0 1
## 	North Charlesbury	0 1
## 	North Christopher	1 0
## 	North Daniel	1 1
## 	North Debra	1 0
## 	North Debrashire	0 1
## 	North Derekville	0 1
## 	North Destiny	0 1
## 	North Elizabeth	1 0
## 	North Frankstad	1 0
## 	North Garyhaven	1 0
## 	North Isabellaville	1 0
## 	North Jenniferburgh	0 1
## 	North Jeremyport	1 0
## 	North Jessicaville	0 1
## 	North Johnside	1 0
## 	North Johntown	0 1
## 	North Jonathan	0 1
## 	North Joshua	1 0
## 	North Katie	0 1
## 	North Kennethside	1 0
## 	North Kevinside	0 1
## 	North Kimberly	0 1
## 	North Kristine	1 0
## 	North Lauraland	0 1
## 	North Laurenview	1 0
## 	North Leonmouth	1 0
## 	North Lisachester	1 0
## 	North Loriburgh	1 0
## 	North Mark	0 1
## 	North Maryland	0 1
## 	North Mercedes	0 1
## 	North Michael	0 1
## 	North Monicaville	1 0
## 	North Randy	1 0
## 	North Raymond	1 0
```

```
##    North Regina              0 1
##    North Ricardotown         0 1
##    North Richardburgh        0 1
##    North Ronaldshire         1 0
##    North Russellborough      0 1
##    North Samantha            0 1
##    North Sarashire           0 1
##    North Shannon             1 0
##    North Stephanieberg       1 0
##    North Tara                1 0
##    North Tiffany             1 0
##    North Tracyport           1 0
##    North Tylerland           1 0
##    North Virginia            0 1
##    North Wesleychester       1 0
##    Novaktown                 1 0
##    Odomville                 1 0
##    Olsonside                 0 1
##    Olsonstad                 0 1
##    Palmerside                0 1
##    Pamelamouth               2 0
##    Parkerhaven               1 0
##    Patriciahaven             1 0
##    Patrickmouth              1 0
##    Pattymouth                0 1
##    Paulhaven                 1 0
##    Paulport                  1 0
##    Paulshire                 1 0
##    Pearsonfort               1 0
##    Penatown                  0 1
##    Perezland                 1 0
##    Perryburgh                0 1
##    Petersonfurt              0 1
##    Phelpschester             1 0
##    Philipberg                0 1
##    Phillipsbury              0 1
##    Port Aliciabury           1 0
##    Port Angelamouth          0 1
##    Port Anthony              1 0
##    Port Aprilville           0 1
##    Port Beth                 0 1
##    Port Blake                0 1
##    Port Brenda               0 1
##    Port Brian                0 1
##    Port Brianfort            1 0
##    Port Brittanyville        1 0
##    Port Brookeland           0 1
##    Port Calvintown           1 0
##    Port Cassie               0 1
##    Port Chasemouth           1 0
##    Port Christina            0 1
##    Port Christinemouth       1 0
##    Port Christopher          0 1
##    Port Christopherborough 0 1
```

```
##    Port Crystal            0 1
##    Port Daniel             1 0
##    Port Danielleberg       1 0
##    Port Davidland          1 0
##    Port Dennis             0 1
##    Port Derekberg          0 1
##    Port Destiny            1 0
##    Port Douglasborough     0 1
##    Port Elijah             1 0
##    Port Eric               0 1
##    Port Erikhaven          0 1
##    Port Erinberg           0 1
##    Port Eugeneport         1 0
##    Port Georgebury         0 1
##    Port Gregory            1 0
##    Port Jacqueline         1 0
##    Port Jacquelinestad     1 0
##    Port James              1 0
##    Port Jasmine            1 0
##    Port Jason              1 1
##    Port Jefferybury        0 1
##    Port Jeffrey            1 0
##    Port Jennifer           0 1
##    Port Jessica            0 1
##    Port Jessicamouth       1 0
##    Port Jodi               1 0
##    Port Joshuafort         0 1
##    Port Juan               1 1
##    Port Julie              1 1
##    Port Karenfurt          1 0
##    Port Katelynview        0 1
##    Port Kathleenfort       0 1
##    Port Kevinborough       1 0
##    Port Lawrence           0 1
##    Port Maria              1 0
##    Port Mathew             1 0
##    Port Melissaberg        0 1
##    Port Melissastad        1 0
##    Port Michaelmouth       0 1
##    Port Michealburgh       0 1
##    Port Mitchell           0 1
##    Port Patrickton         0 1
##    Port Paultown           0 1
##    Port Rachel             0 1
##    Port Raymondfort        1 0
##    Port Robin              1 0
##    Port Sarahhaven         0 1
##    Port Sarahshire         0 1
##    Port Sherrystad         0 1
##    Port Stacey             1 0
##    Port Stacy              1 0
##    Port Susan              1 0
##    Port Whitneyhaven       1 0
##    Portermouth             1 0
```

```
##    Pottermouth              0 1
##    Princebury               1 0
##    Pruittmouth              1 0
##    Rachelhaven              1 0
##    Ramirezhaven             0 1
##    Ramirezland              1 0
##    Ramirezside              0 1
##    Ramirezton               1 0
##    Ramosstad                1 0
##    Randolphport             1 0
##    Randyshire               1 0
##    Rebeccamouth             0 1
##    Reginamouth              0 1
##    Reneechester             0 1
##    Reyesfurt                1 0
##    Reyesland                1 0
##    Rhondaborough            1 0
##    Richardshire             0 1
##    Richardsland             1 0
##    Richardsonland           0 1
##    Richardsonmouth          1 0
##    Richardsonshire          0 1
##    Richardsontown           1 0
##    Rickymouth               1 0
##    Riggsstad                1 0
##    Rivasland                0 1
##    Robertbury               1 0
##    Robertfurt               0 2
##    Robertmouth              1 0
##    Robertside               0 1
##    Robertsonburgh           0 1
##    Robertstown              0 1
##    Roberttown               0 1
##    Robinsonland             1 0
##    Robinsontown             0 1
##    Rochabury                0 1
##    Rogerburgh               0 1
##    Rogerland                1 0
##    Ronaldport               0 1
##    Ronniemouth              0 1
##    Russellville             0 1
##    Ryanhaven                0 1
##    Sabrinaview              1 0
##    Salazarbury              0 1
##    Samanthaland             0 1
##    Samuelborough            1 0
##    Sanchezland              1 0
##    Sanchezmouth             1 0
##    Sandersland              1 0
##    Sanderstown              0 1
##    Sandraland               1 0
##    Sandrashire              0 1
##    Sandraville              1 0
##    Sarafurt                 1 0
```

```
##    Sarahland               0 1
##    Sarahton                1 0
##    Sellerstown             1 0
##    Shaneland               1 0
##    Sharpberg               1 0
##    Shawnside               1 0
##    Shawstad                1 0
##    Shelbyport              1 1
##    Sherrishire             1 0
##    Shirleyfort             1 0
##    Silvaton                0 1
##    Smithburgh              1 0
##    Smithside               0 1
##    Smithtown               1 0
##    South Aaron             0 1
##    South Adam              0 1
##    South Adamhaven         1 0
##    South Alexisborough     0 1
##    South Blakestad         1 0
##    South Brian             1 0
##    South Cathyfurt         0 1
##    South Christopher       1 0
##    South Corey             1 0
##    South Cynthiashire      0 1
##    South Daniel            0 1
##    South Daniellefort      1 0
##    South Davidhaven        0 1
##    South Davidmouth        0 1
##    South Denise            1 0
##    South Denisefurt        1 0
##    South Dianeshire        1 0
##    South George            0 1
##    South Henry             0 1
##    South Jackieberg        0 1
##    South Jade              0 1
##    South Jaimeview         1 0
##    South Jasminebury       0 1
##    South Jeanneport        0 1
##    South Jennifer          1 0
##    South Jessica           0 1
##    South John              0 1
##    South Johnnymouth       0 1
##    South Kyle              0 1
##    South Lauraton          0 1
##    South Lauratown         0 1
##    South Lisa              0 2
##    South Manuel            1 0
##    South Margaret          0 1
##    South Mark              0 1
##    South Meghan            0 1
##    South Meredithmouth     1 0
##    South Pamela            1 0
##    South Patrickfort       1 0
##    South Peter             0 1
```

```
##    South Rebecca           0 1
##    South Renee             1 0
##    South Robert            1 0
##    South Ronald            1 0
##    South Stephanieport     1 0
##    South Tiffanyton        0 1
##    South Tomside           1 0
##    South Troy              1 0
##    South Vincentchester    0 1
##    South Walter            0 1
##    Staceyfort              0 1
##    Stephenborough          1 0
##    Stewartbury             1 0
##    Suzannetown             0 1
##    Sylviaview              1 0
##    Tammymouth              0 1
##    Tammyshire              0 1
##    Taylorberg              1 0
##    Taylorhaven             0 1
##    Taylormouth             0 1
##    Taylorport              1 0
##    Teresahaven             1 0
##    Thomasstad              1 0
##    Thomasview              1 0
##    Timothyfurt             0 1
##    Timothymouth            0 1
##    Timothyport             0 1
##    Timothytown             1 0
##    Tinachester             1 0
##    Tinaton                 0 1
##    Townsendfurt            1 0
##    Tracyhaven              0 1
##    Tranland                1 0
##    Troyville               1 0
##    Turnerchester           0 1
##    Turnerview              1 0
##    Turnerville             1 0
##    Tylerport               0 1
##    Valerieland             1 0
##    Vanessastad             0 1
##    Vanessaview             0 1
##    Villanuevastad          1 0
##    Villanuevaton           1 0
##    Wademouth               1 0
##    Wadestad                1 0
##    Wagnerchester           1 0
##    Wallacechester          1 0
##    Walshhaven              1 0
##    Waltertown              0 1
##    Watsonfort              1 0
##    Welchshire              0 1
##    Wendyton                1 0
##    Wendyville              0 1
##    West Alice              1 0
```

```
##    West Alyssa           1 0
##    West Amanda           0 2
##    West Andrew           1 0
##    West Angela           1 0
##    West Angelabury       1 0
##    West Annefort         0 1
##    West Aprilport        0 1
##    West Arielstad        1 0
##    West Barbara          1 0
##    West Benjamin         1 0
##    West Brad             0 1
##    West Brandonton       0 1
##    West Brenda           1 0
##    West Carmenfurt       1 0
##    West Casey            0 1
##    West Chloeborough     0 1
##    West Christopher      0 1
##    West Colin            1 0
##    West Connor           0 1
##    West Courtney         1 0
##    West Daleborough      1 0
##    West Dannyberg        1 0
##    West David            0 1
##    West Dennis           1 0
##    West Derekmouth       0 1
##    West Dylanberg        0 1
##    West Eduardotown      0 1
##    West Ericaport        0 1
##    West Ericfurt         0 1
##    West Gabriellamouth   0 1
##    West Gregburgh        1 0
##    West Guybury          1 0
##    West James            0 1
##    West Jane             0 1
##    West Jeremyside       0 1
##    West Jessicahaven     0 1
##    West Jodi             1 0
##    West Joseph           1 0
##    West Julia            0 1
##    West Justin           0 1
##    West Katiefurt        0 1
##    West Kevinfurt        0 1
##    West Lacey            1 0
##    West Leahton          0 1
##    West Lindseybury      0 1
##    West Lisa             1 0
##    West Lucas            1 0
##    West Mariafort        1 0
##    West Melaniefurt      0 1
##    West Melissashire     0 1
##    West Michaelhaven     1 0
##    West Michaelport      1 0
##    West Michaelshire     1 0
##    West Michaelstad      1 0
```

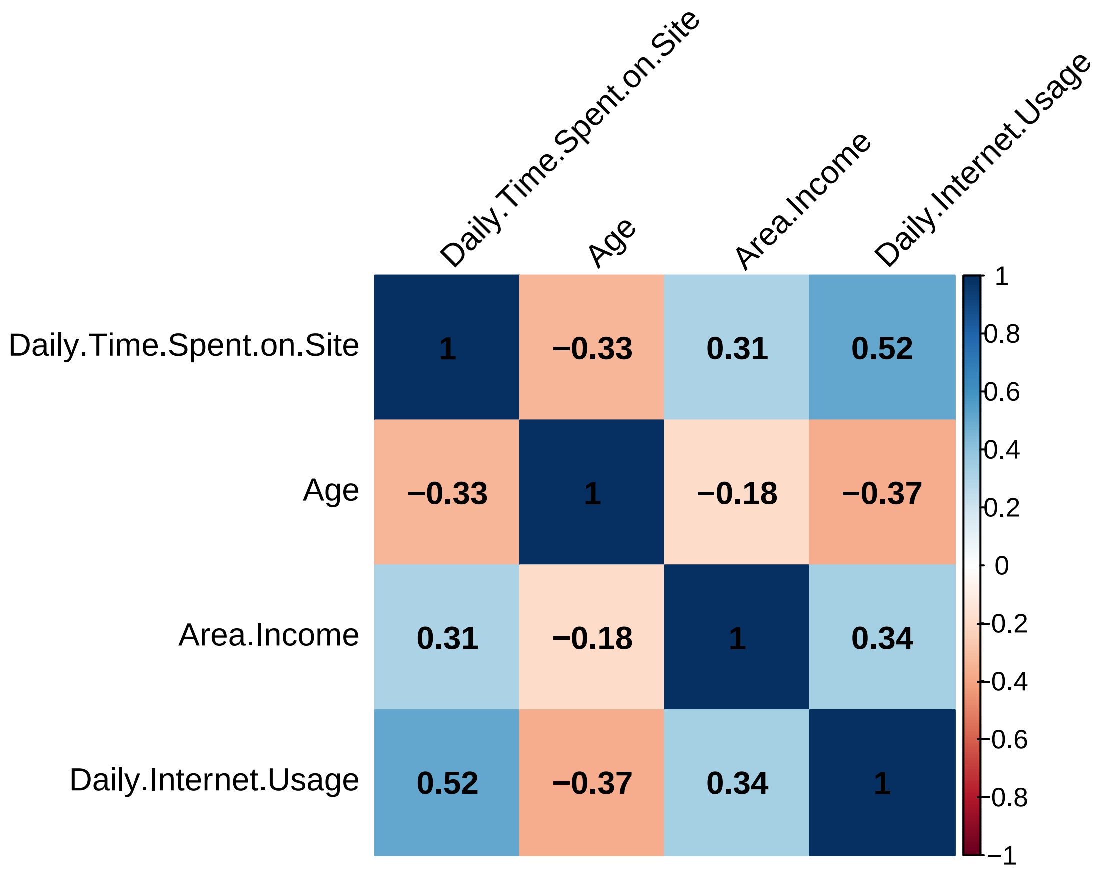
```
##    West Pamela             0 1
##    West Randy              0 1
##    West Raymondmouth       0 1
##    West Rhondamouth        1 0
##    West Ricardo            0 1
##    West Richard            0 1
##    West Robertside         1 0
##    West Roytown            1 0
##    West Russell            1 0
##    West Ryan               0 1
##    West Samantha           1 0
##    West Shannon            0 2
##    West Sharon             1 0
##    West Shaun              1 0
##    West Steven             2 0
##    West Sydney             1 0
##    West Tanner             1 0
##    West Tanya              0 1
##    West Terrifurt          1 0
##    West Thomas             1 0
##    West Tinashire          0 1
##    West Travismouth        0 1
##    West Wendyland          1 0
##    West William            0 1
##    West Zacharyborough     1 0
##    Westshire               0 1
##    Whiteport               0 1
##    Whitneyfort             1 0
##    Wilcoxport              0 1
##    Williammouth            0 1
##    Williamport             1 0
##    Williamsborough         0 1
##    Williamsfort            0 1
##    Williamsmouth           0 1
##    Williamsport            1 2
##    Williamsside            1 0
##    Williamstad             0 1
##    Wilsonburgh             1 0
##    Wintersfort             1 0
##    Wongland                1 0
##    Wrightburgh             2 0
##    Wrightview              0 1
##    Yangside                0 1
##    Youngburgh              1 0
##    Youngfort               0 1
##    Yuton                   0 1
##    Zacharystad             1 0
##    Zacharyton              0 1
```

###Improving the solution: creating a function that returns the highest and lowest values of a specific column so that you do not have to manually go through each individual record.

```
library(corrplot)
```

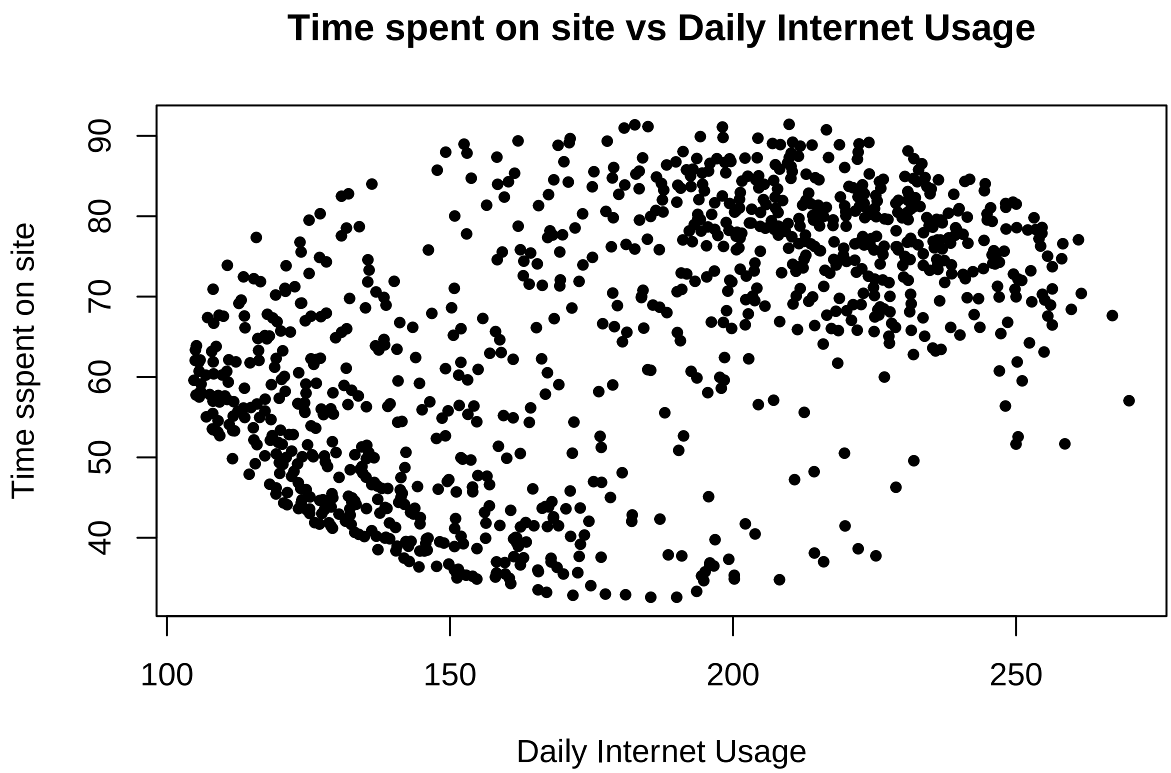
```
## corrplot 0.90 loaded
```

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



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

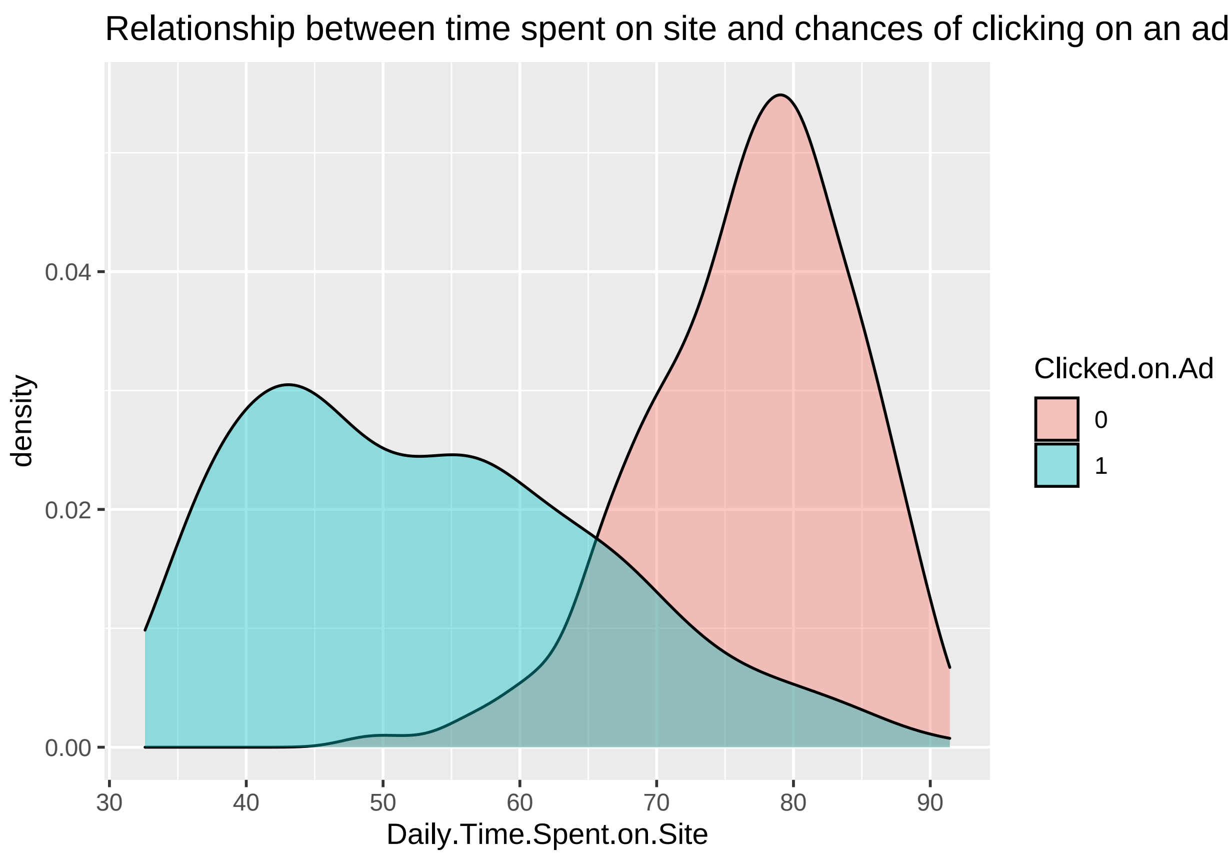
```
x <-df_clean$Daily.Internet.Usage
y <-  df_clean$Daily.Time.Spent.on.Site
# Plot with main and axis titles
# Change point shape (pch = 19) and remove frame.
plot(x, y, main = "Time spent on site vs Daily Internet Usage",
     xlab = "Daily Internet Usage", ylab = "Time sspent on site",
     pch = 20)
```



Scatter Plot

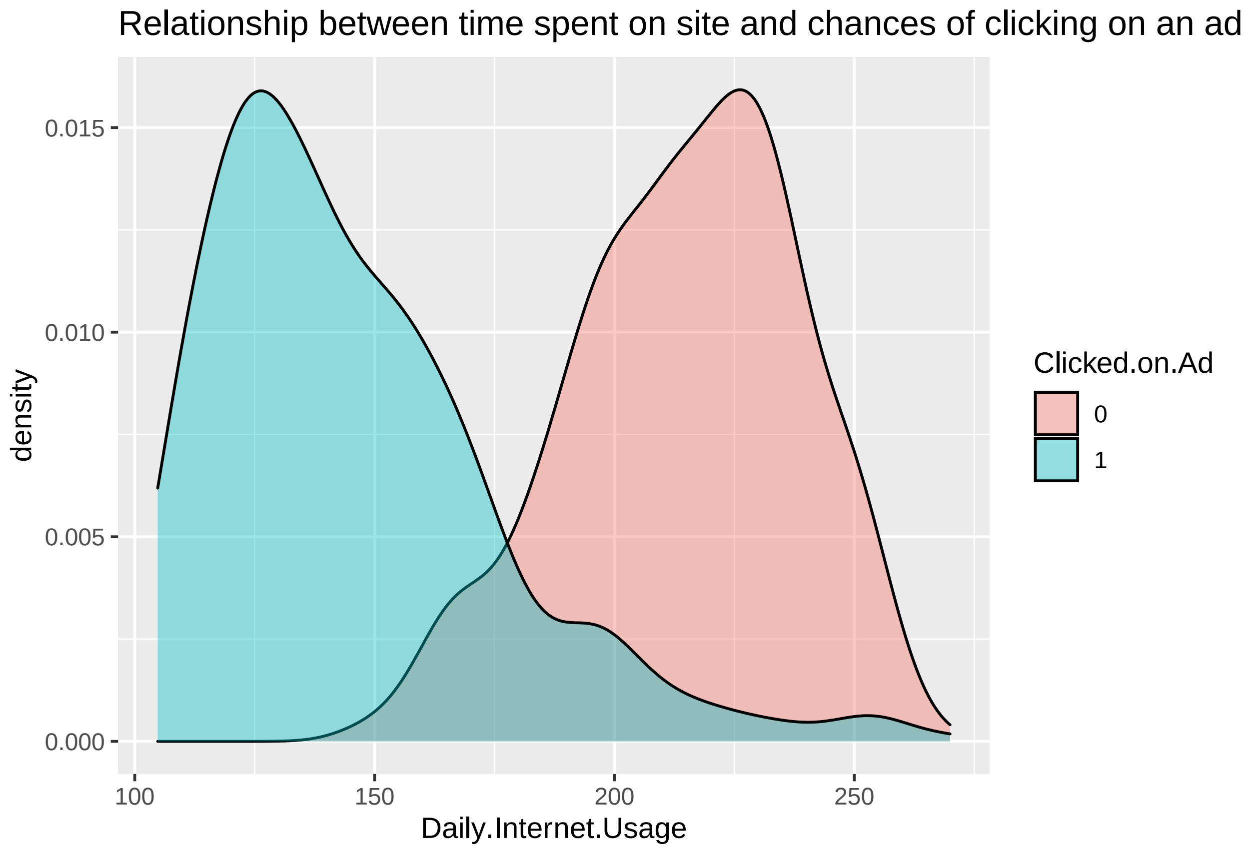
The points are all over but our data points are not highly correlated which explains this. But we can see that people who spend less time on site use less internet. Also, most of the people who use alot of internet per day seem to spend a considerable amount of time on the site.

```
#Time Spent on internet and it's relationship to clicking an ad
ggplot(df_clean,
       aes(x = Daily.Time.Spent.on.Site,
           fill = Clicked.on.Ad)) +
  geom_density(alpha = 0.4) +
  labs(title = "Relationship between time spent on site and chances of clicking on an ad")
```



People who spend less time on the site are likely to click on an ad as compared to those who spend alot of time on the site. .

```
#Internet Usage and it's relationship to clicking an ad
ggplot(df_clean,
       aes(x = Daily.Internet.Usage,
           fill = Clicked.on.Ad)) +
  geom_density(alpha = 0.4) +
  labs(title = "Relationship between time spent on site and chances of clicking on an ad")
```



It seems the longer people spend on the internet, the likelier they are to click on the ads.

Conclusion

i) People who have a daily internet usage of less than 175 are more likely to click on an ad
ii) People who spend less than 70mins on the site are likely to click on ad
iii) People above 40 are more likely to click on an ad
iv) People with an income of less than 60000 are most likely to click on an ad

Challenging the solution

i) It would be great to do some hypothesis testing on the conclusions made from Exploratory Data Analysis, this way we could ascertain the chances of specific person clicking on an ad or not.
ii) Also, it would be necessary to create a predictive model and perform some feature importance selection to choose which variables are most important to use when deciding who will click on an ad or not when using the website.

Feature Engineering

```
print(sapply(df_clean,class))
```

```
## Daily.Time.Spent.on.Site                                  Age              Area.Income
```

```
##                 "numeric"                 "integer"                 "numeric"
##      Daily.Internet.Usage                      City                      Male
##                 "numeric"               "character"                  "factor"
##                   Country             Clicked.on.Ad                      year
##               "character"                  "factor"                  "factor"
##                     month                       day                      hour
##                  "factor"                  "factor"                  "factor"
```

```r
colnames(df_clean)
```

```
##  [1] "Daily.Time.Spent.on.Site" "Age"
##  [3] "Area.Income"              "Daily.Internet.Usage"
##  [5] "City"                     "Male"
##  [7] "Country"                  "Clicked.on.Ad"
##  [9] "year"                     "month"
## [11] "day"                      "hour"
```

```r
df_mod<-df_clean[,c("Daily.Time.Spent.on.Site","Age","Area.Income",     "Daily.Internet.Usage","Male",
colnames(df_mod)
```

```
## [1] "Daily.Time.Spent.on.Site" "Age"
## [3] "Area.Income"              "Daily.Internet.Usage"
## [5] "Male"                     "Clicked.on.Ad"
```

```r
head(df_mod)
```

```
##   Daily.Time.Spent.on.Site Age Area.Income Daily.Internet.Usage Male
## 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
## 6                    59.99  23    59761.56               226.74    1
##   Clicked.on.Ad
## 1             0
## 2             0
## 3             0
## 4             0
## 5             0
## 6             0
```

```r
set.seed(7)

# Randomizing the rows, creates a uniform distribution of 1000
random <- runif(1000)
df_mod <- df_mod[order(random),]

head(df_mod)
```

```
##     Daily.Time.Spent.on.Site Age Area.Income Daily.Internet.Usage Male
## 503                    66.17  26    63580.22               228.70    0
```

```
## 627                     85.77  27    52261.73                191.78    1
## 956                     54.55  44    41547.62                109.04    0
## 630                     73.94  26    55411.06                236.15    1
## 92                      55.79  24    59550.05                149.67    0
## 18                      82.03  41    71511.08                187.53    0
##     Clicked.on.Ad
## 503             0
## 627             0
## 956             1
## 630             0
## 92              1
## 18              0
```

We normalize the data to make the variables comparative

```
# The normalization function is created
normal <-function(x) { (x -min(x))/(max(x)-min(x))   }

# Normalization function is applied to the dataframe
df_normal <- as.data.frame(lapply(df_mod[,c(1,2,3,4)], normal))
```

```
#
require(lattice)
```

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

```
library(caret)
#Create an index for data partitioning
index <- sample(1:nrow(df_normal),0.8 * nrow(df_normal))
```

```
# The training dataset extracted
df_train <- df_normal[index,]

# The test dataset extracted
df_test <- df_normal[-index,]
```

```
 # We also convert ordered factor to normal factor
df_target <- as.factor(df_clean[index,8])

# We compare it with values that will be predicted
# also convert ordered factor to normal factor
test_target <- as.factor(df_clean[-index,8])
```

```
# Running the knn function
library(class)
pr <- knn(df_train,df_test,cl=df_target,k=20)
```

We analyse model perfomance by using confusion matrix and accuracy

```
# Creating the confucion matrix

confusionMatrix(pr,test_target)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0  1
##          0 52 54
##          1 40 54
##
##                Accuracy : 0.53
##                  95% CI : (0.4583, 0.6008)
##     No Information Rate : 0.54
##     P-Value [Acc > NIR] : 0.6392
##
##                   Kappa : 0.0645
##
##  Mcnemar's Test P-Value : 0.1800
##
##             Sensitivity : 0.5652
##             Specificity : 0.5000
##          Pos Pred Value : 0.4906
##          Neg Pred Value : 0.5745
##              Prevalence : 0.4600
##          Detection Rate : 0.2600
##    Detection Prevalence : 0.5300
##       Balanced Accuracy : 0.5326
##
##        'Positive' Class : 0
##
```

```r
# Checking the accuracy
accuracy <- function(x){sum(diag(x)/(sum(rowSums(x)))) * 100}
# Creating the confucion matrix
tb <- table(pr,test_target)

accuracy(tb)
```

```
## [1] 53
```

Decision Tree

Split the data

```r
create_train_test <- function(data, size = 0.8, train = TRUE) {
    n_row = nrow(data)
    total_row = size * n_row
    train_sample <- 1: total_row
    if (train == TRUE) {
        return (data[train_sample, ])
    } else {
        return (data[-train_sample, ])
    }
}
```

```r
data_train <- create_train_test(df_mod, 0.8, train = TRUE)
data_test <- create_train_test(df_mod, 0.8, train = FALSE)
dim(data_train)
```

```
## [1] 800   6
```

```r
dim(data_test)
```

```
## [1] 200   6
```

Checking if the randomization process is correct

```r
prop.table(table(data_train$Clicked.on.Ad))
```

```
##
##       0       1
## 0.50375 0.49625
```

```r
prop.table(table(data_test$Clicked.on.Ad))
```

```
##
##     0     1
## 0.485 0.515
```

```r
library(rpart.plot)
```

```
## Loading required package: rpart
```

```r
data_train <- create_train_test(df_mod, 0.8, train = TRUE)
data_test <- create_train_test(df_mod, 0.8, train = FALSE)
dim(data_train)
```

```
## [1] 800   6
```

```r
dim(data_test)
```

```
## [1] 200   6
```

```r
prop.table(table(data_train$Clicked.on.Ad))
```

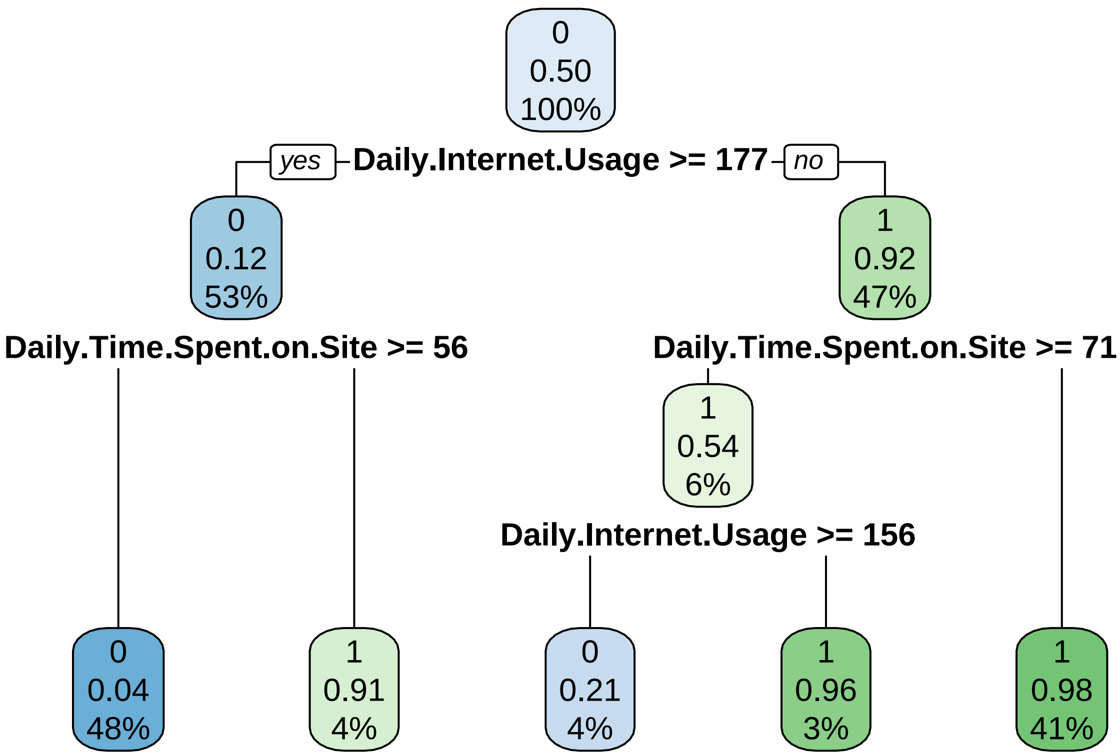
```
##
##       0       1
## 0.50375 0.49625
```

```r
prop.table(table(data_test$Clicked.on.Ad))
```

```
##
##     0     1
## 0.485 0.515
```

We use the class method to predict a class.

```
library(rpart)
library(rpart.plot)
fit <- rpart(Clicked.on.Ad~., data = data_train, method = 'class')
rpart.plot(fit)
```




```
#Factor the Clicked.on.Ad vector in the test dataset
data_test$Clicked.on.Ad <- factor(data_test$Clicked.on.Ad)
```

```
#Using model to predict
TreePredict <- predict(fit, newdata = data_test, type = "class")
x<-confusionMatrix(TreePredict, data_test$Clicked.on.Ad)
x
```

```
## Confusion Matrix and Statistics
##
##             Reference
## Prediction  0  1
##          0 94  9
##          1  3 94
##
##                Accuracy : 0.94
##                  95% CI : (0.8975, 0.9686)
##     No Information Rate : 0.515
##     P-Value [Acc > NIR] : <2e-16
##
```

```
##                   Kappa : 0.8801
##
##  Mcnemar's Test P-Value : 0.1489
##
##             Sensitivity : 0.9691
##             Specificity : 0.9126
##          Pos Pred Value : 0.9126
##          Neg Pred Value : 0.9691
##              Prevalence : 0.4850
##          Detection Rate : 0.4700
##    Detection Prevalence : 0.5150
##       Balanced Accuracy : 0.9408
##
##        'Positive' Class : 0
##
```

Hyperparameter tuning

```
accuracy_tune <- function(fit) {
    predict_unseen <- predict(fit, data_test, type = 'class')
    table_mat <- table(data_test$Clicked.on.Ad, predict_unseen)
    accuracy_Test <- sum(diag(table_mat)) / sum(table_mat)
    accuracy_Test
}
```

We rerun using the tuned parameters

```
control <- rpart.control(minsplit = 4,
    minbucket = round(5 / 3),
    maxdepth = 3,
    cp = 0)
tune_fit <- rpart(Clicked.on.Ad~., data = data_train, method = 'class', control = control)
accuracy_tune(tune_fit)
```

```
## [1] 0.95
```

We have improved the accuracy of the decision tree from from 94 to 95%

SVM

```
#Installing and running the kernlab package
library(kernlab)
```

```
##
## Attaching package: 'kernlab'
```

```
## The following object is masked from 'package:ggplot2':
##
##     alpha
```

```r
#controling all the computational overheads using traincontrol
trctrl <- trainControl(method = "repeatedcv", number = 10, repeats = 3)

#We fit the model using the linear kernel
#Data is also scaled and centered
svm_Linear <- train(Clicked.on.Ad ~., data = data_train, method = "svmLinear",
trControl=trctrl,
preProcess = c("center", "scale"),
tuneLength = 10)

# We then check the result of our train() model
svm_Linear
```

```
## Support Vector Machines with Linear Kernel
##
## 800 samples
##   5 predictor
##   2 classes: '0', '1'
##
## Pre-processing: centered (5), scaled (5)
## Resampling: Cross-Validated (10 fold, repeated 3 times)
## Summary of sample sizes: 720, 720, 720, 720, 720, 721, ...
## Resampling results:
##
##   Accuracy   Kappa
##   0.9708268  0.9416364
##
## Tuning parameter 'C' was held constant at a value of 1
```

```r
#We then predict
test_pred <- predict(svm_Linear, newdata = data_test)
test_pred
```

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

```r
#Print the confusion matrix and statistics
confusionMatrix(table(test_pred, data_test$Clicked.on.Ad))
```

```
## Confusion Matrix and Statistics
##
##
## test_pred  0  1
##         0 95  5
##         1  2 98
##
##                Accuracy : 0.965
```

```
##                  95% CI : (0.9292, 0.9858)
##     No Information Rate : 0.515
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.93
##
##  Mcnemar's Test P-Value : 0.4497
##
##             Sensitivity : 0.9794
##             Specificity : 0.9515
##          Pos Pred Value : 0.9500
##          Neg Pred Value : 0.9800
##              Prevalence : 0.4850
##          Detection Rate : 0.4750
##    Detection Prevalence : 0.5000
##       Balanced Accuracy : 0.9654
##
##        'Positive' Class : 0
##
```

As compared to KNN and Decision Trees, the SVM linear kernel model performs the best. There are fewer misclassifications as shown in the confusion matrix and the model has an accuracy score of 96.5% which is slightly better than the rest.

Conclusion

In conclusion, we advice the owner of the blog to use an SVM model with a linear kernel to predict whether users of the blog will click on an ad or not.