Moringa Core Week 12 Independent Project

Agnes Githiri

2022-05-30

1) Defining the question

a) Specifying the question

Which individuals are most likely to click on the ads?

b) Defining the Metric of Success

The objective is to successfully identify which individuals are most likely to click on the ad.

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. She would now like to employ the services of a Data Science Consultant to help her identify which individuals are most likely to click on her ads.

d) Recording the experimental design

1) Reading the Data
2) Checking the Data
3) Tidying the Dataset
4) Perform Exploratory Descriptive Analysis
5) Conclusions
6) Recommendations

e) Data Relevance

A dataset has been provided with the right data of the previous adverts.

2) Reading the data

```
library(data.table)
Adverts <- fread("http://bit.ly/IPAdvertisingData")
```

3) Checking the data

Lets preview the first 6 records of the data

```
head(Adverts)
```

```
##    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
```

```
colnames(Adverts) <- c('Daily.Time.Spent.on.Site','Age','Area.Income',
                      'Daily.Internet.Usage','Ad.Topic.Line', 'City', 'Male',
                      'Country', 'Timestamp','Clicked.on.Ad')
# printing new data frame
print(Adverts)
```

```
##       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
##   ---
##  996:                    72.97  30    71384.57               208.58
##  997:                    51.30  45    67782.17               134.42
##  998:                    51.63  51    42415.72               120.37
##  999:                    55.55  19    41920.79               187.95
## 1000:                    45.01  26    29875.80               178.35
##                                Ad.Topic.Line           City Male
##    1:    Cloned 5thgeneration orchestration    Wrightburgh    0
##    2:    Monitored national standardization      West Jodi    1
##    3:       Organic bottom-line service-desk       Davidton    0
##    4: Triple-buffered reciprocal time-frame West Terrifurt    1
##    5:         Robust logistical utilization   South Manuel    0
##   ---
##  996:         Fundamental modular algorithm      Duffystad    1
##  997:       Grass-roots cohesive monitoring    New Darlene    1
##  998:          Expanded intangible solution  South Jessica    1
##  999:  Proactive bandwidth-monitored policy    West Steven    0
## 1000:       Virtual 5thgeneration emulation    Ronniemouth    0
##                       Country           Timestamp Clicked.on.Ad
##    1:                Tunisia 2016-03-27 00:53:11             0
##    2:                  Nauru 2016-04-04 01:39:02             0
##    3:             San Marino 2016-03-13 20:35:42             0
##    4:                  Italy 2016-01-10 02:31:19             0
##    5:                Iceland 2016-06-03 03:36:18             0
##   ---
##  996:                Lebanon 2016-02-11 21:49:00             1
##  997: Bosnia and Herzegovina 2016-04-22 02:07:01             1
##  998:               Mongolia 2016-02-01 17:24:57             1
##  999:              Guatemala 2016-03-24 02:35:54             0
## 1000:                 Brazil 2016-06-03 21:43:21             1
```

Lets check the size of the data

The dataset has 1000 rows and 10 columns

```
dim(Adverts)
```

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

Lets check the datatypes of each column

```
str(Adverts)
```

```
## 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>
```

4) Tidying the data

Lets check for missing data

The dataset has no missing values

```
colSums(is.na(Adverts))
```

```
## 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
```

Lets check for duplicates

The dataset has no duplicates

```
duplicates <- Adverts[duplicated(Adverts), ]
duplicates
```

```
## Empty data.table (0 rows and 10 cols): Daily.Time.Spent.on.Site,Age,Area.Income,Daily.Internet.Usage
```

Lets create a function for the numerical columns

```
numerical_cols <- Adverts[,unlist(lapply(Adverts, is.numeric))]
numerical_cols
```

```
## Daily.Time.Spent.on.Site                      Age              Area.Income
##                     TRUE                     TRUE                     TRUE
##     Daily.Internet.Usage            Ad.Topic.Line                     City
##                     TRUE                    FALSE                    FALSE
##                     Male                  Country                Timestamp
##                     TRUE                    FALSE                    FALSE
##            Clicked.on.Ad
##                     TRUE
```

creating a dataframe with numeric columns only so as to plot a boxplot

```
Adverts_numerical_cols <-Adverts[, ..numerical_cols]
```

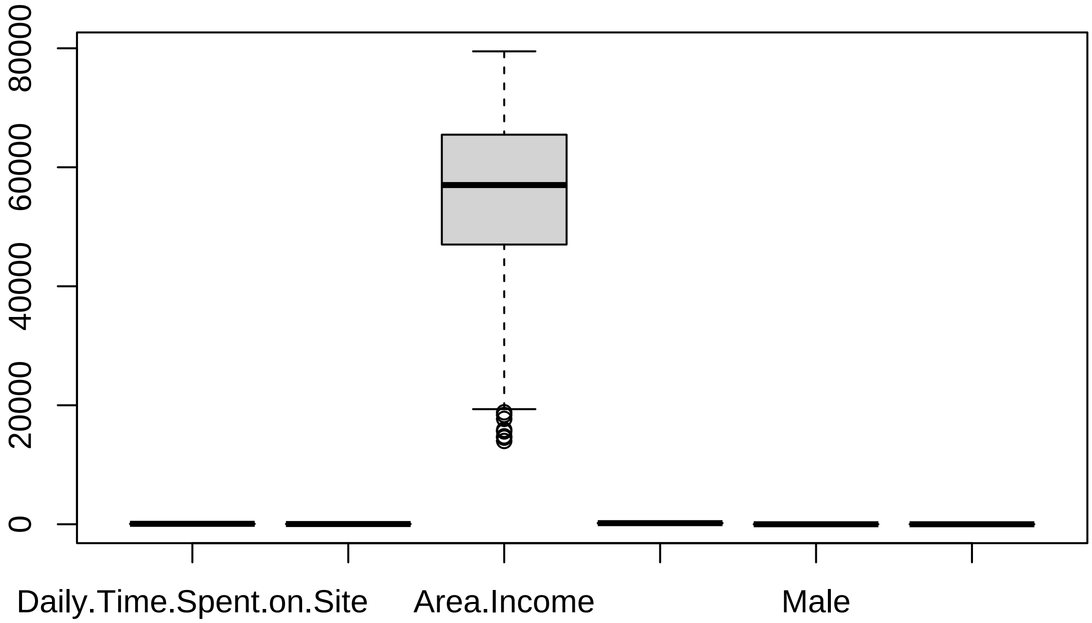
checking the data types

```
str(Adverts_numerical_cols)
```

```
## Classes 'data.table' and 'data.frame':   1000 obs. of  6 variables:
##  $ Daily.Time.Spent.on.Site: num  69 80.2 69.5 74.2 68.4 ...
##  $ Age                     : int  35 31 26 29 35 23 33 48 30 20 ...
##  $ Area.Income             : num  61834 68442 59786 54806 73890 ...
##  $ Daily.Internet.Usage    : num  256 194 236 246 226 ...
##  $ Male                    : int  0 1 0 1 0 1 0 1 1 1 ...
##  $ Clicked.on.Ad           : int  0 0 0 0 0 0 0 1 0 0 ...
##  - attr(*, ".internal.selfref")=<externalptr>
```

Plotting the outliers using boxplot

```
boxplot(Adverts_numerical_cols)
```



Having plotted the above, area income has the most outliers and we shall

plot the variable just to have a vizualization of its values

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

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

Lets check the Z scores of the outliers in all the variables

```
z_scores <- as.data.frame(sapply(numerical_cols, function(numerical_cols) (abs(numerical_cols-mean(nume
z_scores
```

```
##                          sapply(numerical_cols, function(numerical_cols) (abs(numerical_cols - mean(
## Daily.Time.Spent.on.Site
## Age
## Area.Income
## Daily.Internet.Usage
## Ad.Topic.Line
## City
## Male
## Country
## Timestamp
## Clicked.on.Ad
```

Having the above, we shall now remove outliers. Anything below -3 or above 3

will be deemed as an outlier and must be removed.

```
no_outliers <- z_scores[!rowSums(z_scores>3), ]
head(no_outliers)
```

```
## [1] NA NA NA NA NA NA
```

Lets apply the IQR to remove the outliers in the area income variable

```
Area_Income.IQR <- 65471-47032
Area_Income.IQR <-IQR(Adverts$`Area.Income`)
Area_Income.IQR
```

```
## [1] 18438.83
```

Since our data is now clean, we shall create a new variable to save the clean

data that we shall use for analysis

```
New_Ads <- subset(Adverts, Adverts$`Area.Income`> (47032 - 1.5*Area_Income.IQR) & Adverts$`Area.Income`
```

5) Exploratory Data Analysis

5.1) Univariate Analysis

5.1.1) Numerical Data

Lets get the descriptive statistics of the numerical variables

```
summary(numerical_cols)
```

```
##    Mode   FALSE    TRUE
## logical       4       6
```

Lets get the variance of the numerical variables

```
variance <- var(numerical_cols)
variance
```

```
## [1] 0.2666667
```

The descriptive summary we did above does not include the standard deviation

# of the numerical variables therefore, we shall create a function that will

aid in getting the standard deviation of all the numerical variables.

```
sd.function <- function(column) {
  standard.deviations <- sd(column)
  print(standard.deviations)
}
```

Lets get the standard deviation of daily time spent on site

```
sd.function(New_Ads$Daily.Time.Spent.on.Site)
```

```
## [1] 15.9005
```

Lets get the standard deviation of the age column

```
sd.function(New_Ads$Age)
```

```
## [1] 8.804716
```

Lets get the standard deviation of Area Income

```
sd.function(New_Ads$Area.Income)
```

```
## [1] 12961.5
```

Lets get the standard deviation of daily internet useage

```
sd.function(New_Ads$Daily.Internet.Usage)
```

```
## [1] 44.05386
```

5.2) Bivariate Analysis

5.2.1) Covariance

Lets check the covariance of the numerical data

```
cov(Adverts_numerical_cols)
```

```
##                          Daily.Time.Spent.on.Site           Age   Area.Income
## Daily.Time.Spent.on.Site              251.3370949 -4.617415e+01  6.613081e+04
## Age                                   -46.1741459  7.718611e+01 -2.152093e+04
## Area.Income                         66130.8109082 -2.152093e+04  1.799524e+08
## Daily.Internet.Usage                  360.9918827 -1.416348e+02  1.987625e+05
## Male                                   -0.1501864 -9.242142e-02  8.867509e+00
## Clicked.on.Ad                          -5.9331431  2.164665e+00 -3.195989e+03
##                          Daily.Internet.Usage        Male Clicked.on.Ad
## Daily.Time.Spent.on.Site         3.609919e+02 -0.15018639 -5.933143e+00
## Age                             -1.416348e+02 -0.09242142  2.164665e+00
## Area.Income                      1.987625e+05  8.86750903 -3.195989e+03
## Daily.Internet.Usage             1.927415e+03  0.61476667 -1.727409e+01
## Male                             6.147667e-01  0.24988889 -9.509510e-03
## Clicked.on.Ad                   -1.727409e+01 -0.00950951  2.502503e-01
```

There is a positive covariance between the following variables: 1.Area Income and Daily Time Spent on Site 2.Age and Clicking on the Advert. 3.Area Income and Daily Internet Usage. 4.Area Income and Male 5.Daily Internet Usage and Daily Time Spent on Site 6.Male and Daily Internet Usage 7.Clicked on Advert and Age

The rest of the variables exhibit negative covariance.

5.2.2) Correlation

Let's do the correlation of numerical variables

```
cor(Adverts_numerical_cols)
```

```
##                          Daily.Time.Spent.on.Site         Age  Area.Income
## Daily.Time.Spent.on.Site               1.00000000 -0.33151334  0.310954413
## Age                                   -0.33151334  1.00000000 -0.182604955
## Area.Income                            0.31095441 -0.18260496  1.000000000
## Daily.Internet.Usage                   0.51865848 -0.36720856  0.337495533
## Male                                  -0.01895085 -0.02104406  0.001322359
## Clicked.on.Ad                         -0.74811656  0.49253127 -0.476254628
##                          Daily.Internet.Usage         Male Clicked.on.Ad
## Daily.Time.Spent.on.Site           0.51865848 -0.018950855   -0.74811656
## Age                               -0.36720856 -0.021044064    0.49253127
## Area.Income                        0.33749553  0.001322359   -0.47625463
## Daily.Internet.Usage               1.00000000  0.028012326   -0.78653918
## Male                               0.02801233  1.000000000   -0.03802747
## Clicked.on.Ad                     -0.78653918 -0.038027466    1.00000000
```

Visualizing the correlation matrix using a Correlogram

But first we need to install the corrplot package and load the library.

```
#install.packages("corrplot")
```

Loading the corrplot library

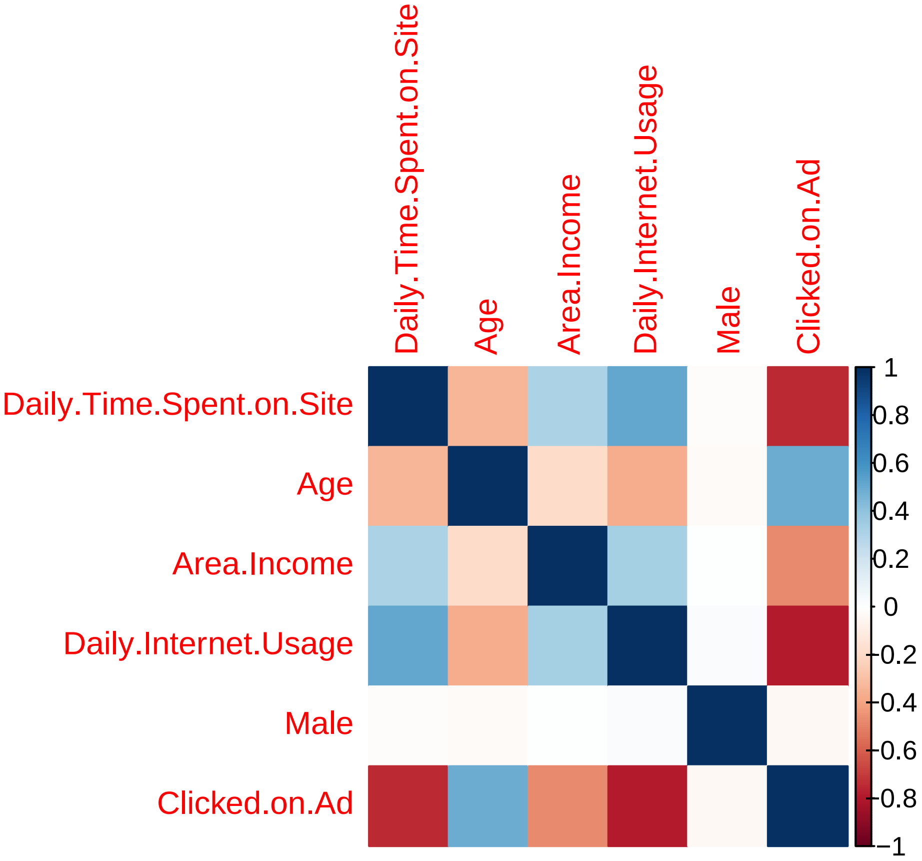
```
library(corrplot)
```

```
## corrplot 0.92 loaded
```

Visualizing the Correlogram

```
corr_ <- cor(Adverts_numerical_cols)

corrplot(corr_, method = 'color')
```



6) Observations

1) The data was clean and contained no outliers nor missing values.
2) Most individuals were between the ages of 19 and 61.
3) Most of the individuals spent around 32 to 91 minutes on the site.
4) The Average Area Income of the individuals was 55,000
5) The Daily Internet Usage ranged between 104 MBS and 270 MBS
6) Most individuals were from Lake Faith and West Ryan cities.
7) Most individuals were from Fiji and Chad.
8) The number of individuals who clicked on the advert and those who didn’t were equal at 500.

7) Conclusions

After completing the analysis, we concluded that the following features would help an individual who is more likely to click on the ads:

1) Daily Time Spent on Site. The higher the time the lower the chances of clicking.

2.Age- Older people have a higher chance of clicking on the ads.

3.Area Income- Individuals with lower income have a higher chance of clicking the ads.

4.Daily Internet Usage- There is a high chance of clicking on the ads of the daily internet usage is lower.

#8) Recommendations

1) There should be focus on old people as they are more likely to click on the ads.
2) As long as the daily internet usage is low, there will be a higher chance of an individual clicking on the ads.
3) People earning a low income are more likely to click on the ads therefore, they should also be a point of focus.
4) Individuals who spend a lot more time on the site should also be focused on as they are more likely to click on the ads.