Google Play Store Analysis

In this study, we expect to analyse three sets of problem:
1. Exploratory analysis of single attribute on the Google Play Store data.
2. Explore the correlation of multiple attributes
3. Use statistical test to examine a few interesting hyphothesis.

1.Dataset Information

The dataset is downloaded from Kaggle (link (https://www.kaggle.com/lava18/google-play-store-apps/downloads/google-play-store-apps.zip/6)). It includes data from roughly 3996 applications. Each row represents one App. There are 13 features including catergory, rating, install numbers, price and so on.

1.1 Variables/Columns in the dataset

- App: Application name
- Category: Category the app belongs to
- Rating: Overall user rating of the app (as when scraped)
- Reviews: Number of user reviews for the app (as when scraped)
- Size: Size of the app (as when scraped)
- Installs: Number of user downloads/installs for the app (as when scraped)
- Type: Paid or Free
- Price: Price of the app (as when scraped)
- Content Rating: Age group the app is targeted at - Children / Mature 21+ / Adult
- Genres:An app can belong to multiple genres (apart from its main category). For eg, a musical family game will belong to Music, Game, Family genres.
- Last Updated: Date when the app was last updated on Play Store (as when scraped)
- Current Ver: Current version of the app available on Play Store (as when scraped)
- Android Ver: Min required Android version (as when scraped)

1.2 Tools and packages used for this analysis

```
options(warn=-1)
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(corrplot)
```

```
## corrplot 0.84 loaded
```

```
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

1.3 Loading dataset

```
options(scipen = 50) # Avoid scientific notation if possible.
data<- read.table("C:/Users/yudan/Desktop/google-play-store-apps/googleplaystore.csv",fill=TRUE,
header=TRUE,sep=',',stringsAsFactors = FALSE)
dim(data)
```

```
## [1] 3765   13
```

```
names(data)
```

```
##  [1] "App"            "Category"       "Rating"         "Reviews"
##  [5] "Size"           "Installs"       "Type"           "Price"
##  [9] "Content.Rating" "Genres"         "Last.Updated"   "Current.Ver"
## [13] "Android.Ver"
```

```
str(data)
```

```
## 'data.frame':    3765 obs. of  13 variables:
##  $ App           : chr  "Photo Editor & Candy Camera & Grid & ScrapBook" "Coloring book moan
a" "U Launcher Lite â\200“ FREE Live Cool Themes, Hide Apps" "Sketch - Draw & Paint" ...
##  $ Category      : chr  "ART_AND_DESIGN" "ART_AND_DESIGN" "ART_AND_DESIGN" "ART_AND_DESIGN"
...
##  $ Rating        : chr  "4.1" "3.9" "4.7" "4.5" ...
##  $ Reviews       : chr  "159" "967" "87510" "215644" ...
##  $ Size          : chr  "19M" "14M" "8.7M" "25M" ...
##  $ Installs      : chr  "10,000+" "500,000+" "5,000,000+" "50,000,000+" ...
##  $ Type          : chr  "Free" "Free" "Free" "Free" ...
##  $ Price         : chr  "0" "0" "0" "0" ...
##  $ Content.Rating: chr  "Everyone" "Everyone" "Everyone" "Teen" ...
##  $ Genres        : chr  "Art & Design" "Art & Design;Pretend Play" "Art & Design" "Art & Desi
gn" ...
##  $ Last.Updated  : chr  "7-Jan-18" "15-Jan-18" "1-Aug-18" "8-Jun-18" ...
##  $ Current.Ver   : chr  "1.0.0" "2.0.0" "1.2.4" "Varies with device" ...
##  $ Android.Ver   : chr  "4.0.3 and up" "4.0.3 and up" "4.0.3 and up" "4.2 and up" ...
```

```
head(data, 10)
```

```
##                                                     App       Category
## 1        Photo Editor & Candy Camera & Grid & ScrapBook ART_AND_DESIGN
## 2                                   Coloring book moana ART_AND_DESIGN
## 3  U Launcher Lite â\200“ FREE Live Cool Themes, Hide Apps ART_AND_DESIGN
## 4                                 Sketch - Draw & Paint ART_AND_DESIGN
## 5                 Pixel Draw - Number Art Coloring Book ART_AND_DESIGN
## 6                            Paper flowers instructions ART_AND_DESIGN
## 7               Smoke Effect Photo Maker - Smoke Editor ART_AND_DESIGN
## 8                                      Infinite Painter ART_AND_DESIGN
## 9                                  Garden Coloring Book ART_AND_DESIGN
## 10                        Kids Paint Free - Drawing Fun ART_AND_DESIGN
##    Rating Reviews Size    Installs Type Price Content.Rating
## 1     4.1     159  19M     10,000+ Free     0       Everyone
## 2     3.9     967  14M    500,000+ Free     0       Everyone
## 3     4.7   87510 8.7M  5,000,000+ Free     0       Everyone
## 4     4.5  215644  25M 50,000,000+ Free     0           Teen
## 5     4.3     967 2.8M    100,000+ Free     0       Everyone
## 6     4.4     167 5.6M     50,000+ Free     0       Everyone
## 7     3.8     178  19M     50,000+ Free     0       Everyone
## 8     4.1   36815  29M  1,000,000+ Free     0       Everyone
## 9     4.4   13791  33M  1,000,000+ Free     0       Everyone
## 10    4.7     121 3.1M     10,000+ Free     0       Everyone
##                       Genres Last.Updated        Current.Ver  Android.Ver
## 1               Art & Design     7-Jan-18              1.0.0 4.0.3 and up
## 2  Art & Design;Pretend Play    15-Jan-18              2.0.0 4.0.3 and up
## 3               Art & Design     1-Aug-18              1.2.4 4.0.3 and up
## 4               Art & Design     8-Jun-18 Varies with device   4.2 and up
## 5    Art & Design;Creativity    20-Jun-18                1.1   4.4 and up
## 6               Art & Design    26-Mar-17                  1   2.3 and up
## 7               Art & Design    26-Apr-18                1.1 4.0.3 and up
## 8               Art & Design    14-Jun-18           6.1.61.1   4.2 and up
## 9               Art & Design    20-Sep-17              2.9.2   3.0 and up
## 10   Art & Design;Creativity     3-Jul-18                2.8 4.0.3 and up
```

2. Data Pre-processing

Since all the variables are in characters in the dataset, we need to convert character to numeric values. Also, we need to remove rows with NA values and duplication apps.

```
original_num_rows <- nrow(data)
original_num_rows
```

```
## [1] 3765
```

```
# Preprocess Rating
#Create a temporary numeric variable of Rating.
tmp <- as.numeric(data$Rating)
# Remove the original Rating column.
data = subset(data, select = -Rating)
#Add the numeric variable of Rating into data.
data$Rating = tmp


# Preprocess Review
tmp2 <- as.numeric(data$Reviews)
data = subset(data, select = -Reviews)
data$Reviews = tmp2


# Preprocess Installs
# Remove "+" sign at the end.
tmp3 <- (substr(data$Installs, 1, nchar(data$Installs)-1))
# Remove "," in the number.
tmp4 <- as.numeric(gsub(",","",tmp3))
# Remove the original Installs column
data = subset(data, select = -Installs)
# Add the numeric variable of Installs into data.
data$Installs = tmp4

# Preprocess Price
tmp5 <- data$Price
# Remove '$' sign if any in the front.
tmp6 <- as.numeric(substr(tmp5, startsWith(tmp5, "$")+1, nchar(tmp5)))
data = subset(data, select = -Price)
data$Price = tmp6

# Remove rows with NA value (for simplicity).
data <- na.omit(data)
current_num_row <- nrow(data)
current_num_row
```

```
## [1] 3104
```

```
# Remove duplication
data <- data %>% distinct(App, Last.Updated, .keep_all = TRUE)
unique_num_row <- nrow(data)
unique_num_row  # after processing, the valid rows left.
```

```
## [1] 2891
```

```
summary(data)
```

```
##      App              Category           Size          
##  Length:2891        Length:2891        Length:2891       
##  Class :character   Class :character   Class :character  
##  Mode  :character   Mode  :character   Mode  :character  
##                                                          
##                                                          
##                                                          
##      Type           Content.Rating       Genres          
##  Length:2891        Length:2891        Length:2891       
##  Class :character   Class :character   Class :character  
##  Mode  :character   Mode  :character   Mode  :character  
##                                                          
##                                                          
##                                                          
##  Last.Updated       Current.Ver        Android.Ver            Rating     
##  Length:2891        Length:2891        Length:2891        Min.   :1.000  
##  Class :character   Class :character   Class :character   1st Qu.:4.000  
##  Mode  :character   Mode  :character   Mode  :character   Median :4.300  
##                                                           Mean   :4.198  
##                                                           3rd Qu.:4.500  
##                                                           Max.   :5.000  
##     Reviews           Installs             Price         
##  Min.   :       1   Min.   :         5   Min.   :  0.0000  
##  1st Qu.:     268   1st Qu.:     10000   1st Qu.:  0.0000  
##  Median :    7149   Median :   1000000   Median :  0.0000  
##  Mean   :  444266   Mean   :  15158733   Mean   :  0.6724  
##  3rd Qu.:   77933   3rd Qu.:   5000000   3rd Qu.:  0.0000  
##  Max.   :78128208   Max.   :1000000000   Max.   :399.9900  
```

3. Single-variable Analysis

3.1 Distribution of App Category

We start by looking into how many apps in each category and we highlight a few most common categories. We find that App with "FAMLY", "GAME" , "TOOLS" and "MEDICAL" cover most of the Category.

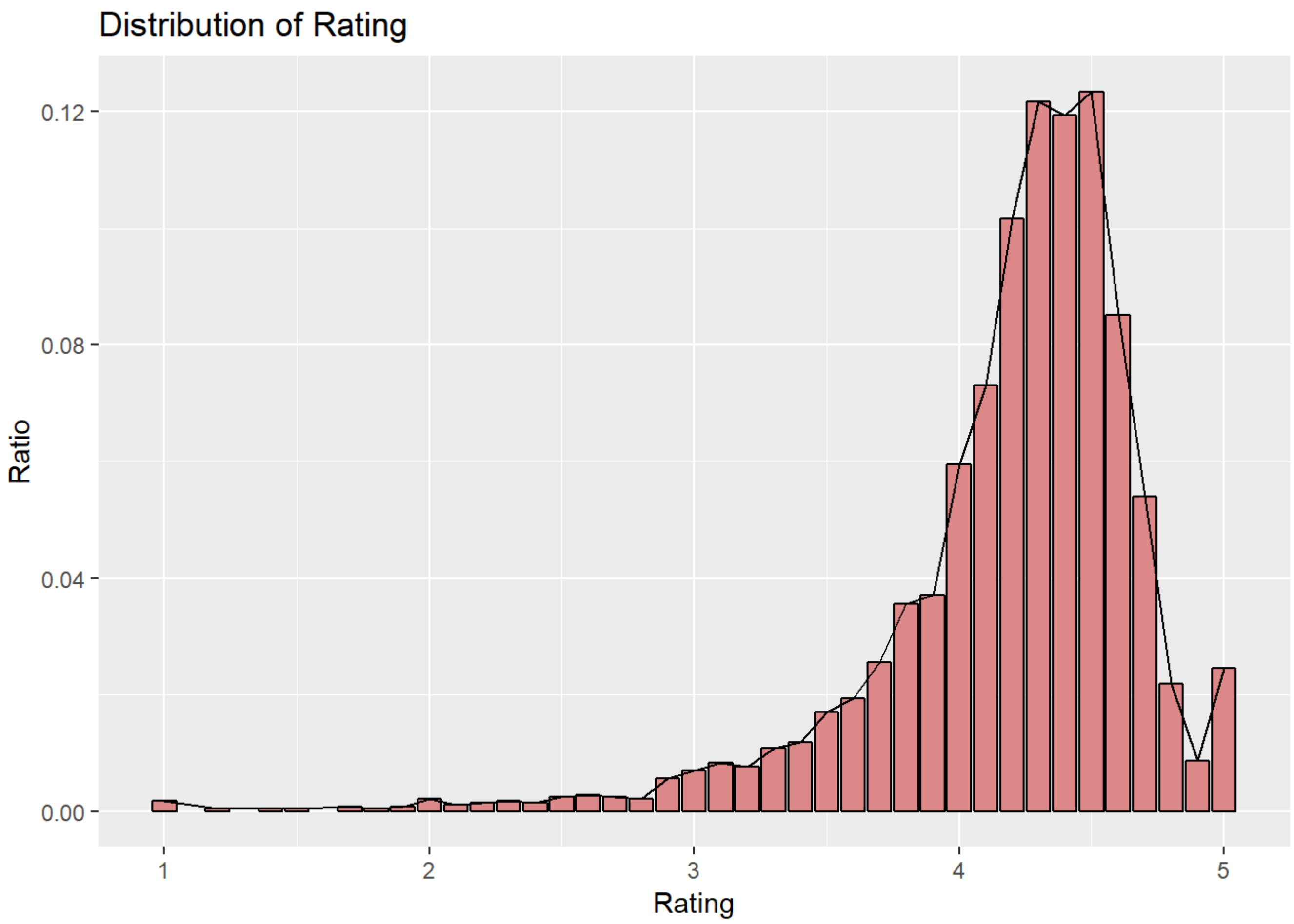
```
total = nrow(data)

Distri_category <- data %>% group_by(Category) %>% summarise(Count=n(), Ratio = n()/total)

# Top-10 Categories
head(arrange(Distri_category, desc(Count)), 10)
```

```
## # A tibble: 10 x 3
##    Category           Count  Ratio
##    <chr>              <int>  <dbl>
##  1 FAMILY               554 0.192
##  2 GAME                 306 0.106
##  3 TOOLS                222 0.0768
##  4 MEDICAL              158 0.0547
##  5 PERSONALIZATION      154 0.0533
##  6 BUSINESS             138 0.0477
##  7 NEWS_AND_MAGAZINES   135 0.0467
##  8 LIFESTYLE            126 0.0436
##  9 PRODUCTIVITY         114 0.0394
## 10 SHOPPING              96 0.0332
```

3.2 Distribution of Rating.

We now study the rating distribution. We want to know what most of ratings locate. From the analysis below, we find that the most common rating range is 4.2~4.7.

```
  rating_data <- data %>% group_by(Rating) %>% summarise(Ratio=n()/total)

  ggplot(rating_data， aes(x=Rating,y=Ratio))+geom_bar(colour="black", fill="#DD8888",stat="ident
ity")+ggtitle("Distribution of Rating")+geom_line()
```



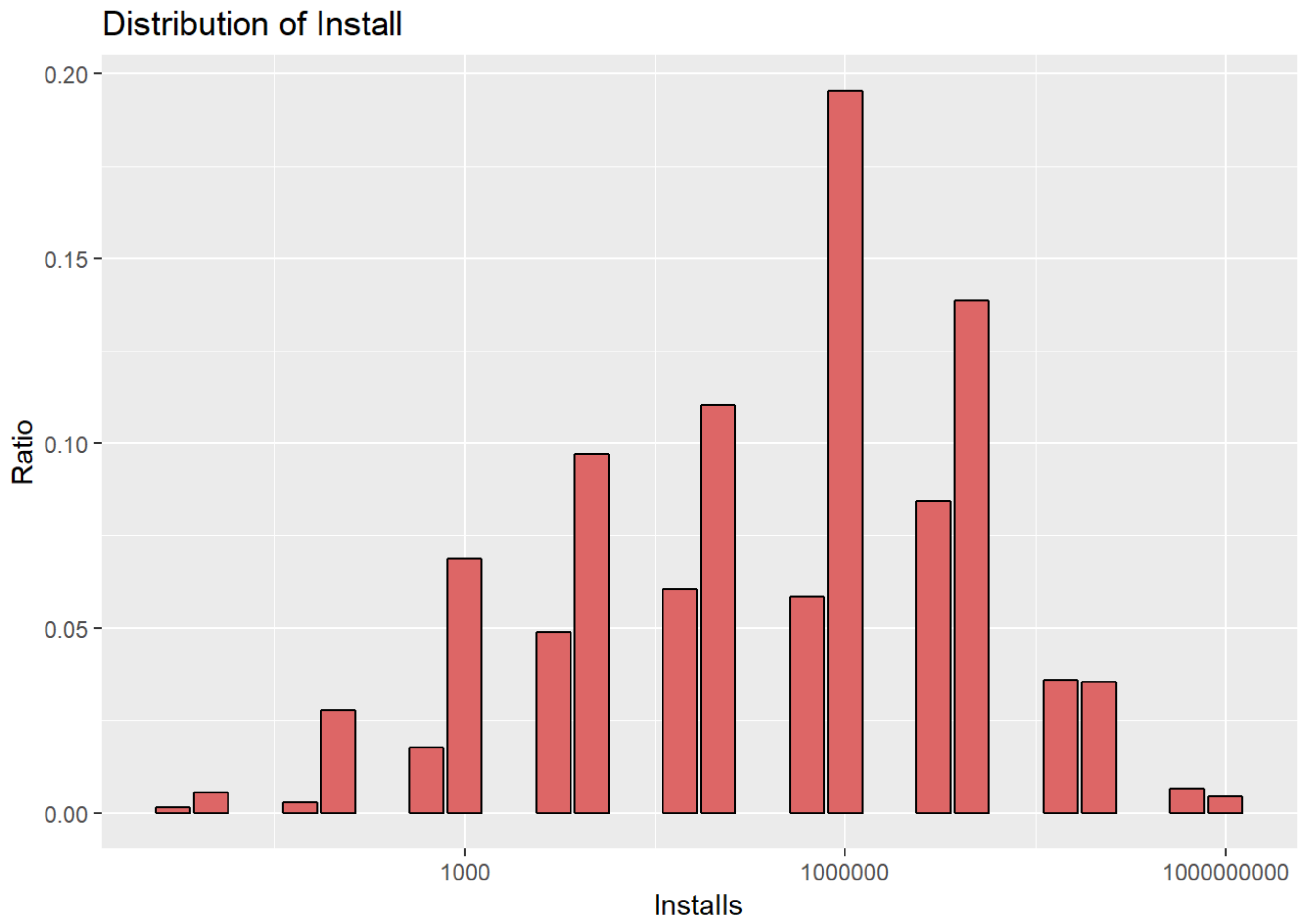
3.3 Distribution of Install Number

We start study distirbution of numbers of install. We find that the most common installation number located on one million.

```
install_data <- data %>% group_by(Installs) %>% summarise(Ratio=n()/total)
install_data
```

```
## # A tibble: 18 x 2
##      Installs   Ratio
##         <dbl>   <dbl>
##  1          5 0.00138
##  2         10 0.00553
##  3         50 0.00277
##  4        100 0.0277
##  5        500 0.0176
##  6       1000 0.0688
##  7       5000 0.0488
##  8      10000 0.0972
##  9      50000 0.0605
## 10     100000 0.110
## 11     500000 0.0585
## 12    1000000 0.195
## 13    5000000 0.0844
## 14   10000000 0.139
## 15   50000000 0.0360
## 16  100000000 0.0353
## 17  500000000 0.00657
## 18 1000000000 0.00450
```

```
ggplot(install_data， aes(x=Installs,y=Ratio))+ geom_bar(colour="black", fill="#DD6666",stat="ide
ntity")+ggtitle("Distribution of Install")  + scale_x_continuous(trans='log10')
```



3.4 Top Apps Analysis

We can find out some interesting things when doing top-apps-analysis.

```
# Top-10 Apps ranked by install numbers
Top_app_install<-data%>%select(App,Category,Installs)%>%arrange(desc(Installs))
head(Top_app_install,10)
```

```
##                          App          Category   Installs
## 1                  Instagram            SOCIAL 1000000000
## 2               Google Drive      PRODUCTIVITY 1000000000
## 3                    YouTube     VIDEO_PLAYERS 1000000000
## 4       Google Play Movies & TV     VIDEO_PLAYERS 1000000000
## 5                Google News NEWS_AND_MAGAZINES 1000000000
## 6             Subway Surfers              GAME 1000000000
## 7           WhatsApp Messenger     COMMUNICATION 1000000000
## 8                   Facebook            SOCIAL 1000000000
## 9  Google Chrome: Fast & Secure     COMMUNICATION 1000000000
## 10                   Google+            SOCIAL 1000000000
```

```
# Top-10 Apps ranked by review numbers
Top_app_review<-data%>%select(App,Category,Reviews)%>%arrange(desc(Reviews))
head(Top_app_review,10)
```

```
##                                             App      Category  Reviews
## 1                                     Facebook        SOCIAL 78128208
## 2                           WhatsApp Messenger COMMUNICATION 69109672
## 3                                    Instagram        SOCIAL 66577446
## 4                               Clash of Clans        FAMILY 44881447
## 5      Clean Master- Space Cleaner & Antivirus         TOOLS 42916526
## 6                               Subway Surfers          GAME 27711703
## 7                                      YouTube VIDEO_PLAYERS 25655305
## 8                                 Clash Royale        FAMILY 23125280
## 9                             Candy Crush Saga          GAME 22430188
## 10 UC Browser - Fast Download Private & Secure COMMUNICATION 17712922
```

Here are the result we observed:
* Top-3 Apps with highest install numbers are dominanted by “Instagram”, “Google Drive” and “YouTube”.
* “Facebook”,“WhatsApp Messenger” and “Instagram” are the apps of highest review numbers.

4. Multiple-variable Analysis

4.1 Correlations Analysis

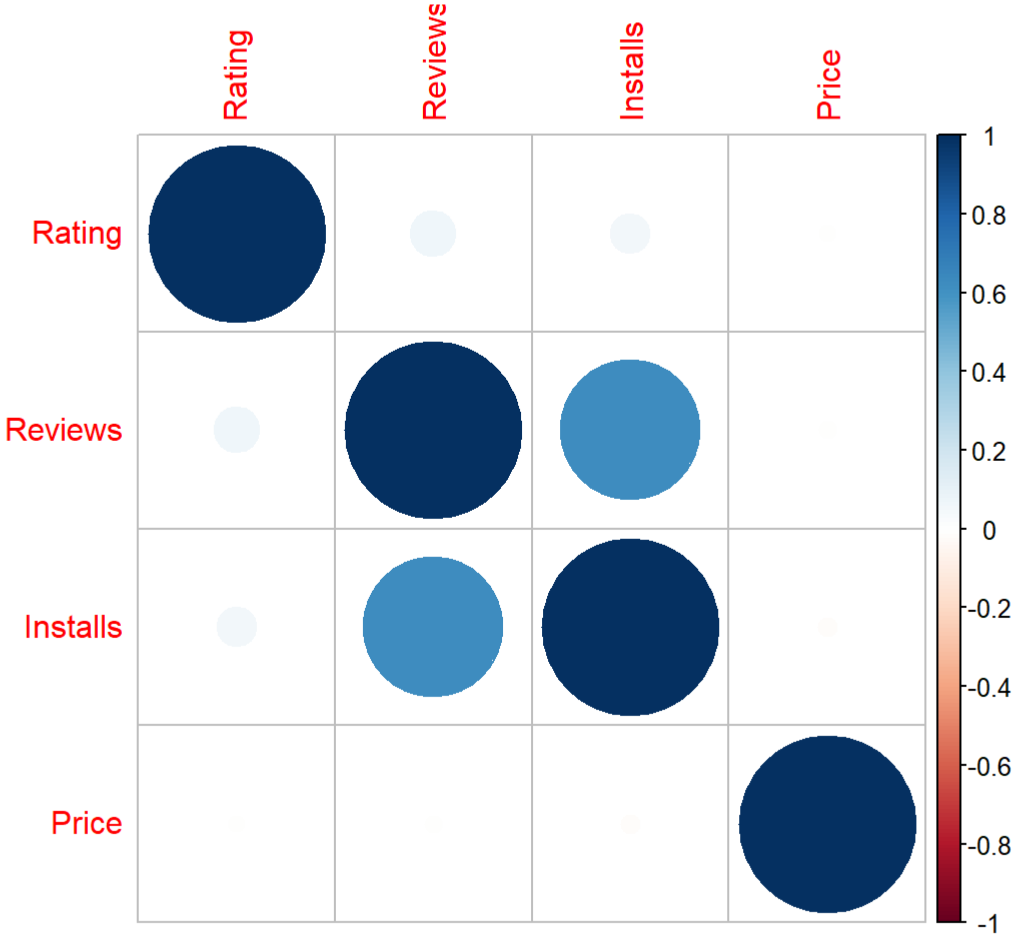
4.11 Correlation Check between Attributes.

We can see obvious correlation between number of reviews and number of installs. No strong correlations between other variables.

```
data_num <- data[,sapply(data, is.numeric)]
str(data_num)
```

```
## 'data.frame':    2891 obs. of  4 variables:
##  $ Rating  : num  4.1 3.9 4.7 4.5 4.3 4.4 3.8 4.1 4.4 4.7 ...
##  $ Reviews : num  159 967 87510 215644 967 ...
##  $ Installs: num  10000 500000 5000000 50000000 100000 50000 50000 1000000 1000000 10000 ...
##  $ Price   : num  0 0 0 0 0 0 0 0 0 0 ...
```

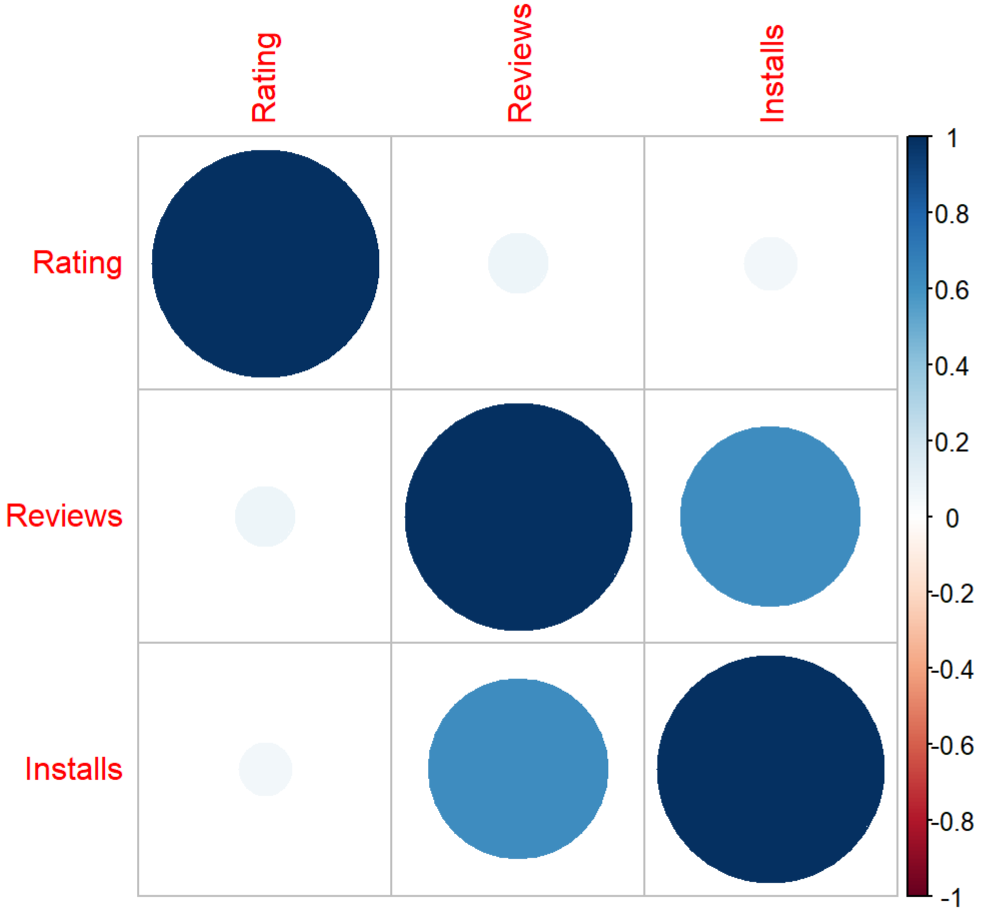
```
Test_rela <-cor(data_num)

corrplot(Test_rela)
```



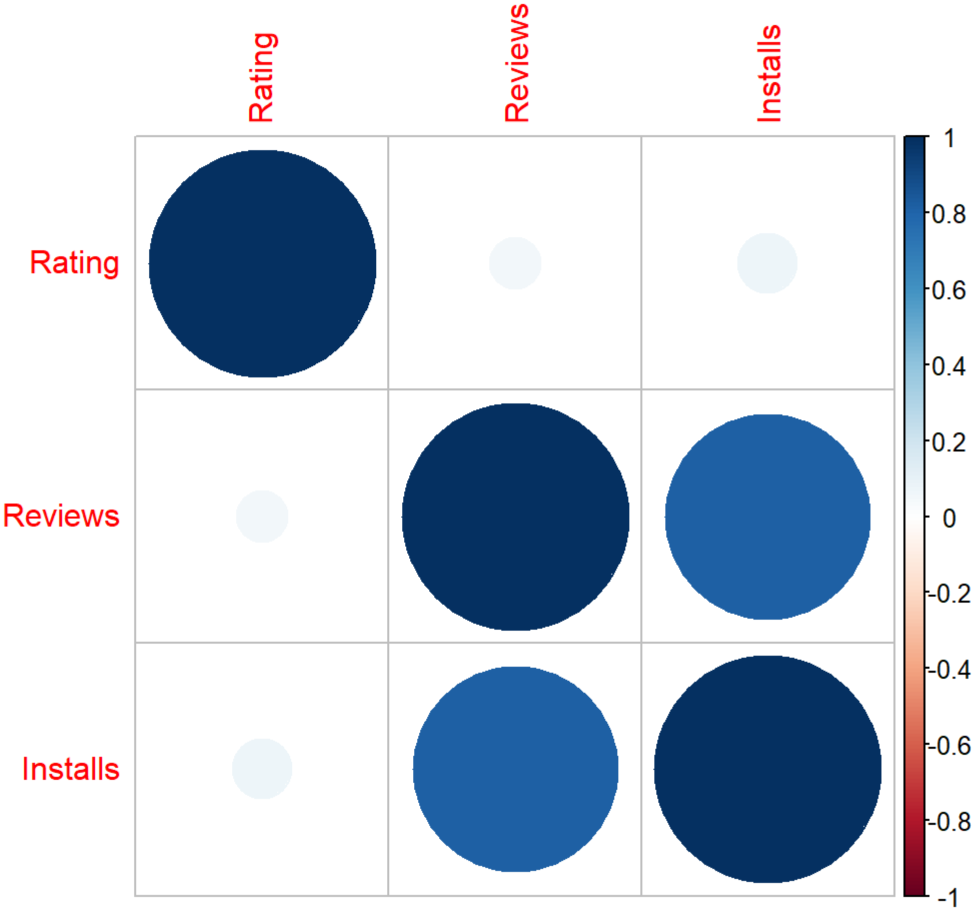
4.1.2 Correlation comparison between Free-apps and Paid-apps.

Their correlation results are very similar (price was dropped). But the correlation between number of reviews and number of installs are slightly higher in Paid Apps. This might indicate paid users are more likely to review.

```
free_app <- data[data$Type=='Free', ]
free_app <- free_app[,sapply(free_app, is.numeric)]
free_app = subset(free_app, select = -Price)
Test_rela_free <-cor(free_app)

corrplot(Test_rela_free)
```



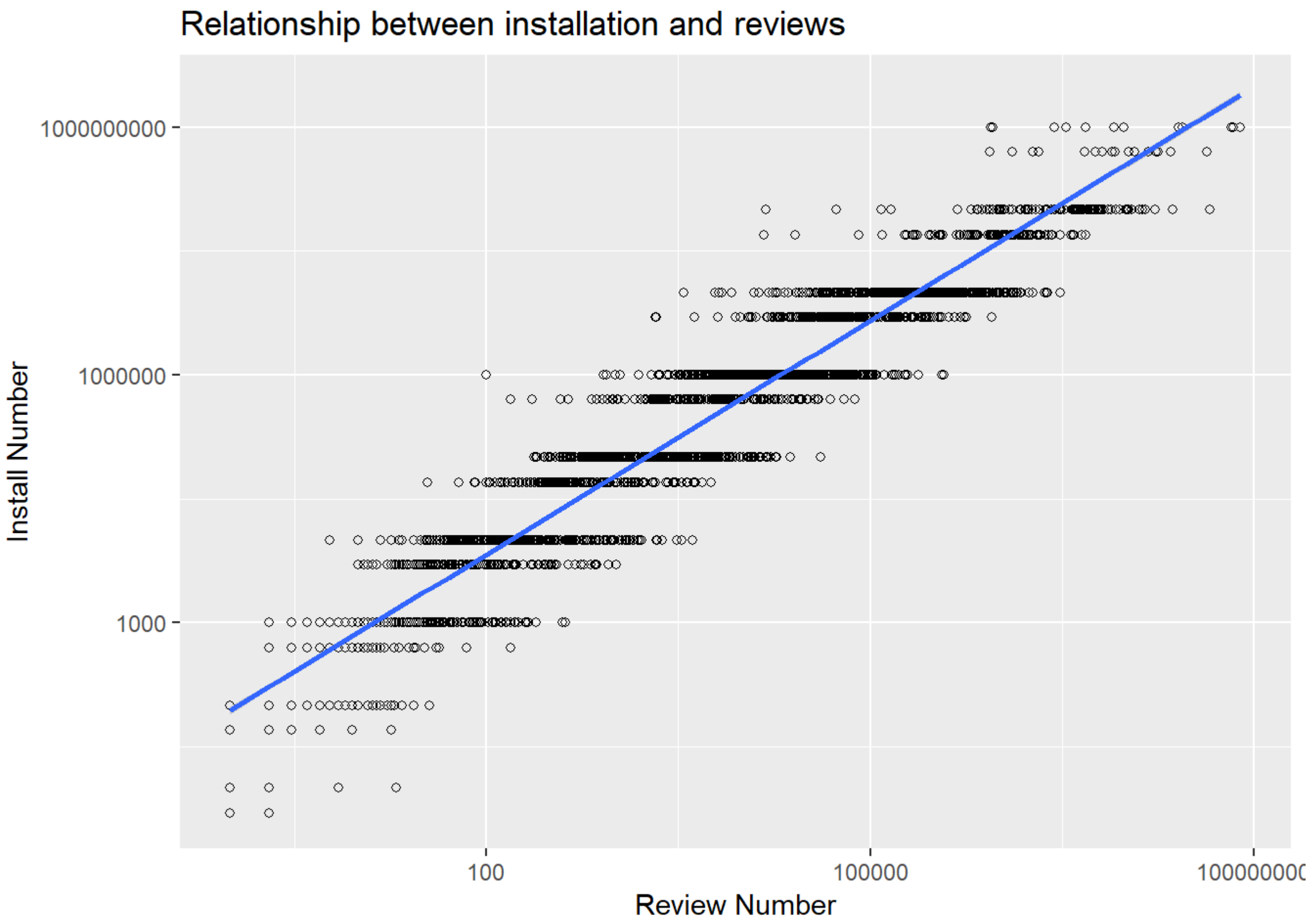
```
paid_app <- data[data$Type=='Paid', ]
paid_app <- paid_app[,sapply(paid_app, is.numeric)]
paid_app = subset(paid_app, select = -Price)
Test_rela_paid <-cor(paid_app)

corrplot(Test_rela_paid)
```



4.2 Relationship between installation and reviews.

We can see the more intuitive relationship between installation and reviews.

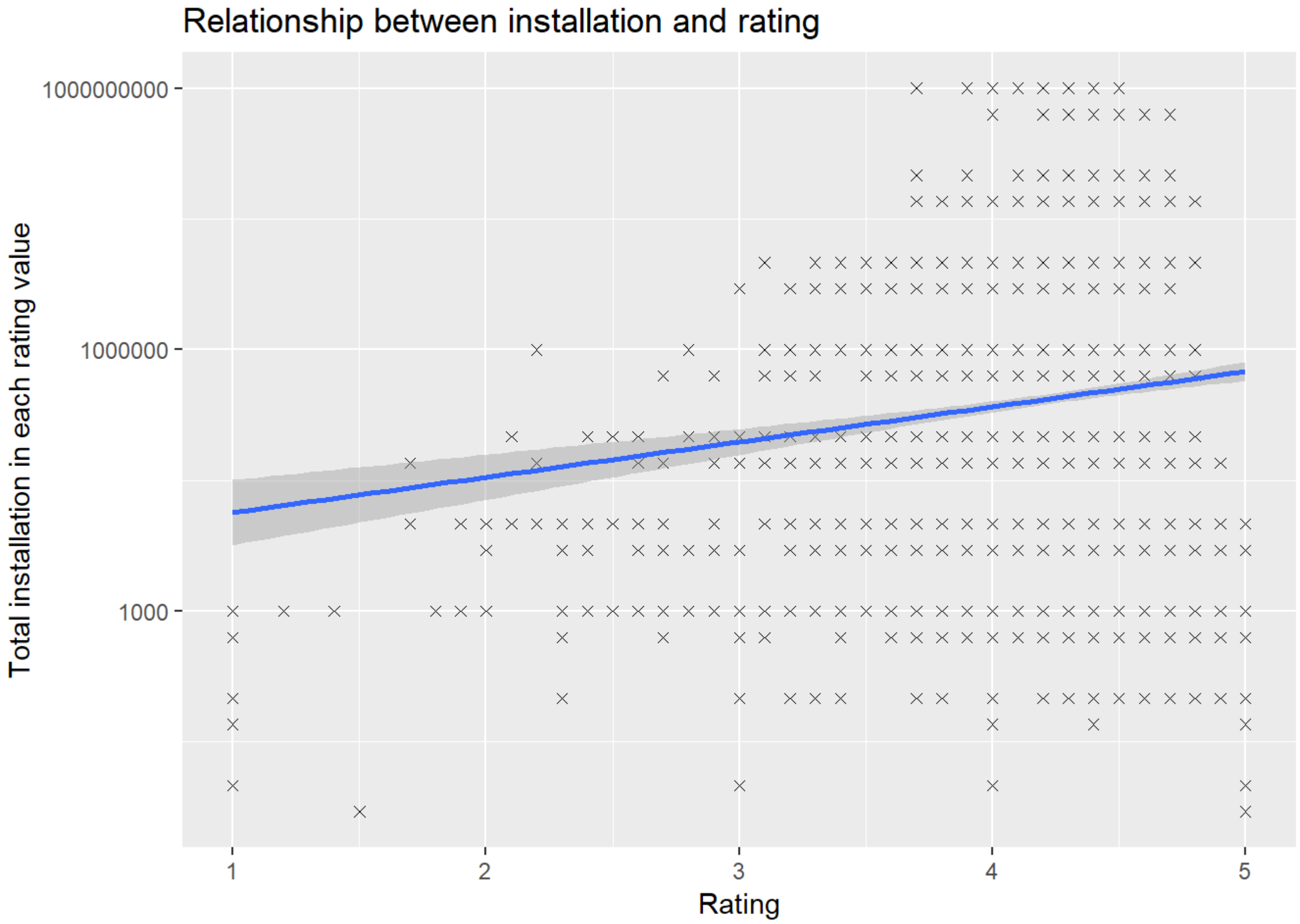
```
G2<-ggplot(data, aes(x=Reviews, y=Installs)) +geom_point(shape=1.5)+geom_smooth(method=lm)+xlab(
"Review Number") +ylab("Install Number") +ggtitle("Relationship between installation and review
s") + scale_x_continuous(trans='log10') + scale_y_continuous(trans='log10')
G2
```



4.3 Relationship between installation and rating.

we can find out slight correlation between installation and rating.

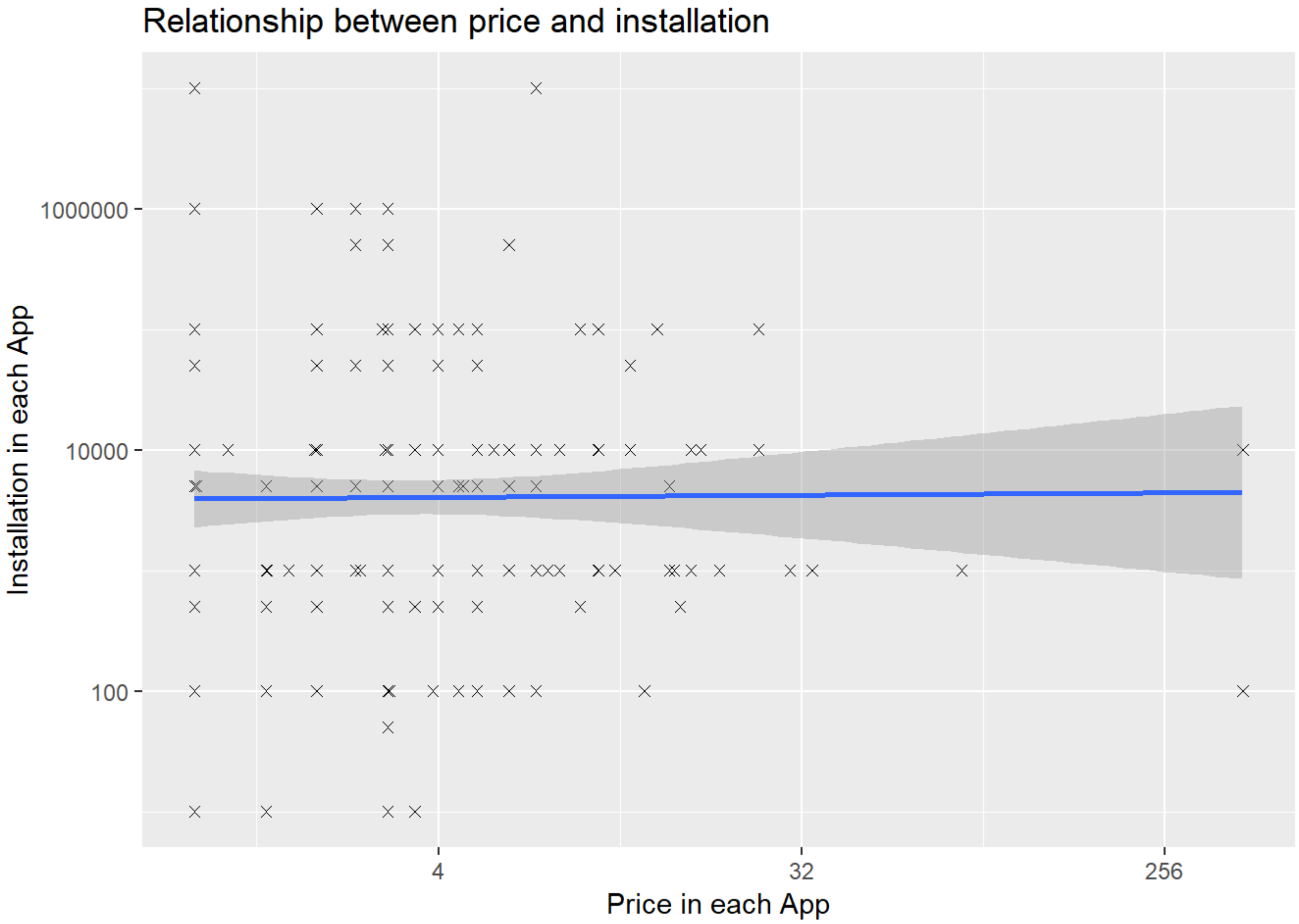
```
G3<-ggplot(data, aes(x=Rating,y=Installs)) +geom_point(shape=4, fill="red")+geom_smooth(method=lm)+xlab("Rating") +ylab("Total installation in each rating value") +ggtitle("Relationship between installation and rating") + scale_y_continuous(trans='log10')
G3
```



4.4 Relationship between Price and Installs

When we select paid apps, we try to find out if there is any relationship between price(number not equal to 0) and installation.
Interesting fact: we always assume the higher price is, the lower is the install. However, it is not true in this dataset analysis!

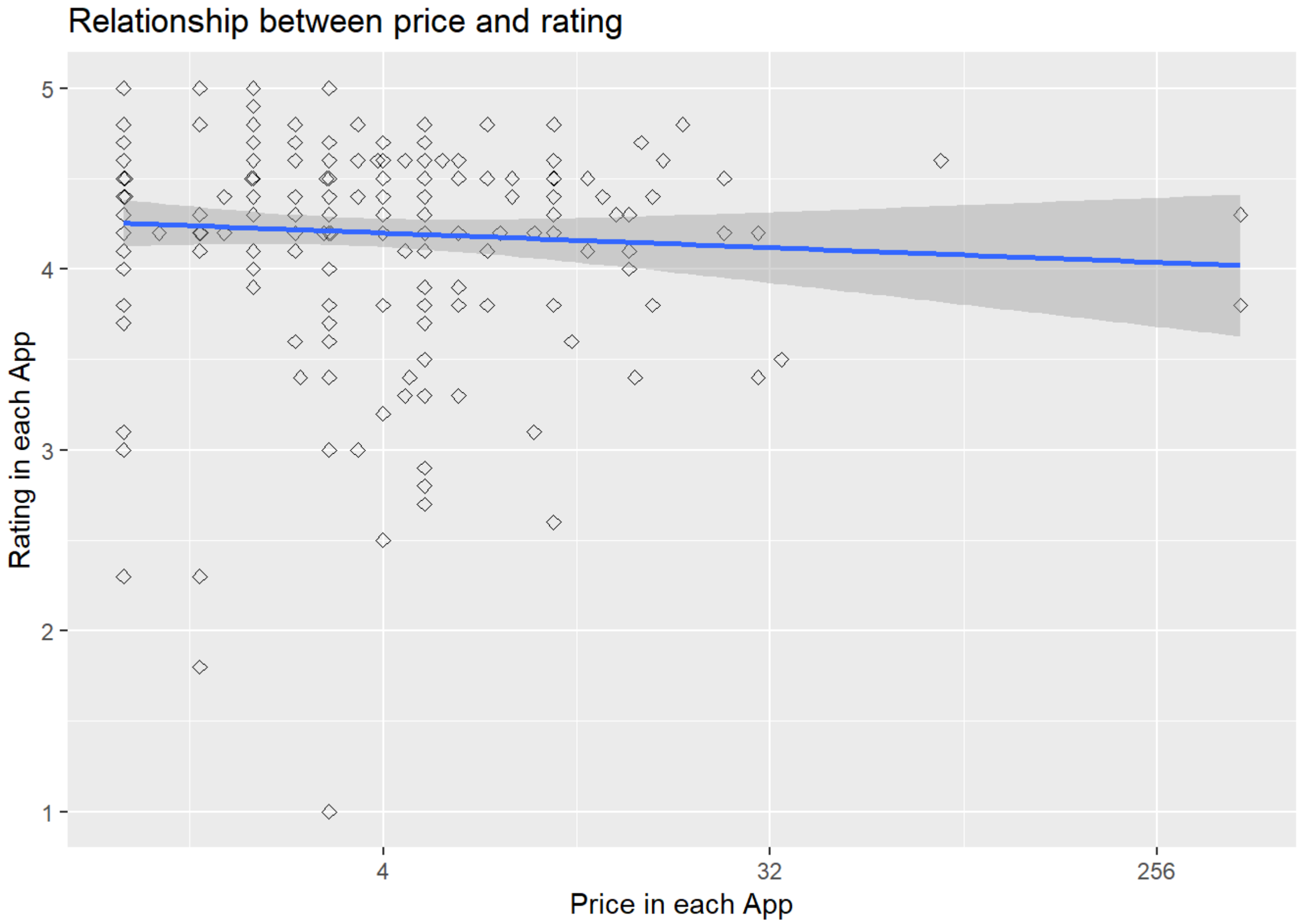
```
sorted_type <- data%>%select(Category,Installs,Type,Price,Rating)%>%arrange(desc(Type))
sorted_type_Paid<-sorted_type[sorted_type$Type=='Paid', ]


GPrice_ins<-ggplot(sorted_type_Paid,aes(x=Price,y=Installs)) +geom_point(shape=4)+geom_smooth(method=lm)+xlab("Price in each App") +ylab("Installation in each App") +ggtitle("Relationship between price and installation") + scale_x_continuous(trans='log2') + scale_y_continuous(trans='log10')
GPrice_ins
```



4.5 Relationship between Price and Rating.

When we want to know whether price affect the rating score or not, we visualize the relationship between price(number not equal to 0) and rating. The plot shows us the higher rating is, the slight lower price is.

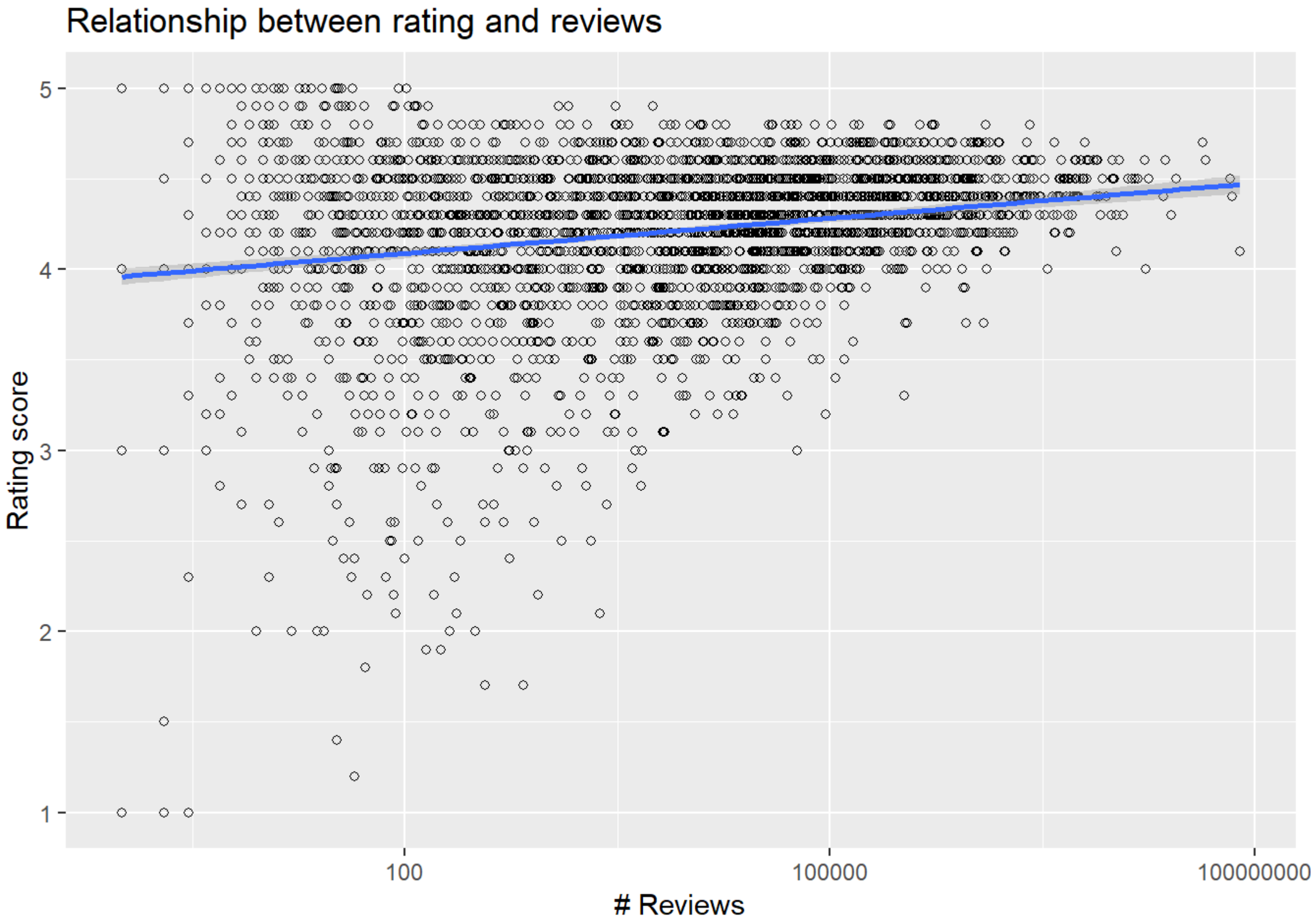
```
GPrice_rat<-ggplot(sorted_type_Paid,aes(x=Price,y=Rating)) +geom_point(shape=5)+geom_smooth(meth
od=lm)+xlab("Price in each App") +ylab("Rating in each App") +ggtitle("Relationship between pric
e and rating") + scale_x_continuous(trans='log2')
GPrice_rat
```



4.6 Relationship between Reviews and Rating.

From the tendency showed in the plot, we can know the more review numbers is, the higher rating score has.

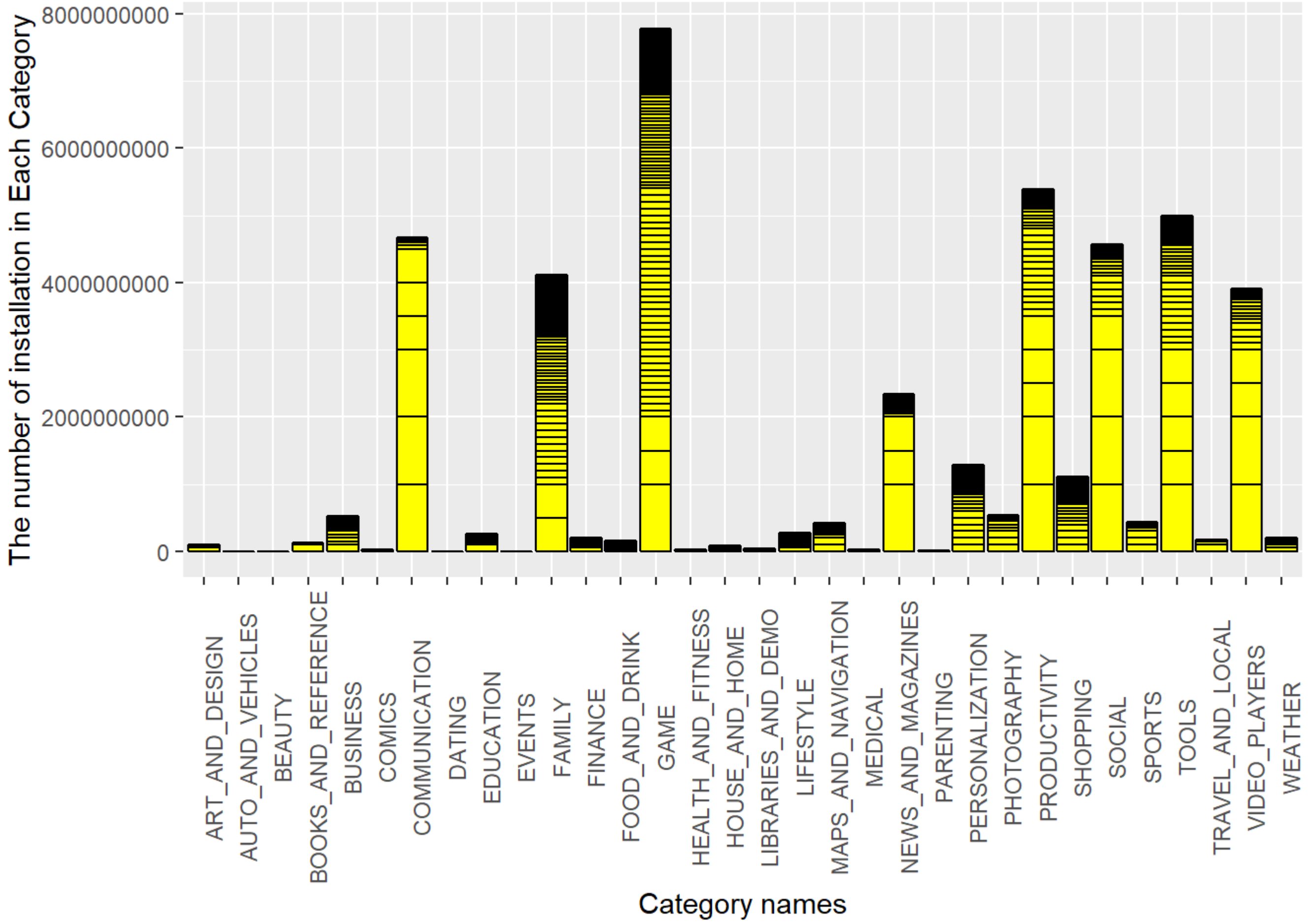
```
rr_plot<-ggplot(data, aes(x=Reviews,y=Rating)) +geom_point(shape=1)+geom_smooth(method=lm)+xlab(
"# Reviews") +ylab("Rating score") +ggtitle("Relationship between rating and reviews") + scale_x
_continuous(trans='log10')
rr_plot
```



4.7 Variation of Install Number in Each Category

Exploring variation in total installation in each category. Game, Social, News are the category of apps with highest number of installs. In each category, a few apps dominate the majority of install numbers. This can be explained by 2/8 principle.

```
Install_data<-data%>%select(App,Category,Installs)%>%arrange(desc(Installs))

G0<-ggplot(Install_data,aes(x=Category, y=Installs)) +geom_bar(stat="identity",fill="yellow", co
lor="black") +theme(axis.text.x =element_text(angle=90)) +xlab("Category names") +ylab("The numb
er of installation in Each Category")
G0
```



4.8 Paid Vs. Free

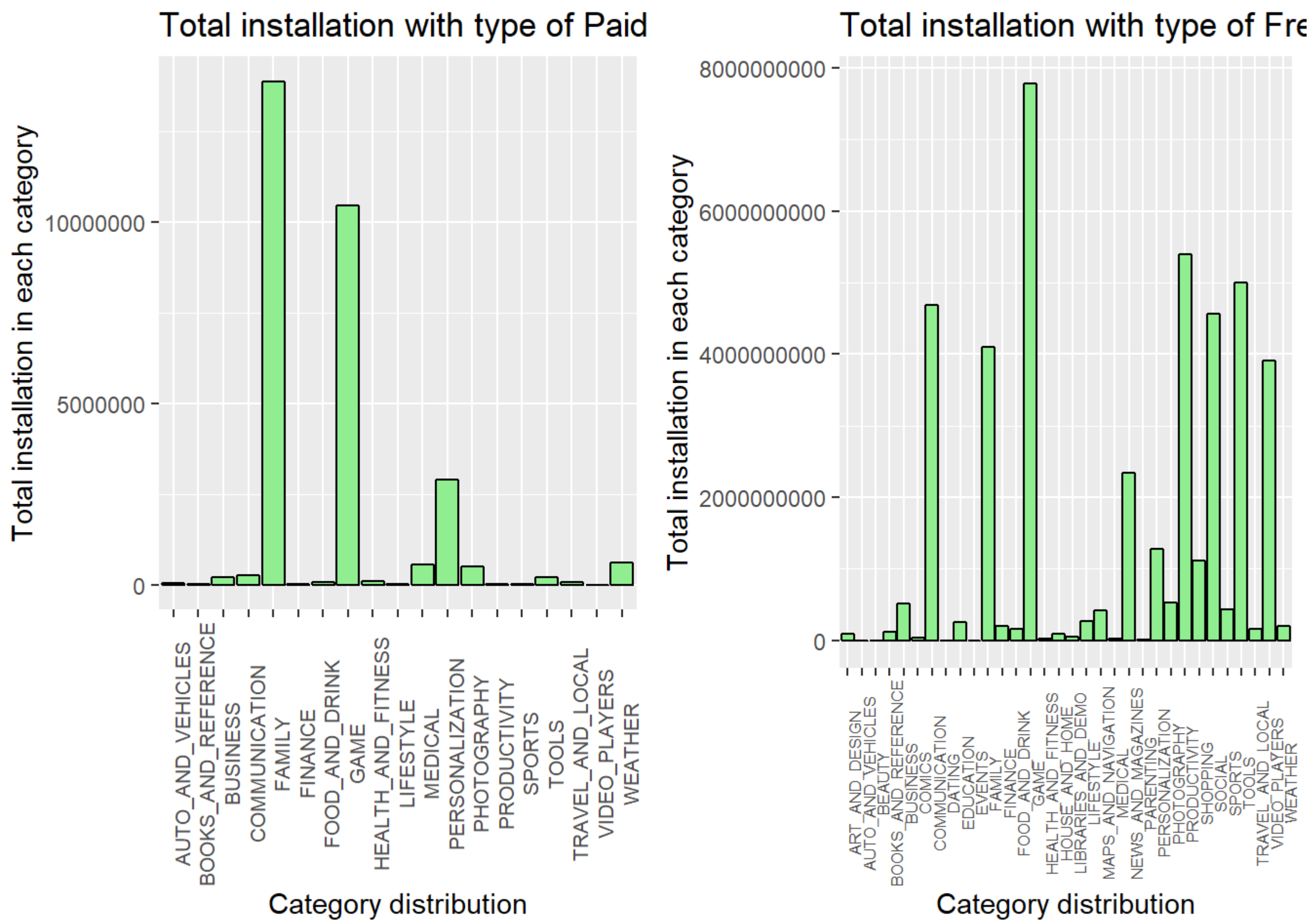
To visualize if there is any difference between installation with paid and installation with free.

```
average_install <- data %>% group_by(Type) %>% summarize(avg_install=sum(Installs)/n())
average_install
```

```
## # A tibble: 2 x 2
##   Type  avg_install
##   <chr>       <dbl>
## 1 Free    16426867.
## 2 Paid      132759.
```

We compare the distribution of install under each category between the paid apps and free apps. Paid-apps are mainly located on “Family” and “Game” while Free-apps are mostly located “Game”, “Productivity”, “Family”,“Tools”,“Communication” and so on. We can conclude that free-apps are more diverse while paid-apps are more focused.

```
type_with_Free <- data[data$Type=='Free', ]
type_with_Paid <- data[data$Type=='Paid', ]

type_with_Paid <-type_with_Paid%>%group_by(Category)%>%summarise(Total_install=sum(Installs))
type_with_Free<-type_with_Free%>%group_by(Category)%>%summarise(Total_install=sum(Installs))

Paid_install <-ggplot(type_with_Paid,aes(x=Category,y=Total_install)) +geom_bar(stat="identity", fill="lightgreen", color="black") +theme(axis.text.x =element_text(angle=90, size=8))+xlab("Category distribution") +ylab("Total installation in each category") +ggtitle("Total installation with type of Paid")

Free_install <-ggplot(type_with_Free,aes(x=Category,y=Total_install)) +geom_bar(stat="identity", fill="lightgreen", color="black") +theme(axis.text.x =element_text(angle=90,size=6))+xlab("Category distribution") +ylab("Total installation in each category") +ggtitle("Total installation with type of Free")

grid.arrange(Paid_install, Free_install, ncol=2)
```



5. Statistical Testing

5.1 Testing for Normality: Shapiro-Wilk test

Shapiro-Wilk test has a maximum sample size limit of 5,000 (sample size must be between 3 and 5000). This test can help us get an intuition from this test by testing the data for normality.

Null-hypothesis: Rating is normally distributed

Alternate-hypothesis: Rating is not normally distributed

Thus if the p-value is less than the chosen alpha level (p<0.05), then the null hypothesis is rejected and there is evidence that the data tested are not from a normally distributed rating value. If the reported p is high,then there is high likelihood that the underlying data is normally distributed.

```
shapiro.test(data$Rating)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  data$Rating
## W = 0.85796, p-value < 0.00000000000000022
```

RESULT: The p-value is very close to zero, so we reject the null hypothesis that the rating is normally distributed.

5.2 Wilcox Two Sample t-test (When data is not normally distributed)

We can use Wilcox's t-test to test the null hypothesis that there is no difference in average rating value between paid-app and free-app by using a two-sided t-test to test whether two rating group have equal means.

Null-hypothesis: There is no difference in rating mean between paid-app and free-app.

Alternate-hypothesis: The rating means are not equal.

```
type_with_Free <- data[data$Type=='Free', ]
type_with_Paid <- data[data$Type=='Paid', ]

x <- type_with_Free$Rating
y <- type_with_Paid$Rating

#Mean of both Rating
mean(x); mean(y)
```

```
## [1] 4.197149
```

```
## [1] 4.204889
```

```
#wilcox test
wilcox.test(x, y, alternative = "two.sided") # x and y are not different.
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  x and y
## W = 278300, p-value = 0.07102
## alternative hypothesis: true location shift is not equal to 0
```

RESULT: Since the p-value is larger than the .05 significance level, we reject the alternative hypothesis.Means are the same. Therefore there is not difference between the value of rating in paid-app and free-app.