FE582_Assignment_1

Mugdha Kamat

9/21/2020

Problem 1

Explore realdirect.com thinking about how buyers and sellers would navigate, and how the website is organized. Use the datasets provided for Bronx, Brooklyn, Manhattan, Queens, and Staten Island.

Load in and clean the data.

Solution:

```
#getwd()
#setwd()
Sys.setenv(JAVA_HOME='C:\\Program Files\\Java\\jre1.8.0_171')
library('plyr')
library('gdata')
```

```
## gdata: read.xls support for 'XLS' (Excel 97-2004) files ENABLED.

##

## gdata: read.xls support for 'XLSX' (Excel 2007+) files ENABLED.

##
## Attaching package: 'gdata'

## The following object is masked from 'package:stats':
##
##     nobs

## The following object is masked from 'package:utils':
##
##     object.size

## The following object is masked from 'package:base':
##
##     startsWith
```

```
library('ggplot2')
library('xlsx')
alldatasets<- c("bronx","brooklyn","manhattan","queens","statenisland")

for (alldatasets in alldatasets){
  df=read.xls(paste("rollingsales_",alldatasets,".xls",sep=""),perl = "C:\\Strawberry\\perl\\bin\\perl.
  names(df)=tolower(names(df))

  #Data cleaning with regular expressions
  df$sale.price.n <-as.numeric(gsub("[^[:digit:]]","",df$sale.price))
  df$gross.sqft <- as.numeric(gsub("[^[:digit:]]","",df$gross.square.feet))
  df$land.sqft<- as.numeric(gsub("[^[:digit:]]","",df$land.square.feet))

  df$borough <- as.character(df$borough)

  df$sale.date <- as.Date(df$sale.date)

  df$year.built = as.numeric(as.character(df$year.built))
  if (alldatasets=="bronx"){ bx=df }
  else if (alldatasets=="brooklyn"){bk=df}
  else if (alldatasets=="manhattan"){mh=df}
  else if (alldatasets=="queens"){qn=df}
  else {si=df}
}
```

Combining all the boroughs data in one Dataframe.

```
allboroughs= rbind(bx,bk,mh,qn,si)
summary(allboroughs)
```

```
##    borough          neighborhood       building.class.category
##  Length:85975       Length:85975       Length:85975
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##  tax.class.at.present     block            lot           ease.ment
##  Length:85975         Min.   :    1   Min.   :   1.0   Length:85975
##  Class :character     1st Qu.: 1052   1st Qu.:  23.0   Class :character
##  Mode  :character     Median : 2157   Median :  51.0   Mode  :character
##                       Mean   : 3661   Mean   : 405.4
##                       3rd Qu.: 5599   3rd Qu.:1009.0
##                       Max.   :16323   Max.   :9117.0
##
##  building.class.at.present   address          apart.ment.number
##  Length:85975              Length:85975       Length:85975
##  Class :character          Class :character   Class :character
##  Mode  :character          Mode  :character   Mode  :character
##
```

```
##
##
##
##     zip.code     residential.units  commercial.units   total.units
##  Min.   :    0   Length:85975       Length:85975       Length:85975
##  1st Qu.:10028   Class :character   Class :character   Class :character
##  Median :11201   Mode  :character   Mode  :character   Mode  :character
##  Mean   :10758
##  3rd Qu.:11238
##  Max.   :11694
##
##  land.square.feet   gross.square.feet    year.built   tax.class.at.time.of.sale
##  Length:85975       Length:85975       Min.   :   0   Min.   :1.000
##  Class :character   Class :character   1st Qu.:1910   1st Qu.:1.000
##  Mode  :character   Mode  :character   Median :1931   Median :2.000
##                                        Mean   :1681   Mean   :1.868
##                                        3rd Qu.:1964   3rd Qu.:2.000
##                                        Max.   :2013   Max.   :4.000
##                                                       NA's   :1
##  building.class.at.time.of.sale  sale.price          sale.date
##  Length:85975                   Length:85975       Min.   :2012-08-01
##  Class :character               Class :character   1st Qu.:2012-11-07
##  Mode  :character               Mode  :character   Median :2013-01-24
##                                                    Mean   :2013-01-30
##                                                    3rd Qu.:2013-05-02
##                                                    Max.   :2013-08-26
##
##   sale.price.n         gross.sqft        land.sqft
##  Min.   :0.000e+00   Min.   :      0   Min.   :      0
##  1st Qu.:0.000e+00   1st Qu.:      0   1st Qu.:      0
##  Median :2.600e+05   Median :    650   Median :   1512
##  Mean   :8.851e+05   Mean   :   5050   Mean   :   3085
##  3rd Qu.:6.200e+05   3rd Qu.:   2344   3rd Qu.:   2625
##  Max.   :1.308e+09   Max.   :2548000   Max.   :7446955
##
```

```
str(allboroughs)
```

```
## 'data.frame':    85975 obs. of  24 variables:
##  $ borough                       : chr  "2" "2" "2" "2" ...
##  $ neighborhood                  : chr  "BATHGATE                 " "BATHGATE                 " "BATH
##  $ building.class.category       : chr  "01  ONE FAMILY HOMES                          " "01  ONE FAMI
##  $ tax.class.at.present          : chr  "1" "1" "1" "1" ...
##  $ block                         : int  3028 3039 3046 3046 2900 2912 2929 3030 3035 3039 ...
##  $ lot                           : int  25 28 39 52 61 158 117 60 27 65 ...
##  $ ease.ment                     : chr  NA NA NA NA ...
##  $ building.class.at.present     : chr  "A5" "A1" "A1" "A1" ...
##  $ address                       : chr  "412 EAST 179TH STREET                    " "2329 WASHINGTON
##  $ apart.ment.number             : chr  "            " "            " "            " "            "
##  $ zip.code                      : int  10457 10458 10457 10457 10457 10457 10457 10457 10457 10458
##  $ residential.units             : chr  "1" "1" "1" "1" ...
##  $ commercial.units              : chr  "0" "0" "0" "0" ...
##  $ total.units                   : chr  "1" "1" "1" "1" ...
##  $ land.square.feet              : chr  "1,842" "1,103" "1,986" "2,329" ...
```

```
##  $ gross.square.feet             : chr  "2,048" "1,290" "1,344" "1,431" ...
##  $ year.built                    : num  1901 1910 1899 1901 1931 ...
##  $ tax.class.at.time.of.sale     : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ building.class.at.time.of.sale: chr  "A5" "A1" "A1" "A1" ...
##  $ sale.price                    : chr  "$355,000" "$474,819" "$210,000" "$343,116" ...
##  $ sale.date                     : Date, format: "2013-07-08" "2013-05-20" ...
##  $ sale.price.n                  : num  355000 474819 210000 343116 0 ...
##  $ gross.sqft                    : num  2048 1290 1344 1431 4452 ...
##  $ land.sqft                     : num  1842 1103 1986 2329 1855 ...
```

Conclusion

From the summary, we can observe the minimum and maximum values for sales price,gross square feet year built,number of units and so on.str() function gives us an overview about the structure of the data.Here we can check the class and mode of all the columns and factorize certain columns for better analysis.

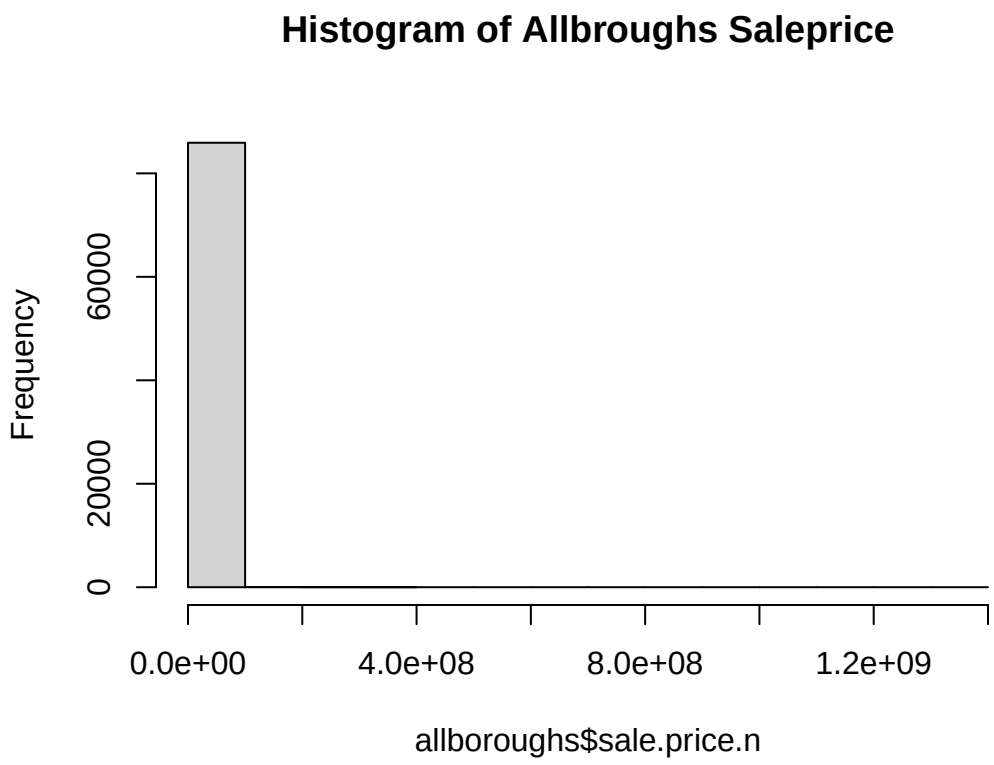
Conduct exploratory data analysis in order to find out where there are outliers or missing values, decide how you will treat them, make sure the dates are formatted correctly, make sure values you think are numerical are being treated as such, etc.

Solution:

```
par("mar")
```

```
## [1] 5.1 4.1 4.1 2.1
```

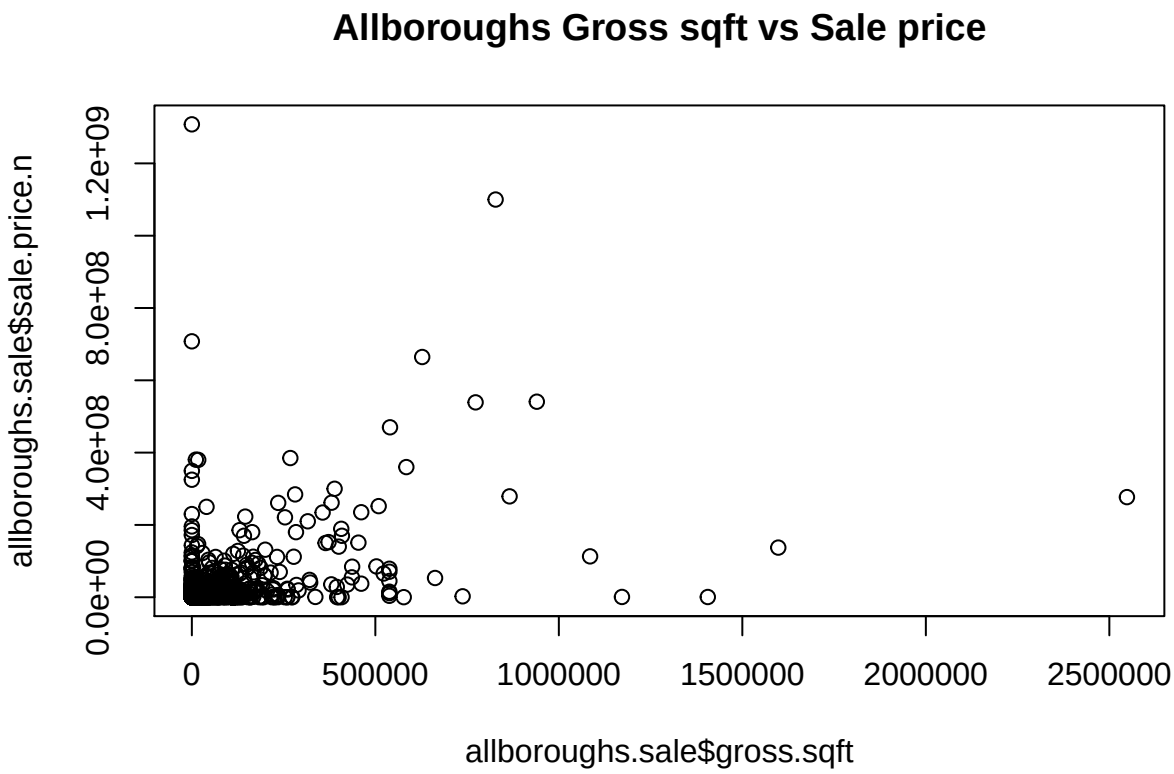
```
par(mar=c(5,5,5,5))
#checking if the data for all boroughs is correct
hist(main = "Histogram of Allbroughs Saleprice",allboroughs$sale.price.n)
```



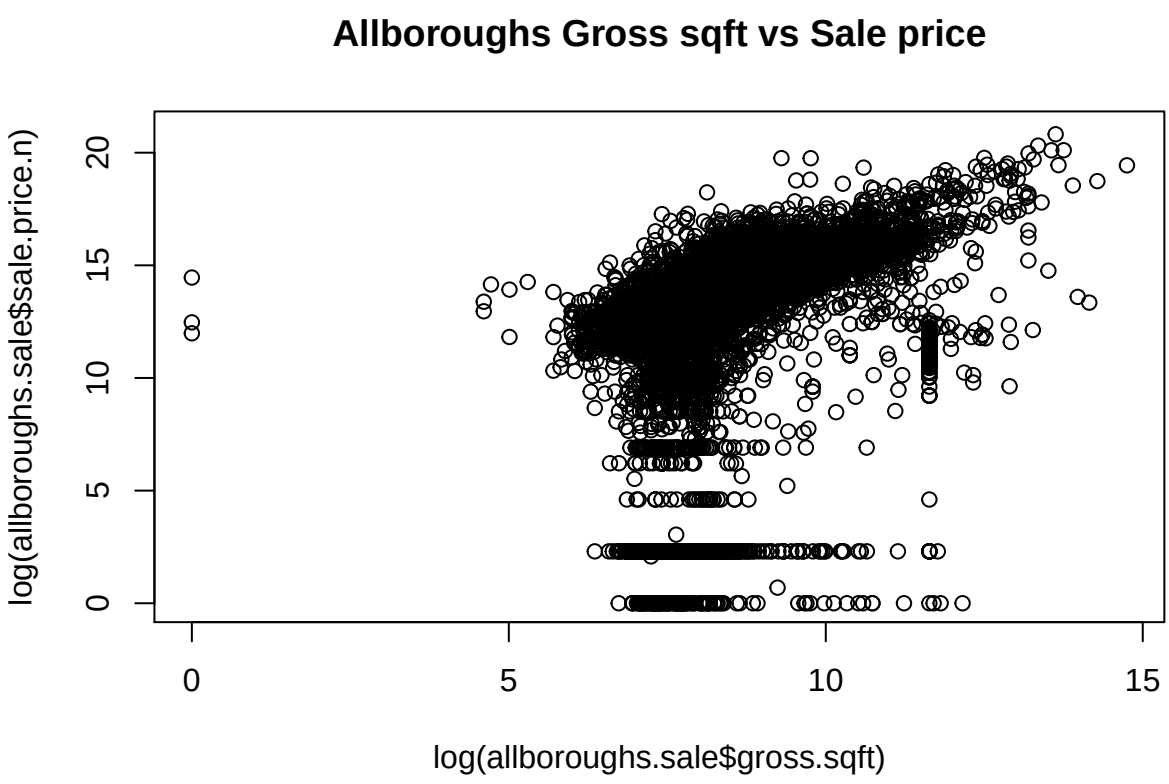
Conclusion

Here we observe that the range of the Sale prices is large, most of them being close to 0.Hence, for better analysis let's keep only the actual sales.

```
allboroughs.sale = allboroughs[allboroughs$sale.price.n!=0,]
plot(allboroughs.sale$gross.sqft,allboroughs.sale$sale.price.n, title(main="Allboroughs Gross sqft vs Sa
```




```
plot(log(allboroughs.sale$gross.sqft),log(allboroughs.sale$sale.price.n),title(main="Allboroughs Gross s
```



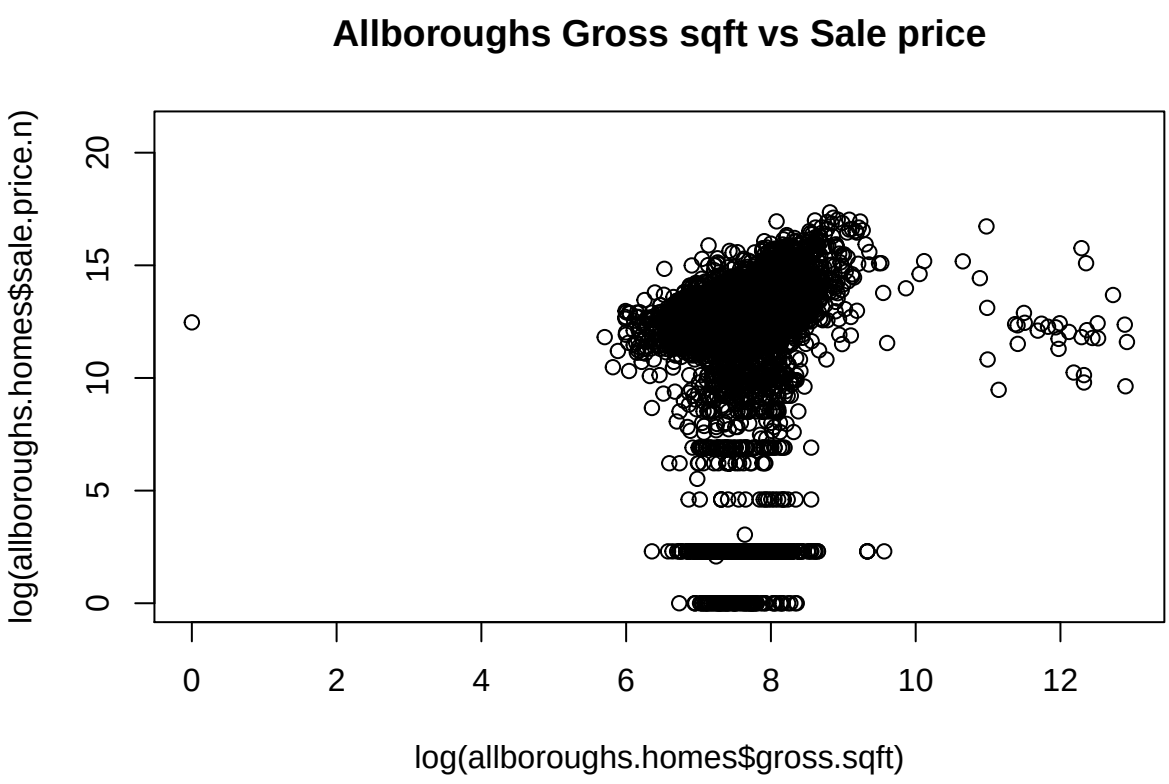
Taking log on both variables as most of the values were concentrated around 0.After taking log, the plot is much better for analysis. But there is noise at some values.

For now,considering only family homes, coops, and condos from the building category.

```
allboroughs.homes = allboroughs.sale [which(grepl("FAMILY | CONDOS | COOPS",
                                          allboroughs.sale$building.class.category)),]
#checking the building class category
unique(allboroughs.homes$building.class.category)
```

```
##  [1] "01  ONE FAMILY HOMES                      "
##  [2] "02  TWO FAMILY HOMES                      "
##  [3] "03  THREE FAMILY HOMES                    "
##  [4] "10  COOPS - ELEVATOR APARTMENTS           "
##  [5] "04  TAX CLASS 1 CONDOS                    "
##  [6] "28  COMMERCIAL CONDOS                     "
##  [7] "09  COOPS - WALKUP APARTMENTS             "
##  [8] "13  CONDOS - ELEVATOR APARTMENTS          "
##  [9] "12  CONDOS - WALKUP APARTMENTS            "
## [10] "15  CONDOS - 2-10 UNIT RESIDENTIAL        "
## [11] "16  CONDOS - 2-10 UNIT WITH COMMERCIAL UNIT "
```

```
plot(log(allboroughs.homes$gross.sqft),log(allboroughs.homes$sale.price.n),title(main="Allboroughs Gross
```

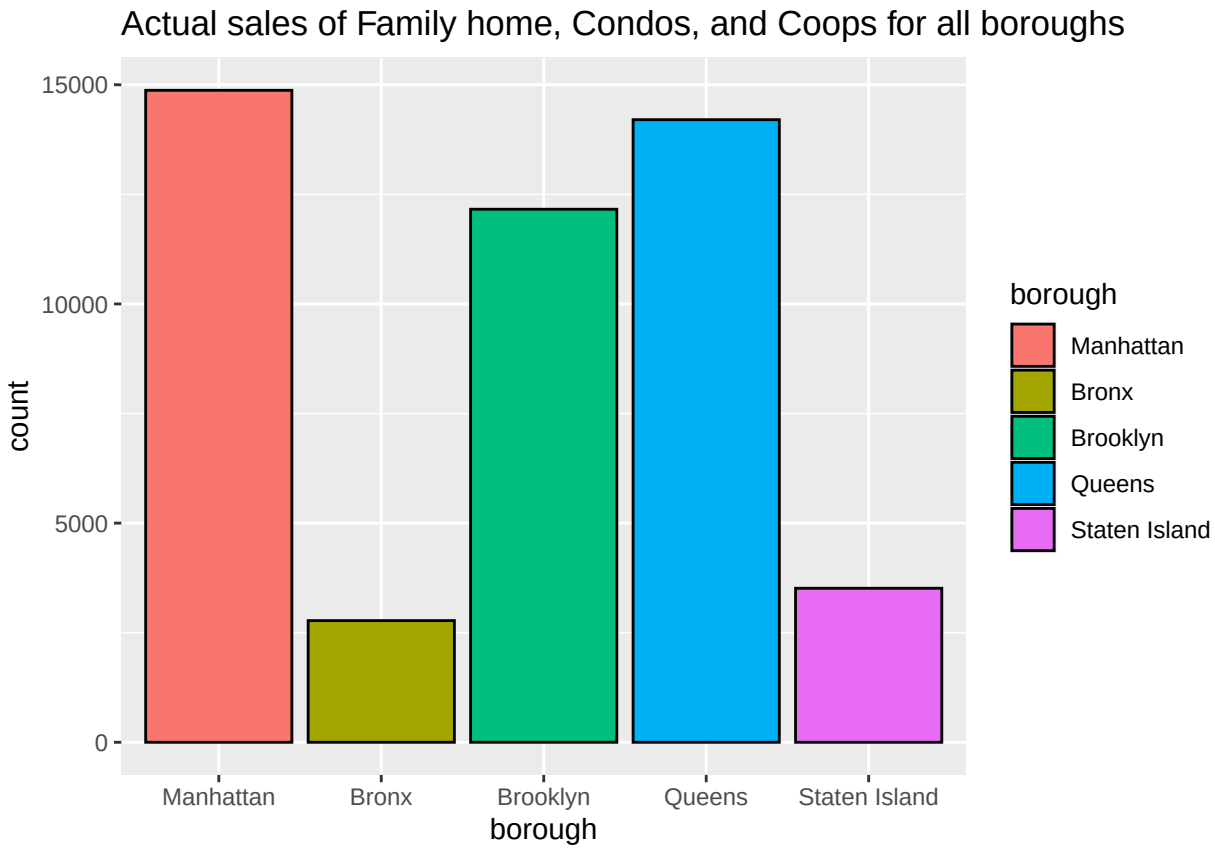



```
#labeling the boroughs
allboroughs.homes$borough <- factor(allboroughs.homes$borough,labels = c("Manhattan","Bronx","Brooklyn"
```

Conclusion

Much less noise appears now in the plot.

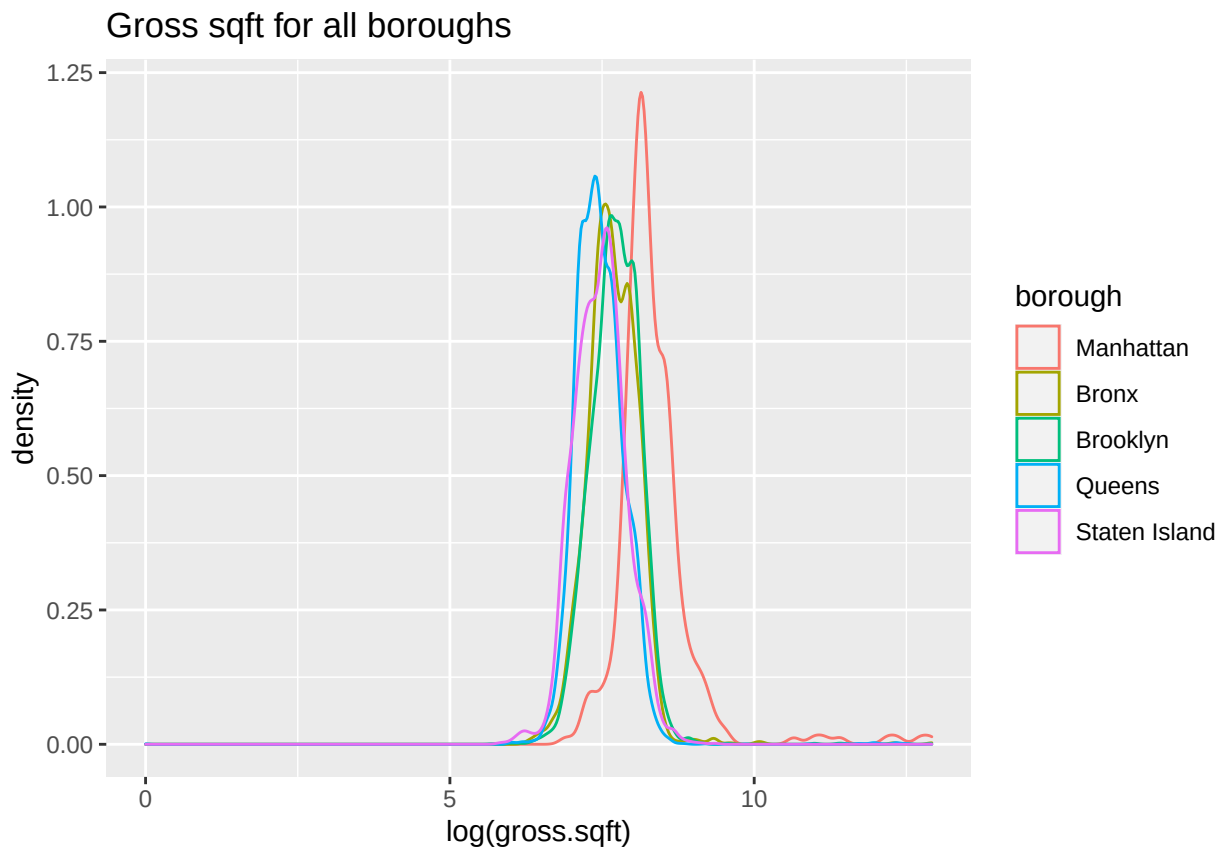
```
library("ggplot2")
ggplot(allboroughs.homes, aes(x=borough, fill=borough))+geom_bar(colour="black")+
  ggtitle("Actual sales of Family home, Condos, and Coops for all boroughs")
```



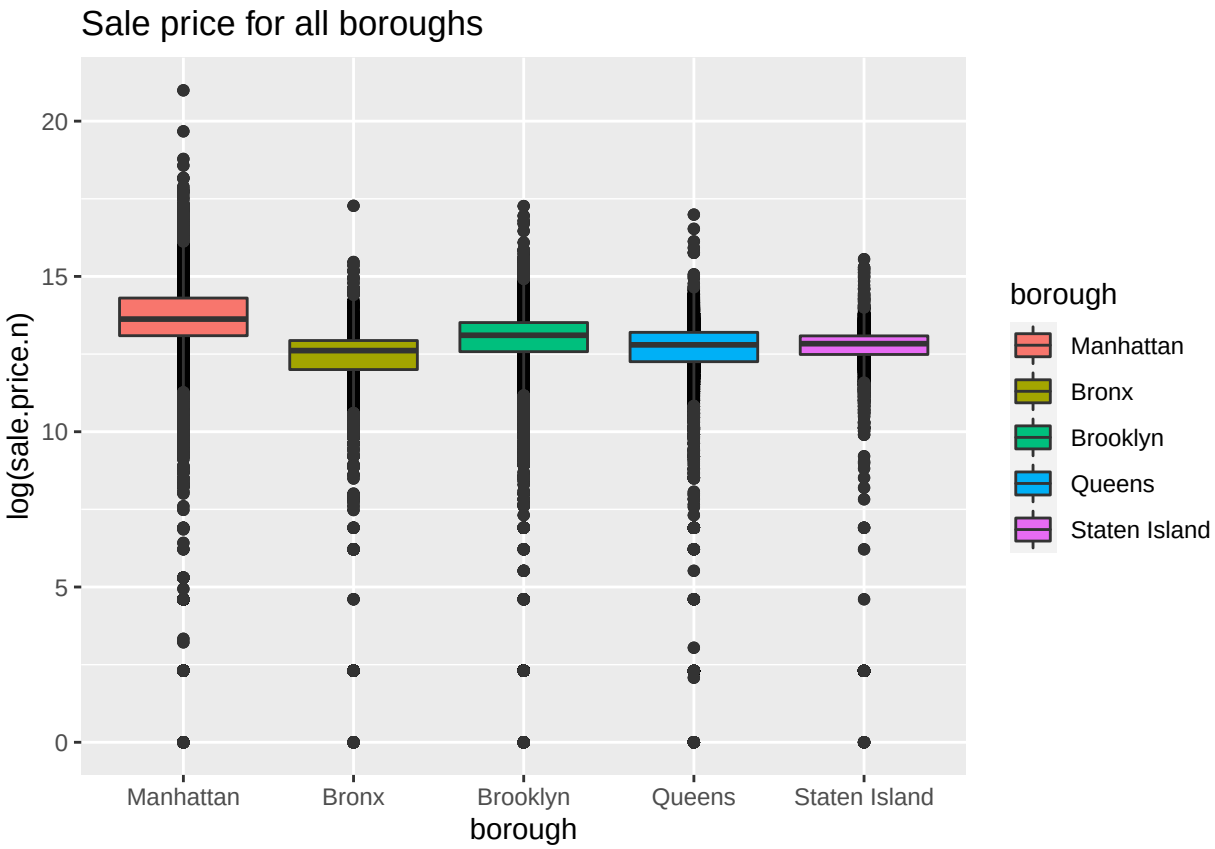
Conclusion

Here, we observe that Manhattan, followed by Queens has the highest actual sales of Family home, Condos, and Coops and Bronx has the lowest sales.

```
ggplot(subset(allboroughs.homes, gross.sqft>0), aes(x=log(gross.sqft),colour=borough))+geom_density()+
  ggtitle("Gross sqft for all boroughs")
```




```
ggplot(allboroughs.homes, aes(x=borough,y=log(sale.price.n),fill=borough))+geom_point()+geom_boxplot()+
  ggtitle("Sale price for all boroughs")
```



Conclusion

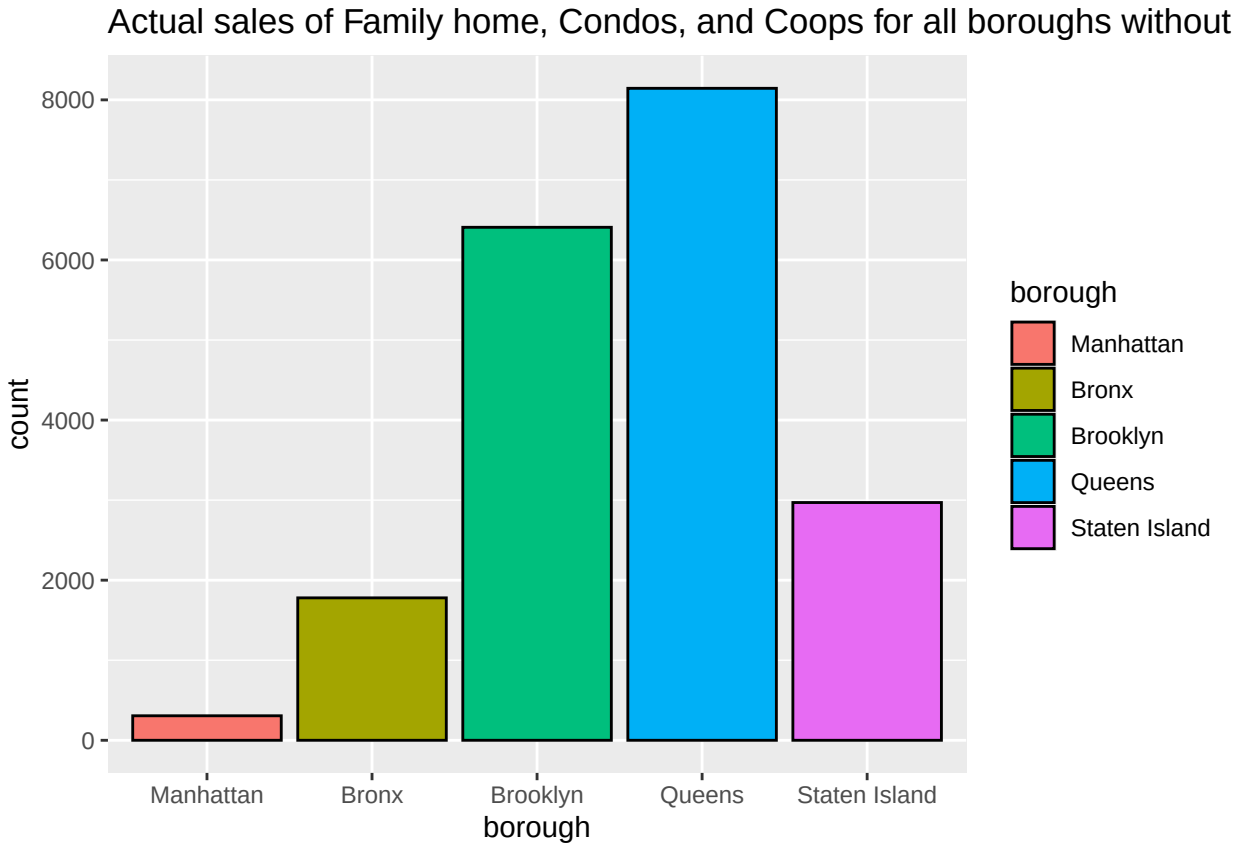
We observe that Manhattan has the highest actual sale price among all boroughs.Manhattan also has the highest mean and high kurtosis.The actual sale price of other boroughs have similar distributions.Now removing all the outliers as they have extreme values.

```
summary(allboroughs.homesnooutliers)
```

```
##           borough     neighborhood       building.class.category
##  Manhattan     : 306   Length:19607       Length:19607
##  Bronx         :1779   Class :character   Class :character
##  Brooklyn      :6408   Mode  :character   Mode  :character
##  Queens        :8144
##  Staten Island:2970
##
##
##  tax.class.at.present     block            lot            ease.ment
##  Length:19607         Min.   :   14   Min.   :   1.00   Length:19607
##  Class :character     1st Qu.: 2786   1st Qu.:  20.00   Class :character
##  Mode  :character     Median : 5293   Median :  39.00   Mode  :character
##                       Mean   : 5647   Mean   :  61.19
##                       3rd Qu.: 7857   3rd Qu.:  64.00
##                       Max.   :16320   Max.   :3601.00
##
##  building.class.at.present   address          apart.ment.number
##  Length:19607              Length:19607       Length:19607
##  Class :character          Class :character   Class :character
```

```
##  Mode  :character          Mode  :character   Mode  :character
##
##
##
##
##     zip.code     residential.units  commercial.units   total.units
##  Min.   :10001   Length:19607       Length:19607       Length:19607
##  1st Qu.:10472   Class :character   Class :character   Class :character
##  Median :11228   Mode  :character   Mode  :character   Mode  :character
##  Mean   :11065
##  3rd Qu.:11375
##  Max.   :11694
##
##  land.square.feet   gross.square.feet    year.built   tax.class.at.time.of.sale
##  Length:19607       Length:19607       Min.   :   0   Min.   :1.000
##  Class :character   Class :character   1st Qu.:1920   1st Qu.:1.000
##  Mode  :character   Mode  :character   Median :1930   Median :1.000
##                                        Mean   :1939   Mean   :1.002
##                                        3rd Qu.:1955   3rd Qu.:1.000
##                                        Max.   :2012   Max.   :2.000
##                                                       NA's   :1
##  building.class.at.time.of.sale  sale.price          sale.date
##  Length:19607                   Length:19607       Min.   :2012-08-01
##  Class :character               Class :character   1st Qu.:2012-11-08
##  Mode  :character               Mode  :character   Median :2013-02-04
##                                                    Mean   :2013-02-02
##                                                    3rd Qu.:2013-05-07
##                                                    Max.   :2013-08-16
##
##   sale.price.n        gross.sqft       land.sqft         outliers
##  Min.   :     250   Min.   :     1   Min.   :   340   Min.   :0
##  1st Qu.:  325000   1st Qu.:  1448   1st Qu.:  1999   1st Qu.:0
##  Median :  455000   Median :  1936   Median :  2500   Median :0
##  Mean   :  597527   Mean   :  2424   Mean   :  3128   Mean   :0
##  3rd Qu.:  649000   3rd Qu.:  2609   3rd Qu.:  3775   3rd Qu.:0
##  Max.   :34350000   Max.   :407920   Max.   :333700   Max.   :0
##
```

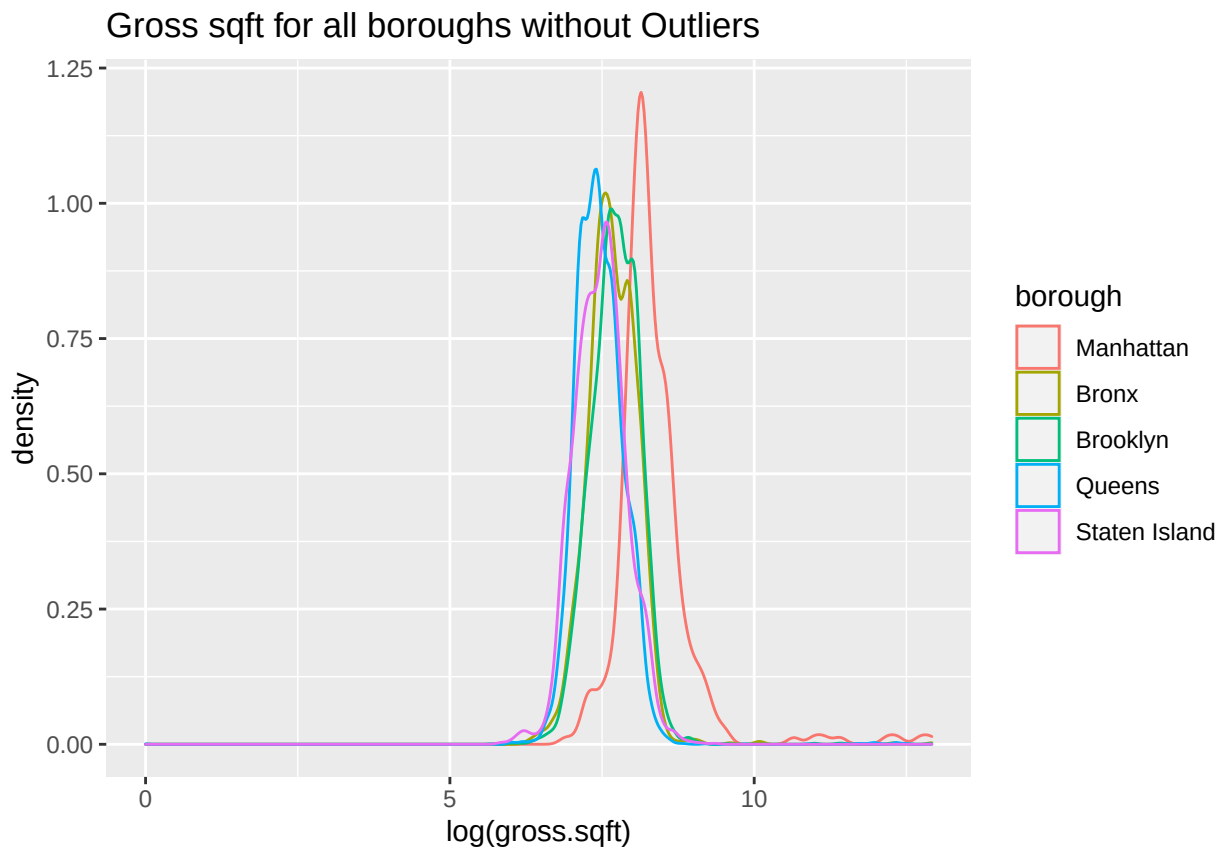
```
ggplot(allboroughs.homesnooutliers, aes(x=borough, fill=borough))+geom_bar(colour="black")+
  ggtitle("Actual sales of Family home, Condos, and Coops for all boroughs without Outliers")
```



Conclusion

After removing outliers,we observe that Manhattan has now the lowest actual sales across the building category and Queens has the highest actual sales now.

```
ggplot(subset(allboroughs.homesnooutliers, gross.sqft>0), aes(x=log(gross.sqft),colour=borough))+geom_d
  ggtitle("Gross sqft for all boroughs without Outliers")
```

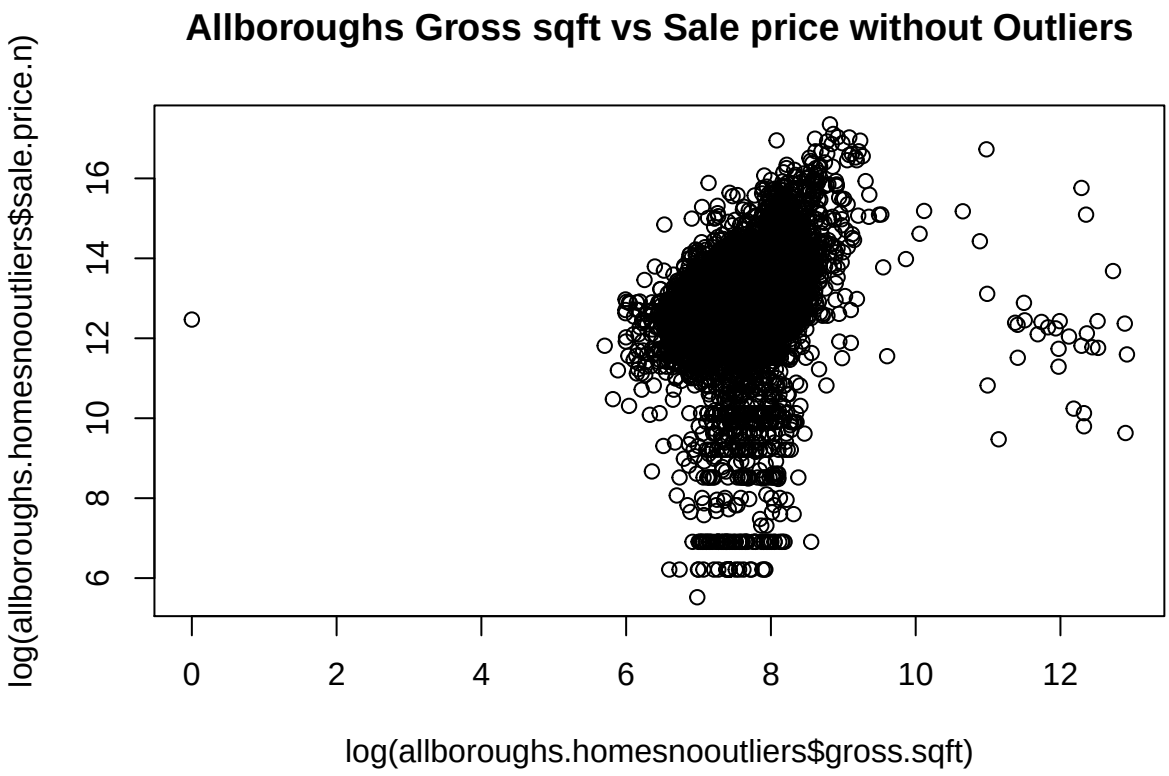



```
ggplot(allboroughs.homesnooutliers, aes(x=borough,y=log(sale.price.n),fill=borough))+geom_point()+geom_
  ggtitle("Sale price for all boroughs without Outliers")
```

Conclusion

The boxplot for Manhattan shows that it has the highest sale price for all quartiles as well as its median is high.

```
plot(log(allboroughs.homesnooutliers$gross.sqft),log(allboroughs.homesnooutliers$sale.price.n),title(ma
```



Conduct exploratory data analysis to visualize and make comparisons for residential building category classes across boroughs and across time

```
summary(allboroughs.homes$year.built)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0    1920    1940    1789    1963    2012
```

Now, categorizing the Year built to visualize the data across time

```
# For years >0
allboroughs.homes<- allboroughs.homes[allboroughs.homes$year.built!=0,]
summary(allboroughs.homes$year.built)
```

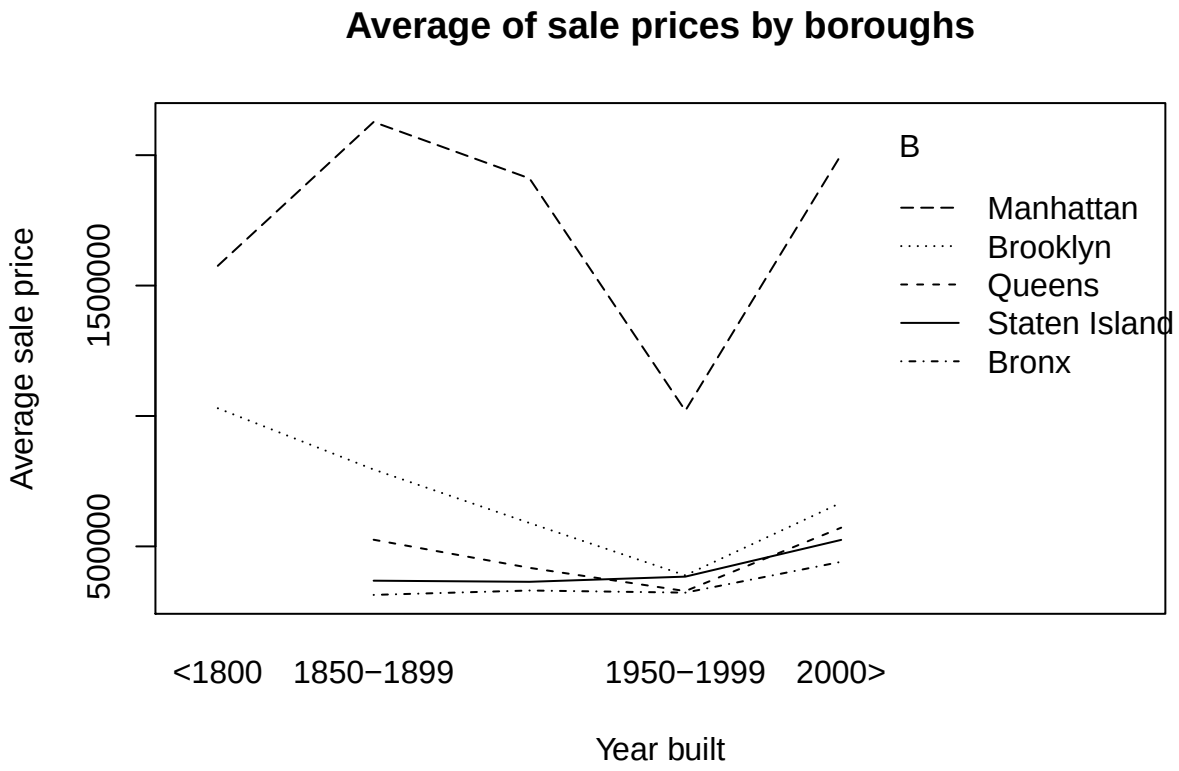
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1800    1925    1945    1948    1965    2012
```

```
allboroughs.homes$yearscat <- cut(allboroughs.homes$year.built,c(0,1850,1900,1950,2000,Inf),labels <-c(
#allboroughs.homes$yearscat
```

Conclusion

Here,after categorizing, now we can analyze the relationship between sale price and year built.

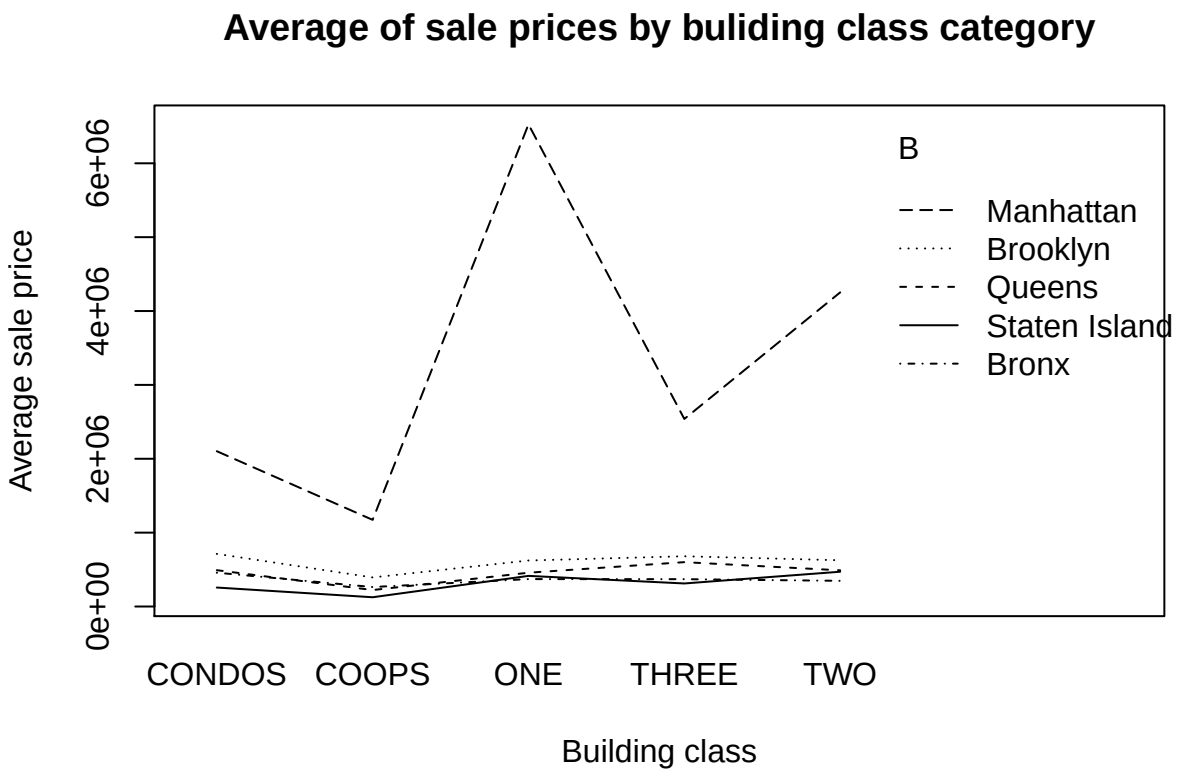
```
means <- with(allboroughs.homes,aggregate(x=list(Y=sale.price.n),by=list(A=yearscat, B=borough),mean))
with(means,interaction.plot(x.factor=A, trace.factor=B, response=Y, type='l',col=B,main ="Average of sa
```



Conclusion

Here,we observe that the average sale price for all the years in which buildings were built is similar for all the boroughs except Manhattan. For Manhattan, the sale price for the buildings built in years 1850 to 1859 is highest and trend continued with some increase from 2000.

```
allboroughs.homes$newbuildingcat = allboroughs.homes$building.class.category
tt = as.vector(allboroughs.homes$newbuildingcat)
tt[grep("COOPS", tt)] = "COOPS"
tt[grep("CONDOS",tt)] = "CONDOS"
tt[grep("ONE FAMILY",tt)] = "ONE"
tt[grep("TWO FAMILY",tt)] = "TWO"
tt[grep("THREE FAMILY",tt)] = "THREE"
allboroughs.homes$newbuildingcat = tt
homes =with(allboroughs.homes,aggregate(x=list(Y=sale.price.n),by=list(A=newbuildingcat, B=borough),mea
with(homes,interaction.plot(x.factor=A, trace.factor=B, response=Y, type='l',
main ="Average of sale prices by buliding class category",xlab = "Building class",ylab = "Average sale
```



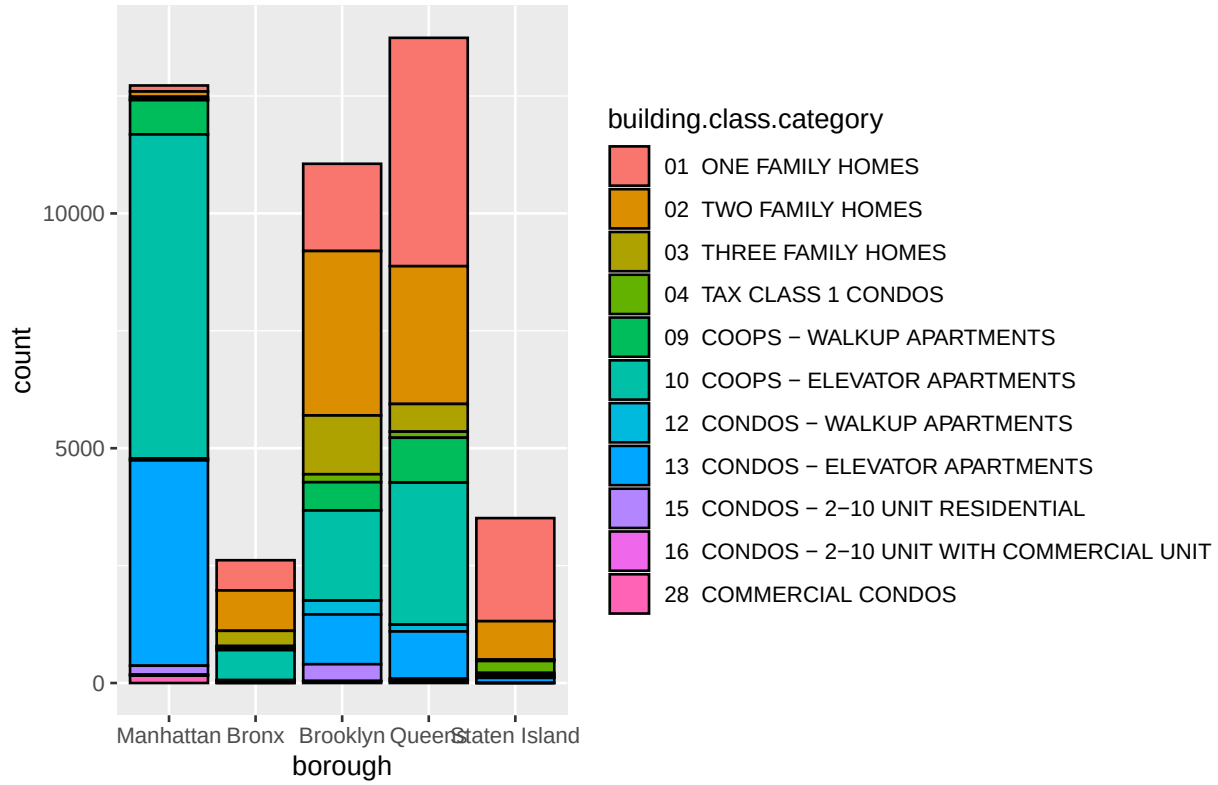
Conclusion

In this plot too we observe that the trend for Manhattan is different than others.

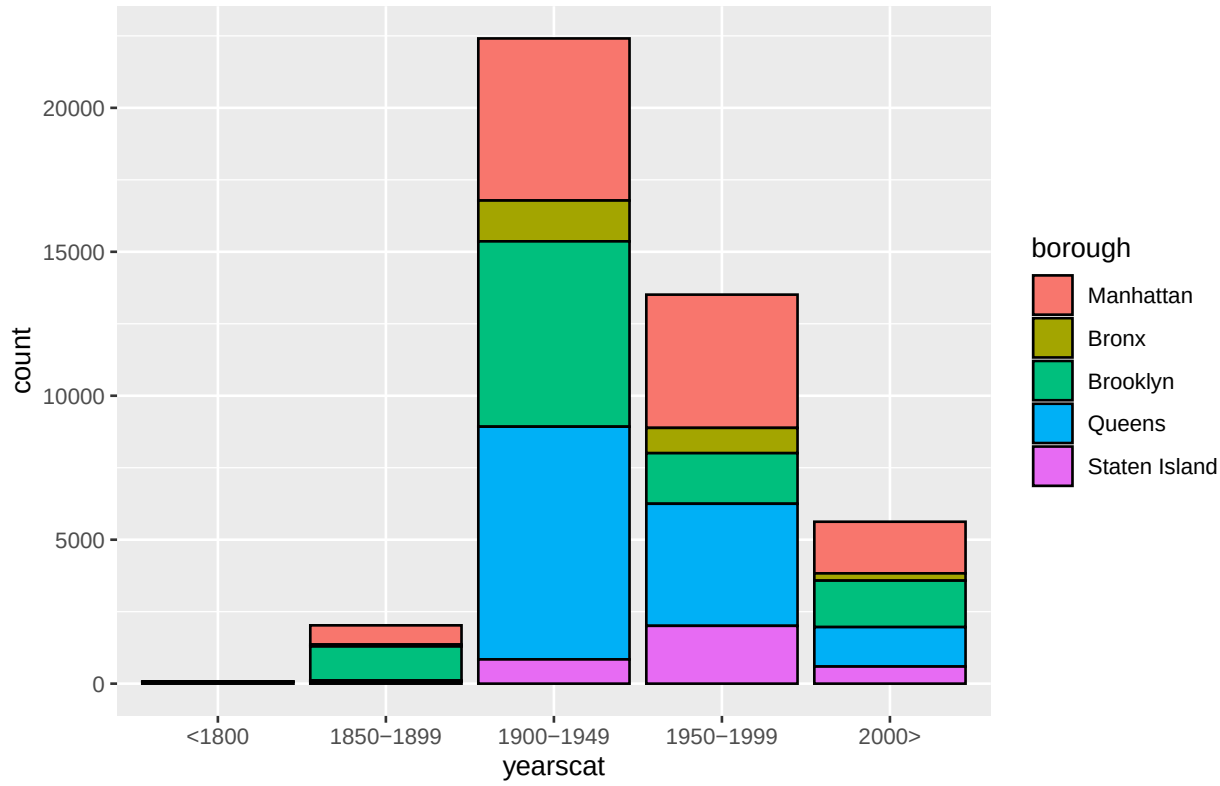
```
par("mar")
```

```
## [1] 5.1 4.1 4.1 2.1
```

```
par(mar=c(6,6,6,6))
ggplot(allboroughs.homes , aes(x=borough, fill=building.class.category))+ geom_bar(colour="black")
```




```
ggplot(allboroughs.homes , aes(x=yearscat, fill=borough))+ geom_bar(colour="black")
```



```
ggplot(allboroughs.homes , aes(x=yearscat, fill=building.class.category))+ geom_bar(colour="black")
```

Conclusion

We can observe the distribution of the boroughs, years built and building class category acorss the years built.Highest number of boroughs are built in the period 1900-1950

Problem 2

The datasets provided nyt1.csv, nyt2.csv, and nyt3.csv represents three (simulated) days of ads shown and clicks recorded on the New York Times homepage. Each row represents a single user. There are 5 columns: age, gender (0=female, 1=male), number impressions, number clicks, and logged-in. Use R to handle this data. Perform some exploratory data analysis:

Create a new variable, age_group, that categorizes users as "<20", "20-29","30-39", "40-49", "50-59", "60-69", and "70+"

Solution:

Loading and Cleaning the data

```
#getwd()
#setwd()
nyt<-c('nyt1','nyt2','nyt3')
for (nyt in nyt){

  data=read.csv(paste(nyt,".csv",sep=""))
```

```
  #categorization of Age  and Signed_In/Not
  data$age_group <- cut(data$Age, c(-Inf,20,29,39,49,59,69,Inf))
  levels(data$age_group)<-c("<20","20-29","30-39","40-49","50-59","60-69","70+")
  data$Signed_In<- factor(data$Signed_In)

  if (nyt=="nyt1"){ data1=data }
  else if (nyt=="nyt2"){data2=data}
  else {data3=data}
}
```

```
head(data1)
```

```
##   Age Gender Impressions Clicks Signed_In age_group
## 1  36      0           3      0         1     30-39
## 2  73      1           3      0         1       70+
## 3  30      0           3      0         1     30-39
## 4  49      1           3      0         1     40-49
## 5  47      1          11      0         1     40-49
## 6  47      0          11      1         1     40-49
```

```
summary(data1)
```

```
##       Age             Gender       Impressions         Clicks        Signed_In
##  Min.   :  0.00   Min.   :0.000   Min.   : 0.000   Min.   :0.00000   0:137106
##  1st Qu.:  0.00   1st Qu.:0.000   1st Qu.: 3.000   1st Qu.:0.00000   1:321335
##  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
##
##  age_group
##  <20  :169204
##  20-29: 51378
##  30-39: 64763
##  40-49: 67565
##  50-59: 54406
##  60-69: 32358
##  70+  : 18767
```

Conclusion

Categorizing users based on Gender: We know that the observations for a person who is not Signed_In are 0 for both Gender and Age group, making it impossible to know the Gender.Hence adding one more category as Unknown along with Male and Female for all observations where the user is not signed in.

```
data1$Gender_code[data1$Gender==0]<-"Female"
data1$Gender_code[data1$Gender>0]<-"Male"
data1$Gender_code[data1$Signed_In==0]<-"Unknown"
data1$Gender_code<- factor(data1$Gender_code)

data2$Gender_code[data2$Gender==0]<-"Female"
```

```
data2$Gender_code[data2$Gender>0]<-"Male"
data2$Gender_code[data2$Signed_In==0]<-"Unknown"
data2$Gender_code<- factor(data2$Gender_code)

data3$Gender_code[data3$Gender==0]<-"Female"
data3$Gender_code[data3$Gender>0]<-"Male"
data3$Gender_code[data3$Signed_In==0]<-"Unknown"
data3$Gender_code<- factor(data3$Gender_code)

summary(data1)
```

```
##       Age              Gender       Impressions         Clicks        Signed_In
##  Min.   :  0.00   Min.   :0.000   Min.   : 0.000   Min.   :0.00000   0:137106
##  1st Qu.:  0.00   1st Qu.:0.000   1st Qu.: 3.000   1st Qu.:0.00000   1:321335
##  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
##
##  age_group       Gender_code
##  <20  :169204   Female :153070
##  20-29: 51378   Male   :168265
##  30-39: 64763   Unknown:137106
##  40-49: 67565
##  50-59: 54406
##  60-69: 32358
##  70+  : 18767
```

```
head(data3)
```

```
##   Age Gender Impressions Clicks Signed_In age_group Gender_code
## 1  46      1           3      0         1     40-49        Male
## 2  75      0           9      0         1       70+      Female
## 3  39      0           2      0         1     30-39      Female
## 4  54      0           4      0         1     50-59      Female
## 5  15      1           3      0         1       <20        Male
## 6  50      0           8      0         1     50-59      Female
```

Conclusion

We can observe that the data is more understandable now that we have categorized users based on Gender.

Define a new variable to segment or categorize users based on their click behavior

Categorizing Users who produced an Impression/Click

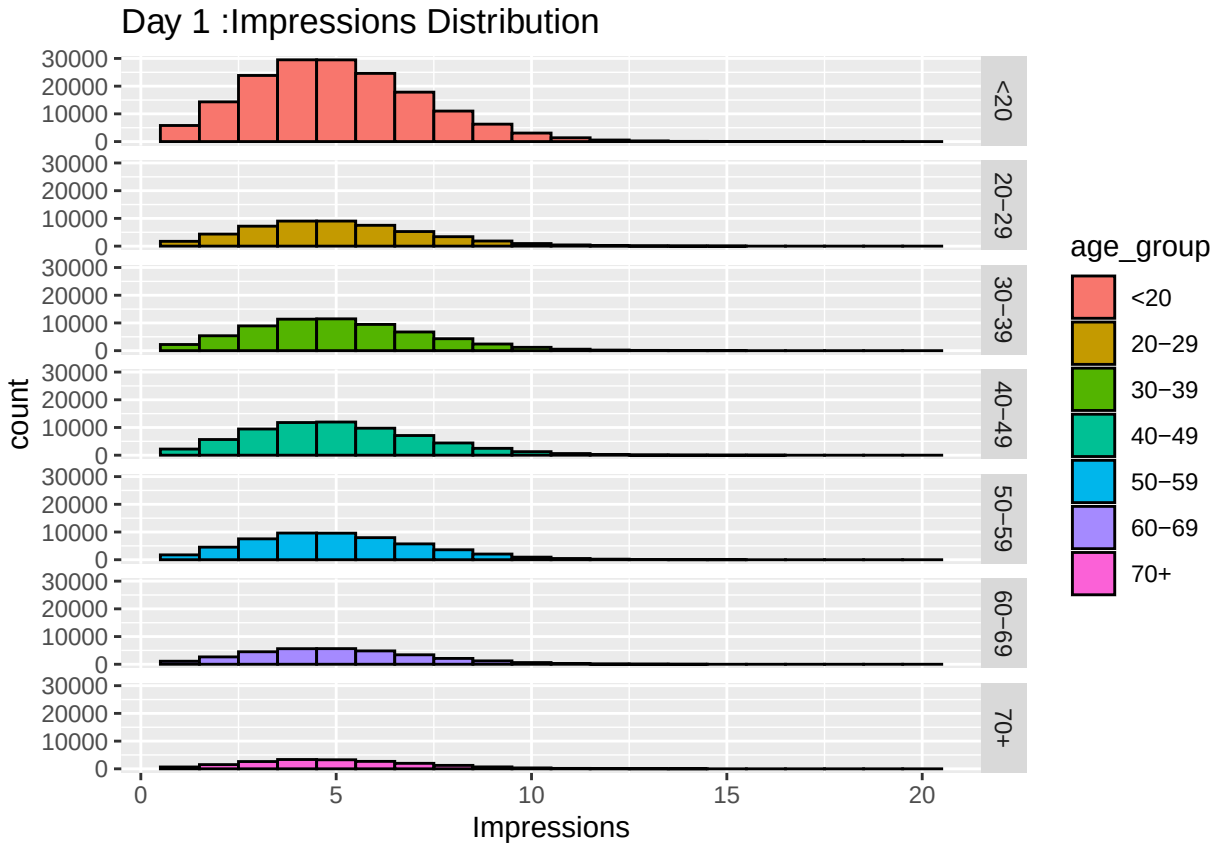
```
data1$scode[data1$Impressions==0] <- "NoImps"
data1$scode[data1$Impressions >0] <- "Imps"
data1$scode[data1$Clicks >0] <- "Clicks"

data2$scode[data2$Impressions==0] <- "NoImps"
```

```
data2$scode[data2$Impressions >0] <- "Imps"
data2$scode[data2$Clicks >0] <- "Clicks"

data3$scode[data3$Impressions==0] <- "NoImps"
data3$scode[data3$Impressions >0] <- "Imps"
data3$scode[data3$Clicks >0] <- "Clicks"

data1$scode <- factor(data1$scode)
data2$scode <- factor(data2$scode)
data3$scode <- factor(data3$scode)

head(data3)
```

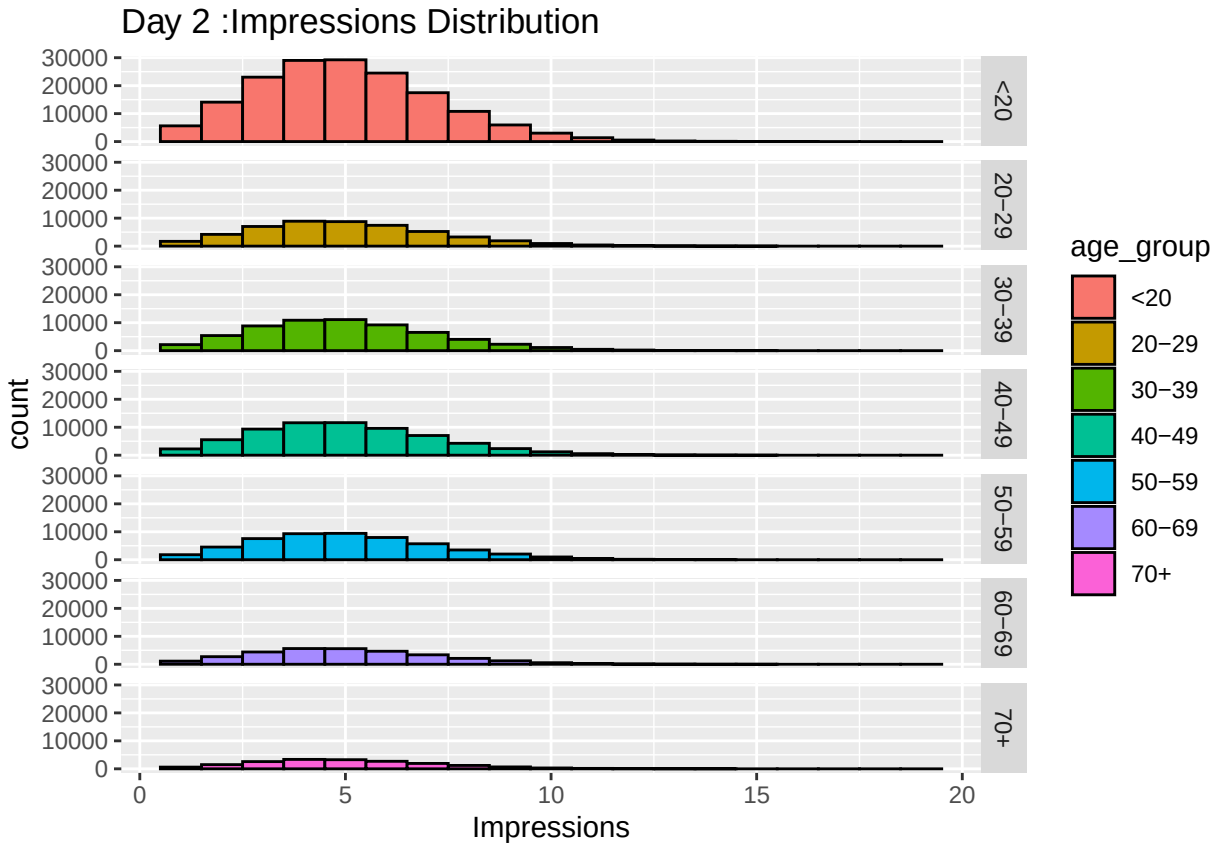
```
##   Age Gender Impressions Clicks Signed_In age_group Gender_code scode
## 1  46      1           3      0         1     40-49        Male  Imps
## 2  75      0           9      0         1       70+      Female  Imps
## 3  39      0           2      0         1     30-39      Female  Imps
## 4  54      0           4      0         1     50-59      Female  Imps
## 5  15      1           3      0         1       <20        Male  Imps
## 6  50      0           8      0         1     50-59      Female  Imps
```

For each day, Plot the distribution of number of impressions and click-through-rate (CTR =clicks/impressions) for these age categories

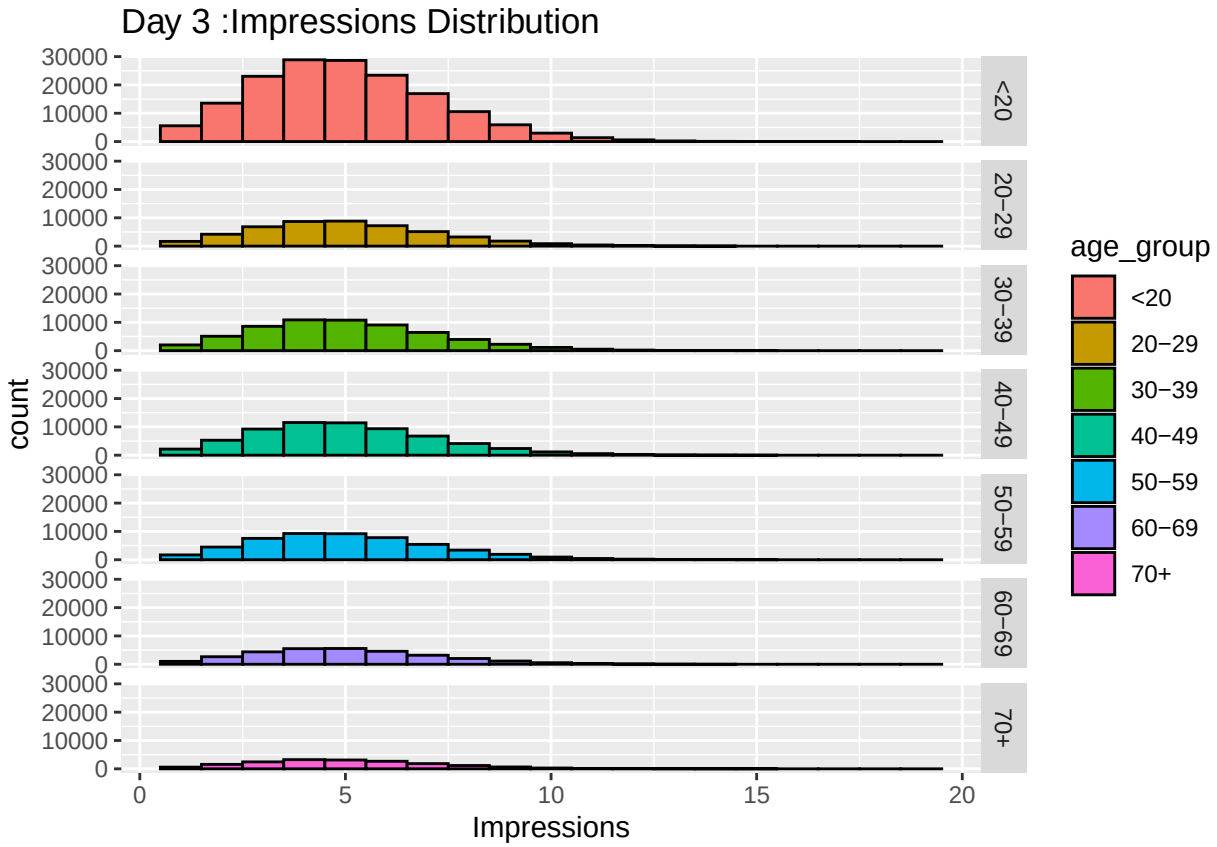
```
library(ggplot2)
ggplot(subset(data1, Impressions>0), aes(x=Impressions,fill=age_group))+
  geom_histogram(position="dodge",binwidth=1,colour="Black")+
  labs(title = "Day 1 :Impressions Distribution") +  facet_grid(age_group ~ .)
```




```
ggplot(subset(data2, Impressions>0), aes(x=Impressions,fill=age_group))+
  geom_histogram(position="dodge",binwidth=1,colour="Black")+
  labs(title = "Day 2 :Impressions Distribution") +  facet_grid(age_group ~ .)
```




```
ggplot(subset(data3, Impressions>0), aes(x=Impressions,fill=age_group))+
  geom_histogram(position="dodge",binwidth=1,colour="Black")+
  labs(title = "Day 3 :Impressions Distribution") +  facet_grid(age_group ~ .)
```

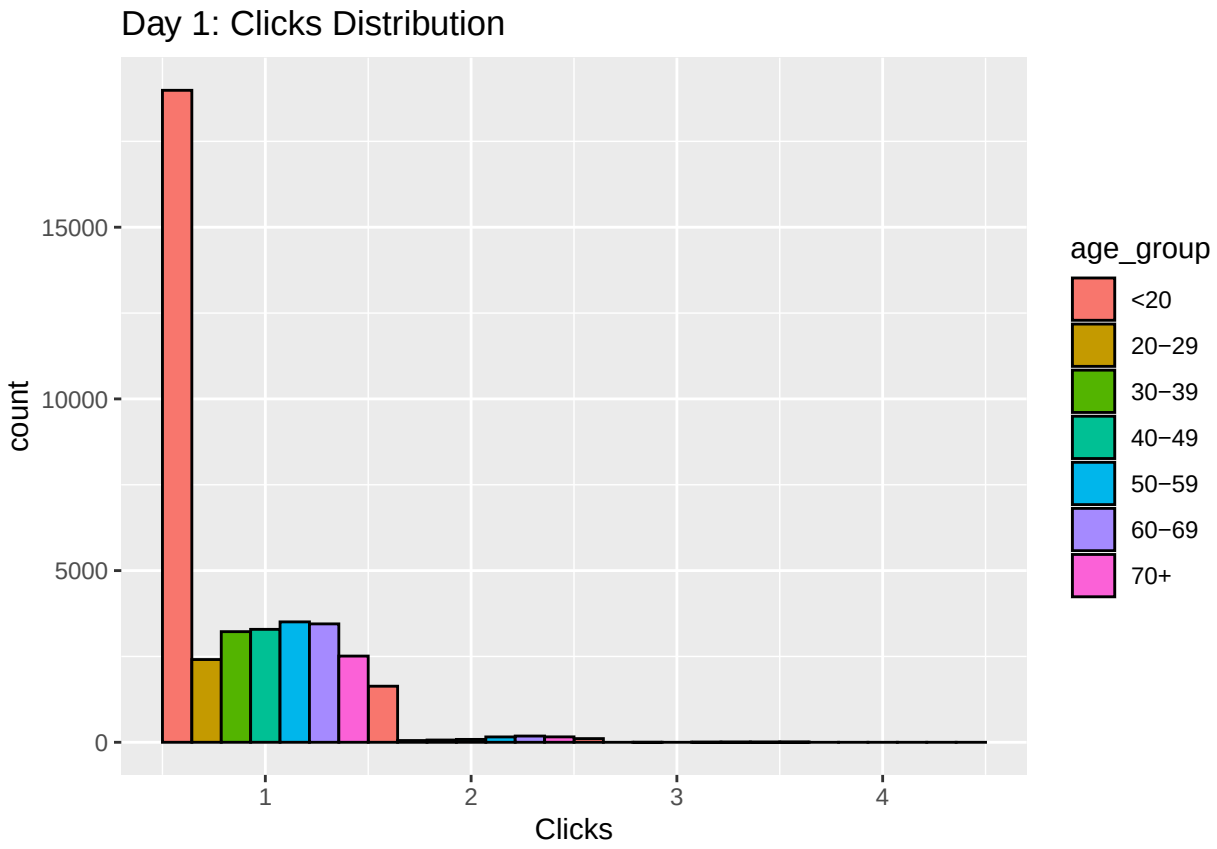


Explore the data and make visual and quantitative comparisons across user segments/demographics

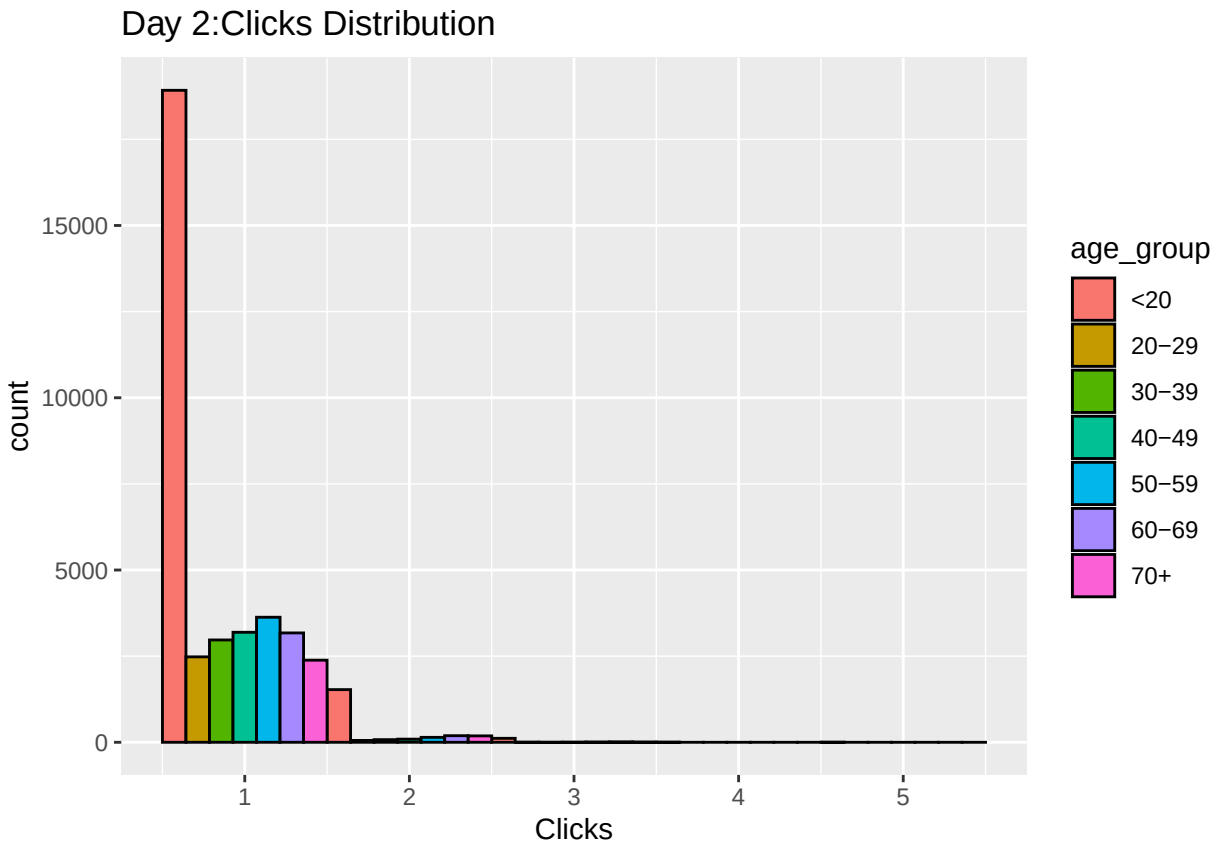
Conclusion

We can conclude that there is Normal Distribution of Impressions for all 3 days and the age group <20 have the maximum Impressions.Also,age group 70+ (senior citizens) have the least Impressions.

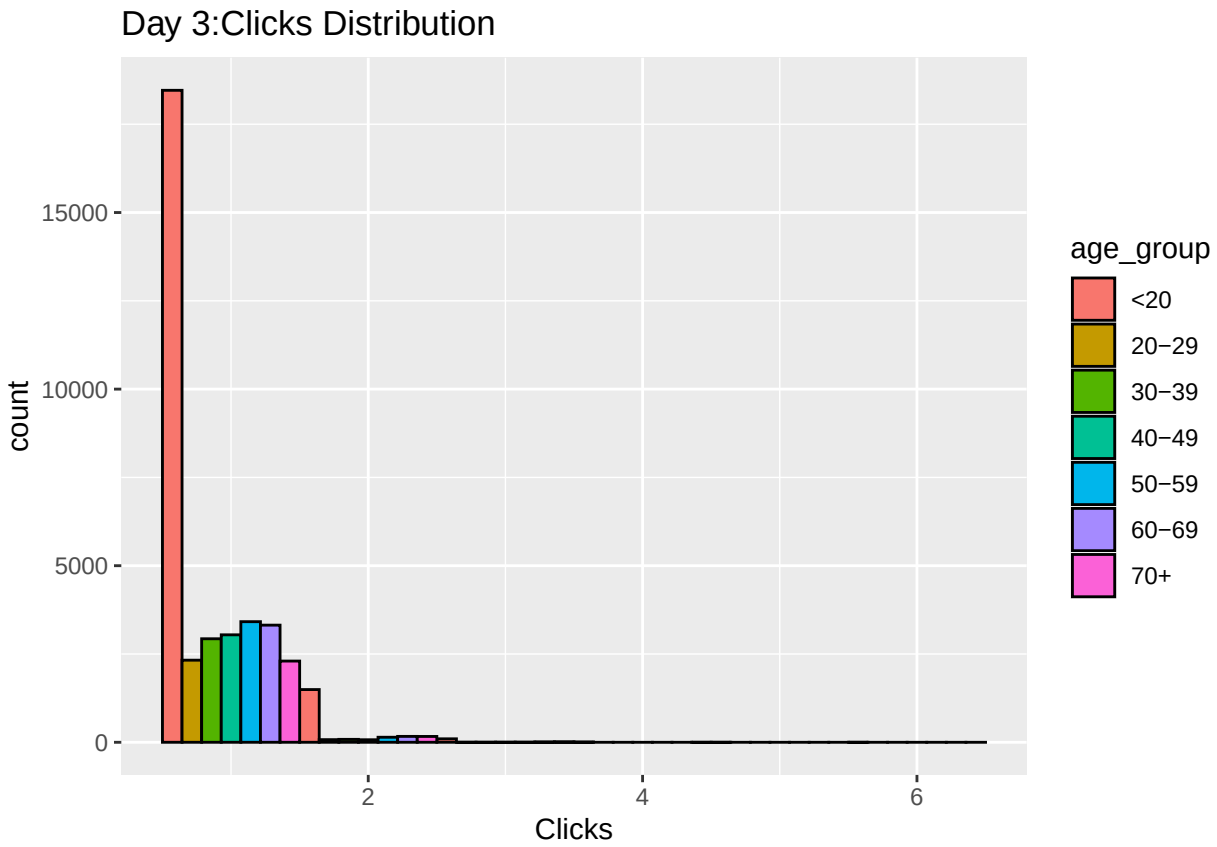
```
#Plotting the distribution of number of Clicks
ggplot(subset(data1, Clicks>0), aes(x=Clicks,fill=age_group))+
  geom_histogram(position="dodge",binwidth=1,colour="Black")+
  labs(title = "Day 1: Clicks Distribution")
```




```
ggplot(subset(data2, Clicks>0), aes(x=Clicks,fill=age_group))+
  geom_histogram(position="dodge",binwidth=1,colour="Black")+
  labs(title = "Day 2:Clicks Distribution")
```



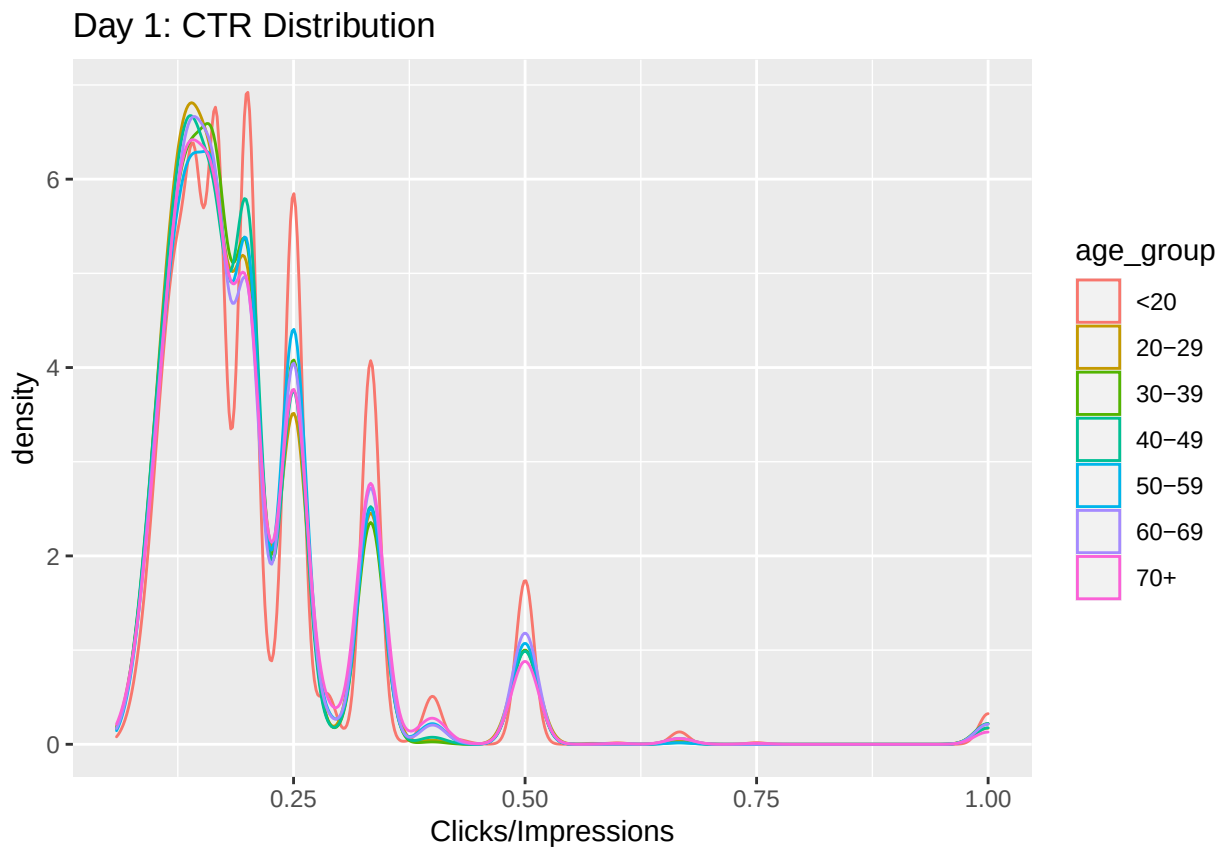

```
ggplot(subset(data3, Clicks>0), aes(x=Clicks,fill=age_group))+
  geom_histogram(position="dodge",binwidth=1,colour="Black")+
  labs(title = "Day 3:Clicks Distribution")
```



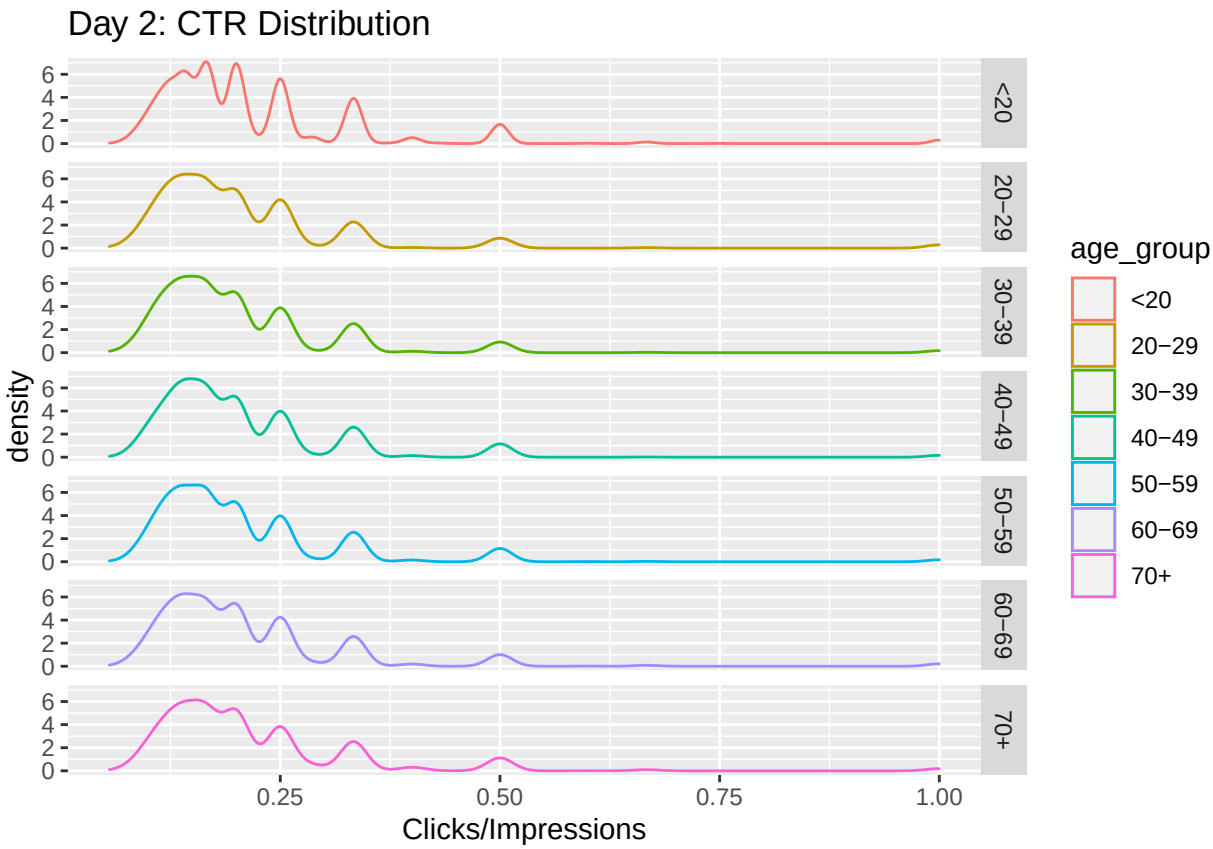
Conclusion

We can conclude that maximum Clicks are from age group <20 for all 3 days.As we have seen earlier that this age group had the most impressions as well,and hence they have clicked most on the ads than the other age groups.

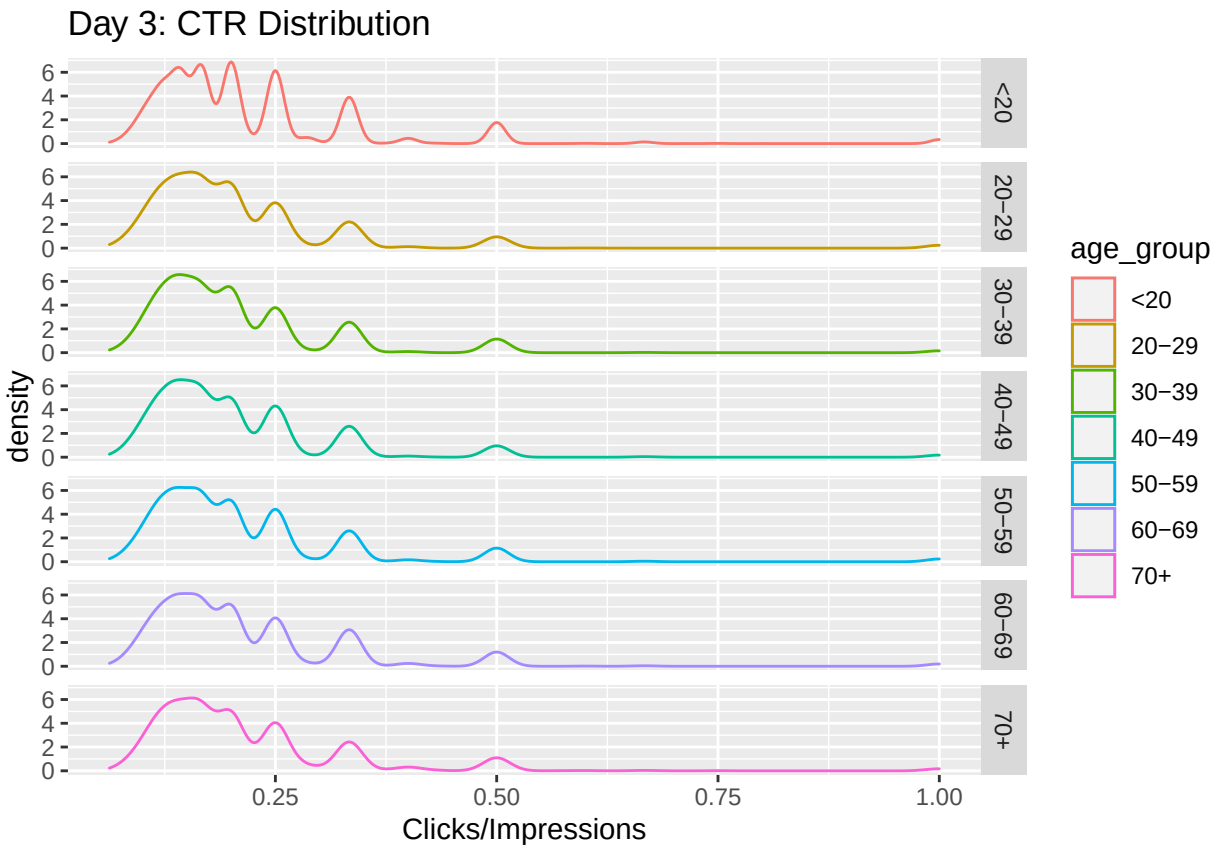
```
#plotting CTR

ggplot(subset(data1, Clicks>0 ), aes(x=Clicks/Impressions,colour=age_group))+
  geom_density() + labs(title = "Day 1: CTR Distribution")
```




```
ggplot(subset(data2, Clicks>0 & Impressions>0), aes(x=Clicks/Impressions,colour=age_group))+
  geom_density() + labs(title = "Day 2: CTR Distribution")  +facet_grid(age_group ~ .)
```




```
ggplot(subset(data3, Clicks>0 & Impressions>0), aes(x=Clicks/Impressions,colour=age_group))+
  geom_density() +  labs(title = "Day 3: CTR Distribution")+ facet_grid(age_group ~ .)
```



Conclusion

The distribution is right skewed. Also,it is evident from the plot that the maximum CTR is 0.25 and hence few people click on any Impressions popping on the Website.

Extend your analysis across days. Visualize some metrics and distributions over time

Combining three data sets into one data frame

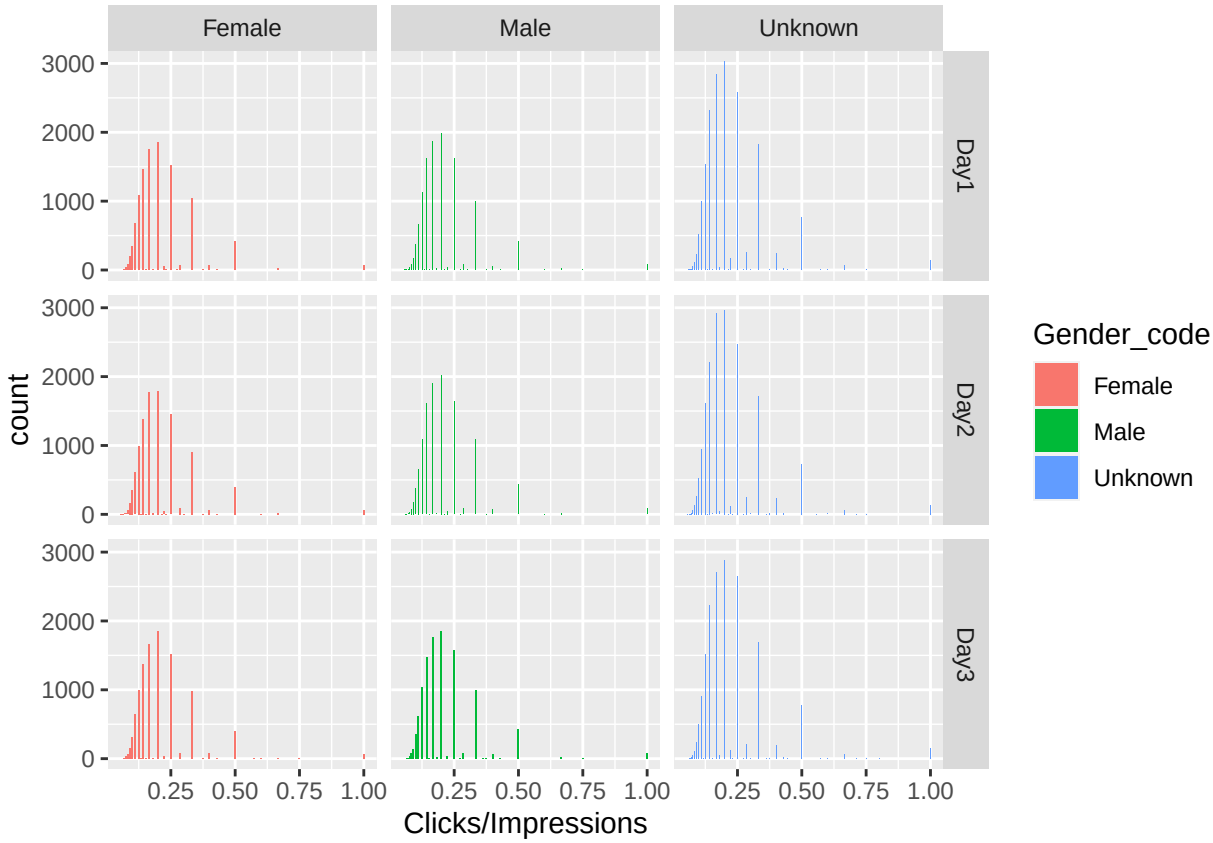
```
data1$Day = 1
data2$Day = 2
data3$Day = 3

alldata<- rbind(data1,data2,data3)

alldata$Day <- factor(alldata$Day,labels=c("Day1","Day2","Day3"))
summary(alldata)
```

```
##       Age             Gender        Impressions         Clicks
##  Min.   :  0.00   Min.   :0.0000   Min.   : 0.000   Min.   :0.00000
##  1st Qu.:  0.00   1st Qu.:0.0000   1st Qu.: 3.000   1st Qu.:0.00000
##  Median : 31.00   Median :0.0000   Median : 5.000   Median :0.00000
##  Mean   : 29.49   Mean   :0.3694   Mean   : 5.001   Mean   :0.09255
##  3rd Qu.: 48.00   3rd Qu.:1.0000   3rd Qu.: 6.000   3rd Qu.:0.00000
##  Max.   :111.00   Max.   :1.0000   Max.   :20.000   Max.   :6.00000
```

```
##
##  Signed_In  age_group       Gender_code        scode           Day
##  0:403761   <20  :498315   Female :446721   Clicks: 117143   Day1:458441
##  1:944985   20-29:151238   Male   :498264   Imps  :1222482   Day2:449935
##             30-39:189026   Unknown:403761   NoImps:   9121   Day3:440370
##             40-49:198390
##             50-59:160944
##             60-69: 95611
##             70+  : 55222
```

Now, we have combined data of 3 days so that we can visualize across time.

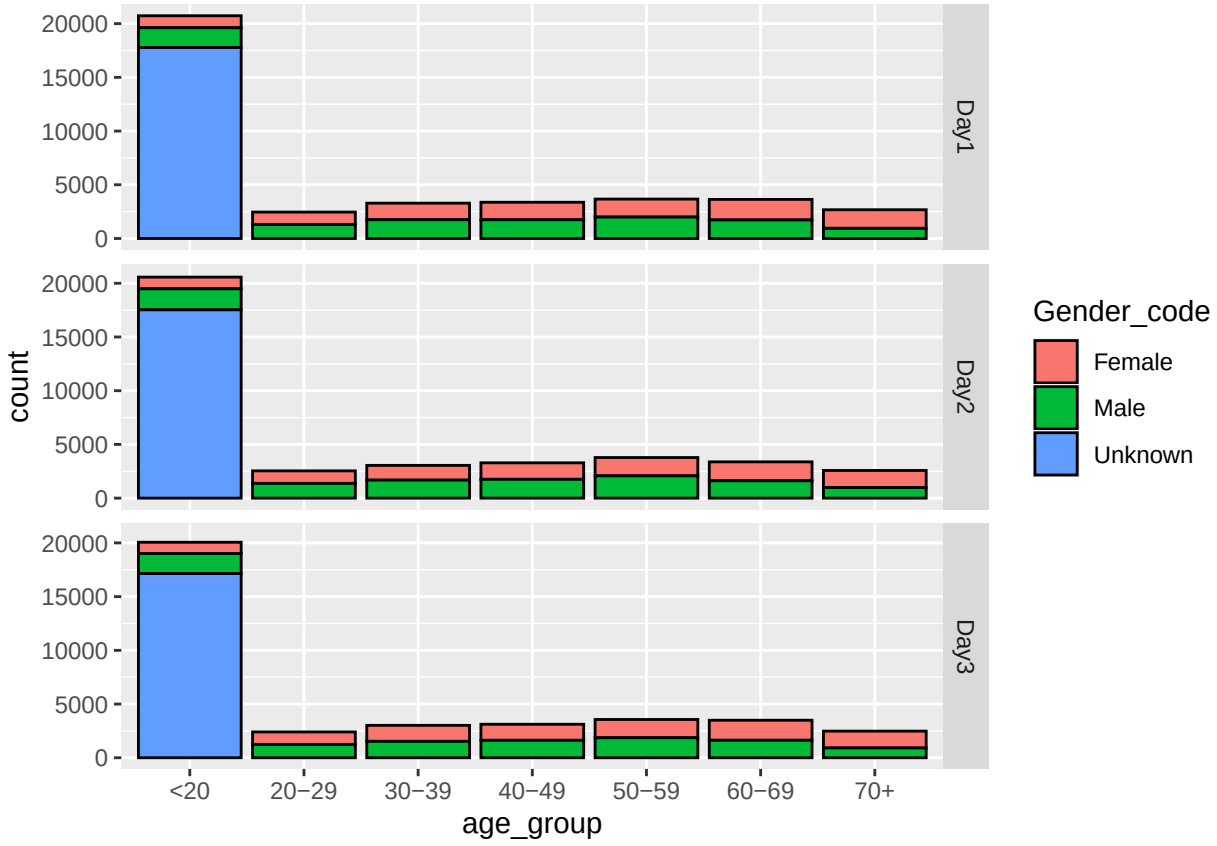
```
ggplot(subset(alldata,Clicks>0 & Impressions>0),aes(x=Clicks/Impressions, fill=Gender_code))+
  geom_bar()+  facet_grid(Day~Gender_code)
```




```
par("mar")
```

```
## [1] 5.1 4.1 4.1 2.1
```

```
par(mar=c(6.5,6,4,1), mgp=c(5,1,0))
ggplot(subset(alldata, Clicks>0),aes(x=age_group, fill=Gender_code))+
  geom_bar(colour="Black") +facet_grid(Day~.)
```



Conclusion

Here we observe that more number of males have clicked on the ads than females for the first 2 days.Also there are a lot of users in the Unknown category across days as they haven't signed in but clicked on the ads.

Also, if we compare across the age categories,more males in the age group <20 have clicks and are signed in for all days. But in the age category, 70 +,more females have signed in and clicked the ads than males.

For the rest age categories, the count is almost same for both male and female.

So the ads should be targeted such that it caters to the above the missing age groups and Gender accordingly.

```
dayImp = aggregate(alldata$Impressions, by=list(Day=alldata$Day),sum)
dayclk = aggregate(alldata$Clicks, by=list(Day=alldata$Day),sum)

dayImp
```

```
##    Day       x
## 1 Day1 2295559
## 2 Day2 2249486
## 3 Day3 2200239
```

```
dayclk
```

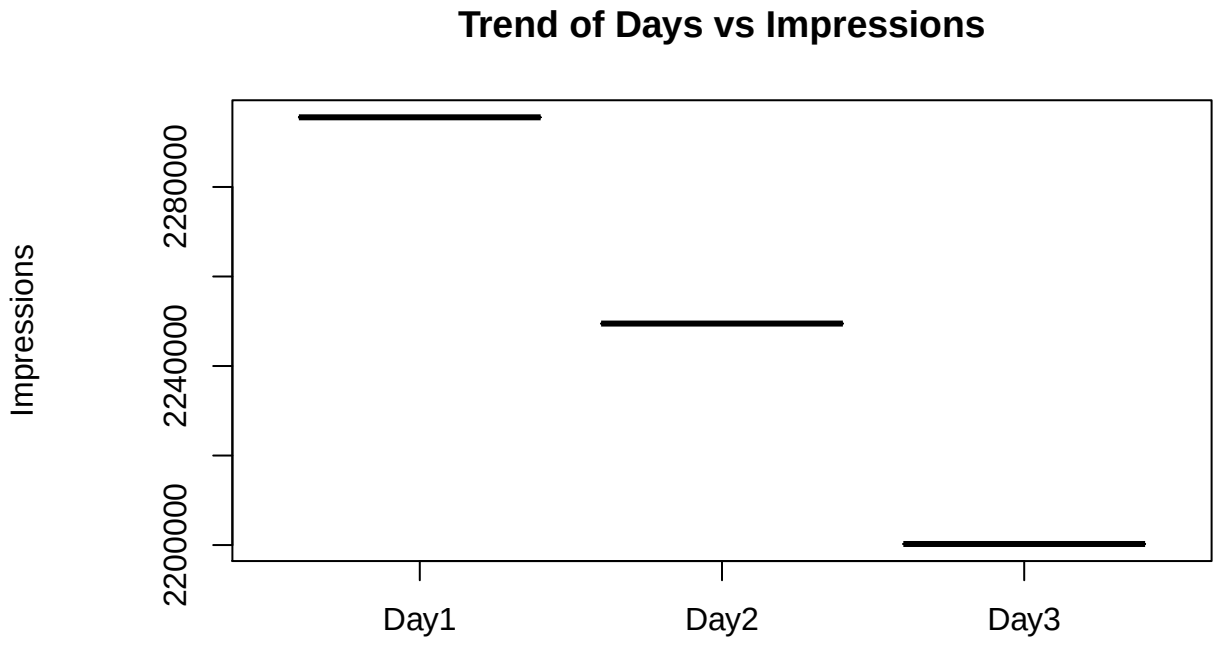
```
##    Day     x
```

```
## 1 Day1 42449
## 2 Day2 41752
## 3 Day3 40630
```

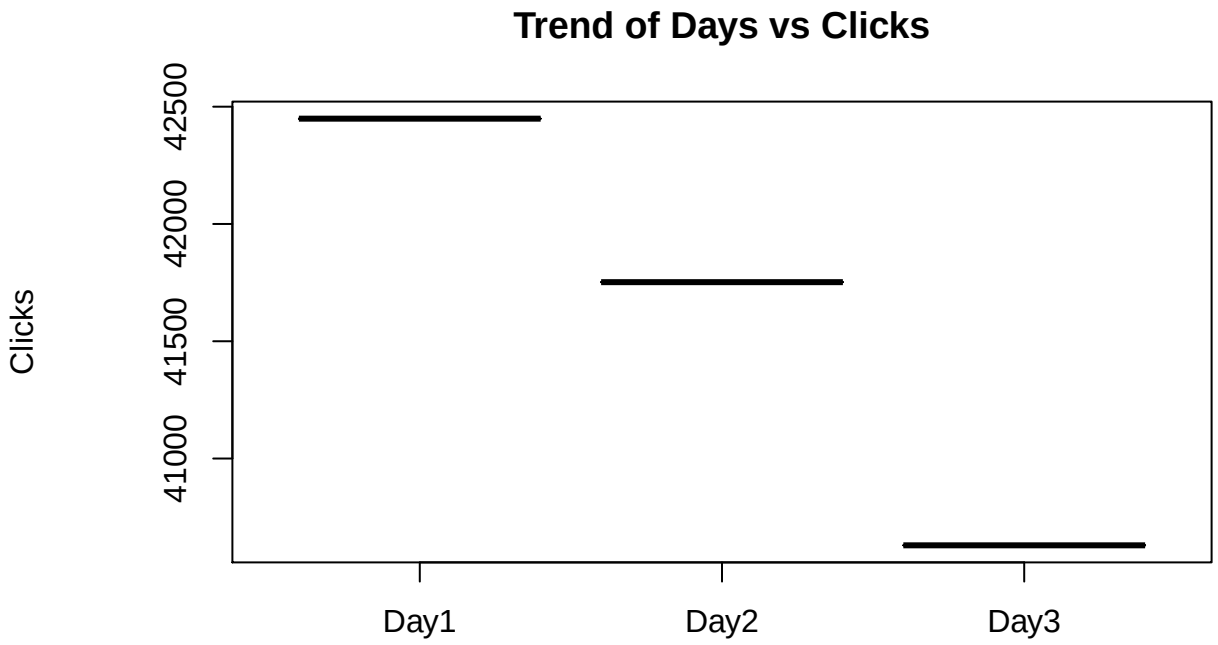
```
par("mar")
```

```
## [1] 5.1 4.1 4.1 2.1
```

```
#par(mar=c(5,5,5,5))
par(mar=c(6.5,6,4,1), mgp=c(5,1,0))

plot(dayImp, main="Trend of Days vs Impressions",xlab(""),ylab("Impressions"))
```




```
plot(dayclk ,main="Trend of Days vs Clicks",xlab(""),ylab("Clicks"))
```




```
summary(alldata$Impressions)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   3.000   5.000   5.001   6.000  20.000
```

```
summary(alldata$Clicks)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.00000 0.00000 0.00000 0.09255 0.00000 6.00000
```

Conclusion

Here,we can observe from the above statistics that the number of impressions and clicks have reduced from Day 1 to Day3, Day 3 being the lowest.So the quality and relevance of the ads for each age group and Gender category should be modified accordingly.