Live Session Unit 10 Assignment

M Nepal

July 21, 2017

Introduction

This is R markdown document for keeping track of assignment submitted for **MSDS-6306@SMU** as an example of **Exploratory Data Analysis (EDA)** based on dataset provided by Department of Statistics Columbia University in the City of New York.

There are 32 data sets named nyt0.csv, nyt1.csv,..., nyt31.csv, which can be downloaded from **here**.

Each csv represents one (simulated) days worth of ads shown and clicks recorded on the New York Times homepage in 2012. Each row in the csv represents a single user.

This information is taken from RPubs.

I have chosen **nyt1.csv** for EDA.

Required packages

- ggplot2
- plyr
- dplyr

Install and/or load these packages before trying the code below.

```
library(ggplot2)
library(plyr)
library(dplyr)

#Get the data from url
fileLocation <- "http://stat.columbia.edu/~rachel/datasets/nyt1.csv"
data1 <- read.csv(url(fileLocation))
names(data1) # This will help to know variable names.

## [1] "Age"         "Gender"      "Impressions" "Clicks"      "Signed_In"

# str function provides the variable types.
str(data1)

## 'data.frame':    458441 obs. of  5 variables:
##  $ Age        : int  36 73 30 49 47 47 0 46 16 52 ...
##  $ Gender     : int  0 1 0 1 1 0 0 0 0 0 ...
##  $ Impressions: int  3 3 3 3 11 11 7 5 3 4 ...
##  $ Clicks     : int  0 0 0 0 0 1 1 0 0 0 ...
##  $ Signed_In  : int  1 1 1 1 1 1 0 1 1 1 ...
```

Exploratory Data Analysis

```
#Let's find summary statistics of data set, just to make a good start for EDA
summary(data1)
```

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

Create a new variable named ageGroup, that categorizes age into following groups:

<18, 18-24, 25-34, 35-44, 45-54, 55-64, and 65+

```
# categorizes age groups
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
```

```
data1$ageGroup <- cut(data1$Age, c(-Inf, 18, 24, 34, 44, 54, 64, Inf))
levels(data1$ageGroup) <- c("<18", "18-24", "25-34", "35-44", "45-54", "55-
64", "65+")
summary(data1)
```

```
##       Age              Gender       Impressions         Clicks
##  Min.   :  0.00   Min.   :0.000   Min.   : 0.000   Min.   :0.00000
##  1st Qu.:  0.00   1st Qu.:0.000   1st Qu.: 3.000   1st Qu.:0.00000
##  Median : 31.00   Median :0.000   Median : 5.000   Median :0.00000
##  Mean   : 29.48   Mean   :0.367   Mean   : 5.007   Mean   :0.09259
##  3rd Qu.: 48.00   3rd Qu.:1.000   3rd Qu.: 6.000   3rd Qu.:0.00000
##  Max.   :108.00   Max.   :1.000   Max.   :20.000   Max.   :4.00000
##
##    Signed_In       ageGroup
##  Min.   :0.0000   <18  :156358
##  1st Qu.:0.0000   18-24: 35270
##  Median :1.0000   25-34: 58174
##  Mean   :0.7009   35-44: 70860
```

```
##  3rd Qu.:1.0000   45-54: 64288
##  Max.   :1.0000   55-64: 44738
##                   65+  : 28753
```

Use sub set of data called "ImpSub" where Impressions > 0

```
ImpSub <- subset(data1, Impressions > 0) # new variable ImpSub
head(ImpSub)
```

```
##   Age Gender Impressions Clicks Signed_In ageGroup
## 1  36      0           3      0         1    35-44
## 2  73      1           3      0         1      65+
## 3  30      0           3      0         1    25-34
## 4  49      1           3      0         1    45-54
## 5  47      1          11      0         1    45-54
## 6  47      0          11      1         1    45-54
```

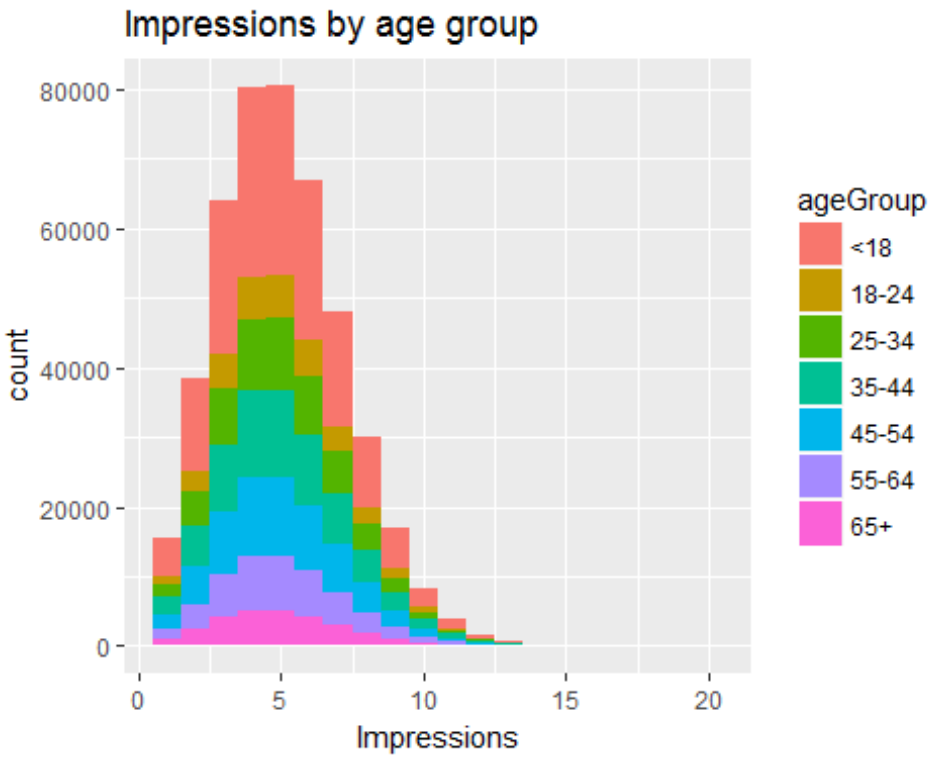
Create new variable called click-through-rate(CTR = click/impression)

```
ImpSub$CTR <- ImpSub$Clicks/ImpSub$Impressions
head(ImpSub)
```

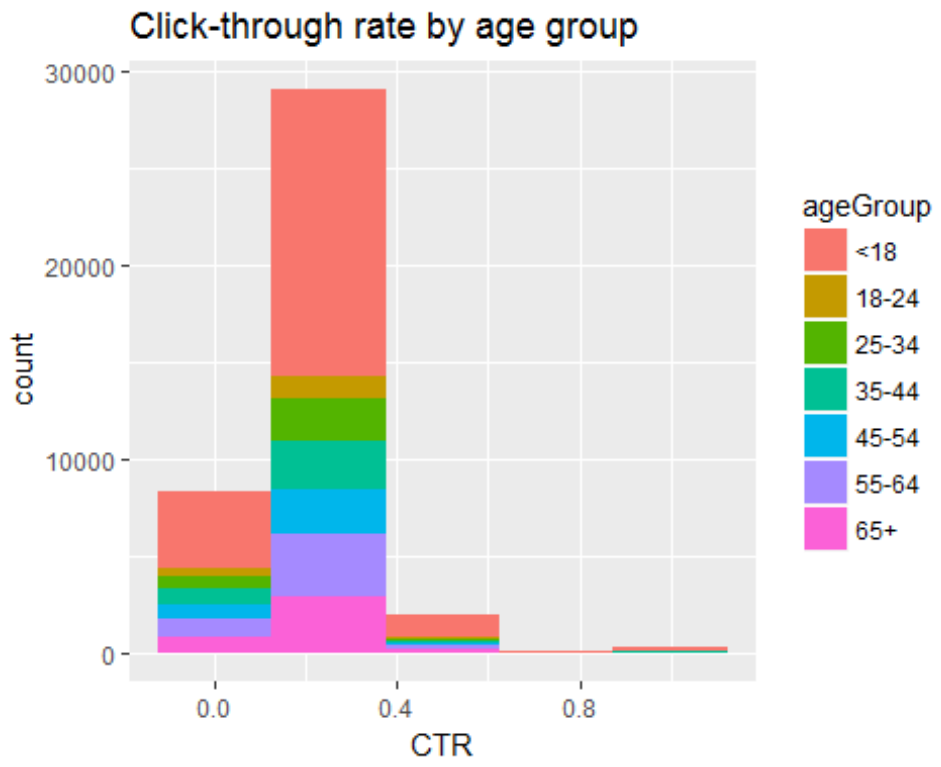
```
##   Age Gender Impressions Clicks Signed_In ageGroup        CTR
## 1  36      0           3      0         1    35-44 0.00000000
## 2  73      1           3      0         1      65+ 0.00000000
## 3  30      0           3      0         1    25-34 0.00000000
## 4  49      1           3      0         1    45-54 0.00000000
## 5  47      1          11      0         1    45-54 0.00000000
## 6  47      0          11      1         1    45-54 0.09090909
```

Plot distributions of number impressions and click-through-rate (CTR = click/impressions) for the age groups

```
#Plot the distribution of Impressions>0, grouped by ageGroup
ggplot(subset(ImpSub, Impressions > 0), aes(x=Impressions, fill=ageGroup))+
  labs(title="Impressions by age group")+
  geom_histogram(binwidth=1)
```




```
#Plot the distribution of CTR>0, grouped by ageGroup
ggplot(subset(ImpSub, CTR > 0), aes(x=CTR, fill=ageGroup))+
  labs(title="Click-through rate by age group")+
  geom_histogram(binwidth=0.25)
```



Define a new variable to segment users based on click -through-rate (CTR) behavior.

CTR< 0.2, 0.2 <= CTR < 0.4, 0.4 <= CTR < 0.6, 0.6 <= CTR <0.8, CTR >0.8

```
ImpSub$CTR_Behavior <- cut(ImpSub$CTR, c(-Inf, 0.2, 0.4, 0.6, 0.8, Inf))
levels(ImpSub$CTR_Behavior) <- c("CTR < 0.2", "0.2 <= CTR < 0.4", "0.4 <= CTR
< 0.6", "0.6 <= CTR < 0.8", "CTR  > 0.8")
```

7) Get the total number of Male, Impressions, Clicks and Signed_In

(0=Female, 1=Male)

```
str(ImpSub)

## 'data.frame':    455375 obs. of  8 variables:
##  $ Age         : int  36 73 30 49 47 47 0 46 16 52 ...
##  $ Gender      : int  0 1 0 1 1 0 0 0 0 0 ...
##  $ Impressions : int  3 3 3 3 11 11 7 5 3 4 ...
##  $ Clicks      : int  0 0 0 0 0 1 1 0 0 0 ...
##  $ Signed_In   : int  1 1 1 1 1 1 0 1 1 1 ...
##  $ ageGroup    : Factor w/ 7 levels "<18","18-24",..: 4 7 3 5 5 5 1 5 1 5
...
##  $ CTR         : num  0 0 0 0 0 ...
##  $ CTR_Behavior: Factor w/ 5 levels "CTR < 0.2","0.2 <= CTR < 0.4",..: 1 1
1 1 1 1 1 1 1 ...

sapply(ImpSub[c(2,3,4,5)],sum)
```

```
##      Gender Impressions      Clicks   Signed_In
##      167146     2295559       42449      319198

# The sum of gender also works here as Male=1 and Female=0, Gender represents
total male since female = 0
```

Get the mean of Age, Impressions, Clicks, CTR and percentage of males and signed_In

```
#Before
ImpSubPer <- sapply(ImpSub[c(1,3,4,7)],mean)
ImpSubPer

##         Age Impressions      Clicks         CTR
## 29.48400988  5.04102992  0.09321768  0.01847053

#Create percentage variables and combined with ImpSubPer
percentageOfMaleAndSigned_In <-
c((sapply(ImpSub[c(2,5)],sum)/sapply(ImpSub[c(2,5)],length)*100))
percentageOfMaleAndSigned_In

##    Gender Signed_In
##  36.70513  70.09564

ImpSubCombined <- c(ImpSubPer,percentageOfMaleAndSigned_In)
#after combining abd before cleaning col names
ImpSubCombined

##         Age Impressions      Clicks         CTR      Gender   Signed_In
## 29.48400988  5.04102992  0.09321768  0.01847053 36.70513313 70.09563547

##combined vector after cleaning the header for question 8
names(ImpSubCombined)<-
c("Age_mean","Impressions_mean","Clicks_mean","CTR_mean","% of  Males","% of
signed_in")
ImpSubCombined

##         Age_mean Impressions_mean      Clicks_mean         CTR_mean
##      29.48400988       5.04102992       0.09321768       0.01847053
##      % of  Males   % of signed_in
##      36.70513313      70.09563547
```

Get the means of Impressions, Clicks, CTR and percentage of males and signed_In by AgeGroup.

```
meansByAgeGroup <-
aggregate(cbind(ImpSub$Impressions,ImpSub$Clicks,ImpSub$CTR)~ageGroup,FUN =
mean,ImpSub,na.rm = TRUE)
colnames(meansByAgeGroup) <-
c("ageGroup","Impressions_mean","Clicks_mean","CTR_mean")
meansByAgeGroup

##   ageGroup Impressions_mean Clicks_mean    CTR_mean
## 1      <18         5.033534  0.14167788 0.028141310
## 2    18-24         5.043240  0.04880905 0.009720481
```

```
## 3    25-34         5.026055  0.05081227 0.010146329
## 4    35-44         5.054749  0.05202148 0.010286330
## 5    45-54         5.045172  0.05062260 0.009957612
## 6    55-64         5.053484  0.10246952 0.020306816
## 7      65+         5.046925  0.15233226 0.029802702
```

```
#using dplyr/plyr package
sumOfMaleByAgeGroup <- ddply(ImpSub, "ageGroup", summarise,
No_Of_Males=sum(Gender))
sumOfMaleByAgeGroup
```

```
##   ageGroup No_Of_Males
## 1      <18       12279
## 2    18-24       18697
## 3    25-34       30750
## 4    35-44       37429
## 5    45-54       33788
## 6    55-64       23830
## 7      65+       10373
```

```
sumOfSignedInAgeGroup <- ddply(ImpSub, "ageGroup", summarise,
No_Of_Signed_In=sum(Signed_In))
#Incase you want to display
sumOfSignedInAgeGroup
```

```
##   ageGroup No_Of_Signed_In
## 1      <18           19126
## 2    18-24           35014
## 3    25-34           57801
## 4    35-44           70394
## 5    45-54           63845
## 6    55-64           44462
## 7      65+           28556
```

```
combinedMaleandSign <-
merge(sumOfMaleByAgeGroup,sumOfSignedInAgeGroup,by="ageGroup")
#In case if you want to display
combinedMaleandSign
```

```
##   ageGroup No_Of_Males No_Of_Signed_In
## 1      <18       12279           19126
## 2    18-24       18697           35014
## 3    25-34       30750           57801
## 4    35-44       37429           70394
## 5    45-54       33788           63845
## 6    55-64       23830           44462
## 7      65+       10373           28556
```

```
totalRows <- nrow(ImpSub)
totalRows
```

```
## [1] 455375
```

```r
combinedMaleandSign$percentage_Of_Males <-
((combinedMaleandSign$No_Of_Males)/totalRows)*100
combinedMaleandSign
```

```
##   ageGroup No_Of_Males No_Of_Signed_In percentage_Of_Males
## 1      <18       12279           19126            2.696459
## 2    18-24       18697           35014            4.105847
## 3    25-34       30750           57801            6.752676
## 4    35-44       37429           70394            8.219380
## 5    45-54       33788           63845            7.419819
## 6    55-64       23830           44462            5.233050
## 7      65+       10373           28556            2.277903
```

```r
combinedMaleandSign$percentage_of_signed_In <-
((combinedMaleandSign$No_Of_Signed_In)/totalRows)*100
combinedMaleandSign
```

```
##   ageGroup No_Of_Males No_Of_Signed_In percentage_Of_Males
## 1      <18       12279           19126            2.696459
## 2    18-24       18697           35014            4.105847
## 3    25-34       30750           57801            6.752676
## 4    35-44       37429           70394            8.219380
## 5    45-54       33788           63845            7.419819
## 6    55-64       23830           44462            5.233050
## 7      65+       10373           28556            2.277903
##   percentage_of_signed_In
## 1                4.200055
## 2                7.689047
## 3               12.693055
## 4               15.458468
## 5               14.020313
## 6                9.763821
## 7                6.270876
```

```r
cleanedVector <- subset(combinedMaleandSign,select=c(1,4,5))# using dplyr
package
cleanedVector
```

```
##   ageGroup percentage_Of_Males percentage_of_signed_In
## 1      <18            2.696459                4.200055
## 2    18-24            4.105847                7.689047
## 3    25-34            6.752676               12.693055
## 4    35-44            8.219380               15.458468
## 5    45-54            7.419819               14.020313
## 6    55-64            5.233050                9.763821
## 7      65+            2.277903                6.270876
```

```r
mergedvector <-merge(meansByAgeGroup, cleanedVector, by="ageGroup")
mergedvector
```

```
##   ageGroup Impressions_mean Clicks_mean    CTR_mean percentage_Of_Males
## 1      <18         5.033534  0.14167788 0.028141310            2.696459
## 2    18-24         5.043240  0.04880905 0.009720481            4.105847
## 3    25-34         5.026055  0.05081227 0.010146329            6.752676
## 4    35-44         5.054749  0.05202148 0.010286330            8.219380
## 5    45-54         5.045172  0.05062260 0.009957612            7.419819
## 6    55-64         5.053484  0.10246952 0.020306816            5.233050
## 7      65+         5.046925  0.15233226 0.029802702            2.277903
##   percentage_of_signed_In
## 1                4.200055
## 2                7.689047
## 3               12.693055
## 4               15.458468
## 5               14.020313
## 6                9.763821
## 7                6.270876
```

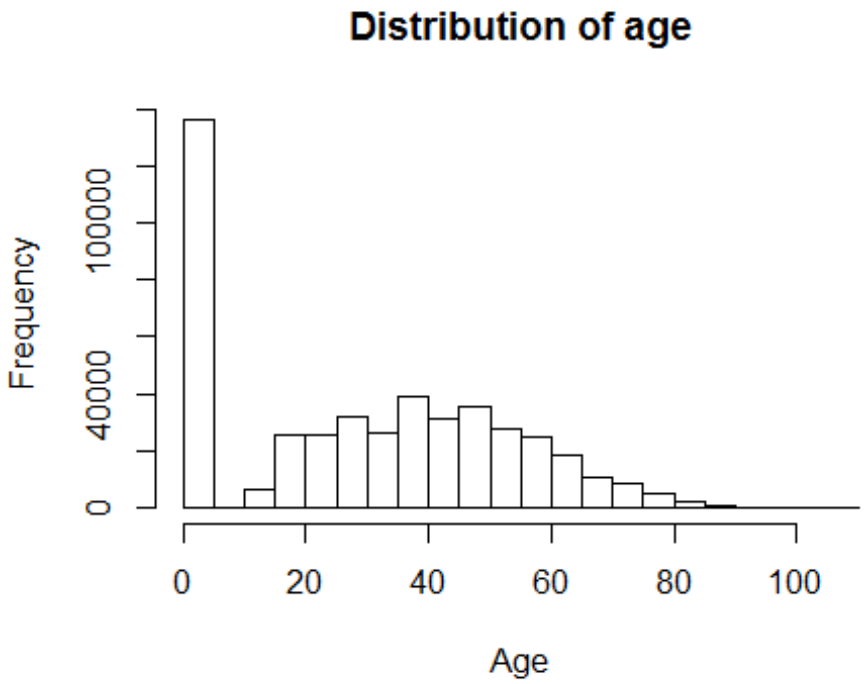
Create a table of CTRGroup vs AgeGroup counts.

```
ctr_age_Table <- table(ImpSub$CTR_Behavior,ImpSub$ageGroup)
ctr_age_Table
```

```
##
##                       <18  18-24  25-34  35-44  45-54  55-64    65+
##   CTR < 0.2        148412  34540  56980  69424  62936  43147  27261
##   0.2 <= CTR < 0.4   5735    391    689    820    776   1104   1108
##   0.4 <= CTR < 0.6    918     68    106    118    113    168    156
##   0.6 <= CTR < 0.8     76      2      7      4      0      7     10
##   CTR  > 0.8          162     13     19     28     20     36     21
```

Let's do One more plot

```
hist(ImpSub$Age, main="Distribution of age", xlab="Age")
```


Distribution of age
Frequency
0
40000
100000
0
20
40
60
80
100
Age