# Who is likely to click the ad

## Defining the Research Question

Can one identify which individuals are most likely to click on a site.

### Metric for Success

Identifying factors that are likely to influence whether a person would click on the ads on the site

### 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 to help her identify which individuals are most likely to click on her ads.

### Experimentaal Design

1. Load the data
2. Data Cleaning
3. Univariate Analysis
4. Bivariate Analysis
5. Recommendation

### Appropriateness of the Data

The data available was not sufficient. More features should be added

Loading the Data and the Libraries

```
library(data.table)
```

```
## Warning: package 'data.table' was built under R version 4.0.4
```

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

```
## Warning: package 'dplyr' was built under R version 4.0.4
```

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

```
## The following objects are masked from 'package:data.table':
##
##     between, first, last

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

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

```
library("ggplot2")
```

```
## Warning: package 'ggplot2' was built under R version 4.0.4
```

```
library(ggstatsplot)
```

```
## Warning: package 'ggstatsplot' was built under R version 4.0.4

## In case you would like cite this package, cite it as:
##      Patil, I. (2018). ggstatsplot: "ggplot2" Based Plots with Statistical Details. CRAN.
##      Retrieved from https://cran.r-project.org/web/packages/ggstatsplot/index.html
```

```
library(vtree)
```

```
## Warning: package 'vtree' was built under R version 4.0.4
```

```
library(Hmisc)
```

```
## Warning: package 'Hmisc' was built under R version 4.0.4

## Loading required package: lattice

## Loading required package: survival

## Loading required package: Formula

##
## Attaching package: 'Hmisc'

## The following objects are masked from 'package:dplyr':
##
##     src, summarize

## The following objects are masked from 'package:base':
##
##     format.pval, units
```

```r
library(moments)
library(modeest)
```

```
## Warning: package 'modeest' was built under R version 4.0.4
```

```
## Registered S3 method overwritten by 'statip':
##   method         from
##   predict.kmeans parameters
```

```
##
## Attaching package: 'modeest'
```

```
## The following object is masked from 'package:moments':
##
##     skewness
```

### 1. Loading the Data

```r
df <- fread('C:\\Users\\Lenovo\\Downloads\\advertising.csv')
```

```r
# Previewing the data: Top
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
```

```r
# Previwing the Bottom records
tail(df)
```

```
##    Daily Time Spent on Site Age Area Income Daily Internet Usage
## 1:                    43.70  28    63126.96               173.01
## 2:                    72.97  30    71384.57               208.58
## 3:                    51.30  45    67782.17               134.42
## 4:                    51.63  51    42415.72               120.37
## 5:                    55.55  19    41920.79               187.95
## 6:                    45.01  26    29875.80               178.35
##                          Ad Topic Line          City Male
## 1:        Front-line bifurcated ability  Nicholasland    0
## 2:        Fundamental modular algorithm     Duffystad    1
## 3:      Grass-roots cohesive monitoring   New Darlene    1
## 4:         Expanded intangible solution South Jessica    1
## 5: Proactive bandwidth-monitored policy   West Steven    0
## 6:      Virtual 5thgeneration emulation   Ronniemouth    0
##                   Country           Timestamp Clicked on Ad
## 1:                Mayotte 2016-04-04 03:57:48             1
## 2:                Lebanon 2016-02-11 21:49:00             1
## 3: Bosnia and Herzegovina 2016-04-22 02:07:01             1
## 4:               Mongolia 2016-02-01 17:24:57             1
## 5:              Guatemala 2016-03-24 02:35:54             0
## 6:                 Brazil 2016-06-03 21:43:21             1
```

### 2. Cleaning the Data

**Standardizing column name** Changing columns to lower and replacing the whitespaces with underscore

```
# Replacing the whitespaces with underscore
names(df) <- gsub(" ","_", names(df))
```

```
# Viewing the column names
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"
```

```
# Changing the column names to lower case
names(df) <- tolower(names(df))
```

```
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"
```

**Duplicate Data**

```r
# Checking for the presence of duplicate values
anyDuplicated(df)
```

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

There are NO duplicated records in this data set

### Missing Values

```r
# Checking for the presence of 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 this data set

### Exploring Outliers in the numerical columns

```r
# Checking for the data types
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 standardi
##  $ city                    : chr  "Wrightburgh" "West Jodi" "Davidton" "West Terrifurt" ...
##  $ male                    : int  0 1 0 1 0 1 0 1 1 1 ...
##  $ country                 : chr  "Tunisia" "Nauru" "San Marino" "Italy" ...
##  $ timestamp               : 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>
```

```r
# Encoding 0 to NO and  1 to YES
df$clicked_on_ad[df$clicked_on_ad == 1] <- "YES"
df$clicked_on_ad[df$clicked_on_ad == 0] <- "NO"
unique(df$clicked_on_ad)
```

**Encoding clicked_on_ad column**

```
## [1] "NO"  "YES"
```

```
# Creating a date column
df$date <- format(as.POSIXct.Date(df$timestamp,format="%Y:%m:%d %H:%M:%S"),"%Y-%m-%d")
df$date <- as.Date(df$date, format = "%Y-%m-%d")
```

```
# Creating a hour column
df$hour <- format(as.POSIXct(df$timestamp, format="%Y:%m:%d %H:%M:%S"),"%H")
df$hour <- as.integer(df$hour)
```

**Feature Extraction**

```
df = subset(df, select = -c(timestamp) )
```

```
str(df)
```

**Dropping the timestamp column after extracting the date**

```
## Classes 'data.table' and 'data.frame':   1000 obs. of  11 variables:
##  $ daily_time_spent_on_site: num  69 80.2 69.5 74.2 68.4 ...
##  $ age                     : int  35 31 26 29 35 23 33 48 30 20 ...
##  $ area_income             : num  61834 68442 59786 54806 73890 ...
##  $ daily_internet_usage    : num  256 194 236 246 226 ...
##  $ ad_topic_line           : chr  "Cloned 5thgeneration orchestration" "Monitored national standardi
##  $ city                    : chr  "Wrightburgh" "West Jodi" "Davidton" "West Terrifurt" ...
##  $ male                    : int  0 1 0 1 0 1 0 1 1 1 ...
##  $ country                 : chr  "Tunisia" "Nauru" "San Marino" "Italy" ...
##  $ clicked_on_ad           : chr  "NO" "NO" "NO" "NO" ...
##  $ date                    : Date, format: NA NA ...
##  $ hour                    : int  0 1 20 2 3 14 20 1 9 1 ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

```
# Encoding 0 with NO and 1 with YES
df$male[df$male == 1] <- "YES"
df$male[df$male == 0] <- "NO"
unique(df$male)
```

**Encoding the male column**

```
## [1] "NO"  "YES"
```

## OUTLIERS

```
# Checking for potential outliers in the daily_time_spent_on_site column
boxplot.stats(df$daily_time_spent_on_site)$out
```

### daily_time_spent_on_site

```
## numeric(0)
```

There is no outlier in the column daily_spent_on_site

```
# Checking for potential outliers in the age column
boxplot.stats(df$age)$out
```

### age

```
## integer(0)
```

There is no outlier in the column age

```
boxplot.stats(df$area_income)$out
```

### area_income

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

This column has outliers

```
# Exploring the extent of the outliers
x <- boxplot.stats(df$area_income)$out
print(paste("Percentage of outliers: ", (length(x)/length(df$area_income)*100), "%"))
```

```
## [1] "Percentage of outliers:  0.8 %"
```

Less than 1% of data in the area_income is considered outliers based on the IQR criterion. Dropping or imputing the outliers requires a further investigation into the data points. This shall be delved into during univariate analysis of the column.

```
x <- boxplot.stats(df$daily_internet_usage)$out
print(paste("Percentage of outliers: ", (length(x)/length(df$daily_internet_usage)*100), "%"))
```

### daily_internet_usage

```
## [1] "Percentage of outliers:  0 %"
```

This dataset has NO outliers

## 3. Univariate Analysis

Creating Summaries of the Data

```
describe(df)
```

```
## df
##
##  11  Variables      1000  Observations
## ---------------------------------------------------------------------------
## daily_time_spent_on_site
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     1000        0      900        1       65    18.11    37.58    41.34
##      .25      .50      .75      .90      .95
##    51.36    68.22    78.55    83.89    86.20
##
## lowest : 32.60 32.84 32.91 32.99 33.21, highest: 90.97 91.10 91.15 91.37 91.43
## ---------------------------------------------------------------------------
## age
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     1000        0       43    0.999    36.01    9.943    23.95    26.00
##      .25      .50      .75      .90      .95
##    29.00    35.00    42.00    49.00    52.00
##
## lowest : 19 20 21 22 23, highest: 57 58 59 60 61
## ---------------------------------------------------------------------------
## area_income
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     1000        0     1000        1    55000    15037    28275    35223
##      .25      .50      .75      .90      .95
##    47032    57012    65471    70506    73601
##
## lowest : 13996.50 14548.06 14775.50 15598.29 15879.10
## highest: 78092.95 78119.50 78520.99 79332.33 79484.80
## ---------------------------------------------------------------------------
## daily_internet_usage
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     1000        0      966        1      180    50.63    113.5    120.5
##      .25      .50      .75      .90      .95
##    138.8    183.1    218.8    236.2    246.7
##
## lowest : 104.78 105.00 105.04 105.15 105.22, highest: 259.76 261.02 261.52 267.01 269.96
## ---------------------------------------------------------------------------
## ad_topic_line
##        n  missing distinct
##     1000        0     1000
##
## lowest : Adaptive 24hour Graphic Interface       Adaptive asynchronous attitude          Adaptive co
## highest: Visionary client-driven installation    Visionary maximized process improvement Visionary m
## ---------------------------------------------------------------------------
## city
##        n  missing distinct
##     1000        0      969
##
```

```
## lowest : Adamsbury     Adamside      Adamsstad     Alanview      Alexanderfurt
## highest: Youngburgh    Youngfort     Yuton         Zacharystad   Zacharyton
## ---------------------------------------------------------------------------
## male
##        n  missing distinct
##     1000        0        2
##
## Value         NO   YES
## Frequency    519   481
## Proportion 0.519 0.481
## ---------------------------------------------------------------------------
## country
##        n  missing distinct
##     1000        0      237
##
## lowest : Afghanistan       Albania           Algeria           American Samoa    Andorra
## highest: Wallis and Futuna Western Sahara    Yemen             Zambia            Zimbabwe
## ---------------------------------------------------------------------------
## clicked_on_ad
##        n  missing distinct
##     1000        0        2
##
## Value       NO YES
## Frequency  500 500
## Proportion 0.5 0.5
## ---------------------------------------------------------------------------
## hour
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##     1000        0       24    0.998    11.66    8.033        1        2
##      .25      .50      .75      .90      .95
##        6       12       18       21       22
##
## lowest :  0  1  2  3  4, highest: 19 20 21 22 23
## ---------------------------------------------------------------------------
##
## Variables with all observations missing:
##
## [1] date
```

The above summary gives a shortcut to a deneral descritive stats of the various data columns. However, as will be demonstrated below we can do a step by step exploration of each data column

**COLUMN: daily_time_spent_on_site  Min, Mean, Qs and Max**

```
summary(df$daily_time_spent_on_site)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   32.60   51.36   68.22   65.00   78.55   91.43
```

**Mode, Skweness and Kurtosis**

```
print(paste("Mode: ", mfv(df$daily_time_spent_on_site)))
```

```
## [1] "Mode:  62.26" "Mode:  75.55" "Mode:  77.05" "Mode:  78.76" "Mode:  84.53"
```

```
print(paste("Skewness: ",skewness(df$daily_time_spent_on_site)))
```

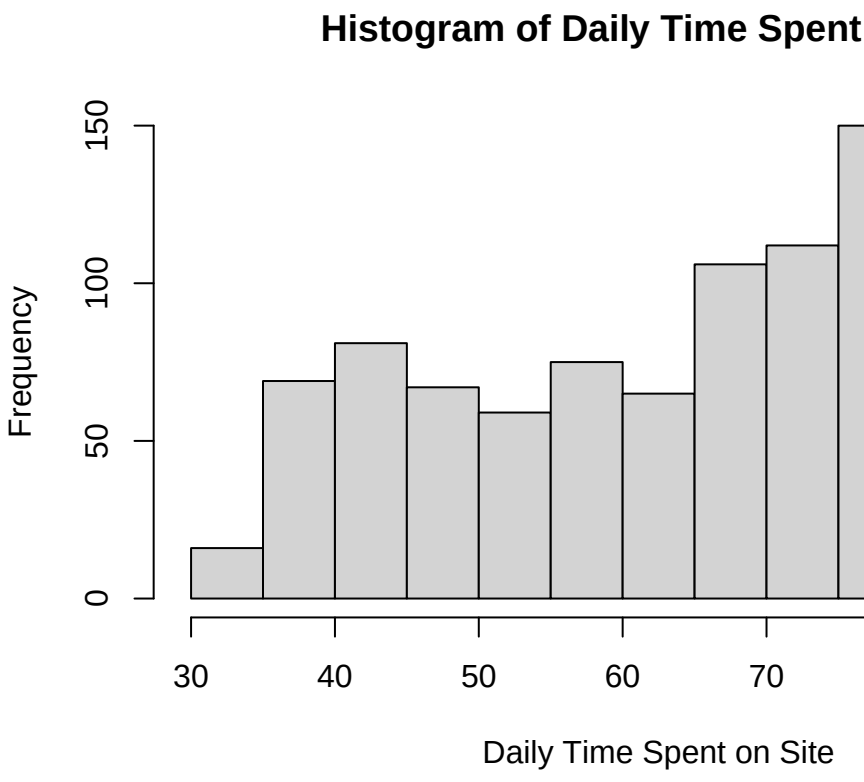
```
## [1] "Skewness:  -0.370645950169329"
```

```
print(paste("Kurtosis: ", kurtosis(df$daily_time_spent_on_site)))
```

```
## [1] "Kurtosis:  1.90394215401081"
```

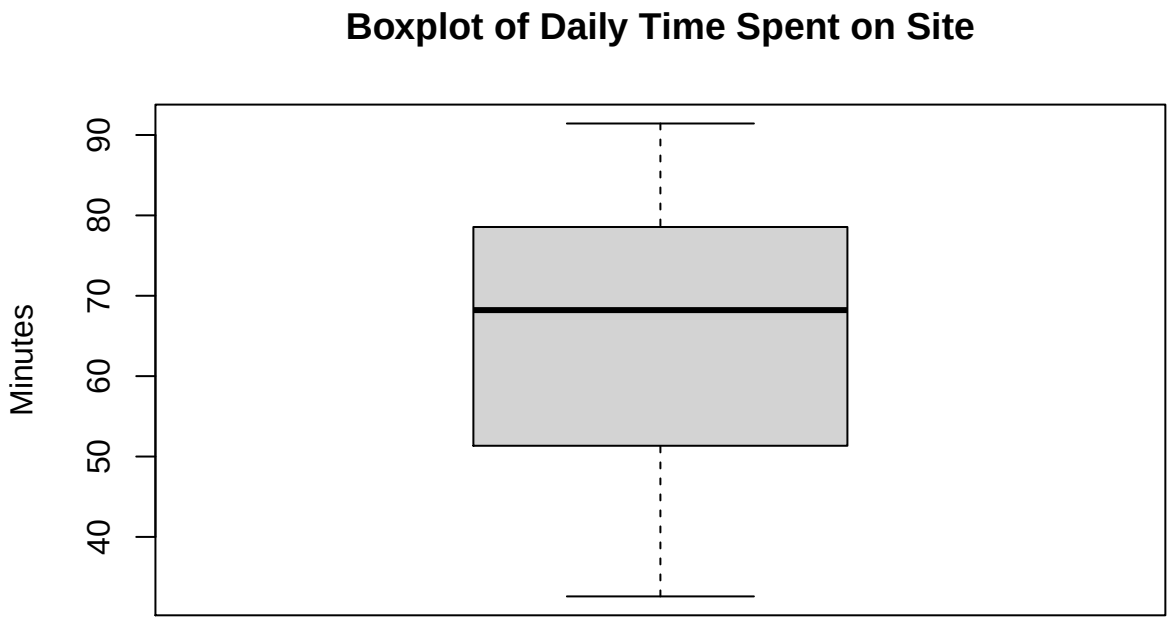
*Obervations* 1. Multi-modal 2. Negatively skewed 3. Leptokurtic

```
hist(df$daily_time_spent_on_site,
  xlab = "Daily Time Spent on Site",
  main = "Histogram of Daily Time Spent on Site",
  breaks = 15)
```




**Histogram of daily_time_spent_on_site**
##### Boxplot of daily_time_spent_on_site

```
boxplot(df$daily_time_spent_on_site, ylab = "Minutes", main = "Boxplot of Daily Time Spent on Site")
```




#### COLUMN: AGE **Min, Mean, Qs and Max**

```
summary(df$age)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   19.00   29.00   35.00   36.01   42.00   61.00
```

**Mode, Skewness and Kurosis**

```
print(paste("Mode: ", mfv(df$age)))
```

```
## [1] "Mode:  31"
```

```
print(paste("Skewness: ",skewness(df$age)))
```

```
## Warning: encountered a tie, and the difference between minimal and
##                     maximal value is > length('x') * 'tie.limit'
## the distribution could be multimodal
```

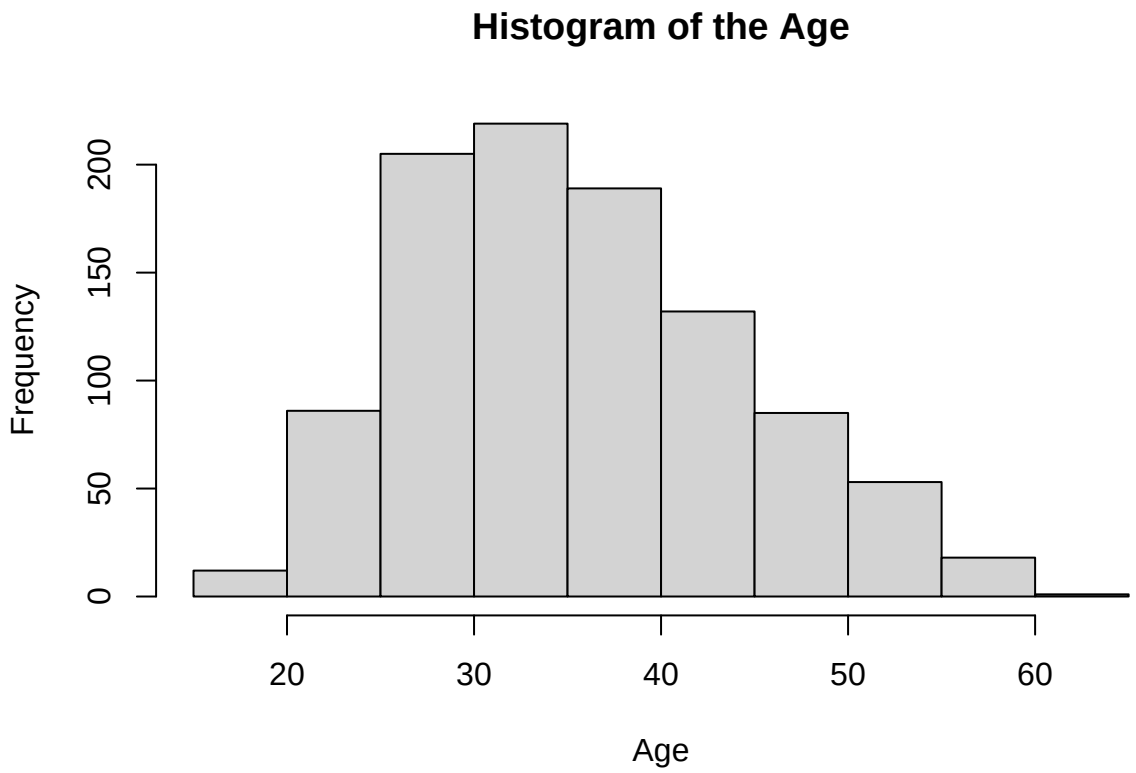
```
## [1] "Skewness:  0.477705221630714"
```

```r
print(paste("Kurtosis: ", kurtosis(df$age)))
```

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

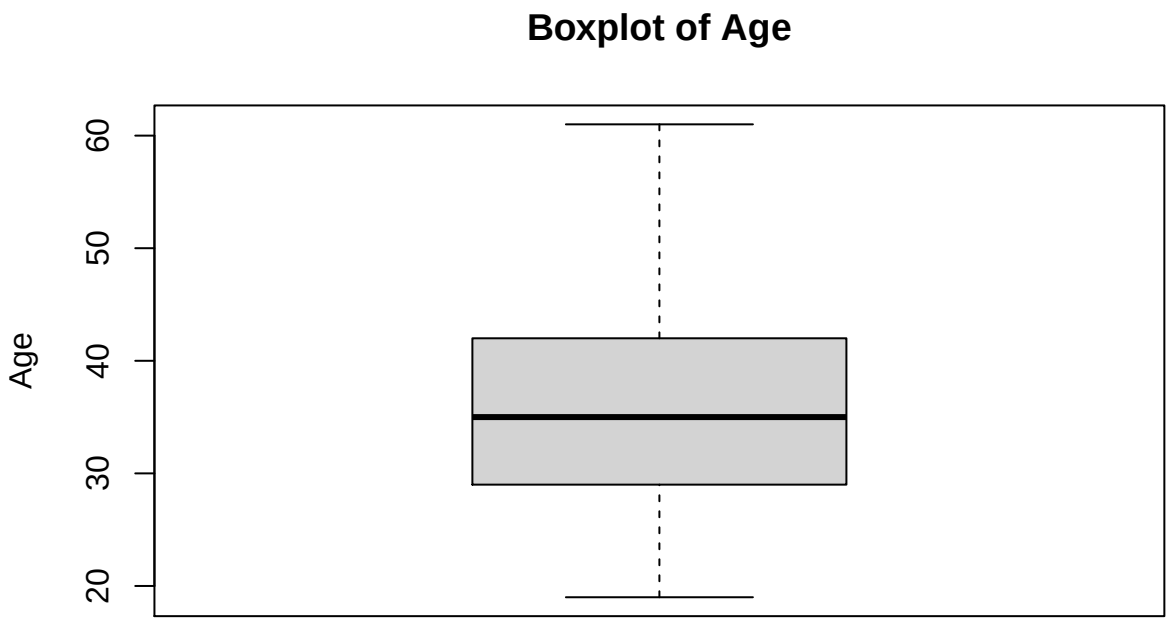
*Observation* 1. the data could be multimodal 2. It is positively skewed 3. It is leptokurtic

```r
hist(df$age,
  xlab = "Age",
  main = "Histogram of the Age",
  breaks = 9,)
```




**Histogram of age**
##### Boxplot of age

```r
boxplot(df$age, ylab = "Age", main = "Boxplot of Age")
```




**COLUMN: area_income  Min, Mean, Qs and Max**

```
summary(df$area_income)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   13996   47032   57012   55000   65471   79485
```

**Mode, Skewness and Kurosis**

```
# print(paste("Mode: ", mfv(df$area_income))) # All the values are unique
print(paste("Skewness: ", skewness(df$area_income)))
```

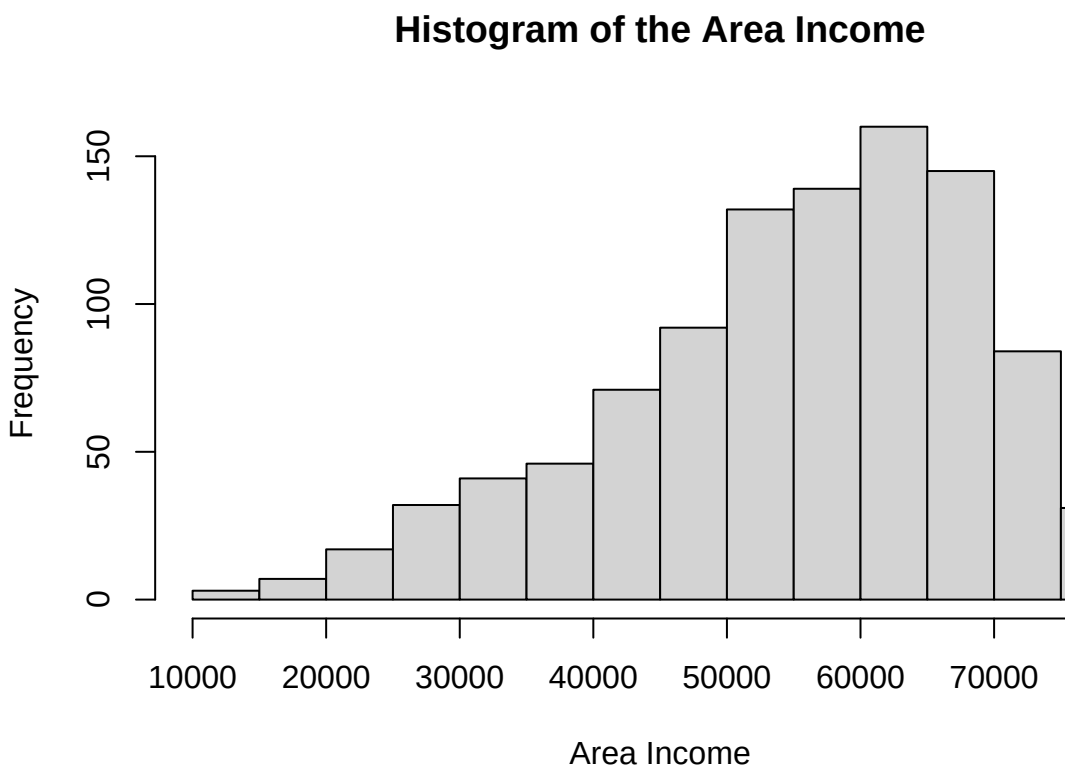
```
## [1] "Skewness:  -0.648422850205901"
```

```
print(paste("Kurtosis: ", kurtosis(df$area_income)))
```

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

*Observations* 1. The data is skewed negatively' 2. The data is leptokurtic

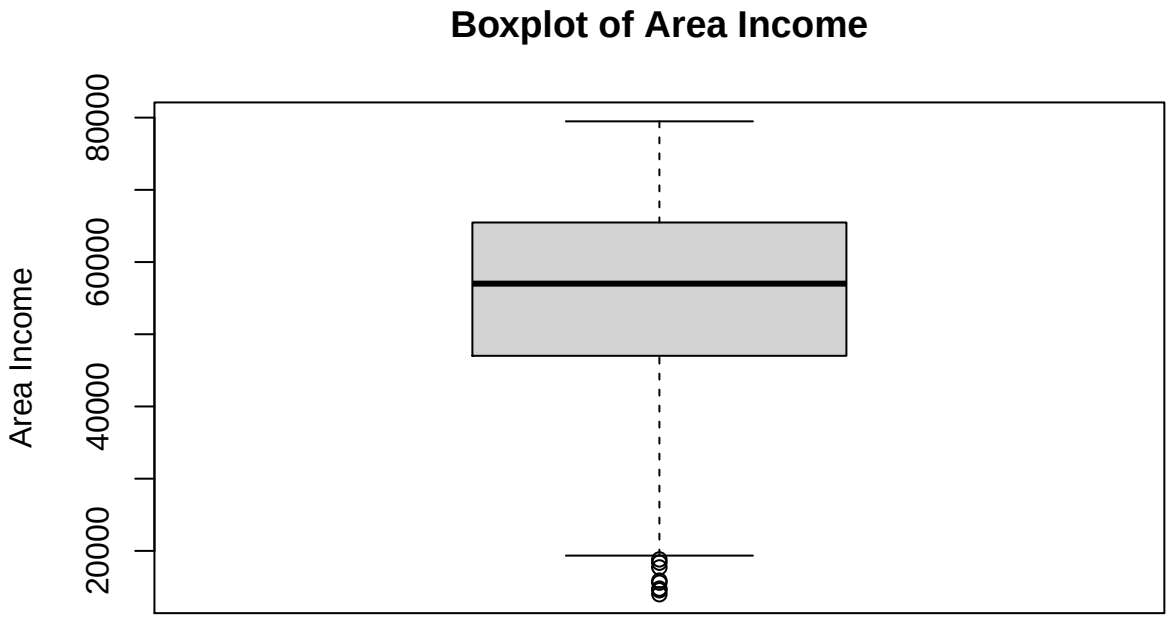
```r
hist(df$area_income,
  xlab = "Area Income",
  main = "Histogram of the Area Income",
  breaks = 10)
```




**Histogram of Area Income**

##### Boxplot of Area Income

```r
boxplot(df$area_income, ylab = "Area Income", main = "Boxplot of Area Income")
```




There are some areas with extremely low incomes that have been classified as outliers

**Exploring the OUTLIERS in the area_income column.** We are going to extraxt the specific rows that contain the outliers for further investigation

```
outlier_income <- boxplot.stats(df$area_income)$out
outlier_income_ind <- which(df$area_income %in% c(outlier_income))
outlier_income_ind
```

```
## [1] 136 511 641 666 693 769 779 953
```

Using the above positions we shall then go ahead to extract the entire row entries

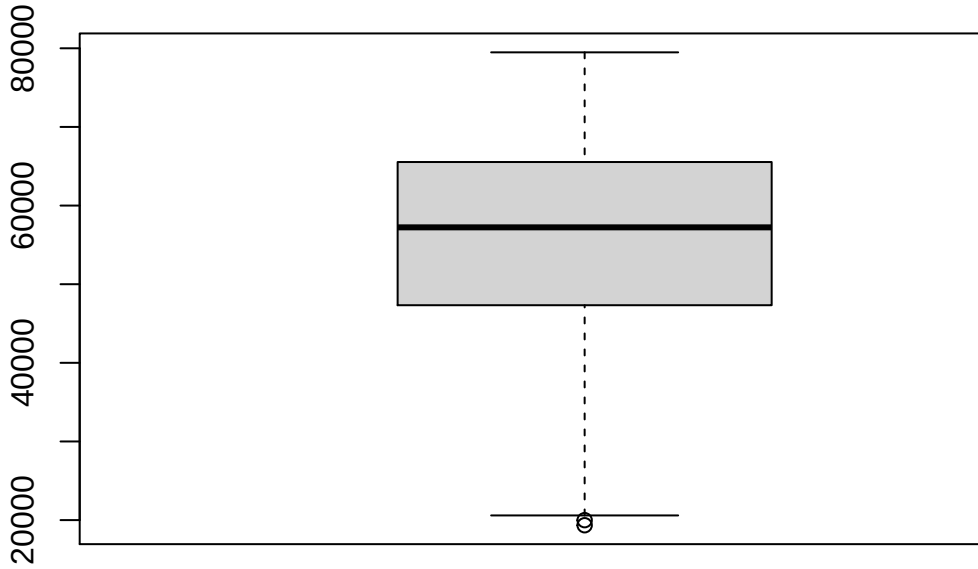
```
df[outlier_income_ind, ]
```

```
##    daily_time_spent_on_site age area_income daily_internet_usage
## 1:                    49.89  39    17709.98               160.03
## 2:                    57.86  30    18819.34               166.86
## 3:                    64.63  45    15598.29               158.80
## 4:                    58.05  32    15879.10               195.54
## 5:                    66.26  47    14548.06               179.04
## 6:                    68.58  41    13996.50               171.54
## 7:                    52.67  44    14775.50               191.26
## 8:                    62.79  36    18368.57               231.87
##                                ad_topic_line            city male
```

```
## 1:           Enhanced system-worthy application     East Michele  YES
## 2:                   Horizontal modular success        Estesfurt   NO
## 3: Triple-buffered high-level Internet solution     Isaacborough  YES
## 4:              Total asynchronous architecture      Sanderstown  YES
## 5:               Optional full-range projection      Matthewtown  YES
## 6:                  Exclusive discrete firmware New Williamville  YES
## 7:     Persevering 5thgeneration knowledge user    New Hollyberg   NO
## 8:                       Total coherent archive        New James  YES
##        country clicked_on_ad date hour
## 1:      Belize           YES <NA>   12
## 2:     Algeria           YES <NA>   17
## 3:  Azerbaijan           YES <NA>    3
## 4:  Tajikistan           YES <NA>   10
## 5:     Lebanon           YES <NA>   19
## 6: El Salvador           YES <NA>   12
## 7:      Jersey           YES <NA>    6
## 8:  Luxembourg           YES <NA>   20
```

Analysis of these rows whose area_income values appear as outliers indicate that all of these had the ad_topic_line clicked on. Notably, they are for Belize, Algeria, Azerbaijan, Tajikistan, Labanon, El Salcador, Jersey and Luxembourg. It would be well to consider dropping these values.

```r
# This has been done using the dplyr library
df <- df %>% slice(-c(outlier_income_ind))
boxplot(df$area_income)
```




**Dropping the outlier values**
After dropping the earlier 8 outliers our new boxplot introduces two new outliers that were not in the ealier list of outliers

**COLUMN: daily_internet_usage Min, Mean, Qs and Max**

```
summary(df$daily_internet_usage)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   104.8   138.6   183.4   180.0   218.8   270.0
```

**Mode, Skewness and Kurosis**

```
print(paste("Mode: ", mfv(df$daily_internet_usage)))
```

```
##  [1] "Mode:  113.53" "Mode:  115.91" "Mode:  117.3"  "Mode:  119.3"
##  [5] "Mode:  120.06" "Mode:  125.45" "Mode:  132.38" "Mode:  135.24"
##  [9] "Mode:  136.18" "Mode:  138.35" "Mode:  158.22" "Mode:  161.16"
## [13] "Mode:  162.44" "Mode:  164.25" "Mode:  167.22" "Mode:  169.4"
## [17] "Mode:  178.75" "Mode:  182.65" "Mode:  190.95" "Mode:  194.23"
## [21] "Mode:  201.15" "Mode:  211.87" "Mode:  214.42" "Mode:  215.18"
## [25] "Mode:  219.72" "Mode:  222.11" "Mode:  223.16" "Mode:  228.81"
## [29] "Mode:  230.36" "Mode:  234.75" "Mode:  235.28" "Mode:  236.96"
## [33] "Mode:  247.05" "Mode:  256.4"
```

```
print(paste("Skewness: ", skewness(df$daily_internet_usage)))
```

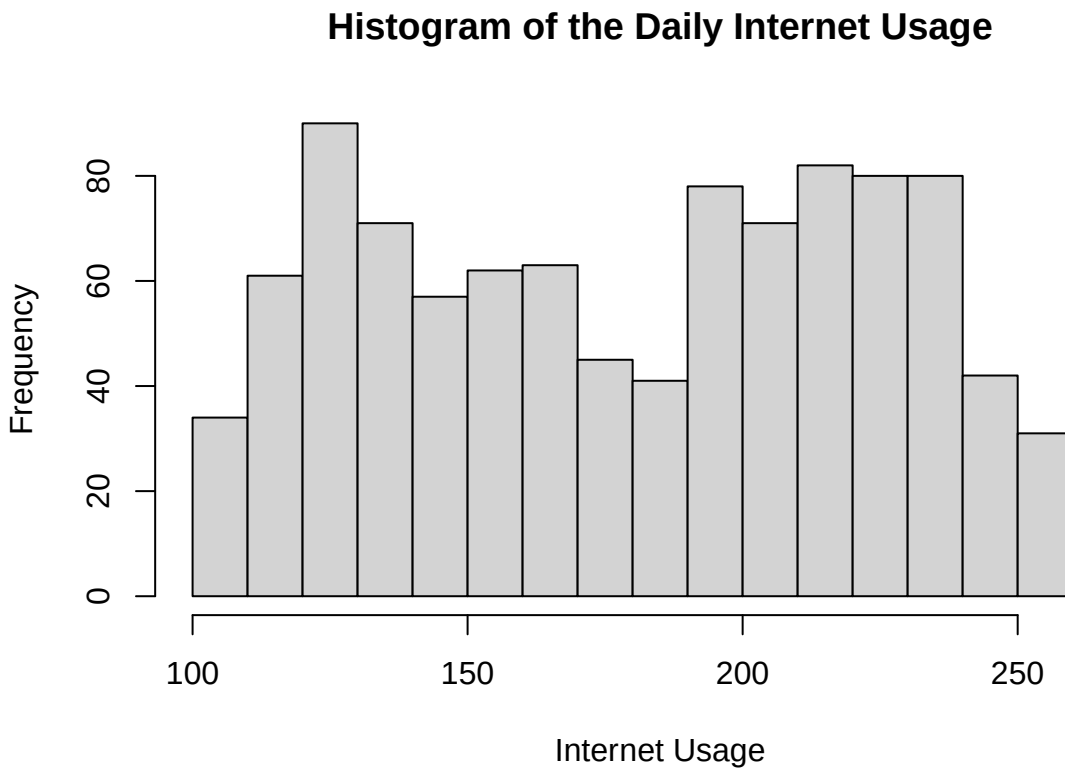
```
## [1] "Skewness:  -0.0338539844521485"
```

```
print(paste("Kurtosis: ", kurtosis(df$daily_internet_usage)))
```

```
## [1] "Kurtosis:  1.71917692942785"
```

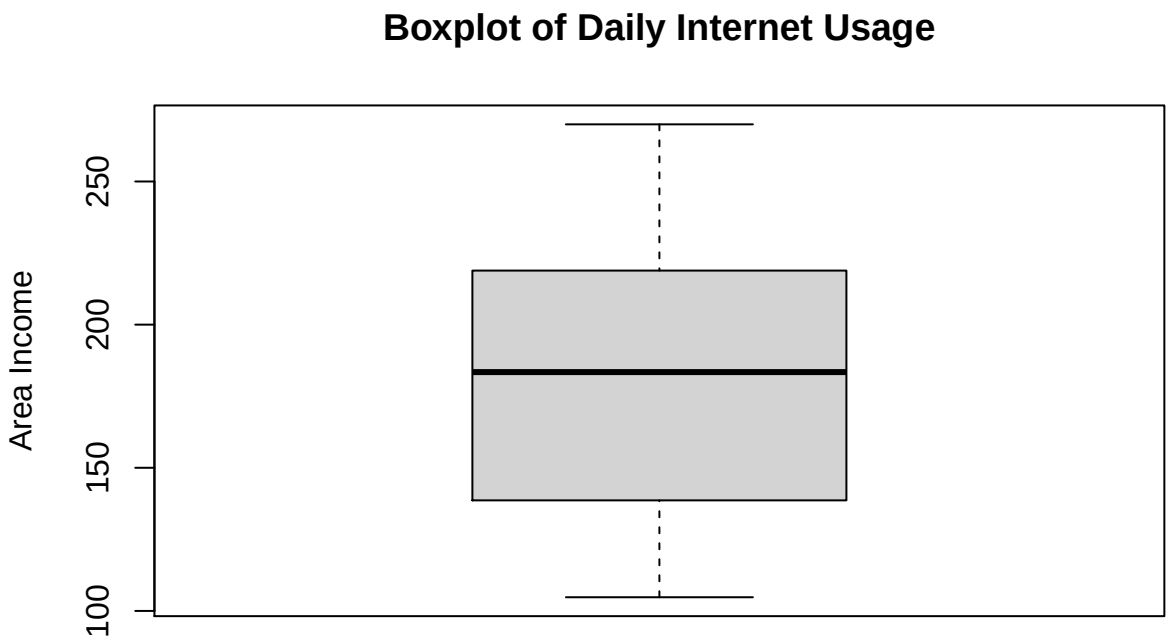
*Observations* 1. This column is multi-modal 2. The data is negatively skewed 3. The data is leptokurtic

```
hist(df$daily_internet_usage,
  xlab = "Internet Usage",
  main = "Histogram of the Daily Internet Usage",
  breaks = 20)
```




**Histogram of Area Income**
##### Boxplot of Daily Internet Usage

```
boxplot(df$daily_internet_usage, ylab = "Area Income", main = "Boxplot of Daily Internet Usage")
```




#### COLUMN: hour **Min, Mean, Qs and Max**

```
summary(df$hour)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    6.00   12.00   11.65   18.00   23.00
```

**Mode, Skweness and Kurtosis**

```
print(paste("Mode: ", mfv(df$hour)))
```

```
## [1] "Mode:  7"
```

```
print(paste("Skewness: ",skewness(df$hour)))
```

```
## Warning: encountered a tie, and the difference between minimal and
##                     maximal value is > length('x') * 'tie.limit'
## the distribution could be multimodal
```

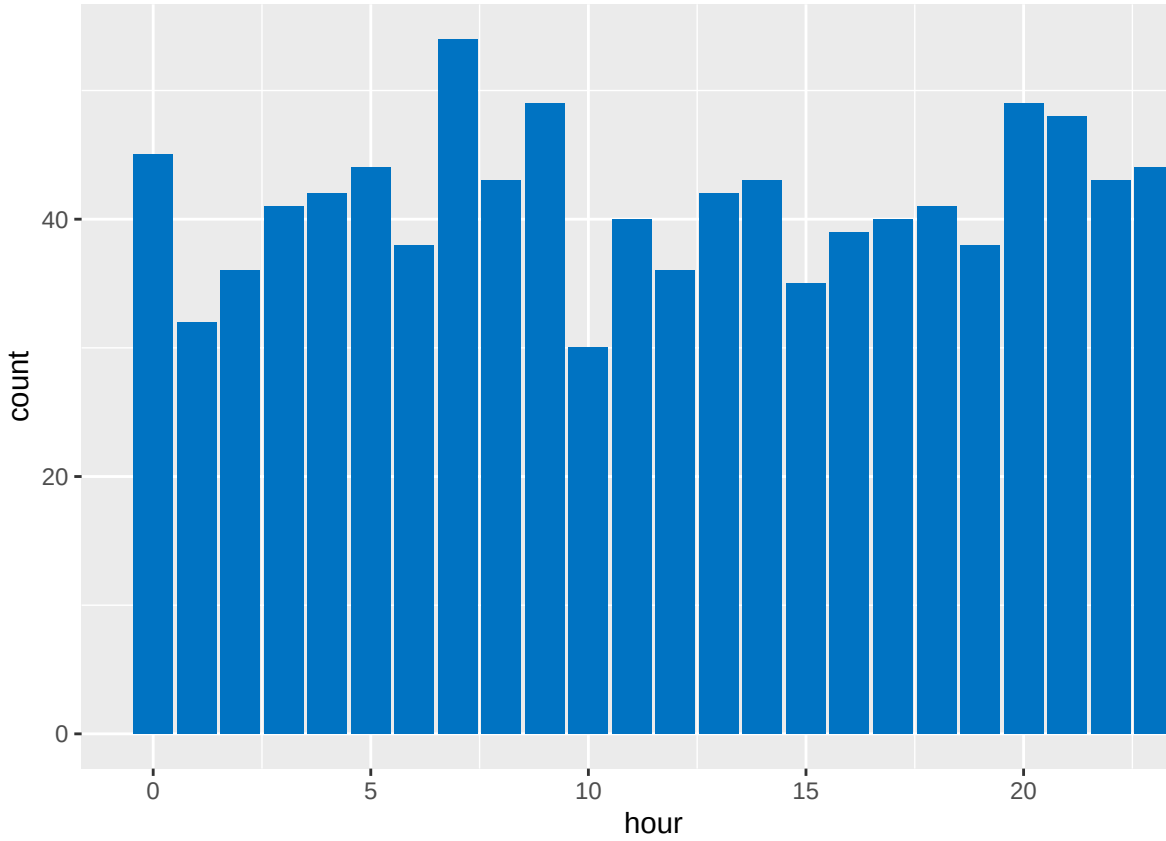
```
## [1] "Skewness:  0.000571936563570611"
```

```
print(paste("Kurtosis: ", kurtosis(df$hour)))
```

```
## [1] "Kurtosis:  1.77408276237894"
```

*Obervations* 1. Multi-modal 2. Negatively skewed 3. Leptokurtic

```
ggplot(df, aes(hour)) +
  geom_bar(fill = "#0073C2FF")
```




**Histogram of hour**
The most popular hour of the day visitting the site is the 7 hour

**COLUMN: ad_topic_line Frequency Table**

```
# This has been done using the dplyr library
topic_line_summary <- df %>%
  count(ad_topic_line, sort = TRUE)

topic_line_summary[1:10]
```

```
##                              ad_topic_line n
##  1:       Adaptive 24hour Graphic Interface 1
##  2:          Adaptive asynchronous attitude 1
##  3:  Adaptive context-sensitive application 1
##  4: Adaptive contextually-based methodology 1
##  5:    Adaptive demand-driven knowledgebase 1
##  6:             Adaptive uniform capability 1
##  7:              Advanced 24/7 productivity 1
##  8:       Advanced 5thgeneration capability 1
##  9:        Advanced didactic conglomeration 1
## 10: Advanced disintermediate data-warehouse 1
```

The ad topic lines are unique.

### COLUMN: city  Frequency Table

```
# This has been done using the dplyr library
city_line_summary <- df %>%
  count(city, sort = TRUE)

city_line_summary[1:10]
```

```
##                city n
##  1:       Lisamouth 3
##  2:    Williamsport 3
##  3: Benjaminchester 2
##  4:       East John 2
##  5:    East Timothy 2
##  6:        Johnstad 2
##  7:        Joneston 2
##  8:      Lake David 2
##  9:      Lake James 2
## 10:       Lake Jose 2
```

Lisamouth and Williamsport are cities with the most visitors to the site.

### COLUMN: male  Frequency Table

```
as.data.frame(table(df$male))
```

```
##   Var1 Freq
## 1   NO  517
## 2  YES  475
```

```

# Using the vtree library
vtree(df, "male")
```


**Visualizing the above information**

52% of the visitors of the sight were from the female gender whreas 48% were of the male gender

**COLUMN: country  Frequency Table**

```
# This has been done using the dplyr library
country_summary <- df %>%
  count(country, sort = TRUE)

country_summary[1:10]
```

```
##            country n
##  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
```

Czech Republic and France produced the most visitors to the site

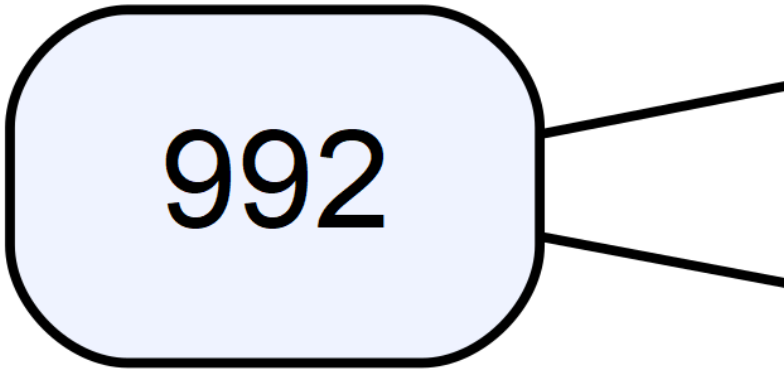
**COLUMN: clicked_on_ad  Frequency Table**

```
clicked_on_add_summary <- df %>%
  count(clicked_on_ad, sort = TRUE)

clicked_on_add_summary
```

```
##    clicked_on_ad   n
## 1:            NO 500
## 2:           YES 492
```

```
# Using the vtree library
vtree(df, "clicked_on_ad")
```




**Visualizing the above information**

Almost 50% (49.60%) of the site visitors clicked on the adds

## 4. Bivariate Analysis

Here we are going to compare other features with whether the individual clicked on the ad

### Correlational Analysis

```
num_cols <- unlist(lapply(df, is.numeric)) # Identifying numeric columns
num_cols
```

```
## daily_time_spent_on_site                     age             area_income
##                     TRUE                    TRUE                    TRUE
##     daily_internet_usage            ad_topic_line                    city
##                     TRUE                   FALSE                   FALSE
##                     male                 country            clicked_on_ad
##                    FALSE                   FALSE                   FALSE
##                     date                    hour
##                    FALSE                    TRUE
```

```
data_num <- subset(df, select=num_cols)                    # Subset numeric columns of data
data_num[1:10]
```

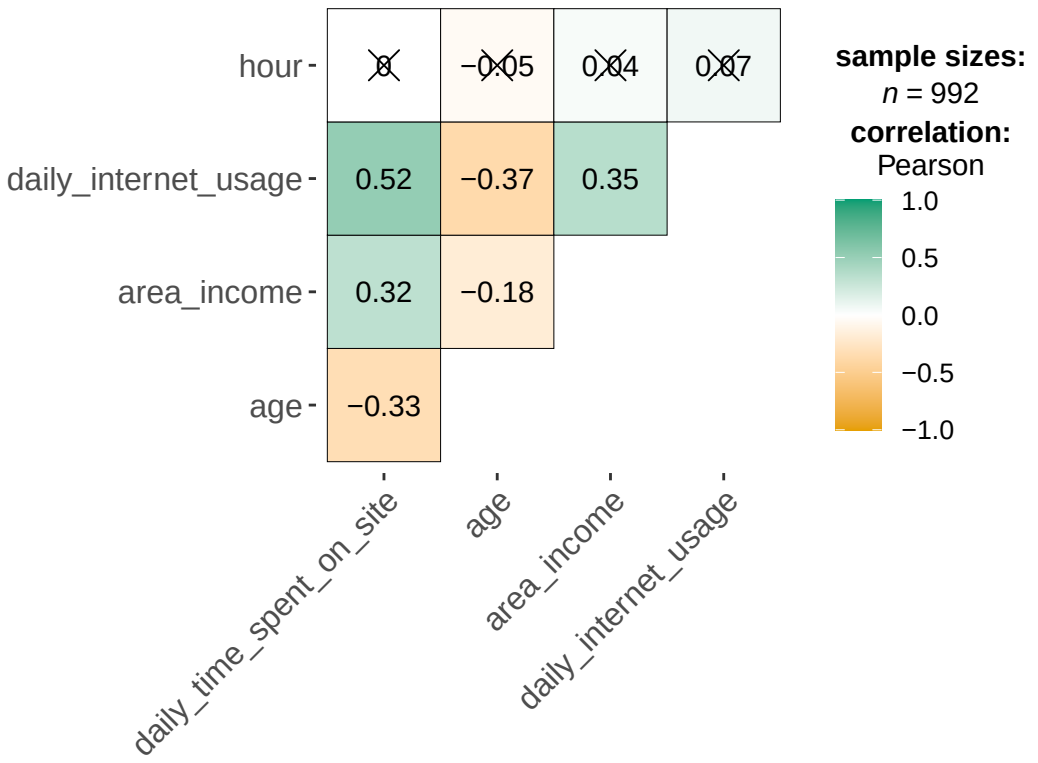
```
##     daily_time_spent_on_site age area_income daily_internet_usage hour
##  1:                    68.95  35    61833.90                256.09    0
##  2:                    80.23  31    68441.85                193.77    1
##  3:                    69.47  26    59785.94                236.50   20
##  4:                    74.15  29    54806.18                245.89    2
##  5:                    68.37  35    73889.99                225.58    3
##  6:                    59.99  23    59761.56                226.74   14
##  7:                    88.91  33    53852.85                208.36   20
##  8:                    66.00  48    24593.33                131.76    1
##  9:                    74.53  30    68862.00                221.51    9
## 10:                    69.88  20    55642.32                183.82    1
```

```
# Correlation Matrix
cor(data_num)
```

```
##                          daily_time_spent_on_site         age area_income
## daily_time_spent_on_site             1.0000000000 -0.33227617  0.31503738
## age                                 -0.3322761740  1.00000000 -0.18011099
## area_income                          0.3150373761 -0.18011099  1.00000000
## daily_internet_usage                 0.5197228251 -0.36793576  0.35082219
## hour                                 0.0005972223 -0.04908514  0.03801942
##                          daily_internet_usage          hour
## daily_time_spent_on_site           0.51972283  0.0005972223
## age                               -0.36793576 -0.0490851356
## area_income                        0.35082219  0.0380194158
## daily_internet_usage               1.00000000  0.0731832677
## hour                               0.07318327  1.0000000000
```

Visualizing the above result

```
ggcorrmat(data_num)
```




*Observations* 1. There is a *strong positive* correlation between daily_internet_usage and daily_time_spent_on_site 2. There is a moderate positive correlation between daily_time_spent_on_site and area_income, daily_internet_usage and area_income 3. There is a moderate negative correlation between age and daily_time_spent_on_site, daily_internet_usage and age 4. There is a weak negative correlation between area_income and age

### Covariance Analysis

```
for (i in colnames(data_num)){
  print(paste(toupper(i)))
  for(j in colnames(data_num)){
    print(paste("Covariance between",i,":",j,cov(df$daily_time_spent_on_site,df[[j]])))
  }
  print(paste("*******************************"))
}
```

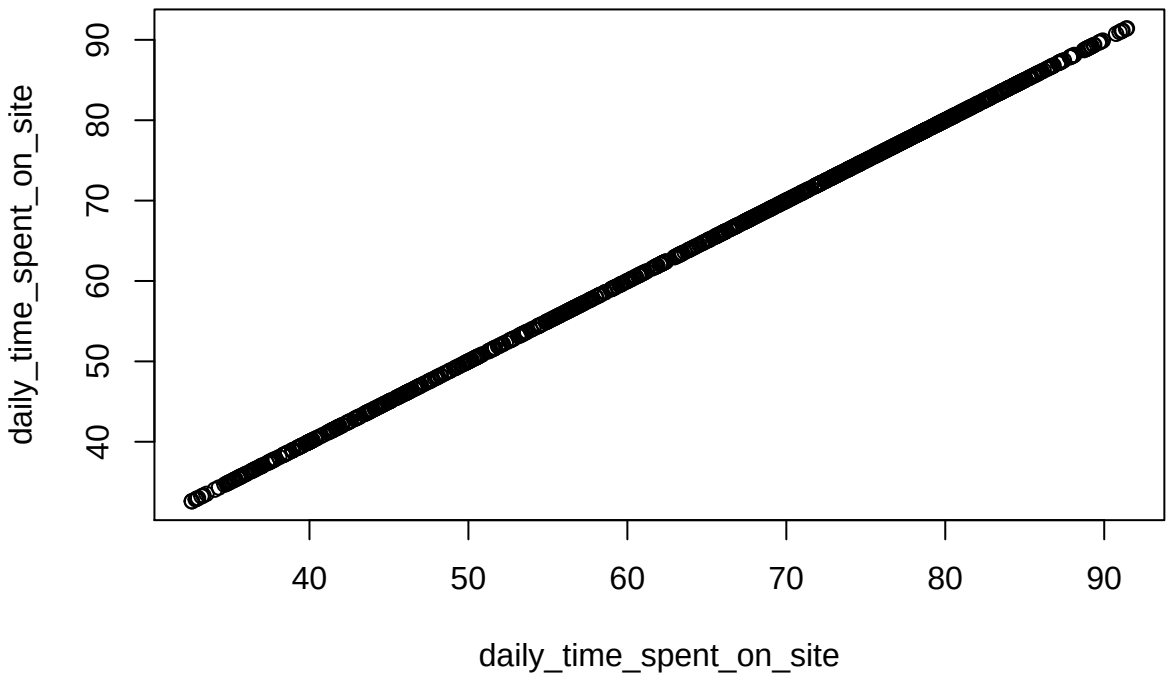
```
## [1] "DAILY_TIME_SPENT_ON_SITE"
## [1] "Covariance between daily_time_spent_on_site : daily_time_spent_on_site 252.860887240711"
## [1] "Covariance between daily_time_spent_on_site : age -46.5009017142183"
## [1] "Covariance between daily_time_spent_on_site : area_income 65151.2825671155"
## [1] "Covariance between daily_time_spent_on_site : daily_internet_usage 363.896103792601"
## [1] "Covariance between daily_time_spent_on_site : hour 0.0661931272582283"
```

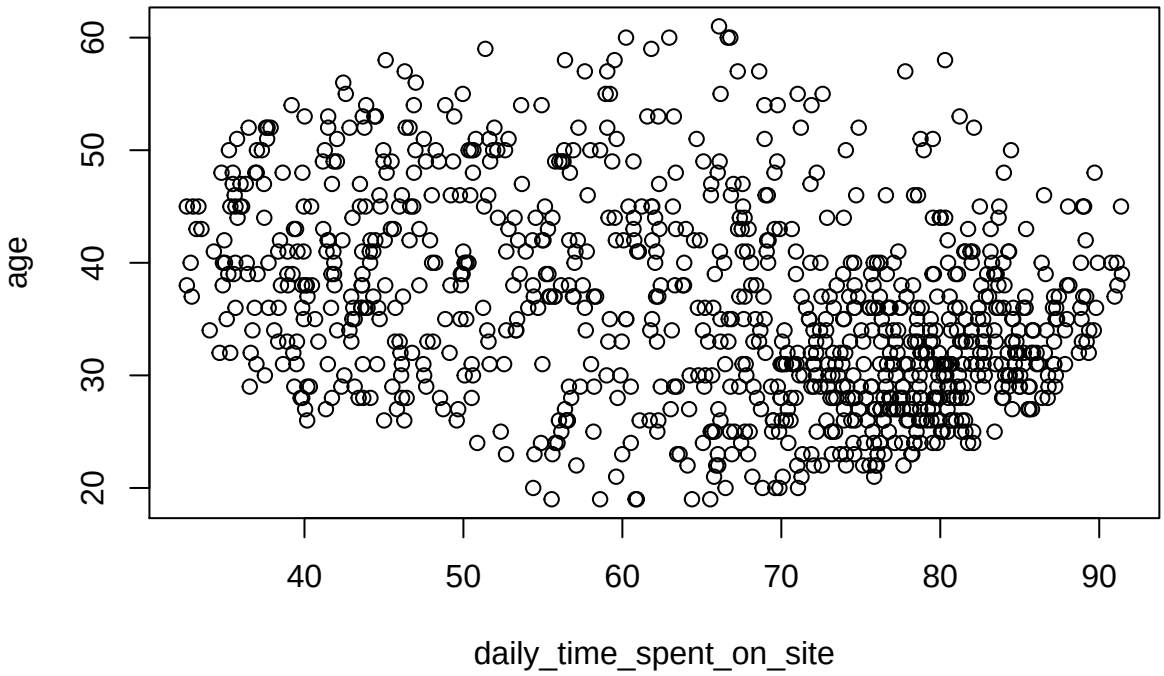
```
## [1] "******************************"
## [1] "AGE"
## [1] "Covariance between age : daily_time_spent_on_site 252.860887240711"
## [1] "Covariance between age : age -46.5009017142183"
## [1] "Covariance between age : area_income 65151.2825671155"
## [1] "Covariance between age : daily_internet_usage 363.896103792601"
## [1] "Covariance between age : hour 0.0661931272582283"
## [1] "******************************"
## [1] "AREA_INCOME"
## [1] "Covariance between area_income : daily_time_spent_on_site 252.860887240711"
## [1] "Covariance between area_income : age -46.5009017142183"
## [1] "Covariance between area_income : area_income 65151.2825671155"
## [1] "Covariance between area_income : daily_internet_usage 363.896103792601"
## [1] "Covariance between area_income : hour 0.0661931272582283"
## [1] "******************************"
## [1] "DAILY_INTERNET_USAGE"
## [1] "Covariance between daily_internet_usage : daily_time_spent_on_site 252.860887240711"
## [1] "Covariance between daily_internet_usage : age -46.5009017142183"
## [1] "Covariance between daily_internet_usage : area_income 65151.2825671155"
## [1] "Covariance between daily_internet_usage : daily_internet_usage 363.896103792601"
## [1] "Covariance between daily_internet_usage : hour 0.0661931272582283"
## [1] "******************************"
## [1] "HOUR"
## [1] "Covariance between hour : daily_time_spent_on_site 252.860887240711"
## [1] "Covariance between hour : age -46.5009017142183"
## [1] "Covariance between hour : area_income 65151.2825671155"
## [1] "Covariance between hour : daily_internet_usage 363.896103792601"
## [1] "Covariance between hour : hour 0.0661931272582283"
## [1] "******************************"
```

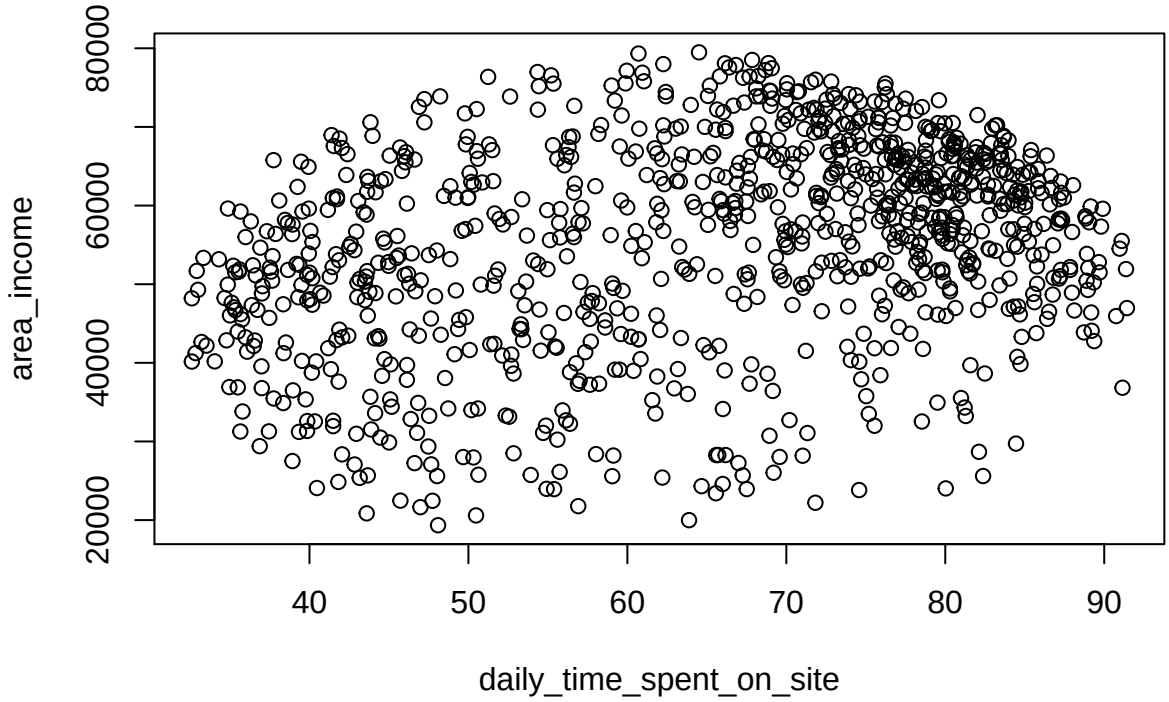
*Observations* 1. There is a very high *positive* covariance between area_income and daily_time_spent_on_site, age and daily_internet_usage 2. There is a *negative* covariance between age and daily_time_spent_on_site, area_income and daily_internet_usage

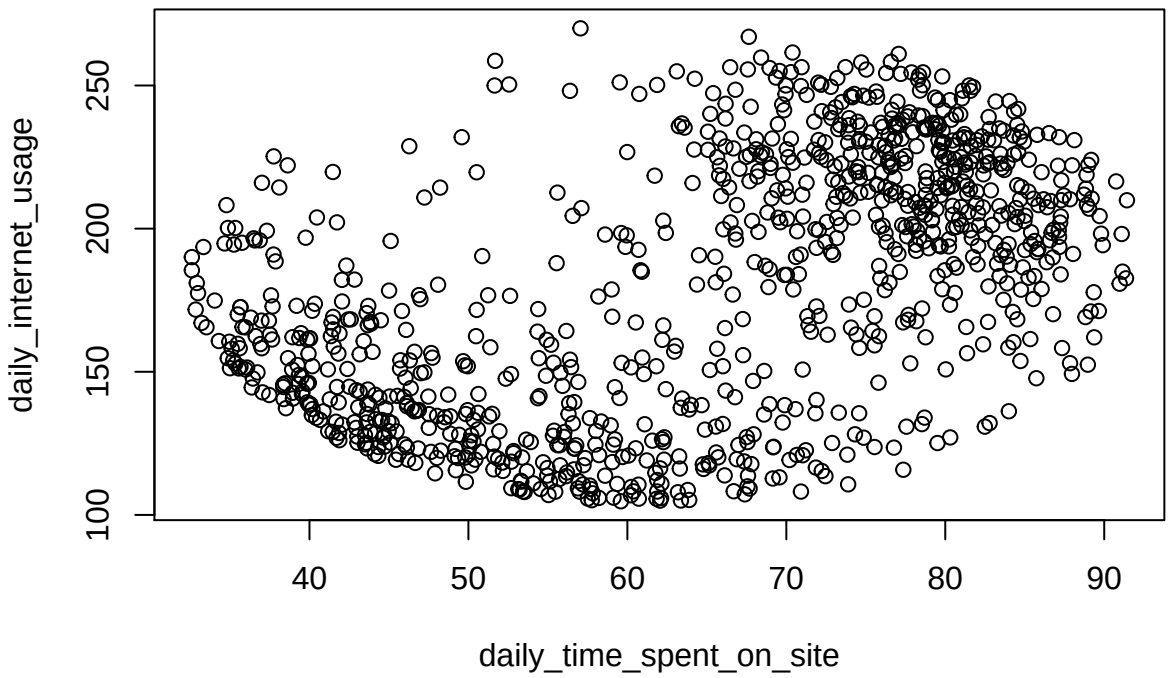
### Scatter Plots

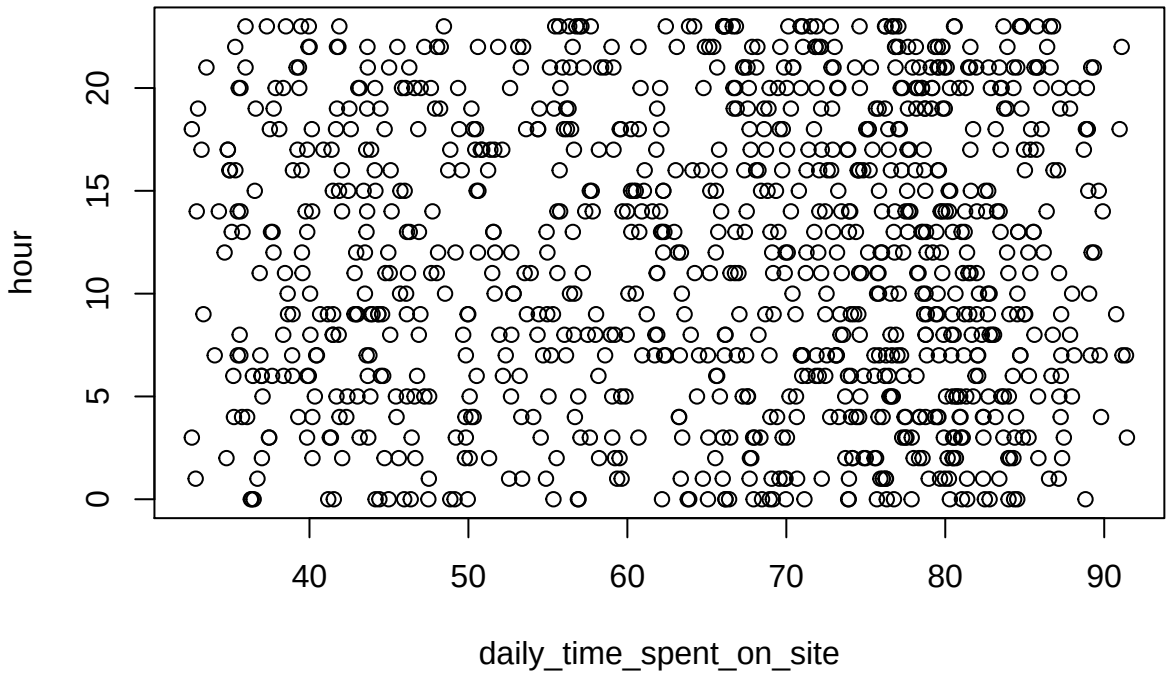
```
for (i in colnames(data_num)){
  for(j in colnames(data_num)){
    plot(data_num[[i]], data_num[[j]], xlab= i, ylab=j)
  }
}
```

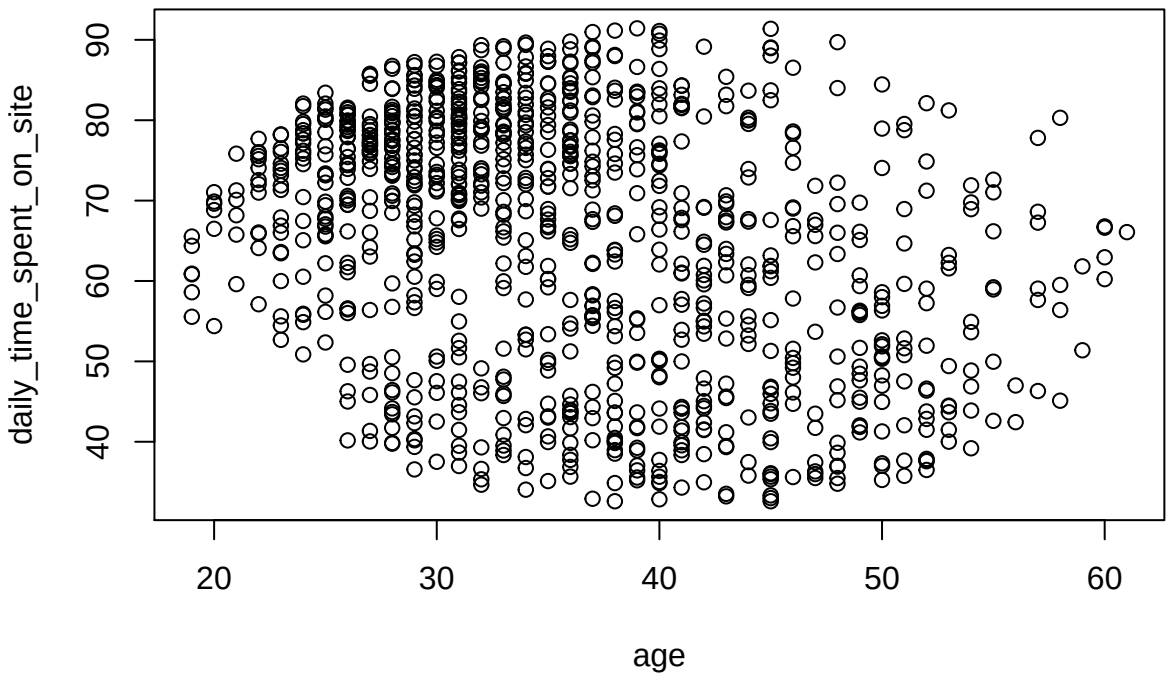

daily_time_spent_on_site
40
50
60
70
80
90
daily_time_spent_on_site
40
50
60
70
80
90

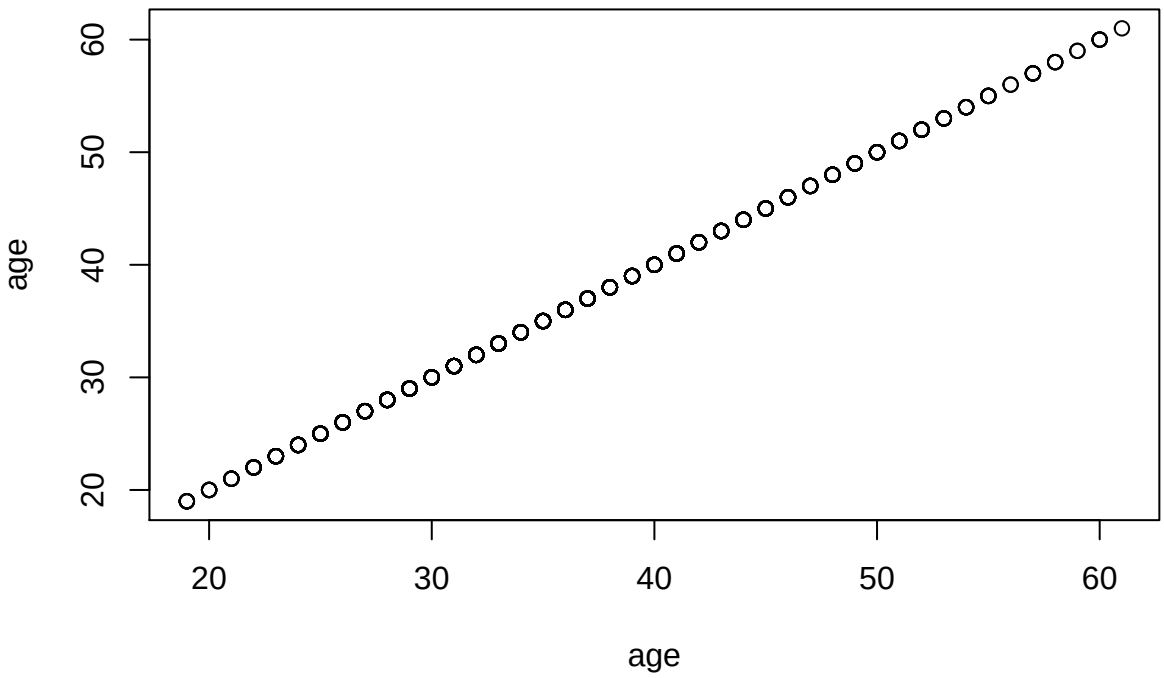

age
20
30
40
50
60
40
50
60
70
80
90
daily_time_spent_on_site

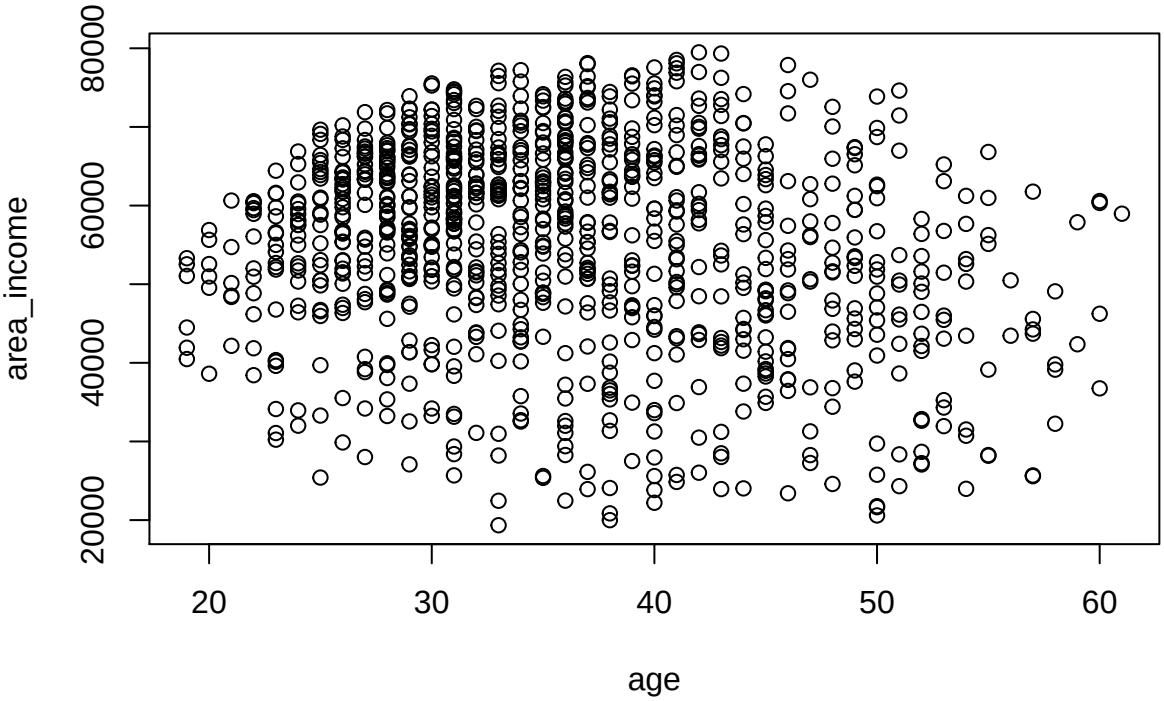

area_income
20000
40000
60000
80000
40
50
60
70
80
90
daily_time_spent_on_site

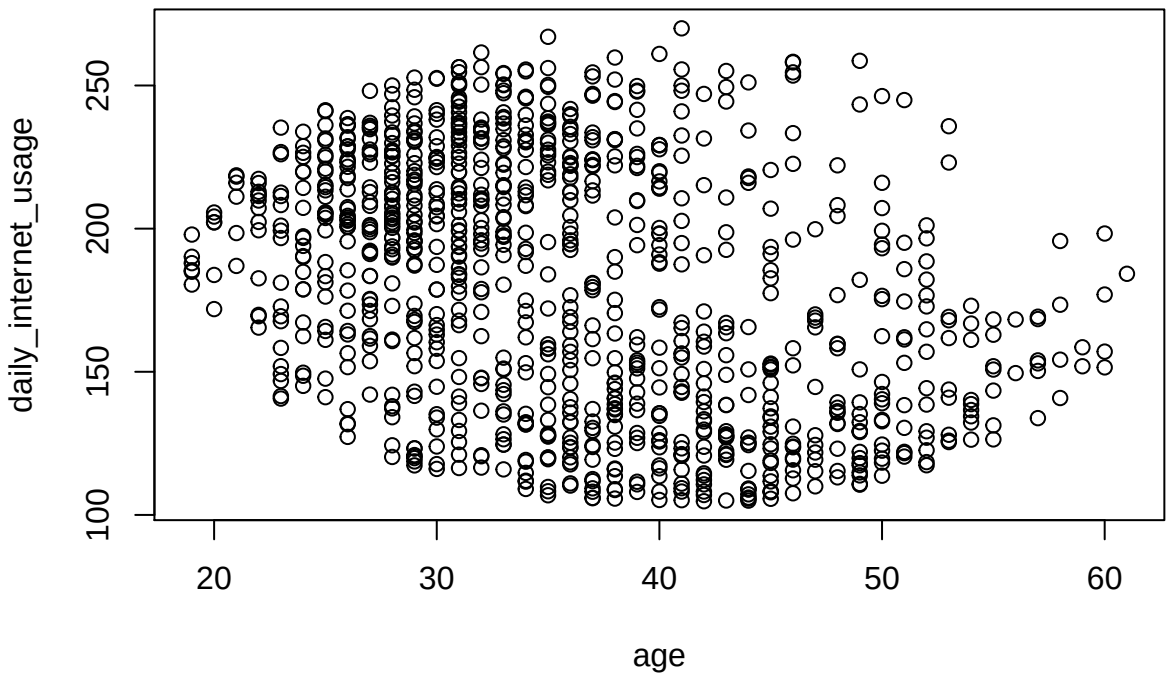

daily_internet_usage
100
150
200
250
40
50
60
70
80
90
daily_time_spent_on_site

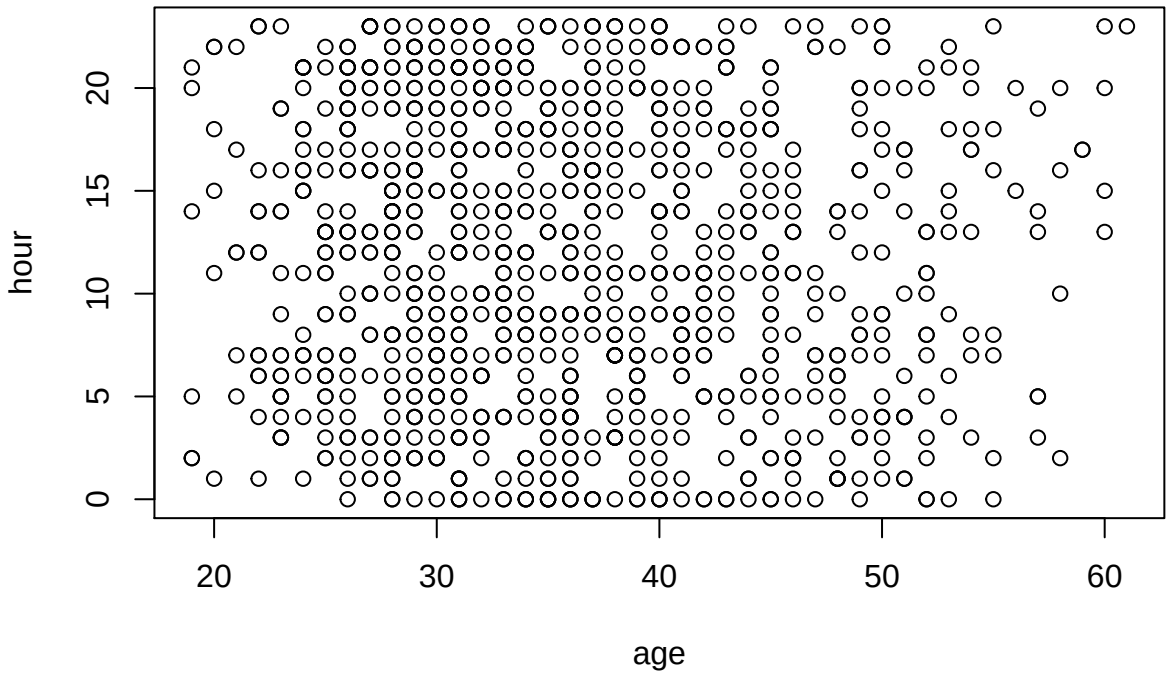

hour
0
5
10
15
20
40
50
60
70
80
90
daily_time_spent_on_site

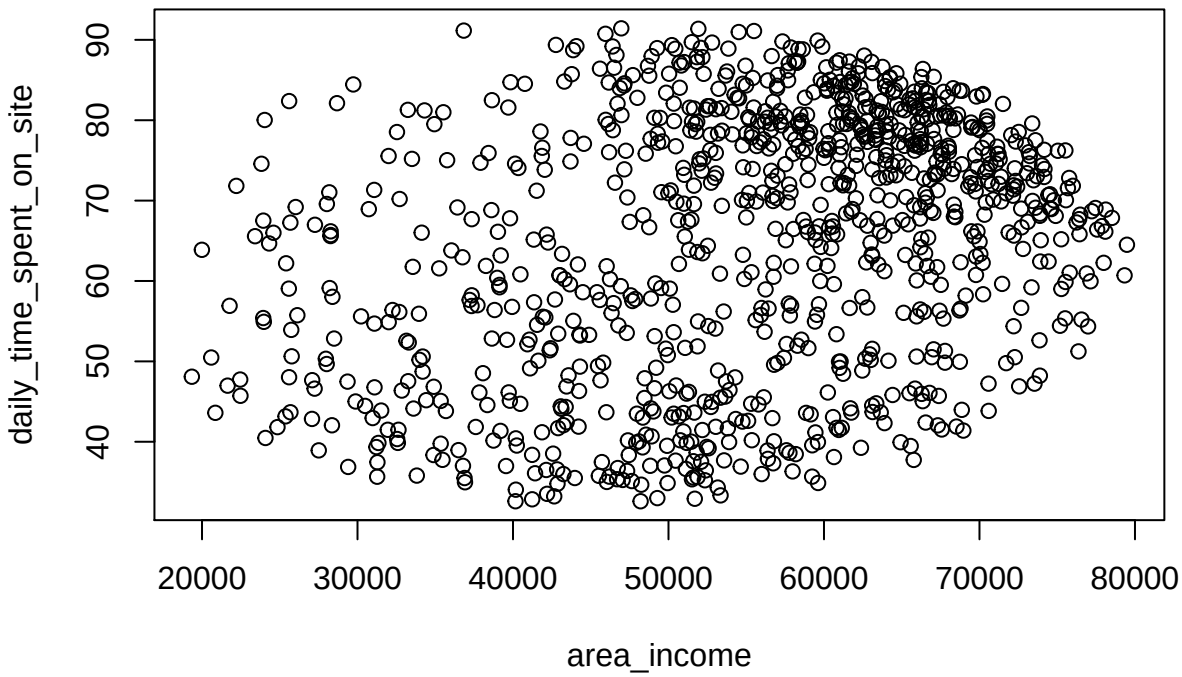

daily_time_spent_on_site
40
50
60
70
80
90
20
30
40
50
60
age

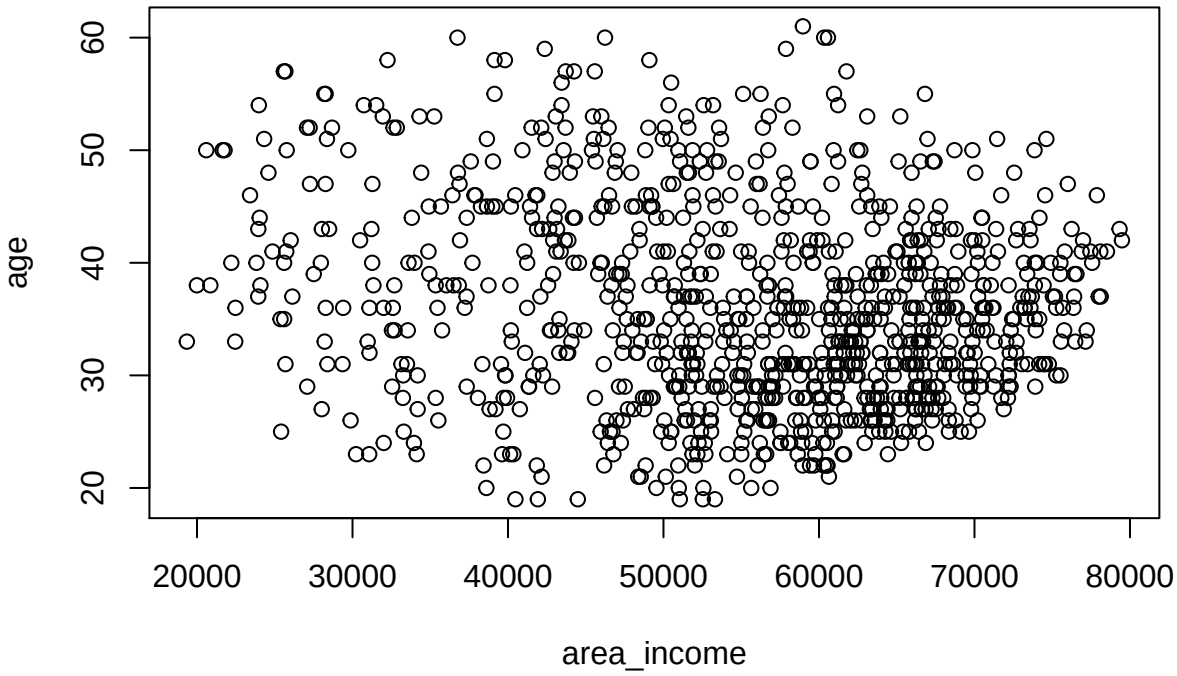

age
20
30
40
50
60
20
30
40
50
60
age

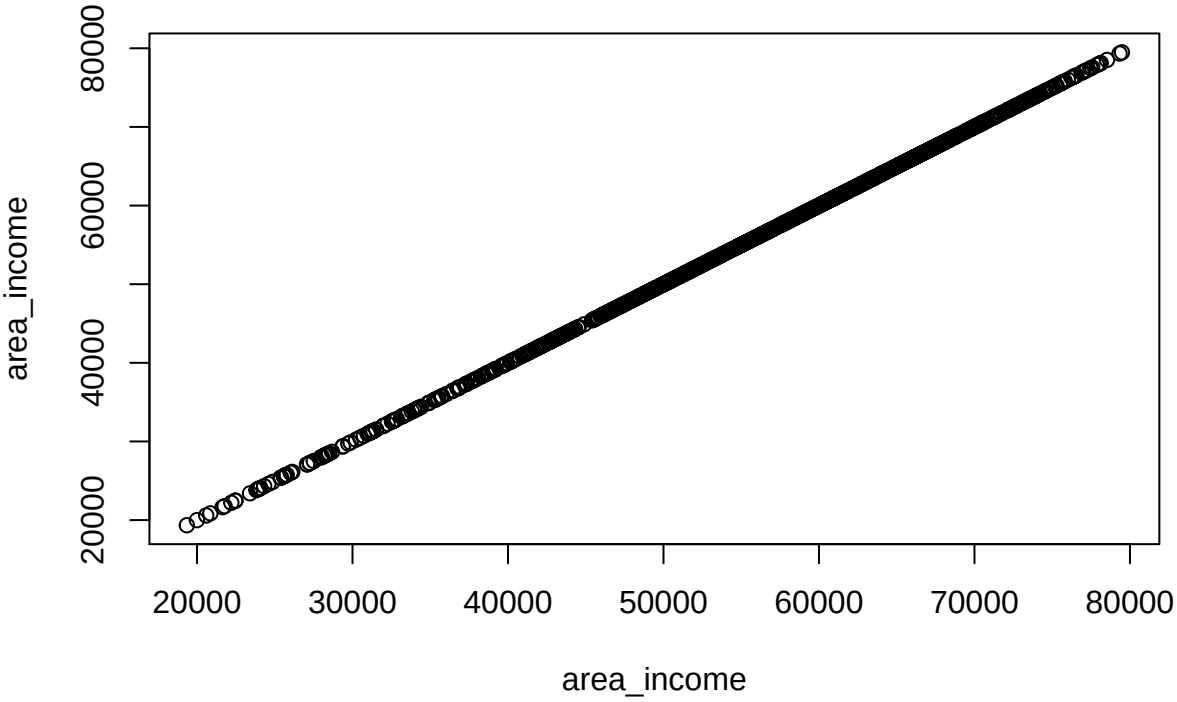

area_income
20000
40000
60000
80000
20
30
40
50
60
age

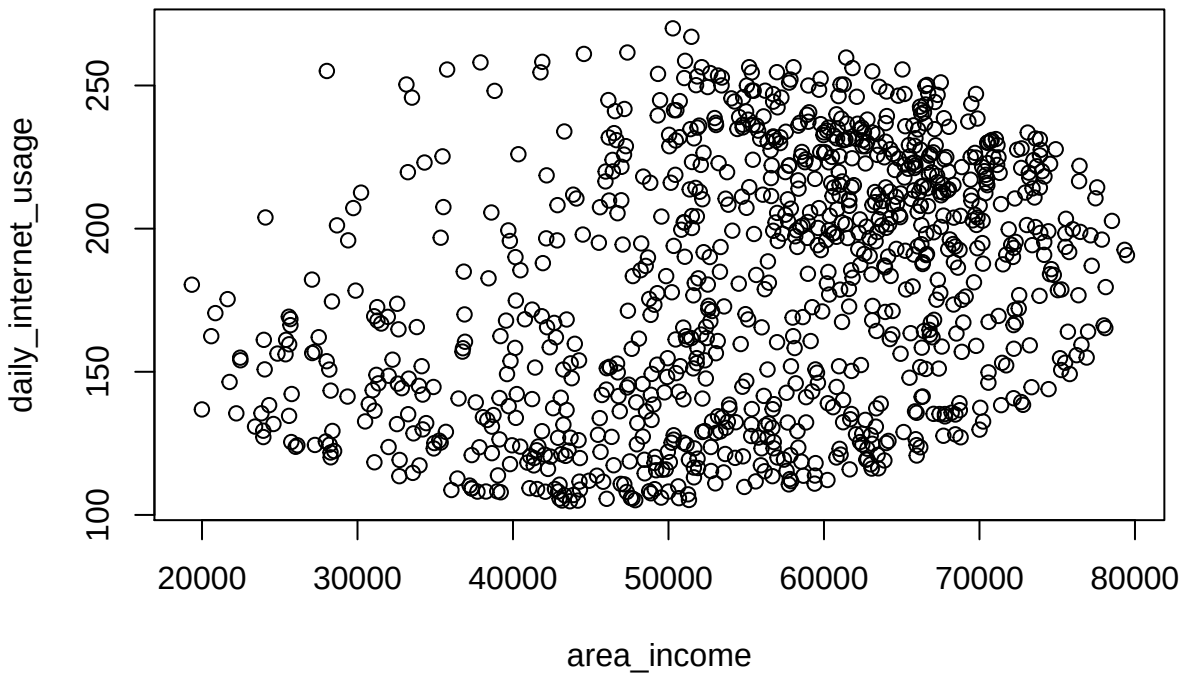

daily_internet_usage
100
150
200
250
20
30
40
50
60
age

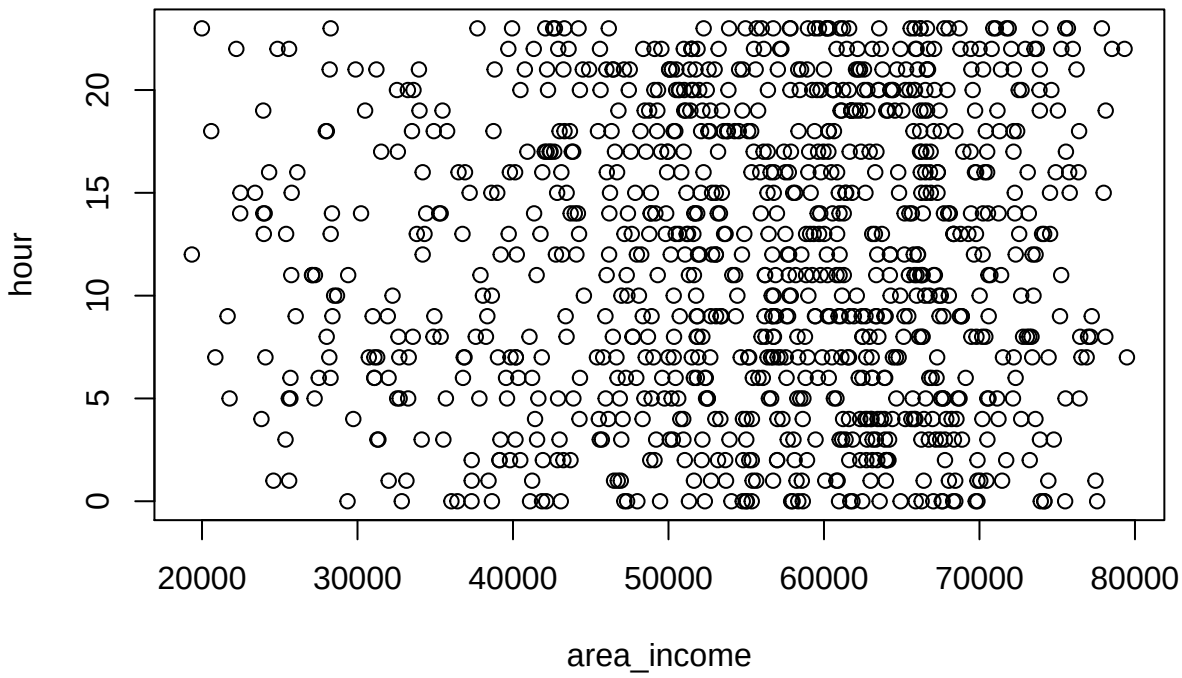

hour
0
5
10
15
20
20
30
40
50
60
age

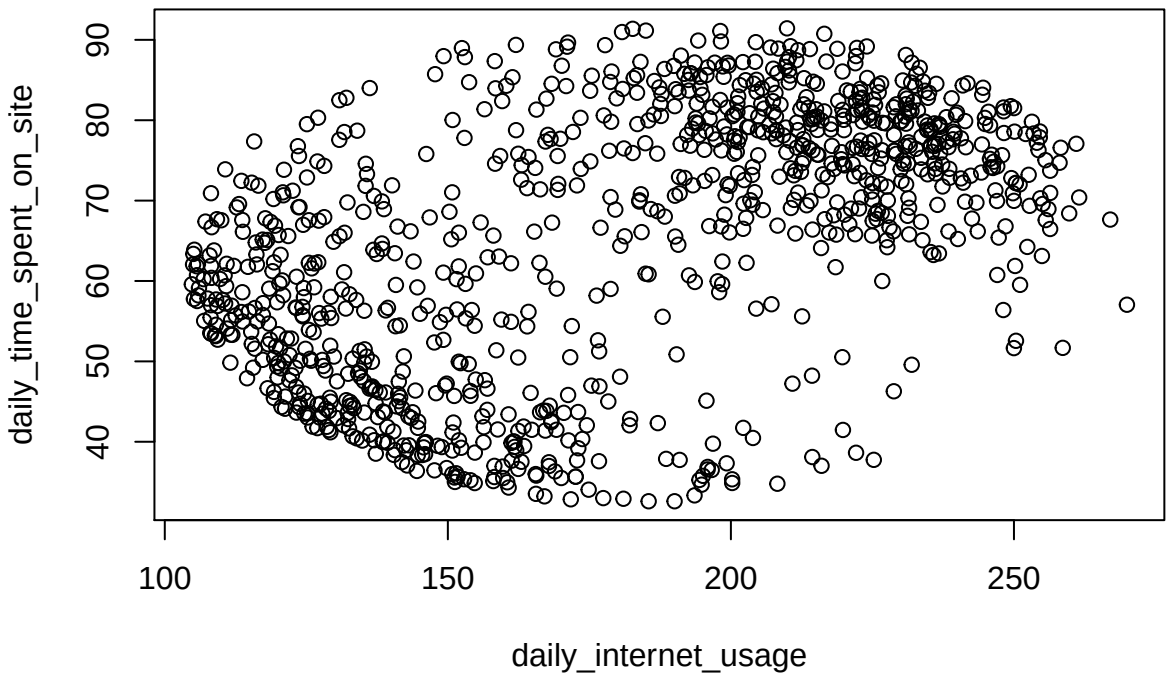

daily_time_spent_on_site
40
50
60
70
80
90
20000
30000
40000
50000
60000
70000
80000
area_income

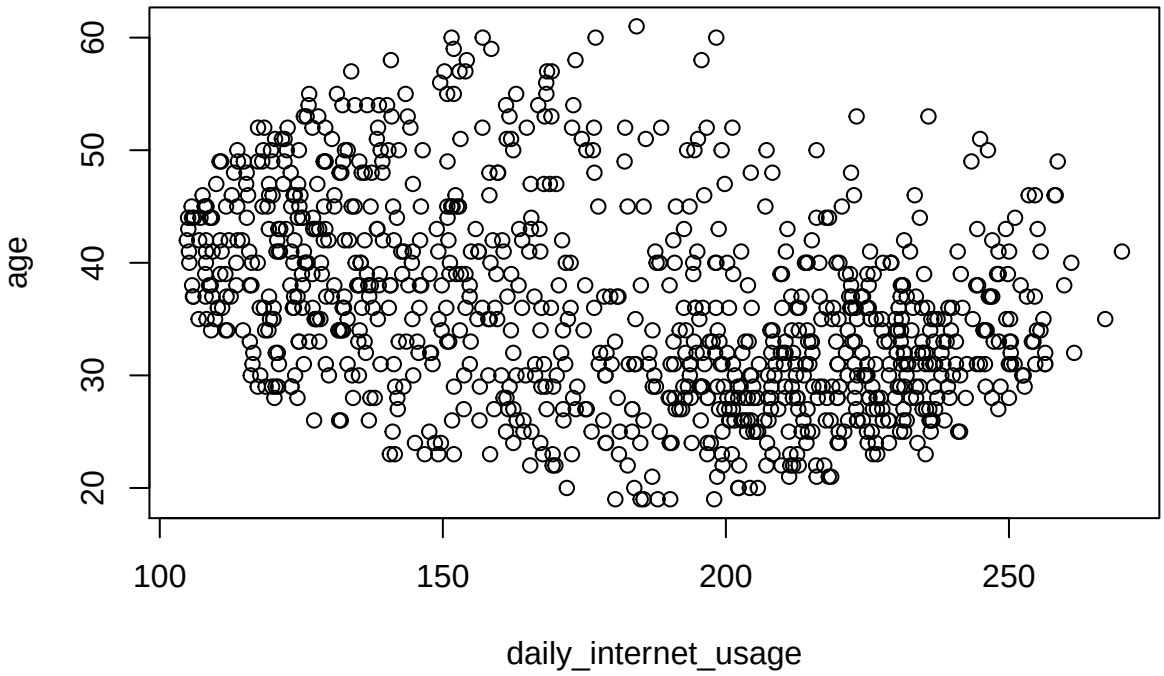

age
20
30
40
50
60
20000
30000
40000
50000
60000
70000
80000
area_income

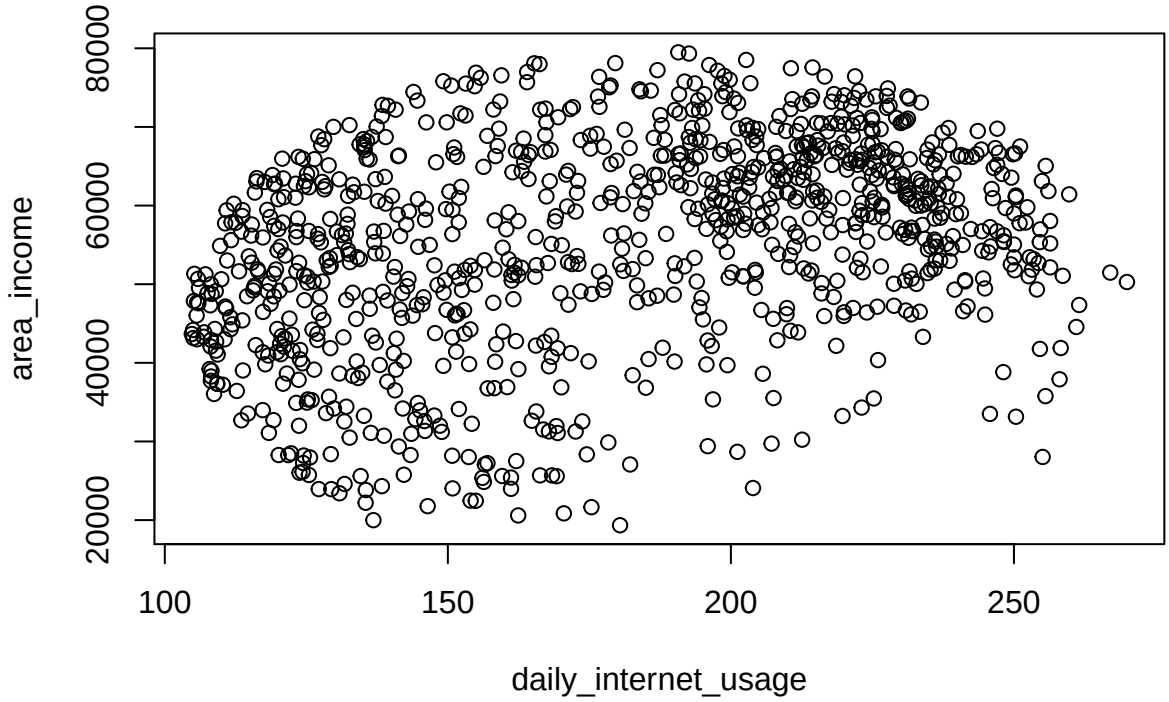

area_income
20000
40000
60000
80000
20000
30000
40000
50000
60000
70000
80000
area_income

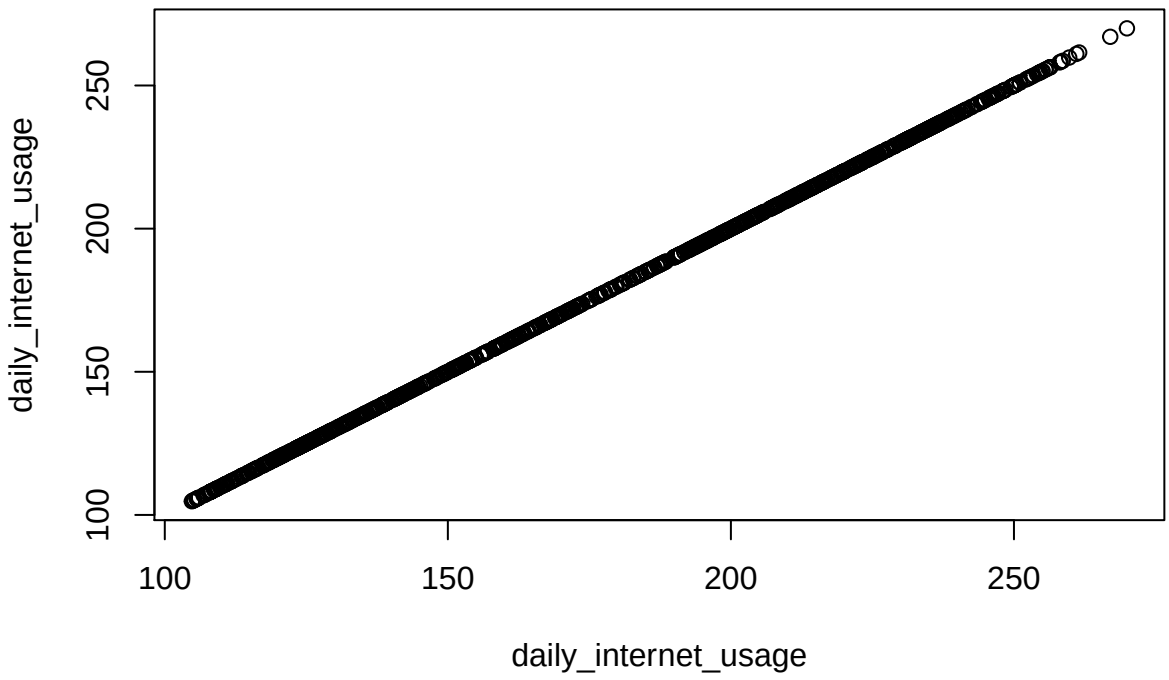

daily_internet_usage
100
150
200
250
20000
30000
40000
50000
60000
70000
80000
area_income

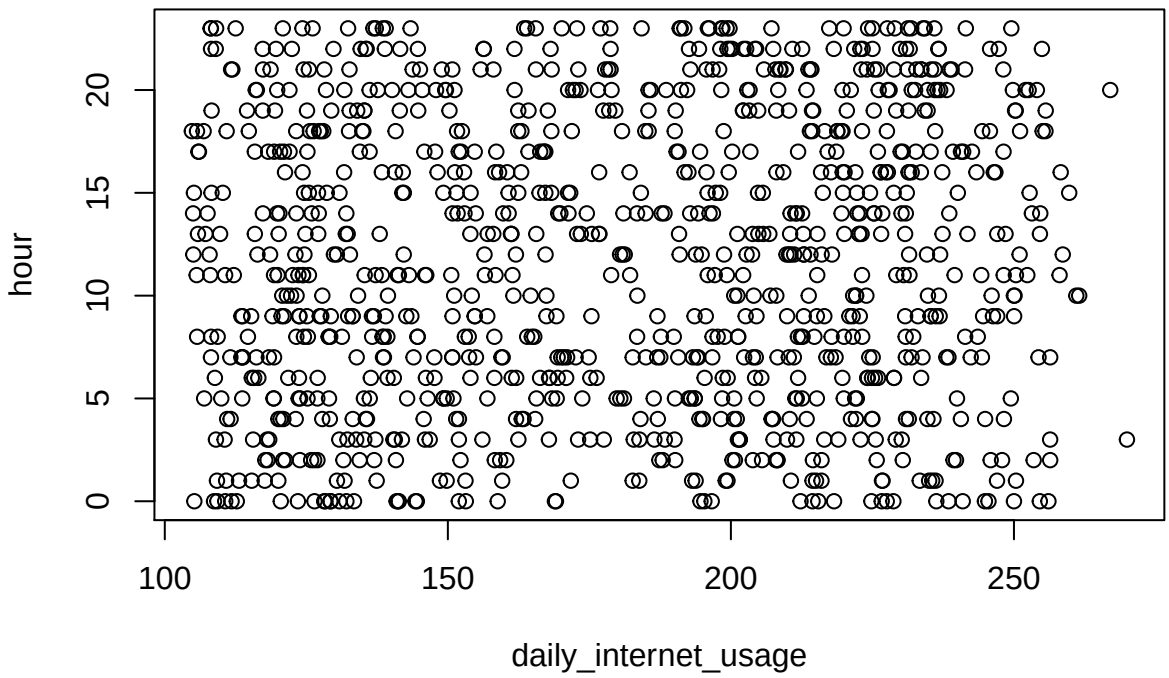

hour
0
5
10
15
20
20000
30000
40000
50000
60000
70000
80000
area_income

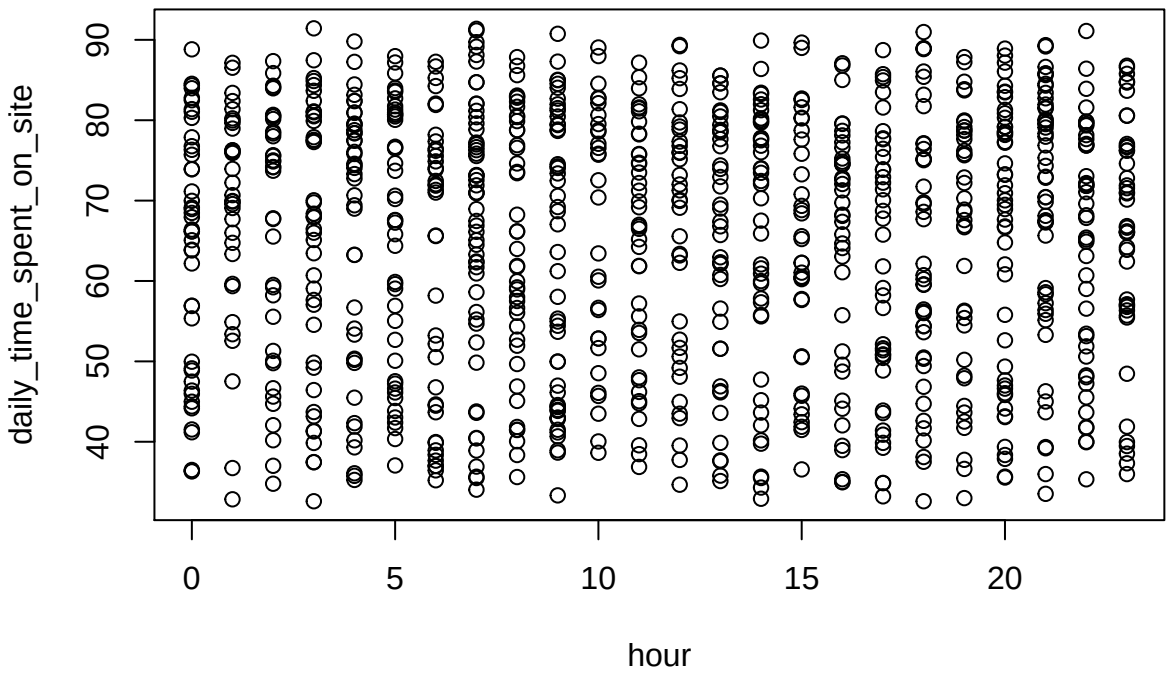

daily_time_spent_on_site
40
50
60
70
80
90
100
150
200
250
daily_internet_usage

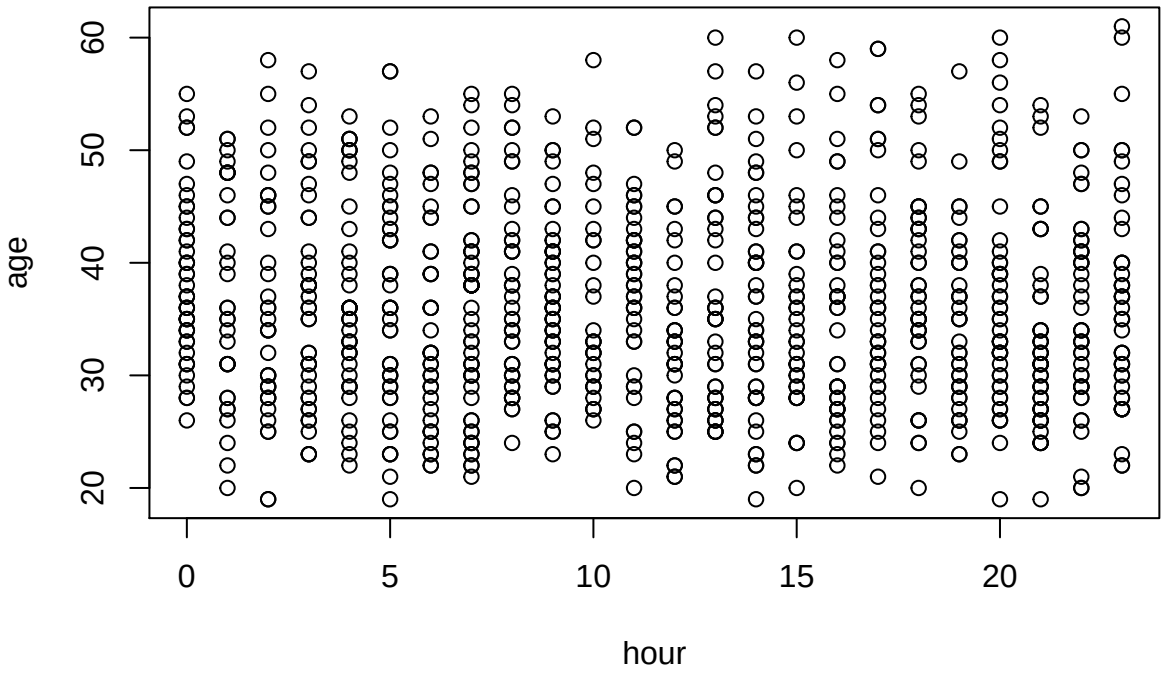

age
20
30
40
50
60
100
150
200
250
daily_internet_usage

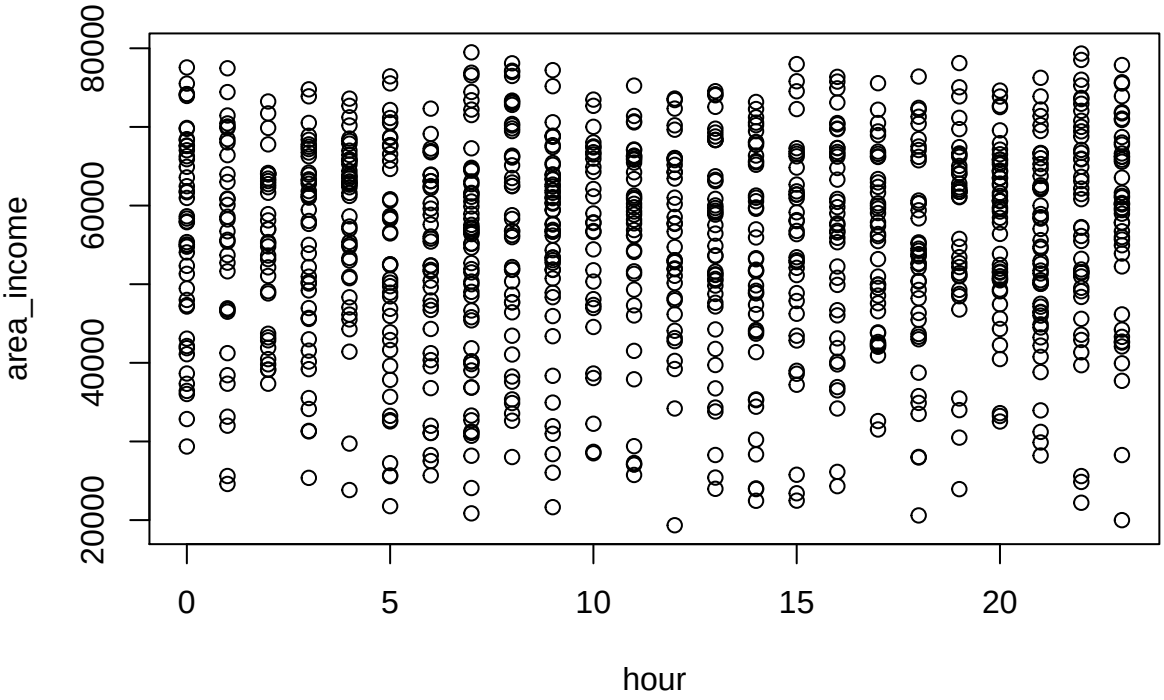

area_income
20000
40000
60000
80000
100
150
200
250
daily_internet_usage

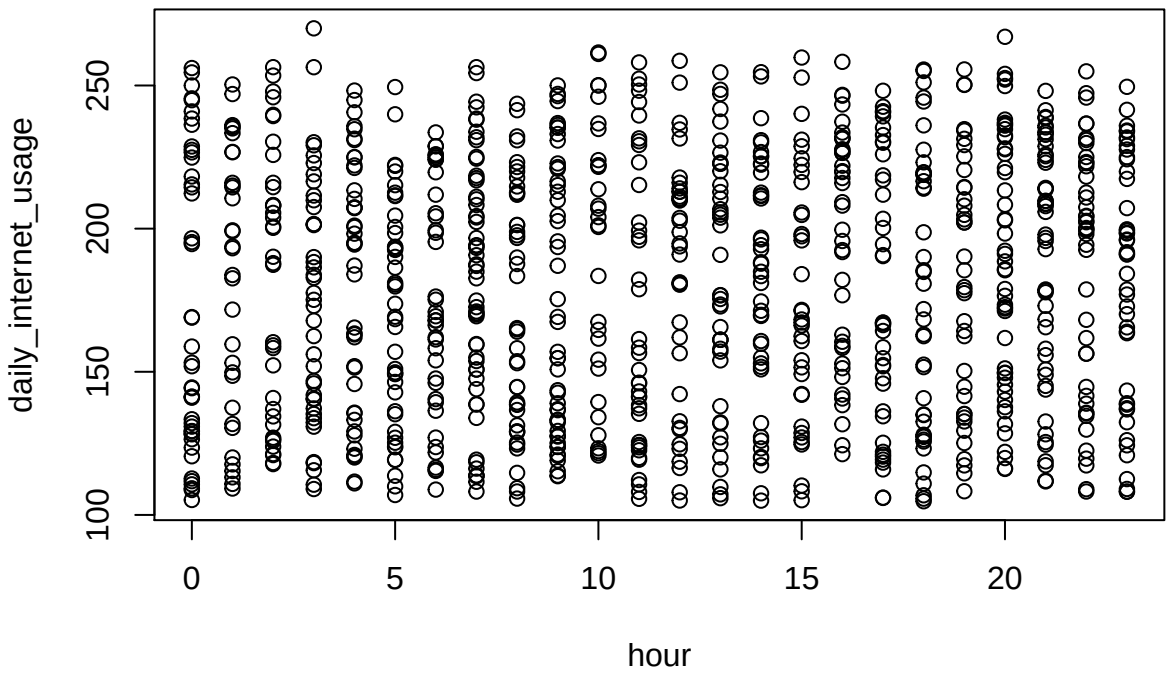

daily_internet_usage
100
150
200
250
100
150
200
250
daily_internet_usage

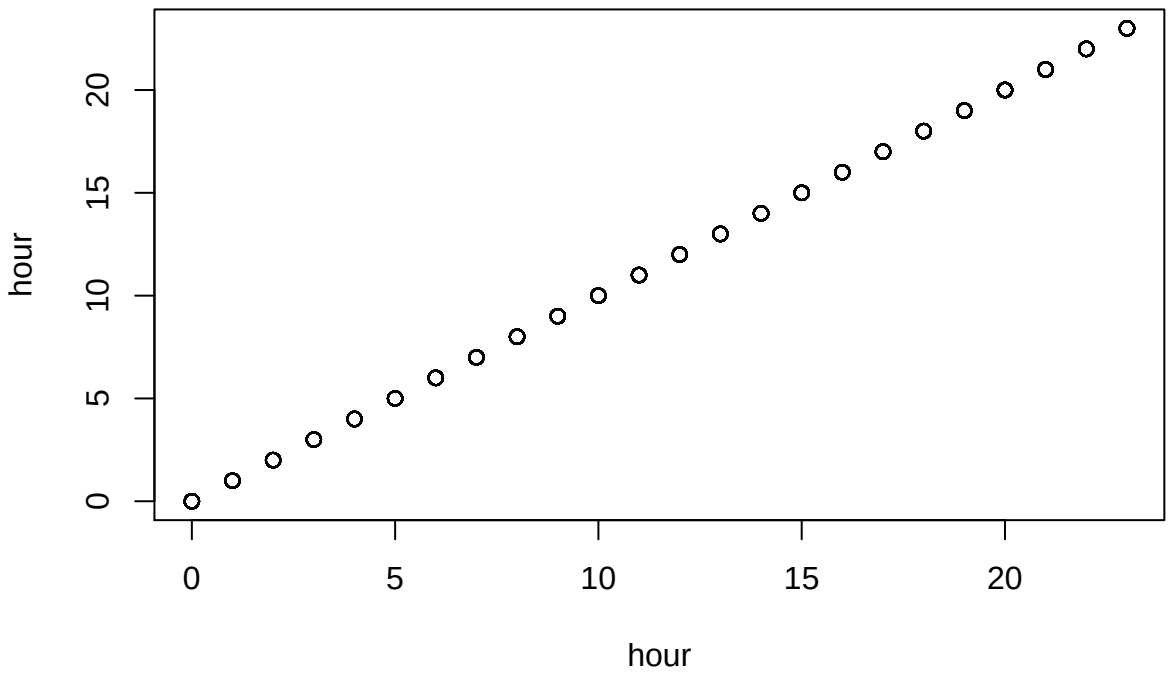

hour
0
5
10
15
20
100
150
200
250
daily_internet_usage


daily_time_spent_on_site
40
50
60
70
80
90
0
5
10
15
20
hour


age
20
30
40
50
60
0
5
10
15
20
hour


area_income
20000
40000
60000
80000
0
5
10
15
20
hour

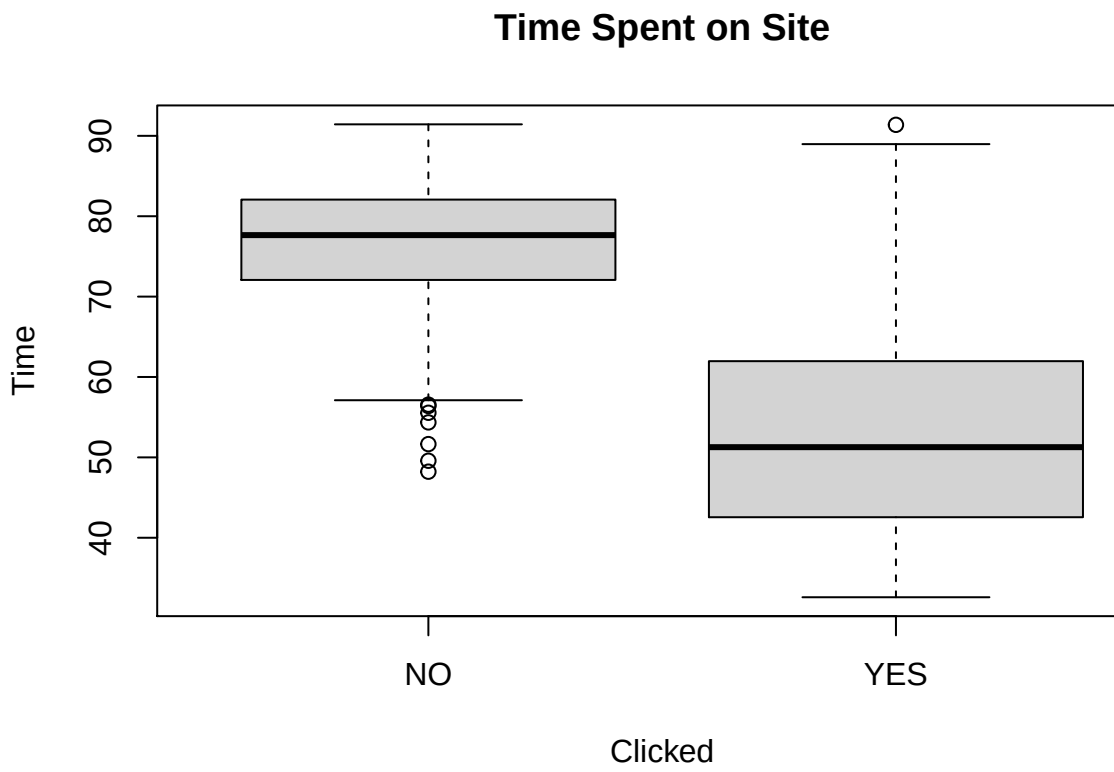

daily_internet_usage
100
150
200
250
0
5
10
15
20
hour




*Observations* The graphs do not indicate linear relationship between different features.

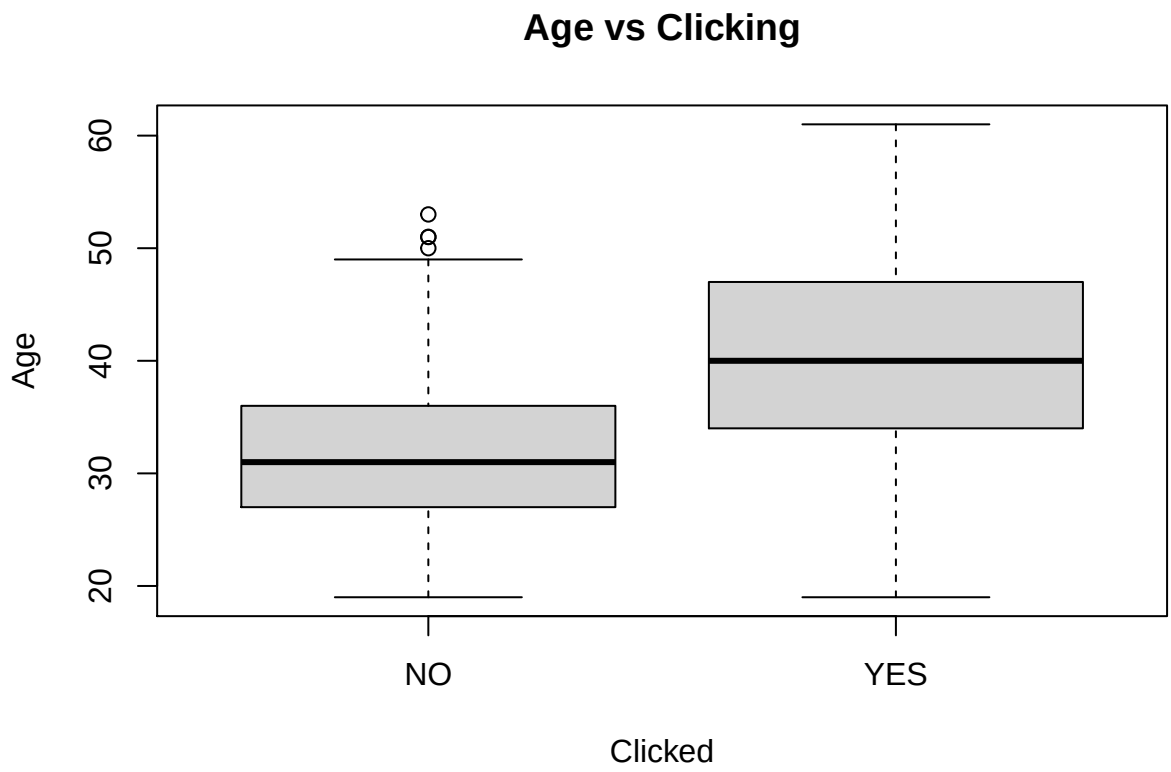
**Incoporating some categorical variables**

```
# Plot the chart.
boxplot(daily_time_spent_on_site ~ clicked_on_ad, data = df, xlab = "Clicked",
   ylab = "Time", main = "Time Spent on Site")
```



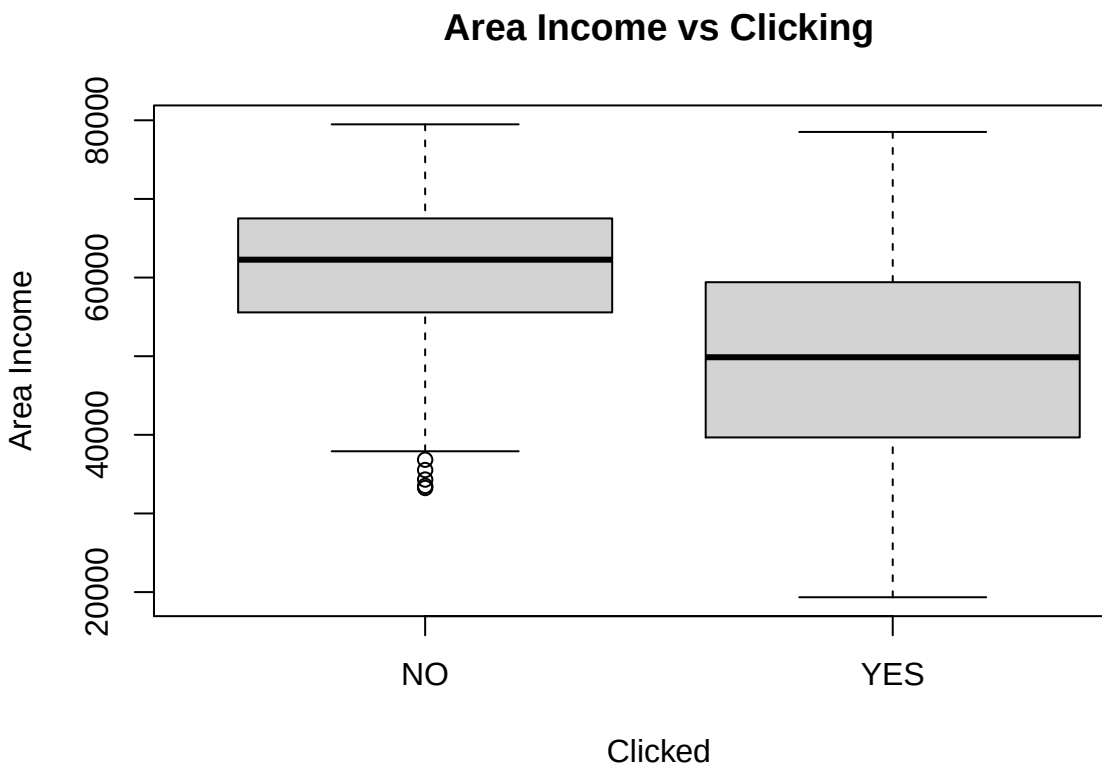

**Clicked vs Daily Times**
On average those who spend shorter times on the internet are likely to click on the ad

```
# Plot the chart.
boxplot(age ~ clicked_on_ad, data = df, xlab = "Clicked",
   ylab = "Age", main = "Age vs Clicking")
```



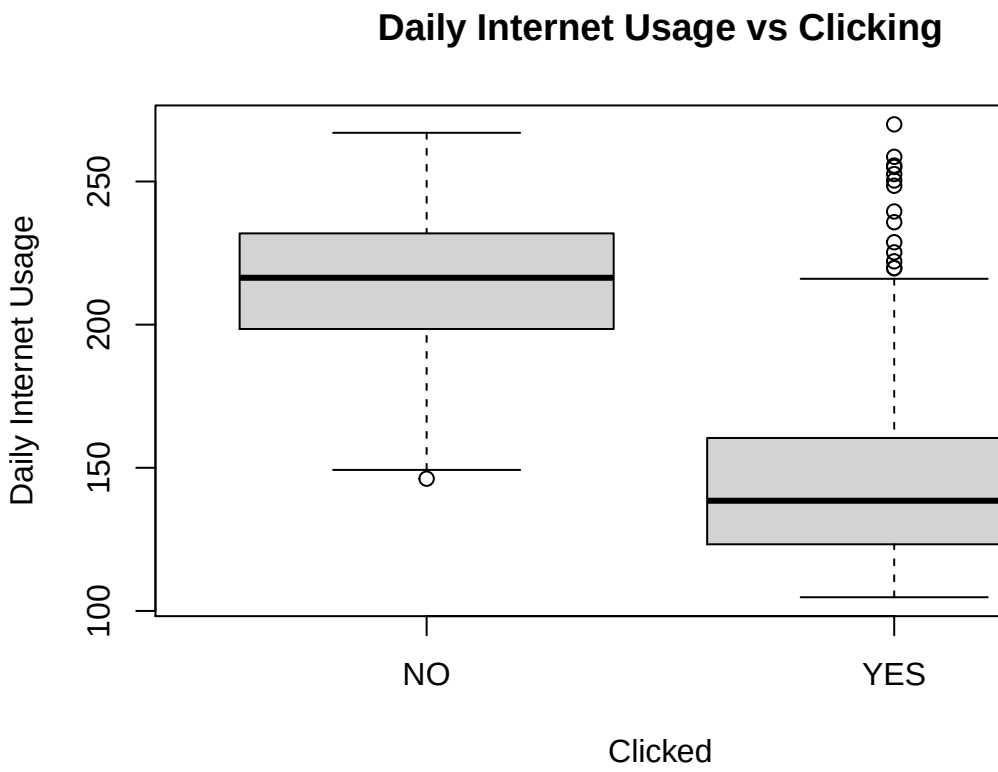

**Clicked vs Age**
On average older people click ads as compared to younger people

```
# Plot the chart.
boxplot(area_income ~ clicked_on_ad, data = df, xlab = "Clicked",
   ylab = "Area Income", main = "Area Income vs Clicking")
```



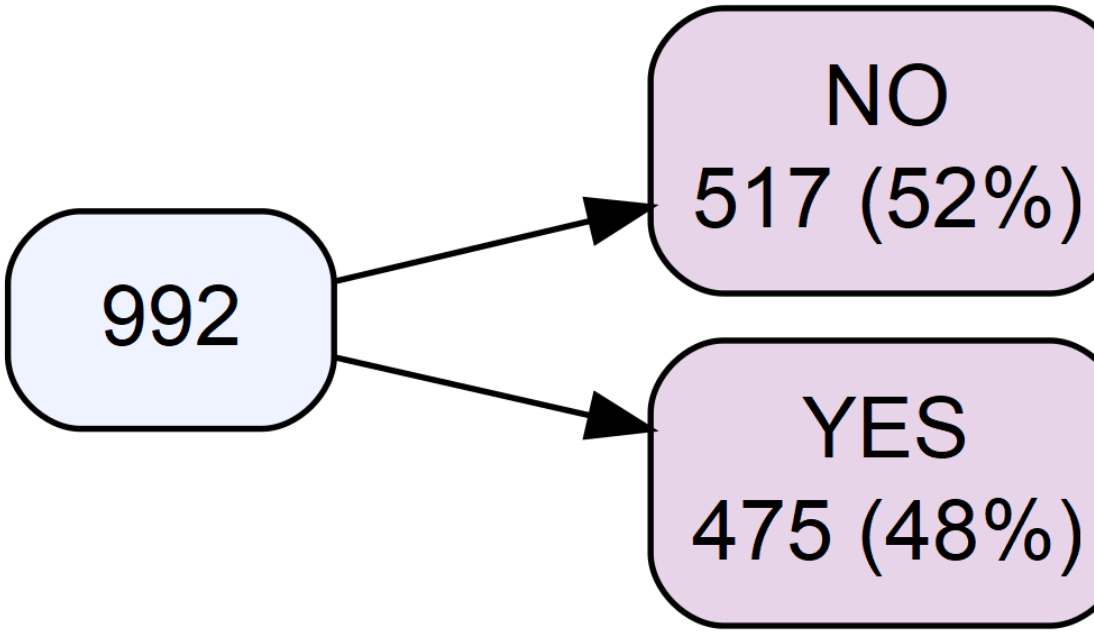

**Clicked vs Area Income**
Those from high income areas are less likely to click on the ad

```
# Plot the chart.
boxplot(daily_internet_usage ~ clicked_on_ad, data = df, xlab = "Clicked",
   ylab = "Daily Internet Usage", main = "Daily Internet Usage vs Clicking")
```




**Clicked vs Daily Internet Usage**
Those who spend less time on the internet are more likely click on the ad

```
vtree(df, c("male", "clicked_on_ad"),
   fillcolor = c( male = "#e7d4e8", clicked_on_ad = "#99d8c9"))
```




male

**Clicked vs Male**

**Observations* 1. Most of the visitors to the site were Female 2. Females are more likely to click on the

ad than males

## 5. Recommendations

1. The most popular hour of the day to run advert is 7 am
2. Targeting the female audience will lead to increased clicks