Project -1

Problem 2 – EDA on NYT Data Set

CSE – 587

Instructor – Bina Ramammurthy

Submitted By:-

NALIN KUMAR

Person - 50170479

Problems

1. Create a new variable, age_group, that categorizes users as "<18", "18-24", "25-34", "35-44", "45-54", "55-64", and "65+".
2. For a single day:

- Plot the distributions of number impressions and clickthrough-rate (CTR=# clicks/# impressions) for these six age categories.
- Define a new variable to segment or categorize users based on their click behavior.
- Explore the data and make visual and quantitative comparisons across user segments/demographics (<18-year-old males versus < 18-year-old females or logged-in versus not, for example).
- Create metrics/measurements/statistics that summarize the data. Examples of potential metrics include CTR, quantiles, mean, median, variance, and max, and these can be calculated across the various user segments. Be selective. Think about what will be important to track over time—what will compress the data, but still capture user behavior.

3. Now extend your analysis across days. Visualize some metrics and distributions over time.

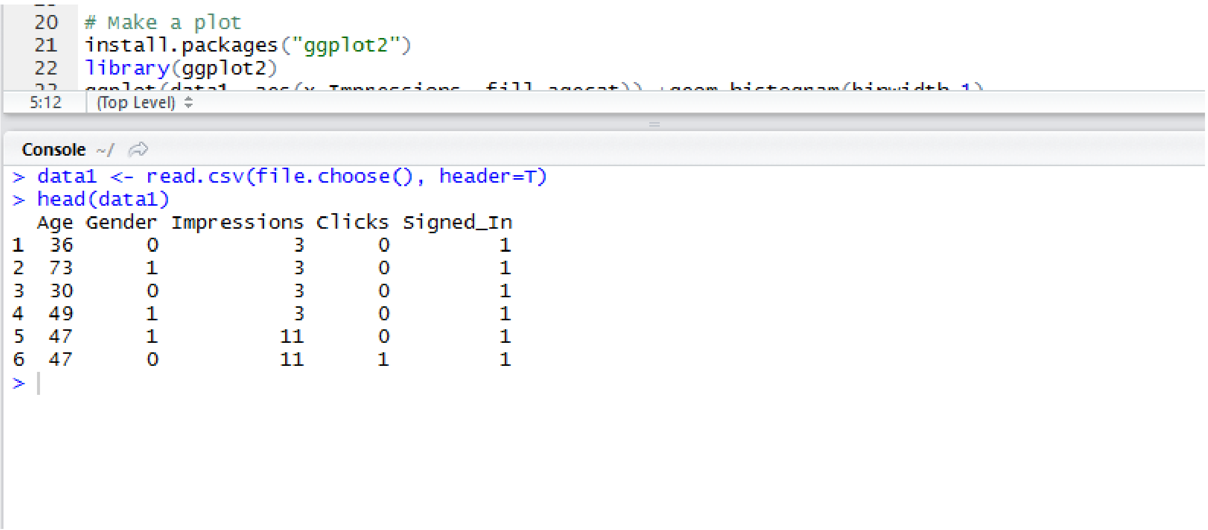
Solution

The data is manipulated in two parts. In the first part, a single file is read and EDA is performed on that single file data and saved into NYTP2nalinkum.R. In the next part, multiple files are read at once, EDA is performed on the combined data in all the files and the results are plotted in NYTP2Exnalinkum.R. These two analysis based on the questions in the textbook (Pages 38-39) are described as follows:-

NYTP2nalinkum.R

- **EDA on single file Dataset**
  Initially, we read the data from nyt1.csv file and analyse the head of data i.e. the first 6 rows of data as can be seen in the following image

  data1 <- read.csv(file.choose(), header=T)
  head(data1)

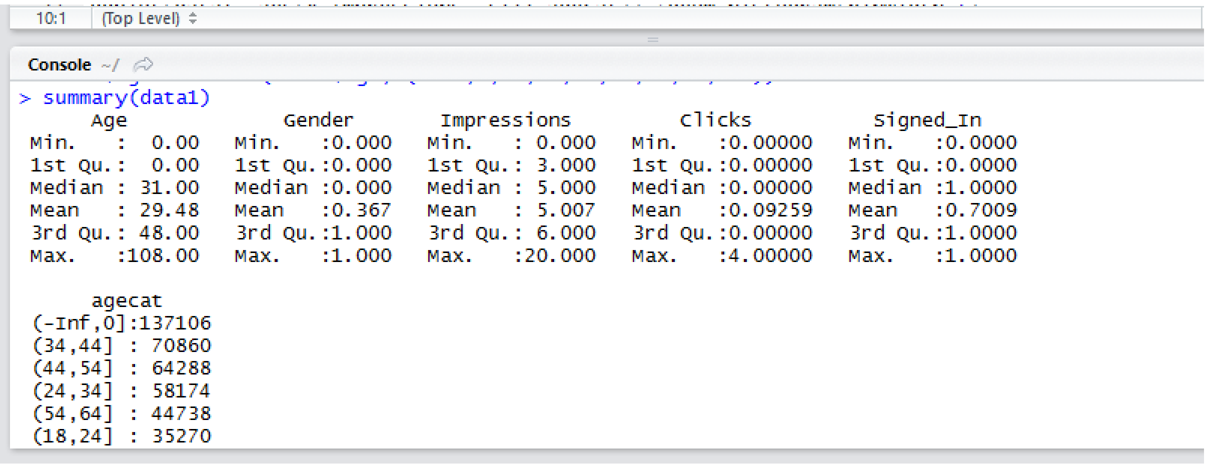


```
# Make a plot
install.packages("ggplot2")
library(ggplot2)
ggplot(data1, aes(x=Impressions, fill=agecat)) +geom_histogram(binwidth=1)
```

```
> data1 <- read.csv(file.choose(), header=T)
> head(data1)
  Age Gender Impressions Clicks Signed_In
1  36      0           3      0         1
2  73      1           3      0         1
3  30      0           3      0         1
4  49      1           3      0         1
5  47      1          11      0         1
6  47      0          11      1         1
> 
```

- Following this, we analyse the summary of columns such as Age, Gender , Clicks, Impressions etc. We also added a new column to the above data called as agecat which categorizes the whole data into different age categories. The parameters Min, Median, Max, Mean etc. on the above columns are visualized as follows

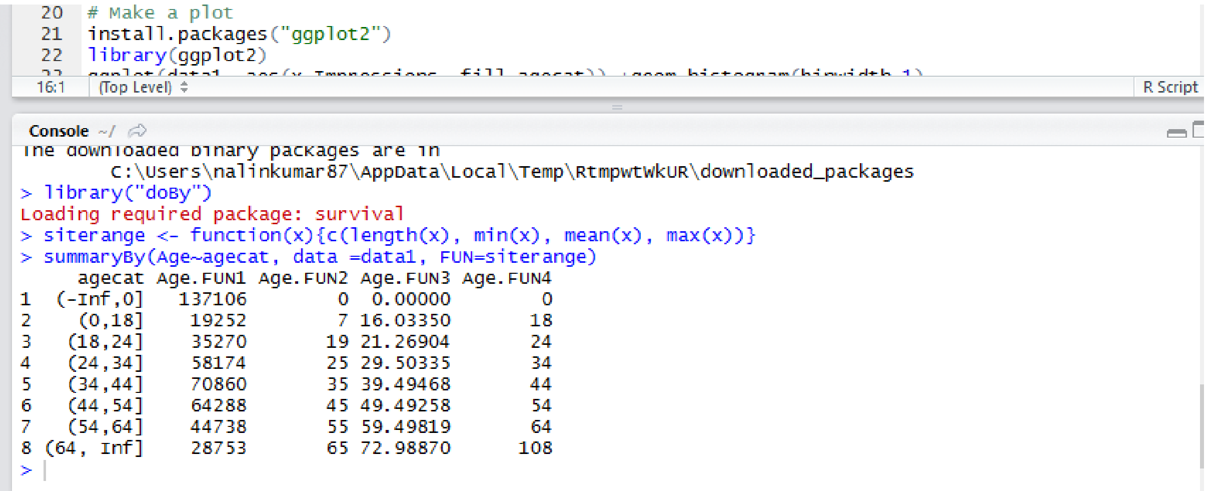
data1$agecat <-cut(data1$Age,c(-Inf,0,18,24,34,44,54,64,Inf))
summary(data1)



```
> summary(data1)
      Age             Gender        Impressions         Clicks          Signed_In     
 Min.   :  0.00   Min.   :0.000   Min.   : 0.000   Min.   :0.00000   Min.   :0.0000  
 1st Qu.:  0.00   1st Qu.:0.000   1st Qu.: 3.000   1st Qu.:0.00000   1st Qu.:0.0000  
 Median : 31.00   Median :0.000   Median : 5.000   Median :0.00000   Median :1.0000  
 Mean   : 29.48   Mean   :0.367   Mean   : 5.007   Mean   :0.09259   Mean   :0.7009  
 3rd Qu.: 48.00   3rd Qu.:1.000   3rd Qu.: 6.000   3rd Qu.:0.00000   3rd Qu.:1.0000  
 Max.   :108.00   Max.   :1.000   Max.   :20.000   Max.   :4.00000   Max.   :1.0000  

      agecat      
 (-Inf,0]:137106  
 (34,44] : 70860  
 (44,54] : 64288  
 (24,34] : 58174  
 (54,64] : 44738  
 (18,24] : 35270  
```

- In the next part, we use a package called as doBy and analyse the summary based on agecat and different attributes of Age column such as length, min, max and mean using a function siterange(x) as follows

install.packages("doBy")
library("doBy")
siterange <- function(x){c(length(x), min(x), mean(x), max(x))}
summaryBy(Age~agecat, data =data1, FUN=siterange)

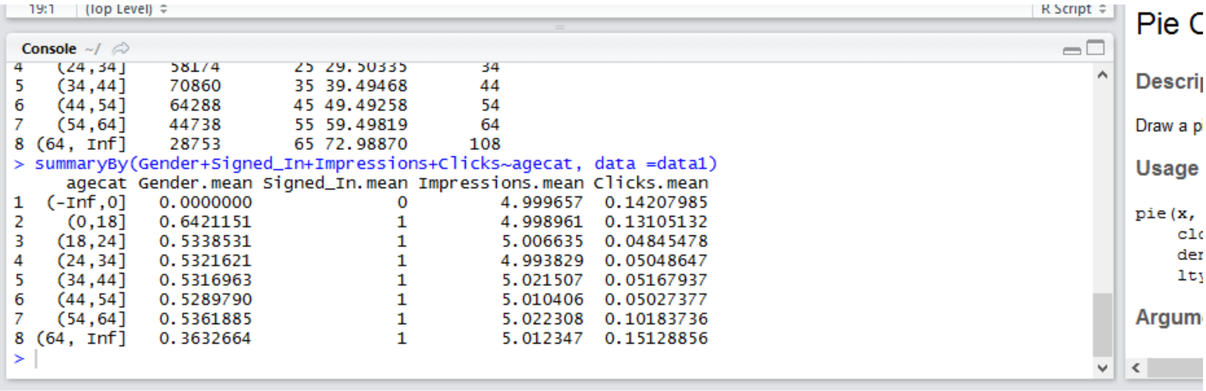


```
# Make a plot
install.packages("ggplot2")
library(ggplot2)
ggplot(data1, aes(x=Impressions, fill=agecat)) +geom_histogram(binwidth=1)
```

```
The downloaded binary packages are in
        C:\Users\nalinkumar87\AppData\Local\Temp\RtmpwtWkUR\downloaded_packages
> library("doBy")
Loading required package: survival
> siterange <- function(x){c(length(x), min(x), mean(x), max(x))}
> summaryBy(Age~agecat, data =data1, FUN=siterange)
    agecat Age.FUN1 Age.FUN2 Age.FUN3 Age.FUN4
1 (-Inf,0]   137106        0  0.00000        0
2   (0,18]    19252        7 16.03350       18
3  (18,24]    35270       19 21.26904       24
4  (24,34]    58174       25 29.50335       34
5  (34,44]    70860       35 39.49468       44
6  (44,54]    64288       45 49.49258       54
7  (54,64]    44738       55 59.49819       64
8 (64, Inf]   28753       65 72.98870      108
> 
```

- Following this, we analyse the summary using the same library doBy based on agecat and different columns such as Gender, Signed_In, Clicks, Impressions as follows

summaryBy(Gender+Signed_In+Impressions+Clicks~agecat, data =data1)

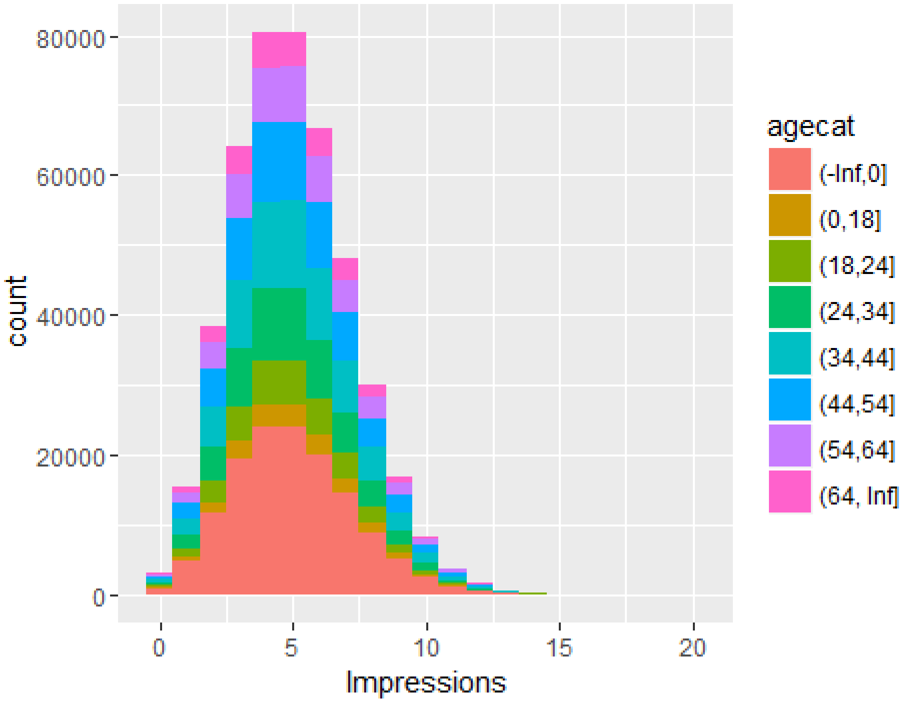


```
4    (24,34]     58174        25 29.50335        34
5    (34,44]     70860        35 39.49468        44
6    (44,54]     64288        45 49.49258        54
7    (54,64]     44738        55 59.49819        64
8  (64, Inf]     28753        65 72.98870       108
> summaryBy(Gender+Signed_In+Impressions+Clicks~agecat, data =data1)
      agecat Gender.mean Signed_In.mean Impressions.mean Clicks.mean
1  (-Inf,0]   0.0000000              0         4.999657  0.14207985
2     (0,18]   0.6421151              1         4.998961  0.13105132
3    (18,24]   0.5338531              1         5.006635  0.04845478
4    (24,34]   0.5321621              1         4.993829  0.05048647
5    (34,44]   0.5316963              1         5.021507  0.05167937
6    (44,54]   0.5289790              1         5.010406  0.05027377
7    (54,64]   0.5361885              1         5.022308  0.10183736
8  (64, Inf]   0.3632664              1         5.012347  0.15128856
>
```

```
pie(x,
    clc
    der
    lty
```

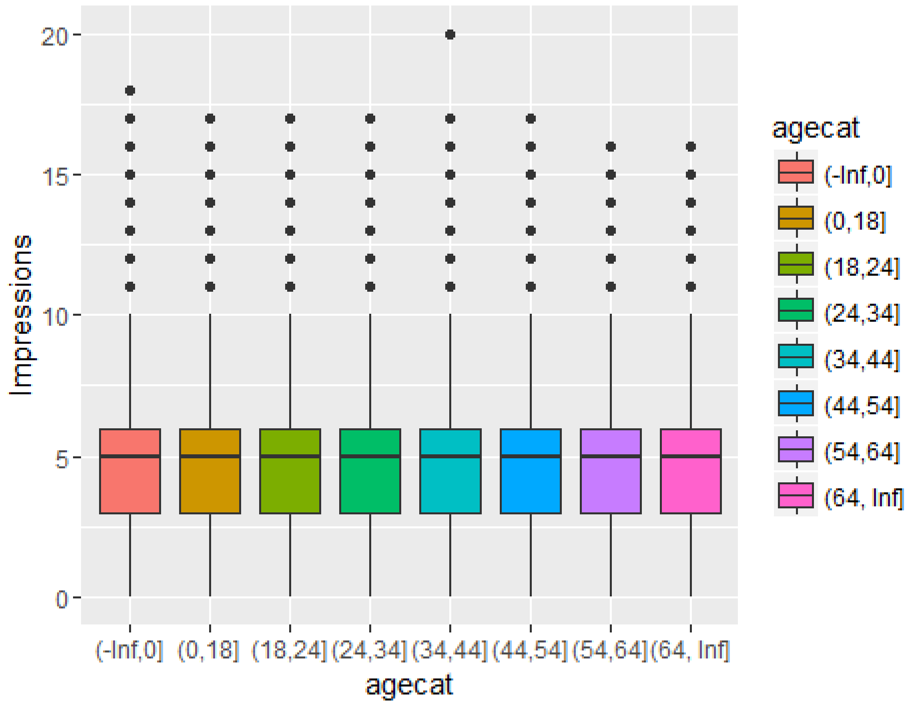
- In the next part, we use a package called as ggplot2, which is used to graphical analysis on the data. We plot Impressions on x-axis and their respective counts and categorize the whole data into different age categories using different colours in the histogram below. This plot tells us how many different impressions are present and in what age categories.

  install.packages("ggplot2")
  library(ggplot2)
  ggplot(data1, aes(x=Impressions, fill=agecat)) +geom_histogram(binwidth=1)



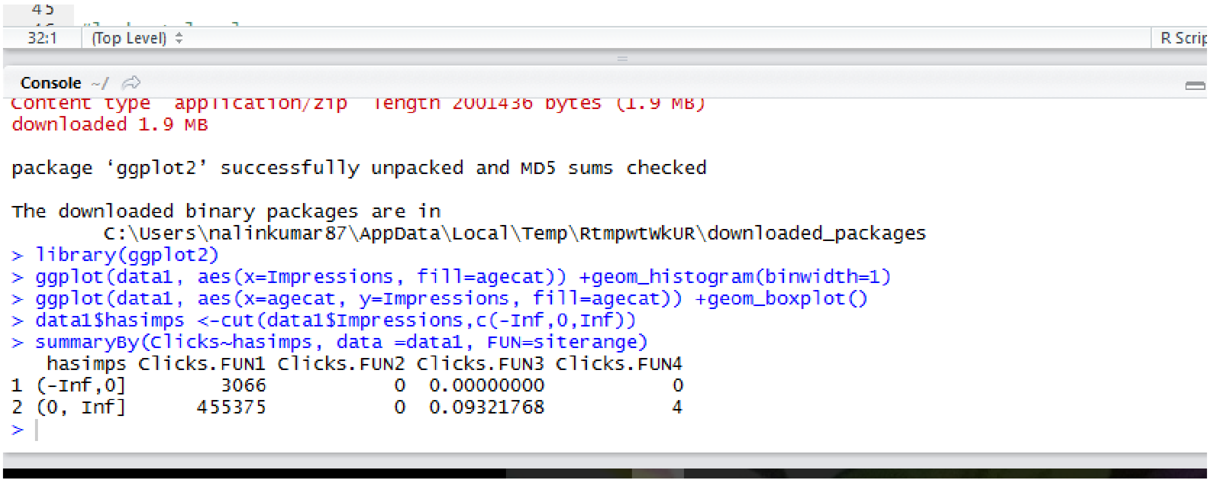
- Next we construct a box-plot of age categories on x-axis and Impressions on y-axis using the same library ggplot2. This plot tells us what all impressions are present in all of the age categories plotted on the x-axis. Box-plot was very helpful in this regard in describing this correlation in a clear and precise manner

  ggplot(data1, aes(x=agecat, y=Impressions, fill=agecat)) +geom_boxplot()



- Subsequently, we categorize the data into two parts based on whether Impressions has a positive value or not. We analyse the summary on this divided data based on different attributes of Clicks column using the same function siterange()

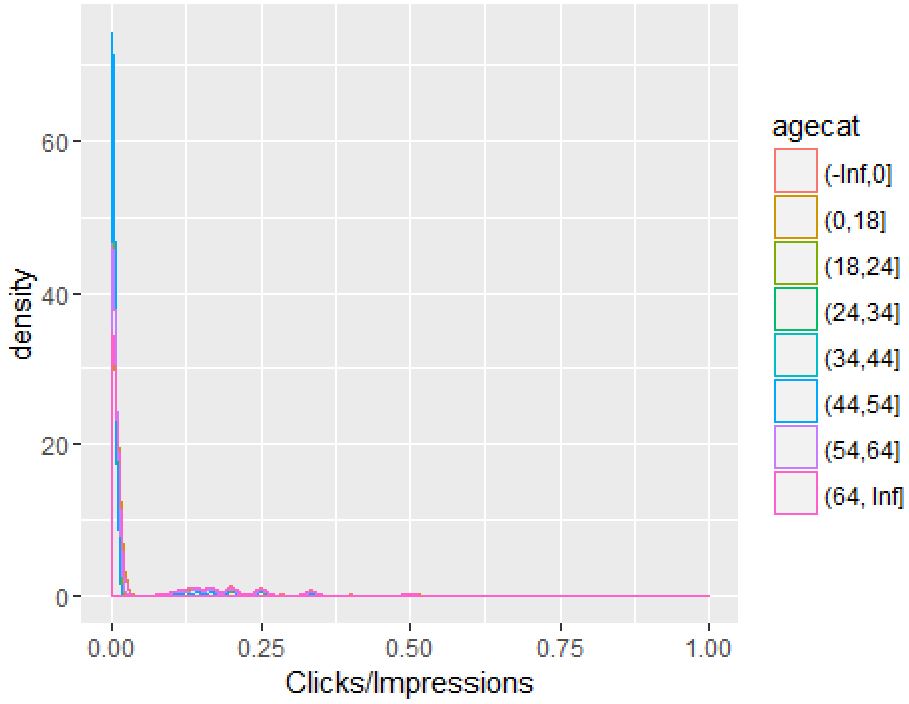
  data1$hasimps <-cut(data1$Impressions,c(-Inf,0,Inf))
  summaryBy(Clicks~hasimps, data =data1, FUN=siterange)



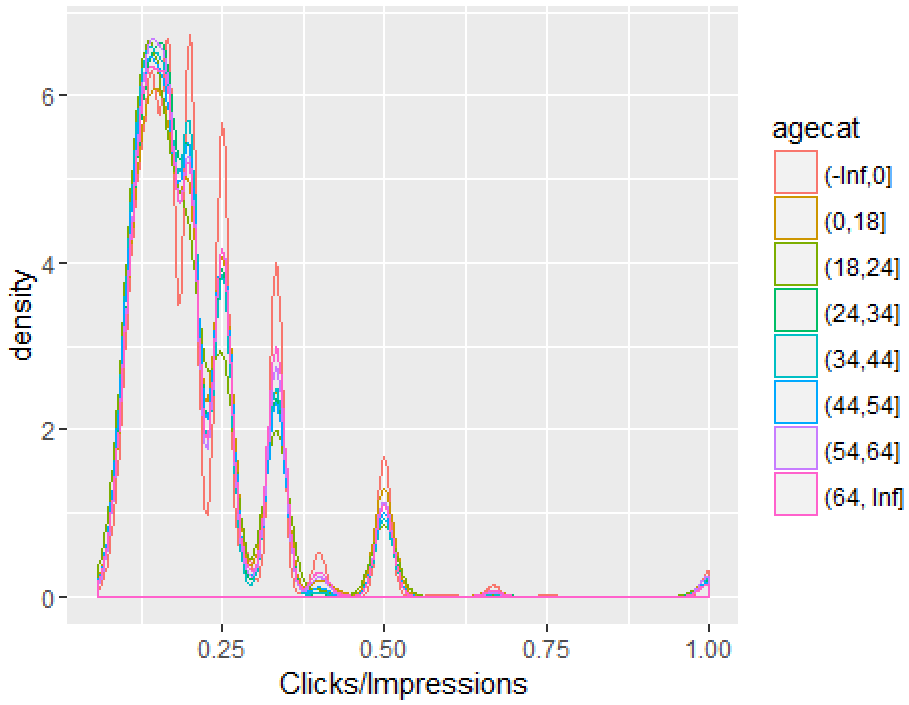
```
content type 'application/zip' length 2001436 bytes (1.9 MB)
downloaded 1.9 MB

package ‘ggplot2’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\nalinkumar87\AppData\Local\Temp\RtmpwtWkUR\downloaded_packages
> library(ggplot2)
> ggplot(data1, aes(x=Impressions, fill=agecat)) +geom_histogram(binwidth=1)
> ggplot(data1, aes(x=agecat, y=Impressions, fill=agecat)) +geom_boxplot()
> data1$hasimps <-cut(data1$Impressions,c(-Inf,0,Inf))
> summaryBy(Clicks~hasimps, data =data1, FUN=siterange)
   hasimps Clicks.FUN1 Clicks.FUN2 Clicks.FUN3 Clicks.FUN4
1 (-Inf,0]        3066           0  0.00000000           0
2 (0, Inf]      455375           0  0.09321768           4
> 
```

- In the next part, we try to visualize click-through rate which is defined as Clicks divided by Impressions and we only care about those clicks where Impressions has a value greater than zero. To visualize this we make four different ggplots. In the first plot, we make a density plot of Clicks/Impressions on the x-axis and their respective density on the y-axis only for those data for which Impressions has a value greater than zero. Along with this, we categorize the graph into different age categories using different colours for each as follows

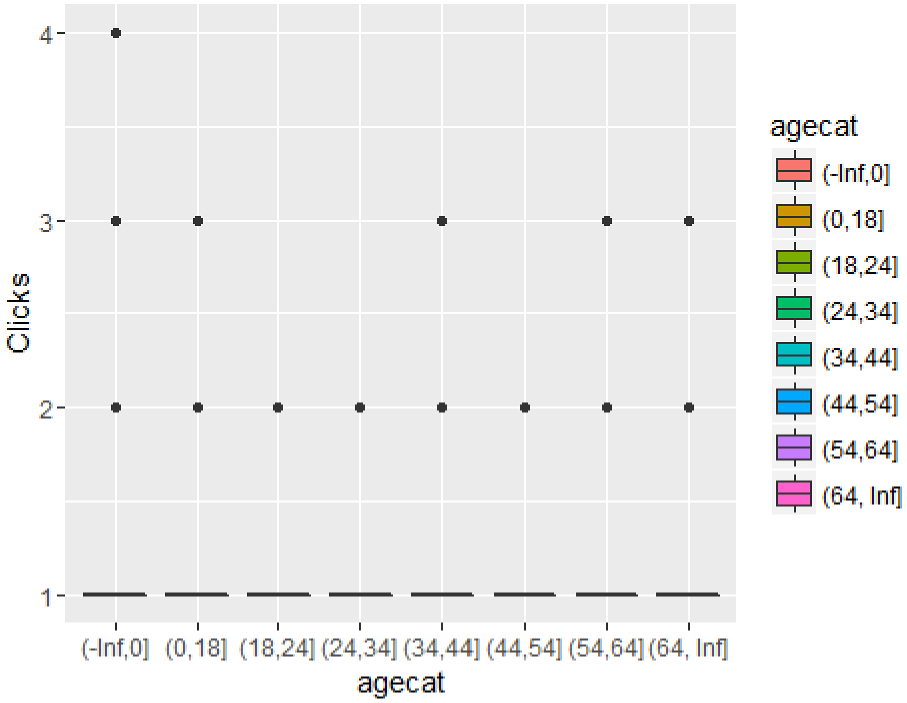
ggplot(subset(data1, Impressions>0), aes(x=Clicks/Impressions, colour=agecat)) + geom_density()



- For the second plot, we make a density plot of Clicks/Impressions on the x-axis and their respective density on the y-axis only for those data for which Clicks has a value greater than zero. Along with this, we categorize the graph into different age categories using different colours for each as follows
ggplot(subset(data1, Clicks>0), aes(x=Clicks/Impressions, colour=agecat)) + geom_density()

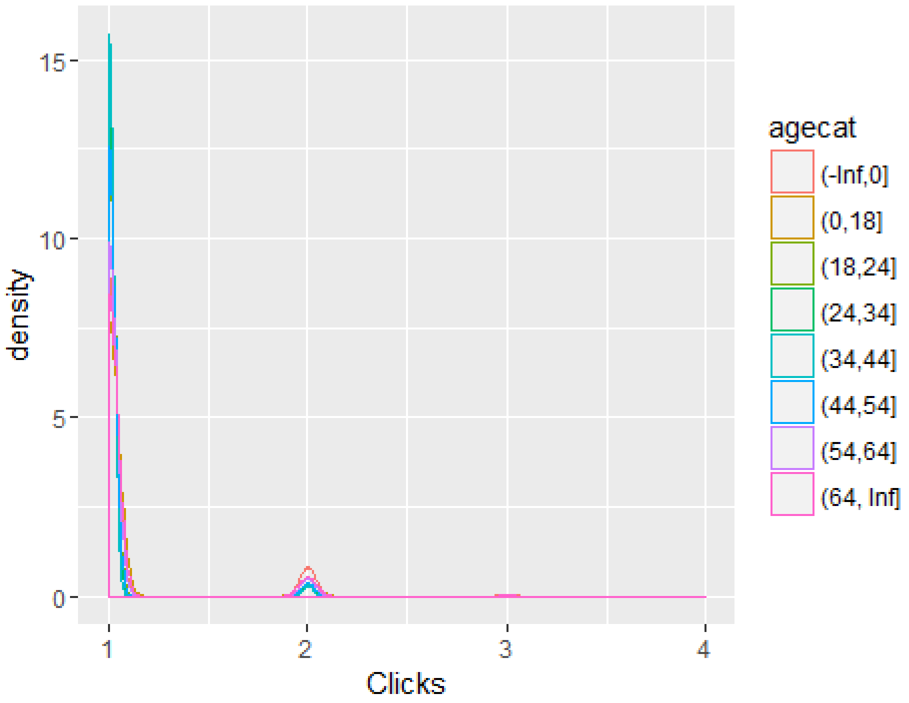


- In the third plot, we make a box plot of age categories on the x-axis and their respective values of Clicks on the y-axis only for those data for which Clicks has a value greater than zero as can be seen in the following image
  ggplot(subset(data1, Clicks>0), aes(x=agecat, y=Clicks, fill=agecat)) + geom_boxplot()



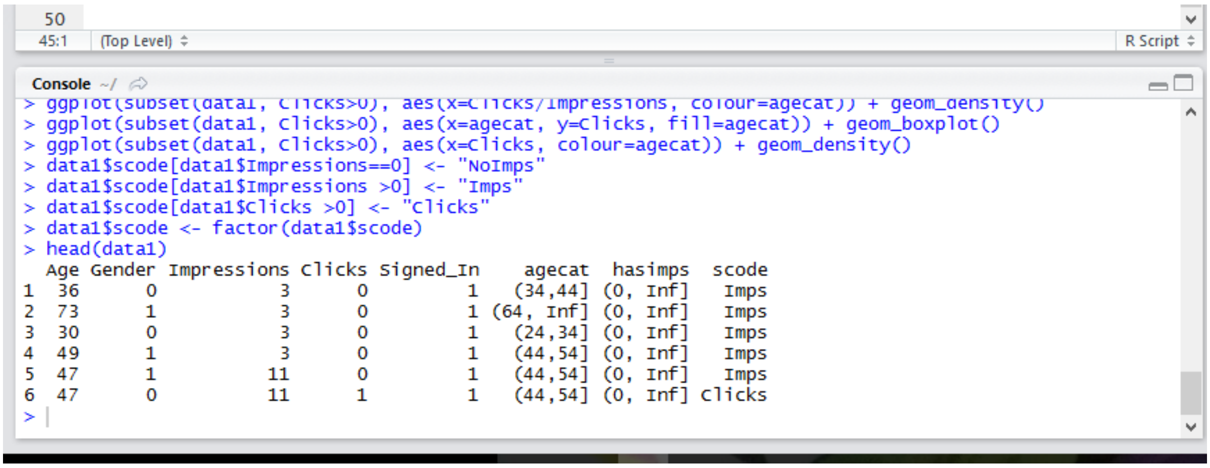
- For the fourth plot, we make a density plot of Clicks on the x-axis and their respective density on the y-axis only for those data for which Clicks has a value greater than zero. Along with this, we categorize the graph into different age categories using different colours for each as follows

  ggplot(subset(data1, Clicks>0), aes(x=Clicks, colour=agecat)) + geom_density()

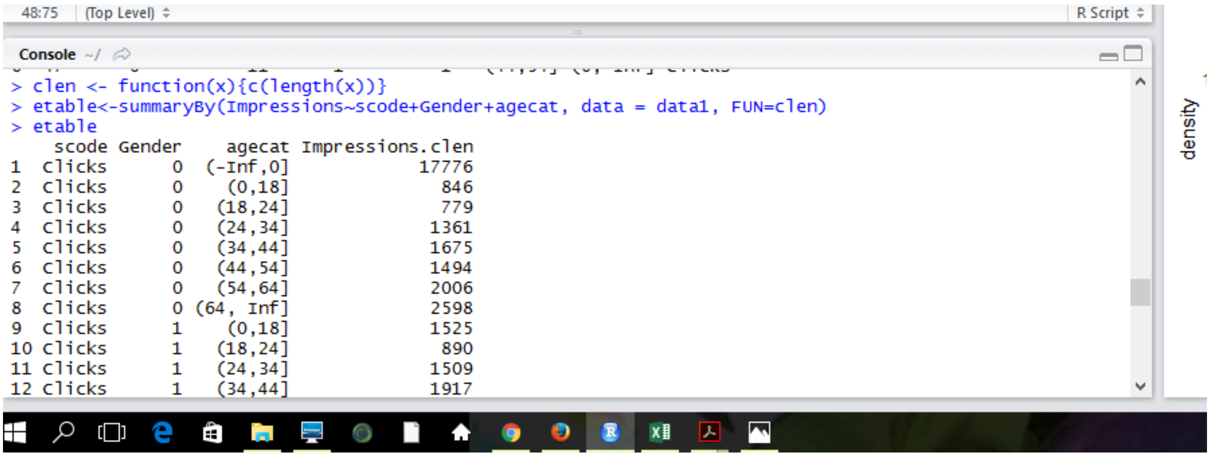


- Finally, they added a new column to the data called as scode which is a factor variable. It takes three set of values based on whether Impressions == 0, Impressions >0 and Clicks >0 and as a result categorizes the whole data into 3 parts based on a particular value scode for a specific row. Subsequently, he performed a summary on the resulting columns of final data such as agecat, scode, Gender and length(Impressions) aa follows

  data1$scode[data1$Impressions==0] <- "NoImps"
  data1$scode[data1$Impressions >0] <- "Imps"
  data1$scode[data1$Clicks >0] <- "Clicks"
  data1$scode <- factor(data1$scode)
  head(data1)
  clen <- function(x){c(length(x))}
  etable<-summaryBy(Impressions~scode+Gender+agecat, data = data1, FUN=clen)
  etable



```
> ggplot(subset(data1, Clicks>0), aes(x=Clicks/Impressions, colour=agecat)) + geom_density()
> ggplot(subset(data1, Clicks>0), aes(x=agecat, y=Clicks, fill=agecat)) + geom_boxplot()
> ggplot(subset(data1, Clicks>0), aes(x=Clicks, colour=agecat)) + geom_density()
> data1$scode[data1$Impressions==0] <- "NoImps"
> data1$scode[data1$Impressions >0] <- "Imps"
> data1$scode[data1$Clicks >0] <- "Clicks"
> data1$scode <- factor(data1$scode)
> head(data1)
  Age Gender Impressions Clicks Signed_In    agecat  hasimps  scode
1  36      0           3      0         1   (34,44] (0, Inf]   Imps
2  73      1           3      0         1 (64, Inf] (0, Inf]   Imps
3  30      0           3      0         1   (24,34] (0, Inf]   Imps
4  49      1           3      0         1   (44,54] (0, Inf]   Imps
5  47      1          11      0         1   (44,54] (0, Inf]   Imps
6  47      0          11      1         1   (44,54] (0, Inf] Clicks
> 
```

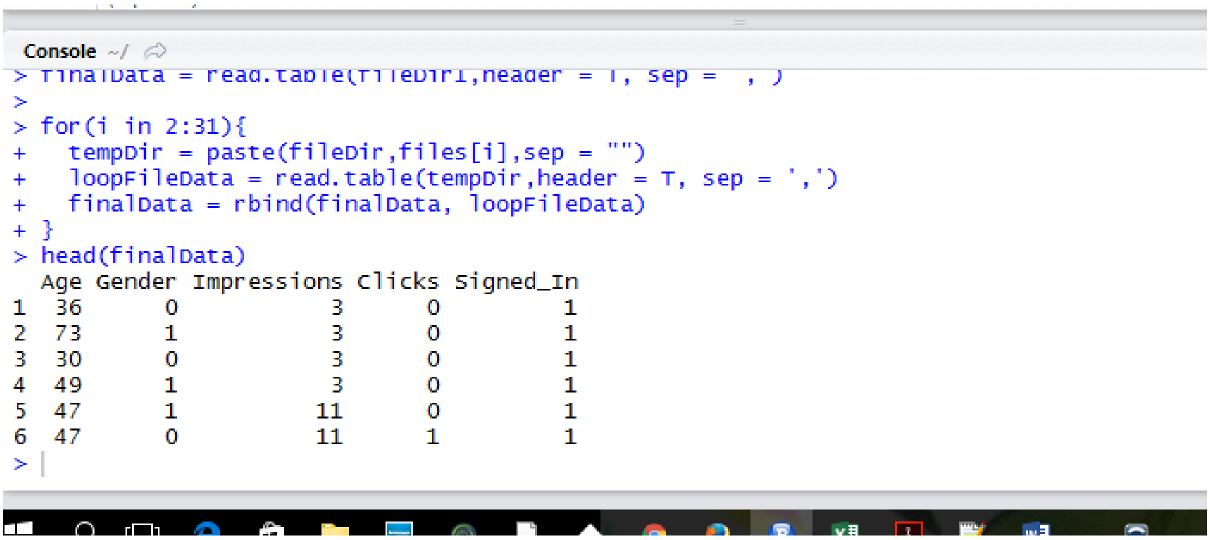


```
> clen <- function(x){c(length(x))}
> etable<-summaryBy(Impressions~scode+Gender+agecat, data = data1, FUN=clen)
> etable
   scode Gender     agecat Impressions.clen
1  Clicks      0  (-Inf,0]            17776
2  Clicks      0    (0,18]              846
3  Clicks      0   (18,24]              779
4  Clicks      0   (24,34]             1361
5  Clicks      0   (34,44]             1675
6  Clicks      0   (44,54]             1494
7  Clicks      0   (54,64]             2006
8  Clicks      0 (64, Inf]             2598
9  Clicks      1    (0,18]             1525
10 Clicks      1   (18,24]              890
11 Clicks      1   (24,34]             1509
12 Clicks      1   (34,44]             1917
```

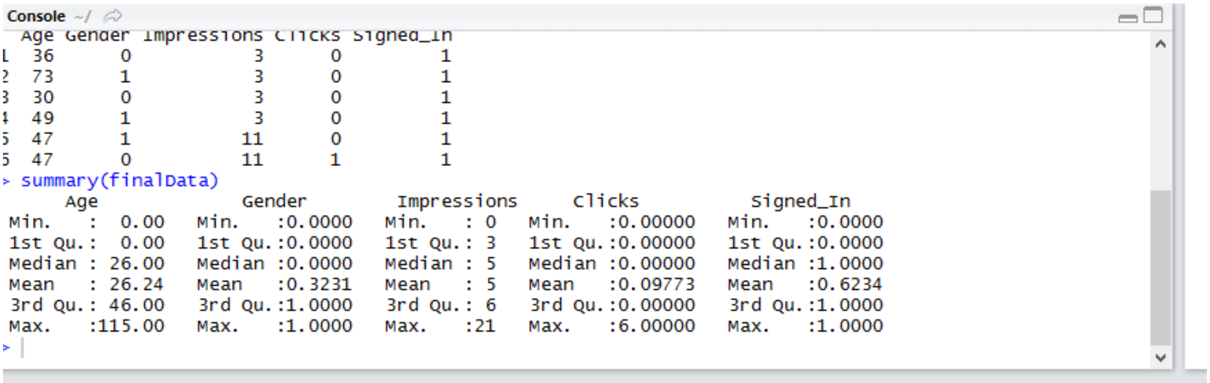
NYTPExnalinKum.R

- Initially, all the data inside 31 files (including file read in the previous part) is read and saved into an object named as finalData. Subsequent to this, we analyse using head and summary commands which somewhat gives us a rough idea about the idea based on various parameters such as Min, Max, Median, Mean etc. as can be seen in the images below

```
fileDir <- 'D:/dic_data/problem2/'
fileDir1 <- 'D:/dic_data/problem2/nyt1.csv'
files <- list.files(path = fileDir, pattern = "\\.csv$")
finalData = read.table(fileDir1,header = T, sep = ',')

for(i in 2:31){
 tempDir = paste(fileDir,files[i],sep = "")
 loopFileData = read.table(tempDir,header = T, sep = ',')
 finalData = rbind(finalData, loopFileData)
}
```



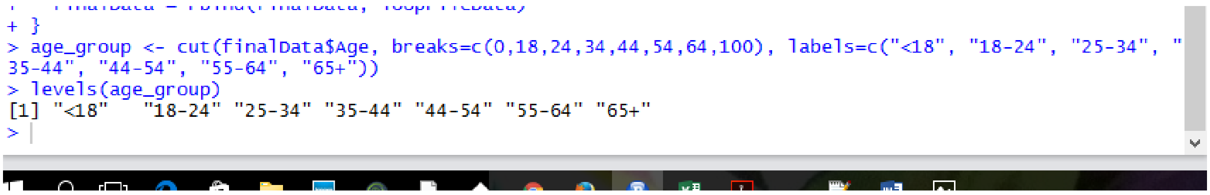
```
> finalData = read.table(fileDir1,header = T, sep = ',')
>
> for(i in 2:31){
+   tempDir = paste(fileDir,files[i],sep = "")
+   loopFileData = read.table(tempDir,header = T, sep = ',')
+   finalData = rbind(finalData, loopFileData)
+ }
> head(finalData)
  Age Gender Impressions Clicks Signed_In
1  36      0           3      0         1
2  73      1           3      0         1
3  30      0           3      0         1
4  49      1           3      0         1
5  47      1          11      0         1
6  47      0          11      1         1
>
```



```
  Age Gender Impressions Clicks Signed_In
1  36      0           3      0         1
2  73      1           3      0         1
3  30      0           3      0         1
4  49      1           3      0         1
5  47      1          11      0         1
6  47      0          11      1         1
> summary(finalData)
      Age             Gender        Impressions       Clicks          Signed_In
 Min.   :  0.00   Min.   :0.0000   Min.   : 0   Min.   :0.00000   Min.   :0.0000
 1st Qu.:  0.00   1st Qu.:0.0000   1st Qu.: 3   1st Qu.:0.00000   1st Qu.:0.0000
 Median : 26.00   Median :0.0000   Median : 5   Median :0.00000   Median :1.0000
 Mean   : 26.24   Mean   :0.3231   Mean   : 5   Mean   :0.09773   Mean   :0.6234
 3rd Qu.: 46.00   3rd Qu.:1.0000   3rd Qu.: 6   3rd Qu.:0.00000   3rd Qu.:1.0000
 Max.   :115.00   Max.   :1.0000   Max.   :21   Max.   :6.00000   Max.   :1.0000
> 
```

- Previously, I gave the answers to the questions on Page 38 of textbook after understanding the code mentioned in the book in the previous part. In this part on extended dataset, I did an analysis on the data and gave answers to all the questions using different code and plots.
- Initially, I created a categorical variable named as age_group which categorized the extended dataset into 6 different age categories based on the value of Age column

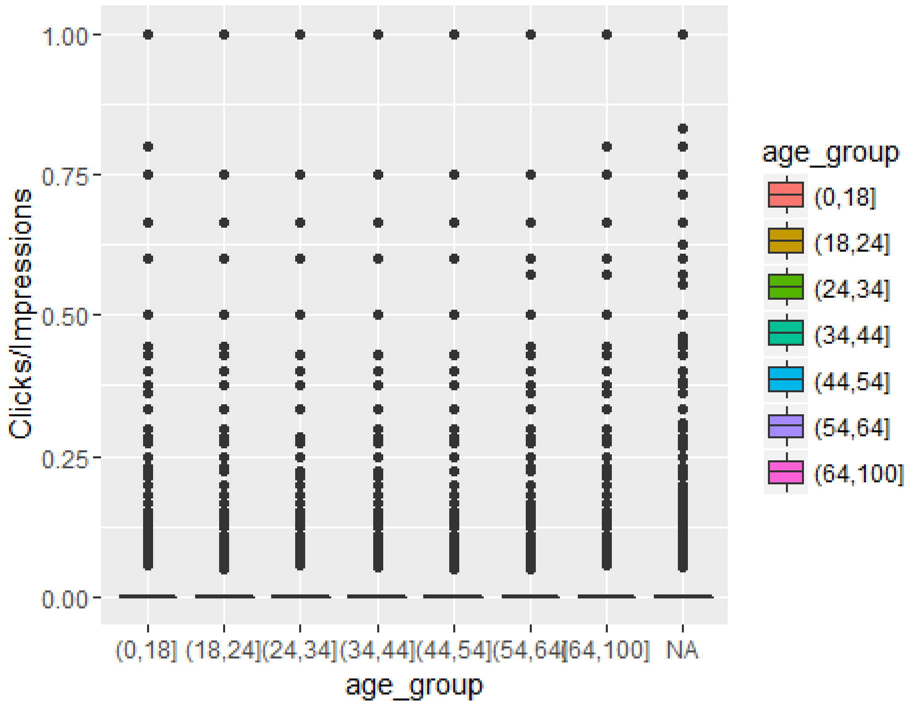
age_group <- cut(finalData$Age, breaks=c(0,18,24,34,44,54,64,100), labels=c("<18", "18-24", "25-34", "35-44", "44-54", "55-64", "65+"))
finalData$age_group <- cut(finalData$Age, breaks=c(0,18,24,34,44,54,64,100))
levels(age_group)



```
+ }
> age_group <- cut(finalData$Age, breaks=c(0,18,24,34,44,54,64,100), labels=c("<18", "18-24", "25-34", "
35-44", "44-54", "55-64", "65+"))
> levels(age_group)
[1] "<18"   "18-24" "25-34" "35-44" "44-54" "55-64" "65+"
> 
```

- In the next part, I defined a new column named as hasImps which categorized the extended dataset into two parts based on the value of Impressions column and read the summary of the resulting dataset. Following this, I used a ggplot to plot age_group on the x-axis and Clicks/Impressions on the y-axis for a subset of finalData for which Impressions >0. As can seen in the image below, the value of Clicks/Impressions for all the age groups lies between 0 and 0.5 for most of the rows in the dataset based on the density of accumulation of data points in the box-plot. This value Clicks/Impressions is defined as the click-through rate which is calculated for Impressions >0

siterange <- function(x){c(length(x), min(x), mean(x), max(x))}
finalData$hasimps <-cut(finalData$Impressions,c(-Inf,0,Inf))
summaryBy(Clicks~hasimps, data =finalData, FUN=siterange)
ggplot(subset(finalData, Clicks>0), aes(x=age_group, y=Clicks/Impressions, fill=age_group)) + geom_boxplot()

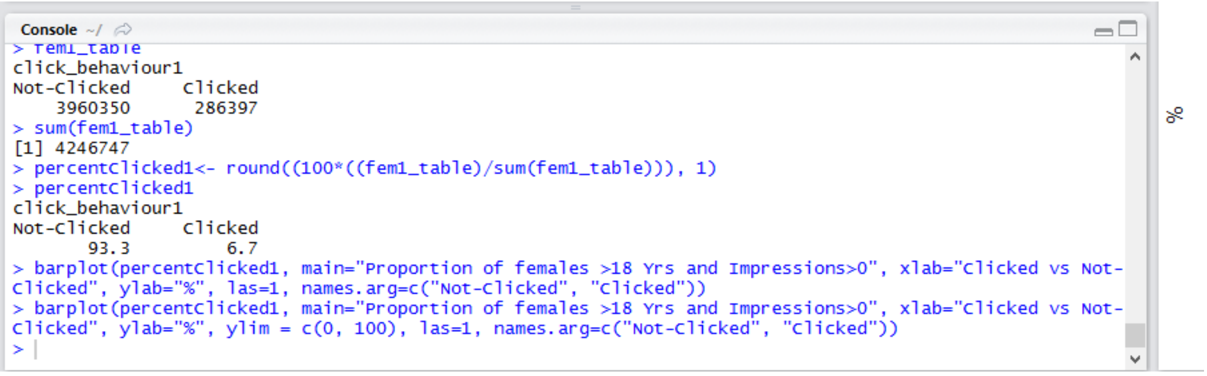


- In the next part, a variable named as click_behaviour is defined which categorizes the extended dataset into two labels namely, Clicked or Not-Clicked based on the value of Clicks column

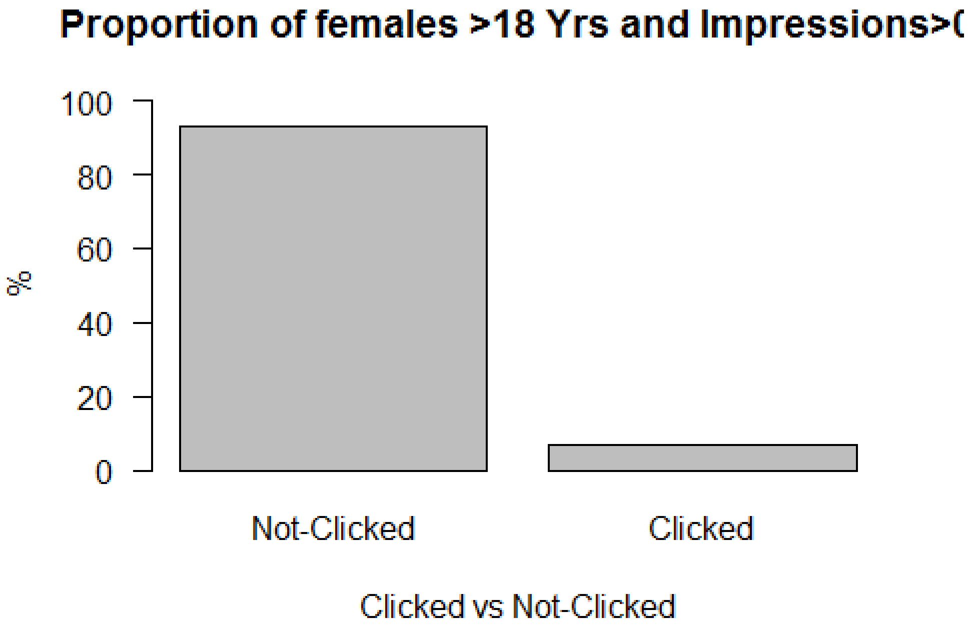
click_behaviour <- cut(finalData$Clicks, breaks=c(-Inf, 0, Inf), labels=c("Not-Clicked", "Clicked"))

- Subsequently, some analysis is performed based on different constraints. In the first analysis, I specifically selected only those rows for which Gender is equal to zero i.e. females greater than 18 years of age and for whom Impressions >0. After this, I categorized data above into two parts based on vales of Clicks column of respective females. Finally, I made a barplot where I analysed what proportion of females(as a percentage of total number of females) greater than 18 years of age and Impressions > 0 have clicked or not

```
fem1 <- finalData[finalData$Gender == 0 & finalData$Age >18 & finalData$Impressions > 0, ]
click_behaviour1 <- cut(fem1$Clicks, breaks=c(-Inf, 0, Inf), labels=c("Not-Clicked", "Clicked"))
fem1_table <- table(click_behaviour1)
fem1_table
sum(fem1_table)
percentClicked1<- round((100*((fem1_table)/sum(fem1_table))), 1)
percentClicked1
barplot(percentClicked1, main="Proportion of females >18 Yrs and Impressions>0", xlab="Clicked vs Not-Clicked", ylab="%", las=1, names.arg=c("Not-Clicked", "Clicked"))
```



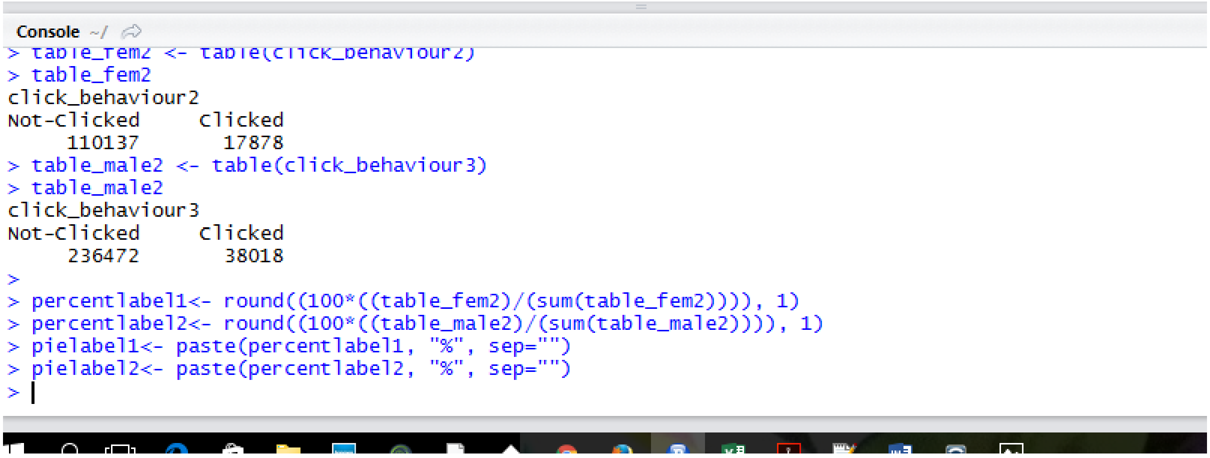
```
> fem1_table
click_behaviour1
Not-Clicked     Clicked
    3960350      286397
> sum(fem1_table)
[1] 4246747
> percentClicked1<- round((100*((fem1_table)/sum(fem1_table))), 1)
> percentClicked1
click_behaviour1
Not-Clicked     Clicked
       93.3         6.7
> barplot(percentClicked1, main="Proportion of females >18 Yrs and Impressions>0", xlab="Clicked vs Not-Clicked", ylab="%", las=1, names.arg=c("Not-Clicked", "Clicked"))
> barplot(percentClicked1, main="Proportion of females >18 Yrs and Impressions>0", xlab="Clicked vs Not-Clicked", ylab="%", ylim = c(0, 100), las=1, names.arg=c("Not-Clicked", "Clicked"))
> |
```



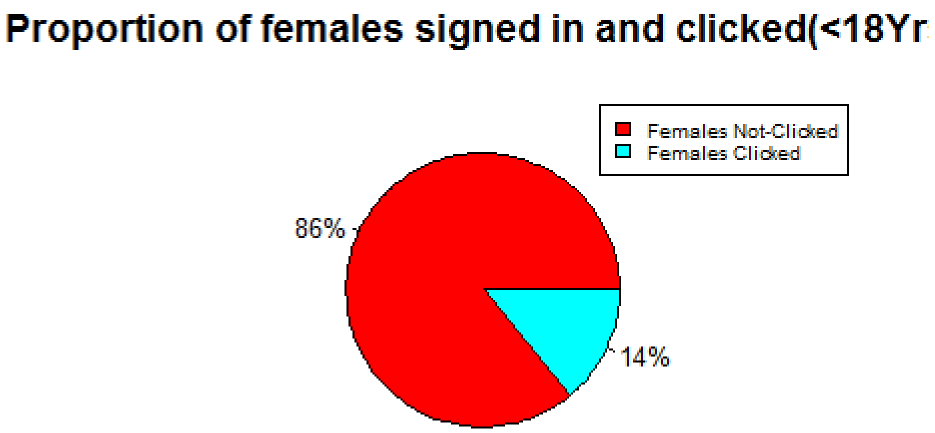
- In the second analysis, I analysed demographically somewhat different kind of perspective. I constructed two pie charts, one for males and other for females. For each of them, I plotted separately only those data for males and females less than 18 years of age who have signed in. Then I calculated the respective percentage population individually based on total number of females for female pie chart and total number of males for male pie chart and plotted the two resulting pie charts separately. The results clearly depicted an interesting picture of the data. Although, the total number of males is much smaller than total number of females in the extended dataset, but still the proportion of males and females(as a percentage of total number of males and females respectively) less than 18 years of age who have signed in as well clicked is very much the same as can be seen in the below images

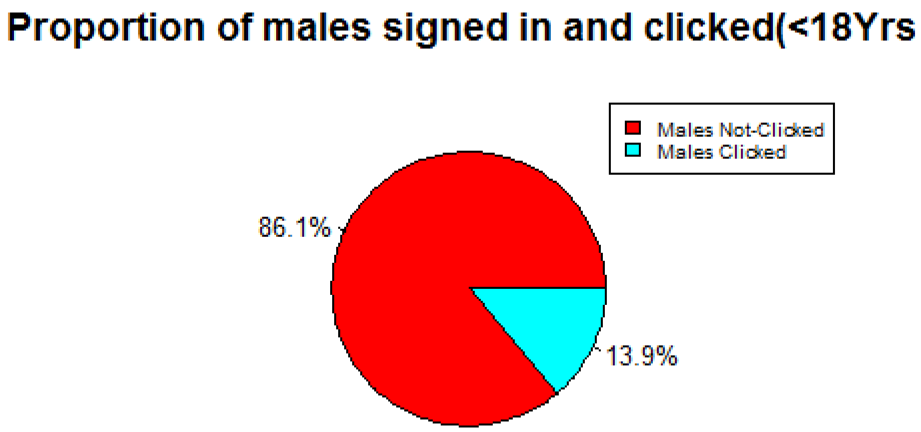
fem2 <- finalData[finalData$Gender == 0 & finalData$Age <18 & finalData$Signed_In == 1,]
male2 <- finalData[finalData$Gender == 1 & finalData$Age <18 & finalData$Signed_In == 1,]

```
click_behaviour2 <- cut(fem2$Clicks, breaks=c(-Inf, 0, Inf), labels=c("Not-Clicked", "Clicked"))
click_behaviour3 <- cut(male2$Clicks, breaks=c(-Inf, 0, Inf), labels=c("Not-Clicked", "Clicked"))
table_fem2 <- table(click_behaviour2)
table_fem2
table_male2 <- table(click_behaviour3)
table_male2

percentlabel1<- round((100*((table_fem2)/(sum(table_fem2)))), 1)
percentlabel2<- round((100*((table_male2)/(sum(table_male2)))), 1)
pielabel1<- paste(percentlabel1, "%", sep="")
pielabel2<- paste(percentlabel2, "%", sep="")

#Pie plots to distinguish between what proportion of males and females signed_in as well as clicked
pie(table_fem2, main="Proportion of females signed in and clicked(<18Yrs)", col=rainbow(length(table_fem2)), labels=pielabel1, cex=0.8)
legend("topright", c("Females Not-Clicked","Females Clicked"), cex=0.6, fill=rainbow(length(table_fem2)))

pie(table_male2, main="Proportion of males signed in and clicked(<18Yrs)", col=rainbow(length(table_male2)), labels=pielabel2, cex=0.8)
legend("topright", c("Males Not-Clicked","Males Clicked"), cex=0.6, fill=rainbow(length(table_male2)))
```



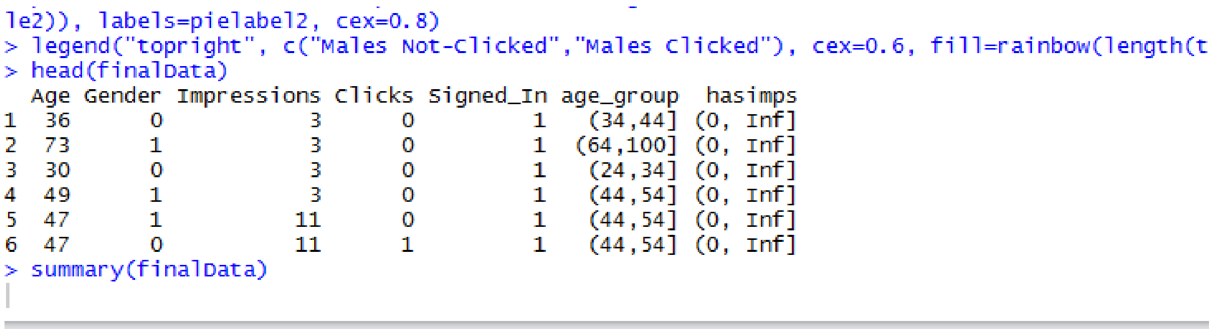
```
> table_fem2 <- table(click_behaviour2)
> table_fem2
click_behaviour2
Not-Clicked     Clicked
     110137       17878
> table_male2 <- table(click_behaviour3)
> table_male2
click_behaviour3
Not-Clicked     Clicked
     236472       38018
>
> percentlabel1<- round((100*((table_fem2)/(sum(table_fem2)))), 1)
> percentlabel2<- round((100*((table_male2)/(sum(table_male2)))), 1)
> pielabel1<- paste(percentlabel1, "%", sep="")
> pielabel2<- paste(percentlabel2, "%", sep="")
> |
```



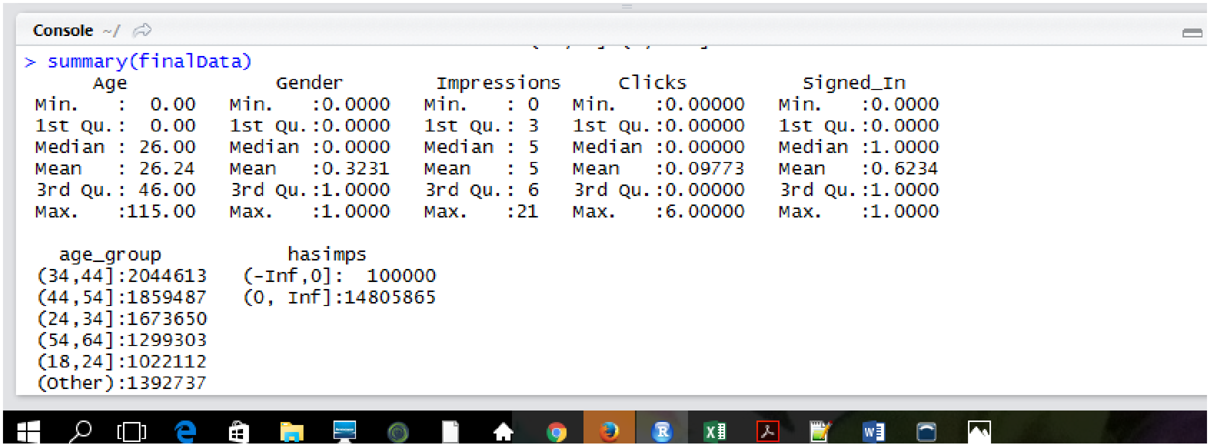


- In the final part we needed to create statistics to summarize the dataset. Initially, I ran head and summary commands on finalData to visualize the min, max, median and mode as follows

```
#summary and analysis of data
head(finalData)
summary(finalData)
```



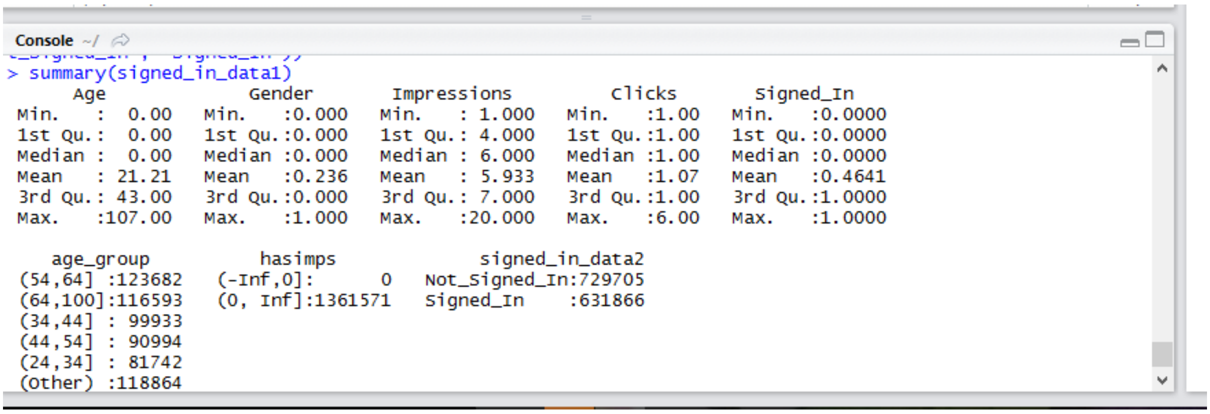
```
le2)), labels=pielabel2, cex=0.8)
> legend("topright", c("Males Not-Clicked","Males Clicked"), cex=0.6, fill=rainbow(length(t
> head(finalData)
  Age Gender Impressions Clicks Signed_In age_group  hasimps
1  36      0           3      0         1   (34,44] (0, Inf]
2  73      1           3      0         1  (64,100] (0, Inf]
3  30      0           3      0         1   (24,34] (0, Inf]
4  49      1           3      0         1   (44,54] (0, Inf]
5  47      1          11      0         1   (44,54] (0, Inf]
6  47      0          11      1         1   (44,54] (0, Inf]
> summary(finalData)
```



```
> summary(finalData)
      Age             Gender          Impressions        Clicks            Signed_In
 Min.   :  0.00   Min.   :0.0000   Min.   : 0    Min.   :0.00000   Min.   :0.0000
 1st Qu.:  0.00   1st Qu.:0.0000   1st Qu.: 3    1st Qu.:0.00000   1st Qu.:0.0000
 Median : 26.00   Median :0.0000   Median : 5    Median :0.00000   Median :1.0000
 Mean   : 26.24   Mean   :0.3231   Mean   : 5    Mean   :0.09773   Mean   :0.6234
 3rd Qu.: 46.00   3rd Qu.:1.0000   3rd Qu.: 6    3rd Qu.:0.00000   3rd Qu.:1.0000
 Max.   :115.00   Max.   :1.0000   Max.   :21    Max.   :6.00000   Max.   :1.0000

   age_group              hasimps
 (34,44]:2044613   (-Inf,0]:  100000
 (44,54]:1859487   (0, Inf]:14805865
 (24,34]:1673650
 (54,64]:1299303
 (18,24]:1022112
 (Other):1392737
```

- Next, I compressed the finalData based on whether Impressions >0 and Clicks >0. I am only interested in those rows for which users have Impressions, Clicked and Signed >0 all at the same time. This data will actually help me in analysing about how many users have actually signed in along with Impressions and Clicks value greater than zero. Then I created a new column in the subset of finalData mentioned above which categorizes the dataset into 2 values namely Signed_In and Not-Signed_in respectively. Finally, I analysed the compressed data by visualizing its summary

signed_in_data1 <- finalData[finalData$Impressions > 0 & finalData$Clicks > 0,]
signed_in_data1$signed_in_data2 <- cut(signed_in_data1$Signed_In, breaks=c(-Inf, 0, Inf), labels=c("Not_Signed_In", "Signed_In"))
summary(signed_in_data1)



```
> summary(signed_in_data1)
      Age            Gender        Impressions         Clicks         Signed_In
 Min.   :  0.00   Min.   :0.000   Min.   : 1.000   Min.   :1.00   Min.   :0.0000
 1st Qu.:  0.00   1st Qu.:0.000   1st Qu.: 4.000   1st Qu.:1.00   1st Qu.:0.0000
 Median :  0.00   Median :0.000   Median : 6.000   Median :1.00   Median :0.0000
 Mean   : 21.21   Mean   :0.236   Mean   : 5.933   Mean   :1.07   Mean   :0.4641
 3rd Qu.: 43.00   3rd Qu.:0.000   3rd Qu.: 7.000   3rd Qu.:1.00   3rd Qu.:1.0000
 Max.   :107.00   Max.   :1.000   Max.   :20.000   Max.   :6.00   Max.   :1.0000

    age_group           hasimps             signed_in_data2
 (54,64] :123682   (-Inf,0]:      0   Not_Signed_In:729705
 (64,100]:116593   (0, Inf]:1361571   Signed_In    :631866
 (34,44] : 99933
 (44,54] : 90994
 (24,34] : 81742
 (Other) :118864
```