Cryptography course advertisement

Whiterose

2022-05-30

Exploratory data analysis

Define the question

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.

The main aim is to help her identify which individuals are most likely to click on her ads.

defining the metric for success

Our success will be determined by the outcome

explaining the context

Cryptography is technique of securing information and communications through use of codes so that only those person for whom the information is intended can understand it and process it. Thus preventing unauthorized access to information. The prefix "crypt" means "hidden" and suffix graphy means "writing".

In Cryptography the techniques which are use to protect information are obtained from mathematical concepts and a set of rule based calculations known as algorithms to convert messages in ways that make it hard to decode it. These algorithms are used for cryptographic key generation, digital signing, verification to protect data privacy, web browsing on internet and to protect confidential transactions such as credit card and debit card transactions.

experimental design

1. installing packages
2. loading packages
3. reading and viewing the data
4. Data wrangling
5. Tidying the dataset
6. Univariate data analysis
7. multivariate data analysis

data source validation

installing needed packages

```
#install.packages('tidyverse')
#install.packages('moderndive')
#install.packages('skimr')
#install.packages('fivethirtyeight')
```

loading the packages

```
#library(tidyverse)
library(moderndive)
library(skimr)
library(dplyr)
```

```
##
## Attaching package: ’dplyr’

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

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

```
library(ggplot2)
library(readr)
library(tidyr)
library(stringr)
library(latexpdf)
```

loading our dataset

```
advertising <- read_csv("advertising.csv")
```

```
## Rows: 1000 Columns: 10
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (3): Ad Topic Line, City, Country
## dbl  (6): Daily Time Spent on Site, Age, Area Income, Daily Internet Usage, ...
## dttm (1): Timestamp
##
## i Use ‘spec()‘ to retrieve the full column specification for this data.
## i Specify the column types or set ‘show_col_types = FALSE‘ to quiet this message.
```

```
View(advertising)
```

The dataset consists 1000 observations and 10 variables describing these observations.

previewing our dataset

```
glimpse(advertising)
```

```
## Rows: 1,000
## Columns: 10
## $ ‘Daily Time Spent on Site‘ <dbl> 68.95, 80.23, 69.47, 74.15, 68.37, 59.99, 8~
## $ Age                        <dbl> 35, 31, 26, 29, 35, 23, 33, 48, 30, 20, 49,~
## $ ‘Area Income‘              <dbl> 61833.90, 68441.85, 59785.94, 54806.18, 738~
## $ ‘Daily Internet Usage‘     <dbl> 256.09, 193.77, 236.50, 245.89, 225.58, 226~
## $ ‘Ad Topic Line‘            <chr> "Cloned 5thgeneration orchestration", "Moni~
## $ City                       <chr> "Wrightburgh", "West Jodi", "Davidton", "We~
## $ Male                       <dbl> 0, 1, 0, 1, 0, 1, 0, 1, 1, 1, 0, 1, 1, 0, 0~
## $ Country                    <chr> "Tunisia", "Nauru", "San Marino", "Italy", ~
## $ Timestamp                  <dttm> 2016-03-27 00:53:11, 2016-04-04 01:39:02, ~
## $ ‘Clicked on Ad‘            <dbl> 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 1, 0, 1, 0, 1~
```

The outcome variable y is the clicked on ad. This variable indicates either True or False that an ad was clicked denoted by 0 for False and 1 for True. There are four explanatory variables: 1. Daily Time Spent on Site 2. Ad Topic Line - explaining the topic the ad displays 3. City 4. Country

There is also a datetime variable that would help in plotting the time-series

data wrangling

renaming columns

```
advertising <- advertising %>%
  rename(daily_time_on_site = 'Daily Time Spent on Site', age = Age, area_income = `Area Income`, daily_

head(advertising)
```

```
## # A tibble: 6 x 10
##   daily_time_on_site   age area_income daily_net_usage ad_topic      city   male
##                <dbl> <dbl>       <dbl>           <dbl> <chr>         <chr> <dbl>
## 1               69.0    35      61834.            256. Cloned 5thge~ Wrig~     0
## 2               80.2    31      68442.            194. Monitored na~ West~     1
## 3               69.5    26      59786.            236. Organic bott~ Davi~     0
## 4               74.2    29      54806.            246. Triple-buffe~ West~     1
## 5               68.4    35      73890.            226. Robust logis~ Sout~     0
## 6               60.0    23      59762.            227. Sharable cli~ Jami~     1
## # ... with 3 more variables: country <chr>, timestamp <dttm>, clicked_ad <dbl>
```

We had to rename the columns for easy analysis.

checking missing data

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

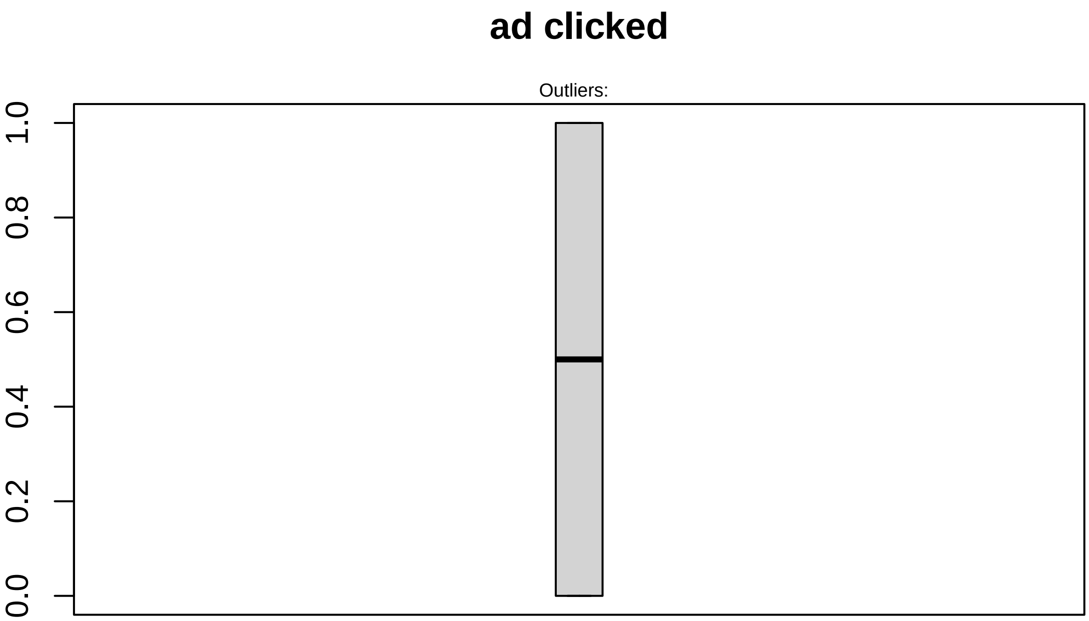
```
## daily_time_on_site                age        area_income    daily_net_usage
##                  0                  0                  0                  0
##           ad_topic               city               male            country
##                  0                  0                  0                  0
##          timestamp         clicked_ad
##                  0                  0
```

The dataset seems to be lacking missing values in every column.

now let us now check for outliers

outliers on our target variable

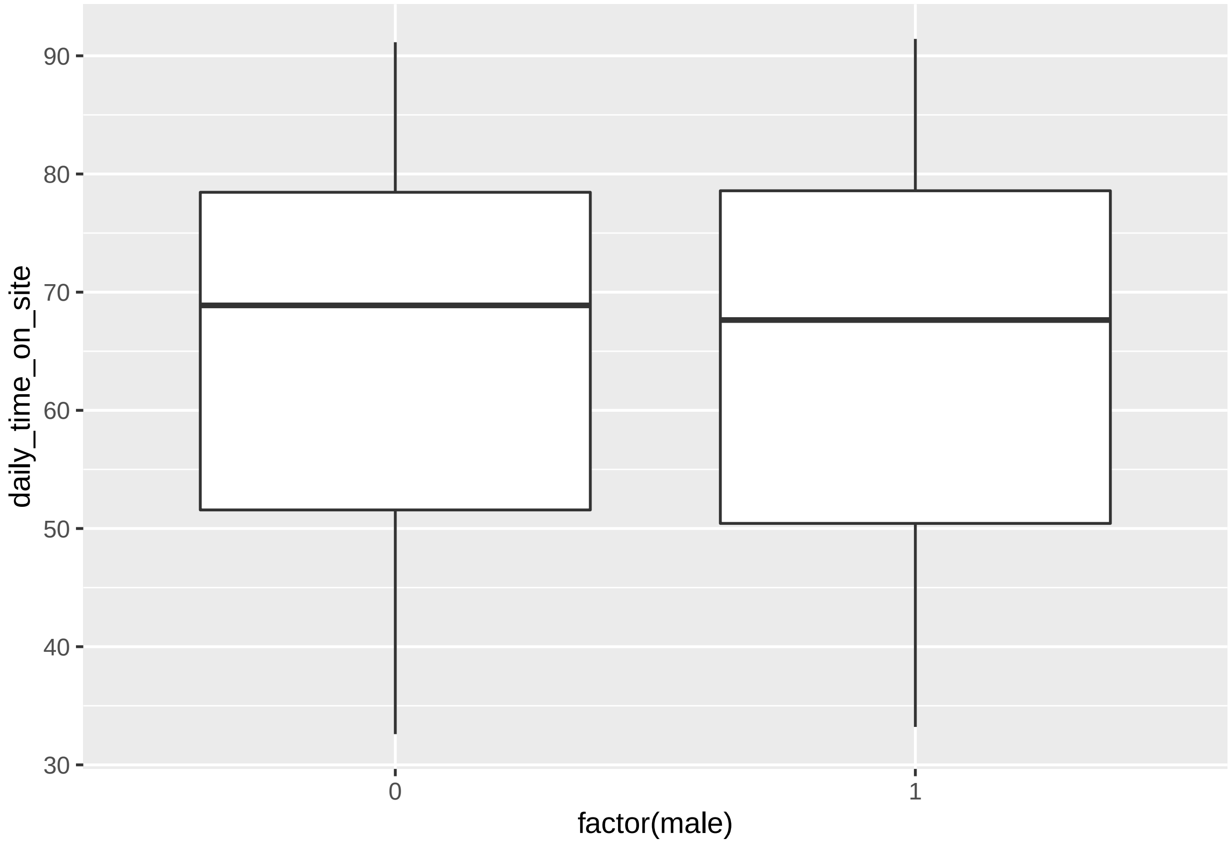
```
outlier_values <- boxplot.stats(advertising$clicked_ad)$out  # outlier values.
boxplot(advertising$clicked_ad, main="ad clicked", boxwex=0.1)
mtext(paste("Outliers: ", paste(outlier_values, collapse=", ")), cex=0.6)
```



our target variable has no outliers from the boxplot above.

outliers based on gender and daily time on site

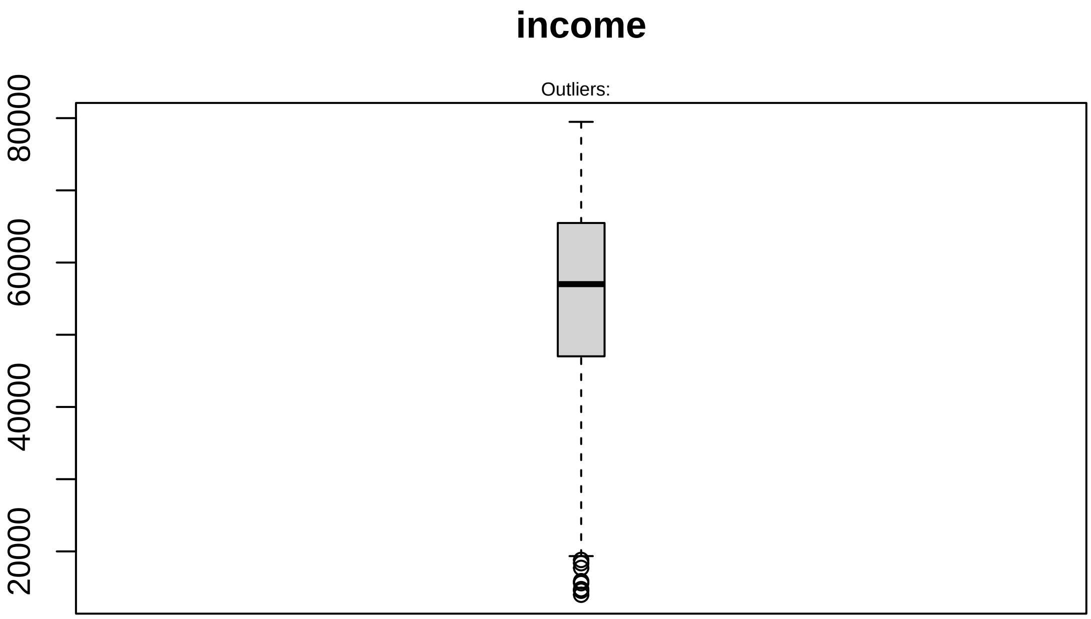
```
ggplot(data = advertising, mapping = aes(x = factor(male), y = daily_time_on_site)) +
  geom_boxplot()
```



The graph above shows no outliers for daily time spent on site.

checking outliers in area income

```
outlier_values <- boxplot.stats(advertising$clicked_ad)$out  # outlier values.
boxplot(advertising$area_income, main="income", boxwex=0.1)
mtext(paste("Outliers: ", paste(outlier_values, collapse=", ")), cex=0.6)
```



The area income variable has outliers for income below 20,000. let us preview these individuals

potential outliers based on area income and iqr criterion

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

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

previewing rows with outliers

```
out <- boxplot.stats(advertising$area_income)$out
out_ind <- which(advertising$area_income %in% c(out))
out_ind
```

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

taking a closer look

```
advertising[out_ind, ]
```

```
## # A tibble: 8 x 10
##   daily_time_on_site   age area_income daily_net_usage ad_topic      city   male
##                <dbl> <dbl>       <dbl>           <dbl> <chr>         <chr> <dbl>
## 1               49.9    39      17710.            160. Enhanced sys~ East~     1
## 2               57.9    30      18819.            167. Horizontal m~ Este~     0
## 3               64.6    45      15598.            159. Triple-buffe~ Isaa~     1
## 4               58.0    32      15879.            196. Total asynch~ Sand~     1
## 5               66.3    47      14548.            179. Optional ful~ Matt~     1
## 6               68.6    41      13996.            172. Exclusive di~ New ~     1
## 7               52.7    44      14776.            191. Persevering ~ New ~     0
## 8               62.8    36      18369.            232. Total cohere~ New ~     1
## # ... with 3 more variables: country <chr>, timestamp <dttm>, clicked_ad <dbl>
```

We can now replace the outliers with the median since they are not too far away and they are less.

replace outlier with median

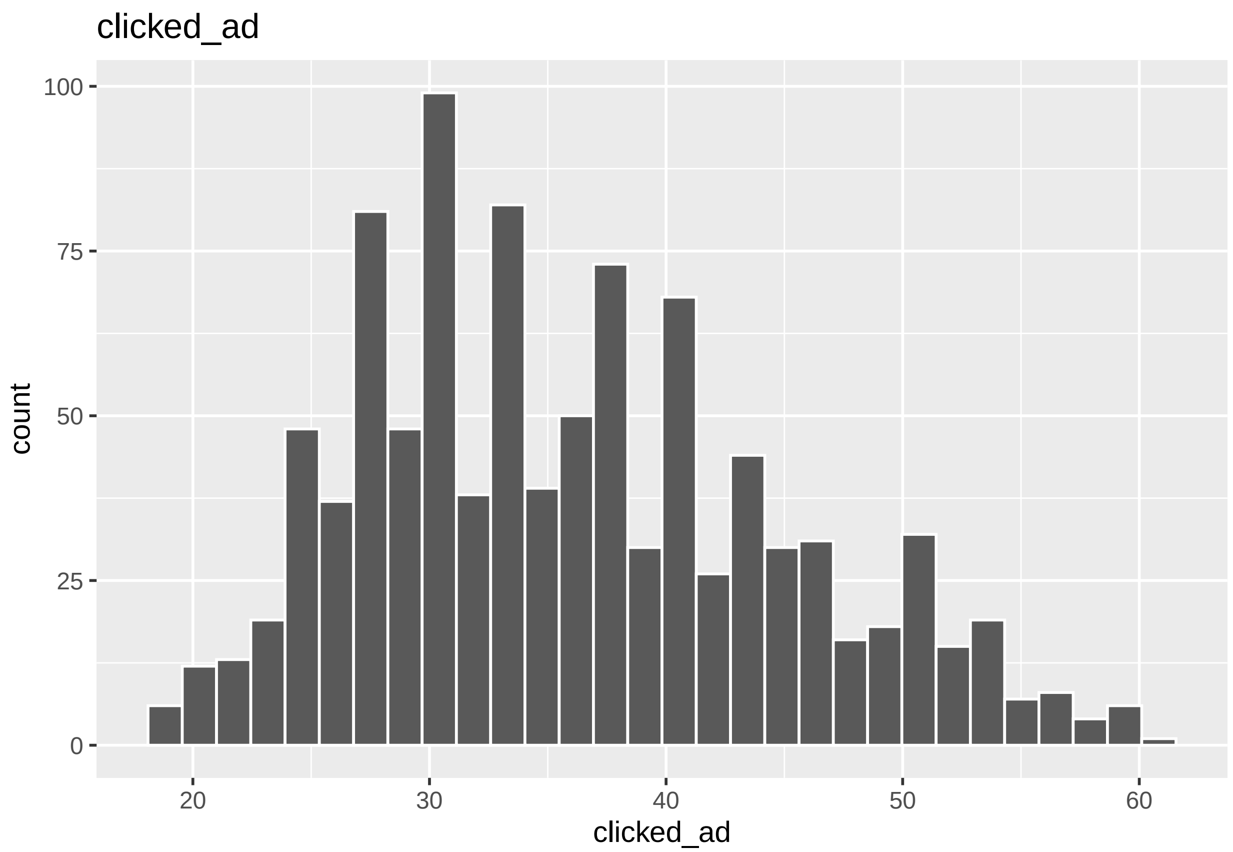
```
advertising$area_income[advertising$area_income %in% out_ind] <- median(advertising$area_income)
```

Exploratory Data Analysis

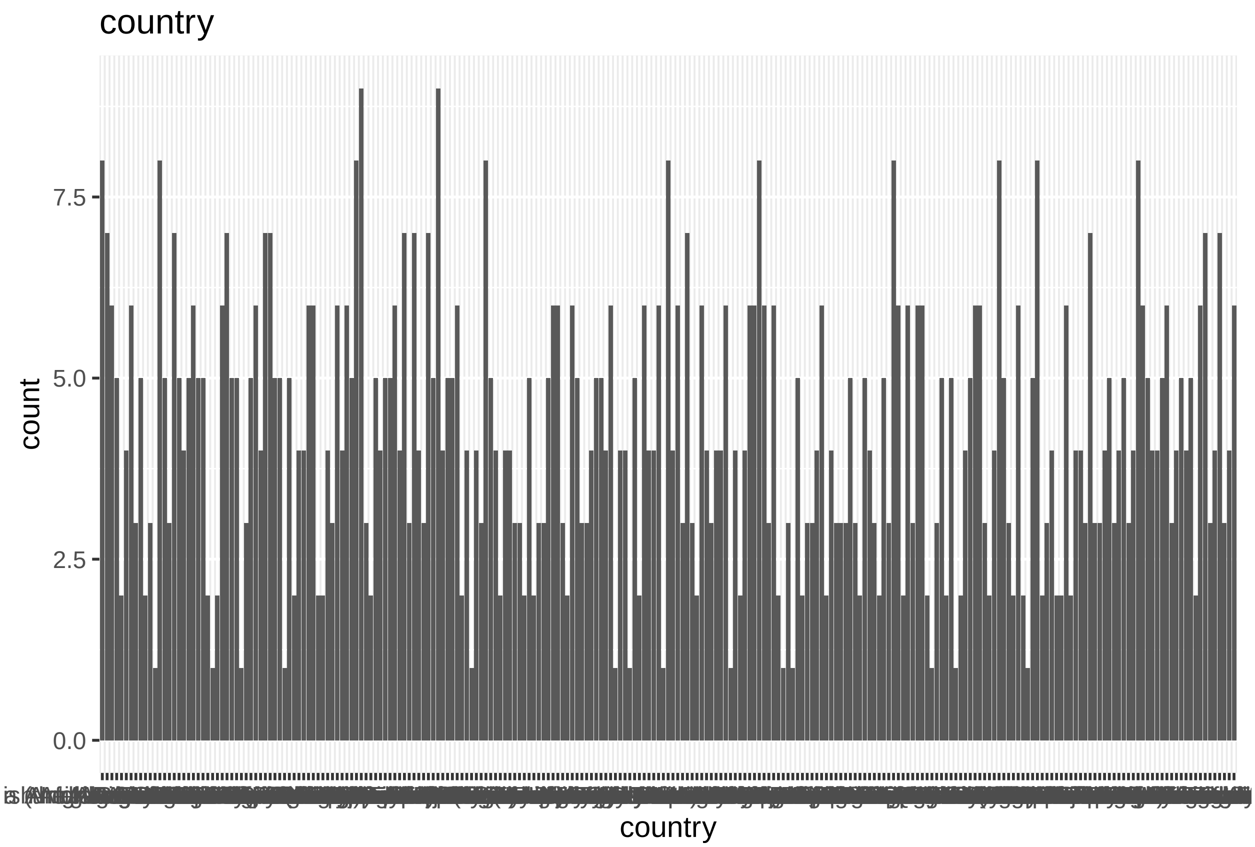
histogram of age

```
ggplot(advertising, aes(x = age)) +
  geom_histogram(color = "white") +
  labs(x = "clicked_ad", title = "clicked_ad")
```

```
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
```



```
# Barplot of country:
ggplot(advertising, aes(x = country)) +
  geom_bar() +
  labs(x = "country", title = "country")
```



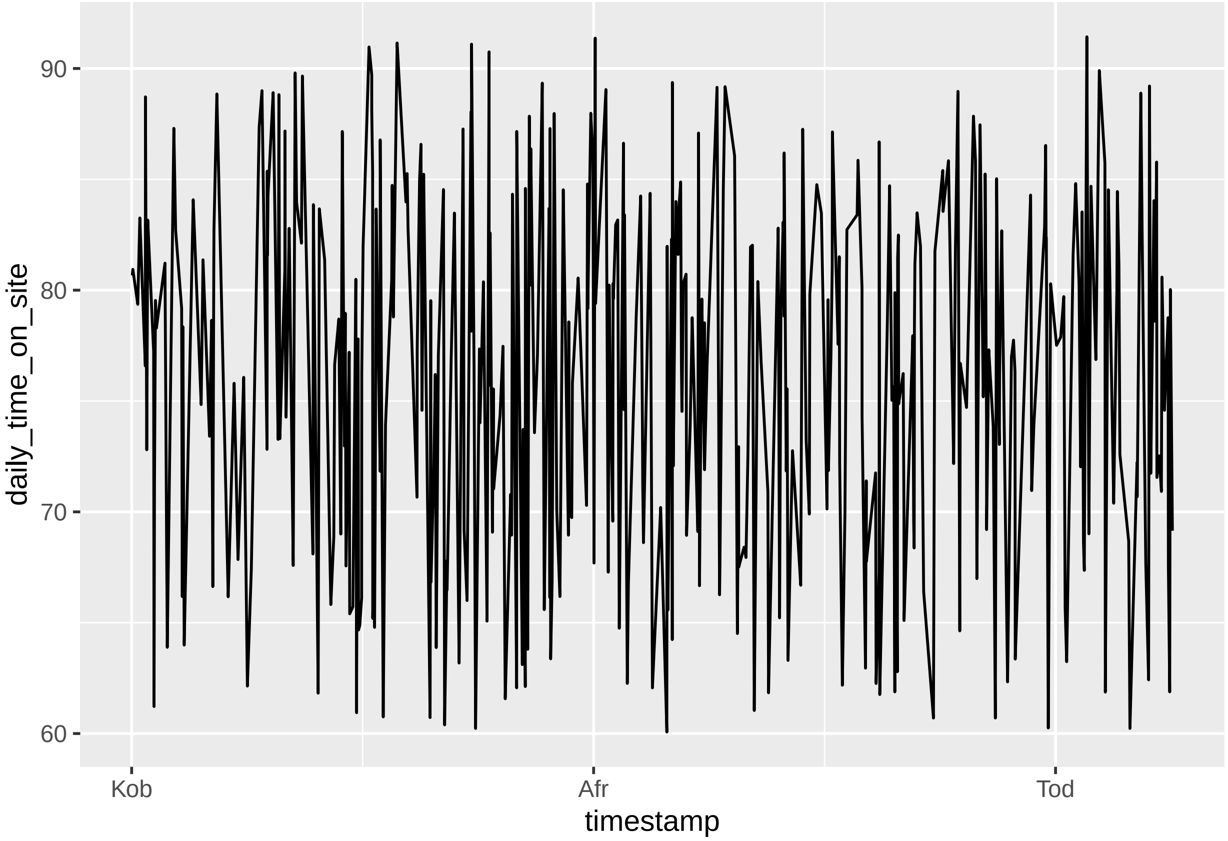
new dataframe for time above 1 hour on site

```r
above_an_hour <- advertising %>%
  filter(daily_time_on_site >= 60 & age >= 30)
above_an_hour
```

```
## # A tibble: 422 x 10
##    daily_time_on_site   age area_income daily_net_usage ad_topic     city   male
##                 <dbl> <dbl>       <dbl>           <dbl> <chr>        <chr> <dbl>
##  1               69.0    35      61834.            256. Cloned 5thg~ Wrig~     0
##  2               80.2    31      68442.            194. Monitored n~ West~     1
##  3               68.4    35      73890.            226. Robust logi~ Sout~     0
##  4               88.9    33      53853.            208. Enhanced de~ Bran~     0
##  5               66      48      24593.            132. Reactive lo~ Port~     1
##  6               74.5    30      68862             222. Configurabl~ West~     1
##  7               83.1    37      62491.            231. Team-orient~ East~     1
##  8               69.6    48      51637.            113. Centralized~ West~     1
##  9               82.0    41      71511.            188. Intuitive d~ Prui~     0
## 10               74.6    40      23822.            136. Advanced 24~ Mill~     1
## # ... with 412 more rows, and 3 more variables: country <chr>,
## #   timestamp <dttm>, clicked_ad <dbl>
```

line graph for above_an_hour

```
ggplot(data = above_an_hour,
       mapping = aes(x = timestamp, y = daily_time_on_site)) +
  geom_line()
```



People who are 30 years and above have an inconsistent internet usage when mapped to usage above 1 hour. let us check for people between 18 an 25 years old.

internet usage for 18-25 years old

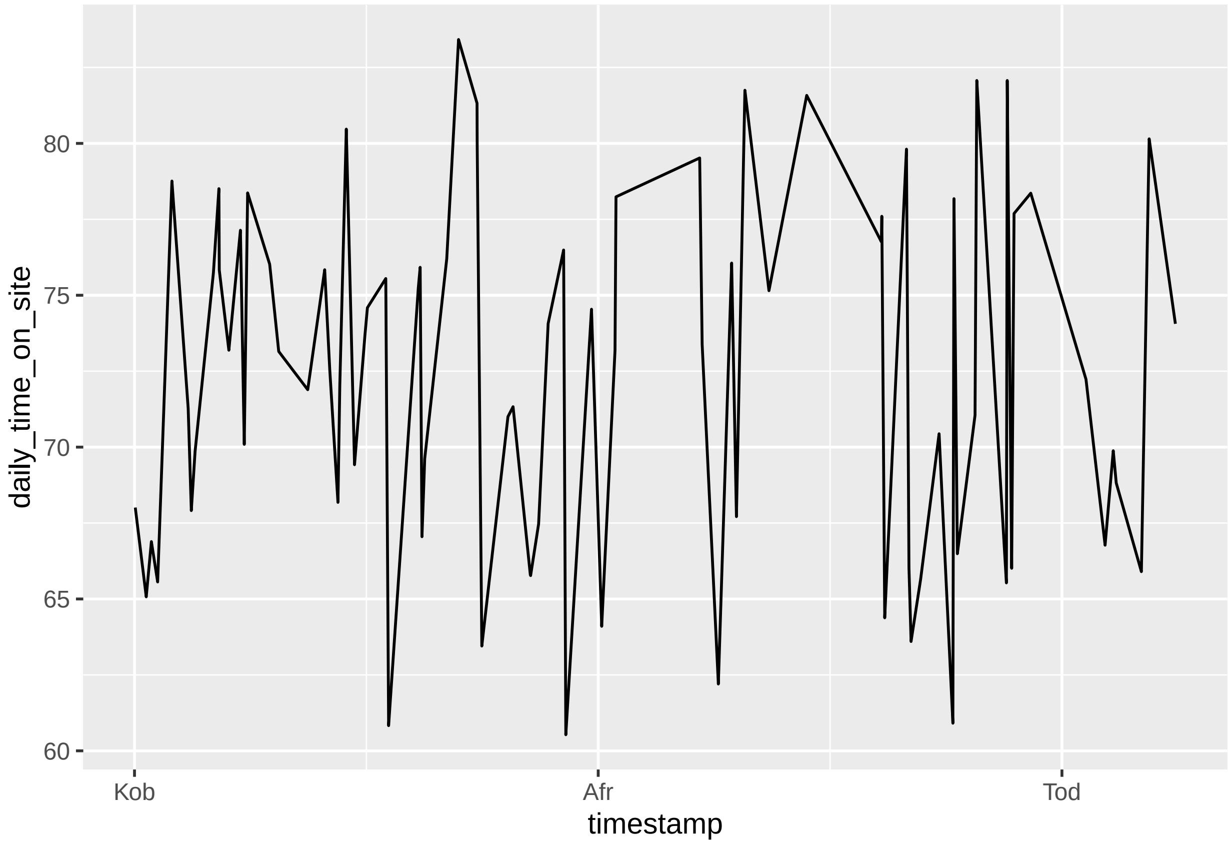
```
young_adult <- advertising %>%
  filter(daily_time_on_site >= 60 & age >= 18 & age <= 25)
young_adult
```

```
## # A tibble: 82 x 10
##    daily_time_on_site   age area_income daily_net_usage ad_topic     city   male
##                 <dbl> <dbl>       <dbl>           <dbl> <chr>        <chr> <dbl>
##  1               69.9    20      55642.            184. Mandatory h~ Rami~     1
##  2               79.5    24      51740.            214. Synergistic~ Nort~     0
##  3               63.4    23      52182.            141. Persistent ~ New ~     1
##  4                 76.0    22      46180.            210. Business-fo~ West~     0
##  5                 80.5    25      57520.            205. Reduced glo~ Jame~     0
##  6                 69.6    20      50984.            202. Business-fo~ New ~     1
##  7                 73.2    23      61526.            197. Organized s~ Holl~     1
##  8                 75.7    25      61006.            215. Ergonomic m~ New ~     1
##  9                 64.1    22      60466.            216. Seamless ob~ East~     0
## 10                 63.6    23      51865.            235. Centralized~ Youn~     1
## # ... with 72 more rows, and 3 more variables: country <chr>, timestamp <dttm>,
## #   clicked_ad <dbl>
```

let us now visualize them in a line plot

lineplot of young_adult

```
ggplot(data = young_adult,
       mapping = aes(x = timestamp, y = daily_time_on_site)) +
  geom_line()
```



From the graph above, we can easily predict people who use internet daily for the ages between 18 and 25 unlike for people aged 30 and above. However, what if we only include the male who actually clicked an ad.

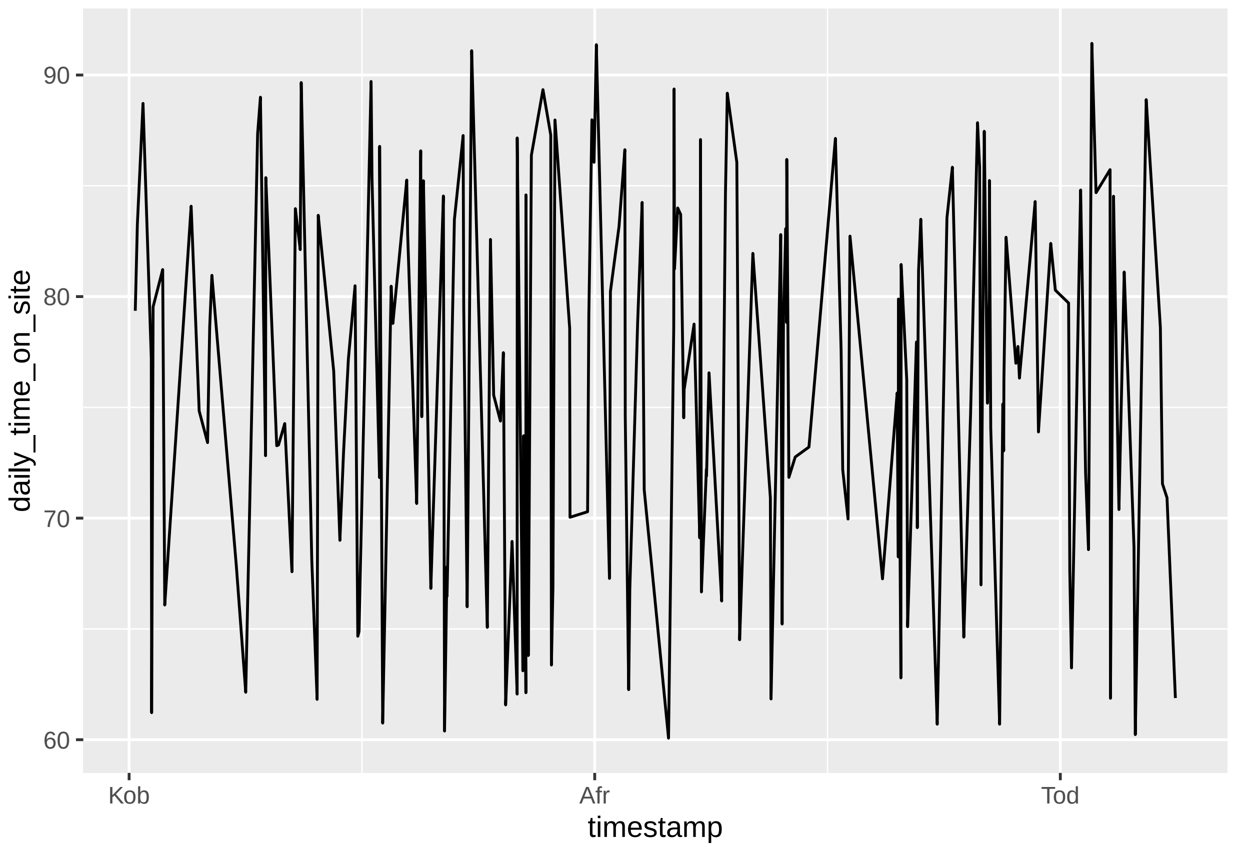
multiple conditions

```
many <- advertising %>%
  filter(daily_time_on_site >= 60 & age >= 30, male==1)
many
```

```
## # A tibble: 194 x 10
##    daily_time_on_site   age area_income daily_net_usage ad_topic     city   male
##                 <dbl> <dbl>       <dbl>           <dbl> <chr>        <chr> <dbl>
##  1               80.2    31      68442.            194. Monitored n~ West~     1
##  2               66      48      24593.            132. Reactive lo~ Port~     1
##  3               74.5    30      68862             222. Configurabl~ West~     1
##  4               83.1    37      62491.            231. Team-orient~ East~     1
##  5               69.6    48      51637.            113. Centralized~ West~     1
##  6               74.6    40      23822.            136. Advanced 24~ Mill~     1
##  7               77.2    30      64802.            224. Object-base~ Port~     1
##  8               84.6    35      60016.            227. Streamlined~ Lake~     1
##  9               87.3    36      61629.            210. Future-proo~ Pame~     1
## 10               67.6    35      51473.            267. Programmabl~ Phel~     1
## # ... with 184 more rows, and 3 more variables: country <chr>,
## #   timestamp <dttm>, clicked_ad <dbl>
```

lineplot of many

```
ggplot(data = many,
       mapping = aes(x = timestamp, y = daily_time_on_site)) +
  geom_line()
```



When we look at the male who access the internet, we still do not get a clear picture of internet usage.

let us now group by only male

determining number male and female

```
male_freq <- advertising %>%
  group_by(male) %>%
  summarize(male_count = n())
male_freq
```

```
## # A tibble: 2 x 2
##    male male_count
##   <dbl>      <int>
## 1     0        519
## 2     1        481
```

The 1 represents male while 0 represents the opposite. There are 481 male and 519 female.

```
male_click <- advertising %>%
  group_by(male) %>%
  filter(clicked_ad == 1) %>%
  summarize(n())
male_click
```

```
## # A tibble: 2 x 2
##    male `n()`
##   <dbl> <int>
## 1     0   269
## 2     1   231
```

Women are the ones who clicked an ad mostly.

country count

```
country_freq <- advertising %>%
  group_by(country) %>%
  summarize(country_count = n())
country_freq
```

```
## # A tibble: 237 x 2
##    country                                        country_count
##    <chr>                                                  <int>
##  1 Afghanistan                                                8
##  2 Albania                                                    7
##  3 Algeria                                                    6
##  4 American Samoa                                             5
##  5 Andorra                                                    2
##  6 Angola                                                     4
##  7 Anguilla                                                   6
##  8 Antarctica (the territory South of 60 deg S)               3
##  9 Antigua and Barbuda                                        5
## 10 Argentina                                                  2
## # ... with 227 more rows
```

The above shows the number of times a contry is appearing. Let us look at the home country of the entrepreneur.

Kenya

```
kenyan <- advertising %>%
  filter(country == 'Kenya' )
kenyan
```

```
## # A tibble: 4 x 10
##   daily_time_on_site   age area_income daily_net_usage ad_topic      city   male
##                <dbl> <dbl>       <dbl>           <dbl> <chr>         <chr> <dbl>
## 1                 37    48      36782.            158. Function-bas~ Jona~     1
## 2               60.2    35      43314.            107. Balanced asy~ New ~     0
## 3               67.6    31      62318.            125. Seamless com~ Mich~     0
## 4               49.4    49      44304.            120. Inverse stab~ Lake~     0
## # ... with 3 more variables: country <chr>, timestamp <dttm>, clicked_ad <dbl>
```

In Kenya only four people were recorded. Out of the four, only 1 was a male.

let us now look at the people who actually clicked an ad

```
ad <- advertising %>%
  filter(clicked_ad == 1 )%>%
  summarize(ad_count = n())
ad
```

```
## # A tibble: 1 x 1
##   ad_count
##      <int>
## 1      500
```

A total of 500 people clicked an ad. This is half the number of observations. let us view for people who are online over an hour, have an income of 30000 and above, age 18+ and clicked an ad

```
ad1 <- advertising %>%
  filter(clicked_ad == 1 , age >= 18, area_income >= 30000, daily_time_on_site >= 60)

ad1
```

```
## # A tibble: 123 x 10
##    daily_time_on_site   age area_income daily_net_usage ad_topic     city   male
##                 <dbl> <dbl>       <dbl>           <dbl> <chr>        <chr> <dbl>
##  1               69.6    48      51637.            113. Centralized~ West~     1
##  2               63.4    23      52182.            141. Persistent ~ New ~     1
##  3               70.2    34      32709.            119. Open-archit~ Palm~     0
##  4               62.3    53      56771.            125. Profound st~ West~     1
##  5               62.3    47      62723.            119. Team-orient~ Aman~     0
##  6               65.2    36      75255.            151. Cross-group~ Garc~     0
##  7               63.9    40      51317.            105. Synchronize~ Jens~     0
##  8               78.5    34      32537.            132. Synergized ~ Nort~     0
##  9               68.9    54      30726.            139. Streamlined~ East~     0
## 10               69.9    43      71393.            138. Down-sized ~ Chri~     0
## # ... with 113 more rows, and 3 more variables: country <chr>,
## #   timestamp <dttm>, clicked_ad <dbl>
```

and what is the count?

count ad1

```
ad1 %>%
  summarise(n())
```

```
## # A tibble: 1 x 1
##   ‘n()‘
##   <int>
## 1   123
```

Out of 500 who clicked an ad, 123 have 18+ years and have an income above 30,000 and stay over an hour on the site.

summary statistics

```
income <- advertising %>%
  summarize(
    min = min(area_income),
    q1 = quantile(area_income, 0.25),
    median = quantile(area_income, 0.5),
    q3 = quantile(area_income, 0.75),
    max = max(area_income),
    mean = mean(area_income),
    sd = sd(area_income),
    missing = sum(is.na(area_income))
  )
income
```

```
## # A tibble: 1 x 8
##      min     q1 median     q3    max   mean     sd missing
##    <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl>   <int>
## 1 13996. 47032. 57012. 65471. 79485. 55000. 13415.       0
```

conclusion

The above analysis shows that the people most likely to click an ad are below the age of 30 and have a lower income than the mean of 55,000. the persons stay online for over an hour and are mostly men.

recommendations

I recommend for use of SEO search engine optimization techniques to increase website availability. also to use personalized ads, those tailored to the person viewing the site.