

EDA on Google Play Store Applications

Report: Data Analysis and Visualization

Table of Contents

Abstract (1pts)	3
1. Introduction and Background (4Pts)	4
2 Overview of the Data Analysis Pipeline (20pts)	5
3 Discussion and Conclusions (5pts)	24
4 References	25

Abstract

Data analysis is the process of cleaning, transforming, calculating and evaluating an enormous amount of collected data, in order to extract a significantly relevant information from collected data. In the process of data analysis, data visualisation enables a huge amount of data putting into a simple and readable chart, graph or other visual format, which enables to interpret and analyse columns of dataset and as well as their correlations. This project tries to analyse a dataset of current Google Play Store applications along with specific details and values of applications, including category, rating, reviews, installs and more information. The objective of the project is to investigate and analyse current trends of applications in Android market. By doing so, this project provides the opportunity and insight for developers and app-making businesses to capture customers' preferences and interests as well as enabling developers to predict and work on future orientations and trends of the application. The process of data analysis in this project starts with importing packages and loading data and then data cleaning and wrangling and subsequently perform exploratory data analysis using visualisation. Also, python packages, such as Pandas, Numpy and Seaborn will be used to help exploratory data analysis. This project will use visualisation techniques, including histograms, box plot, pair plot and correlation plots between variables. Finally, this project aims at finding correlations between variables and particular trends of those collected data of applications in Android market. Especially, rating and reviews and number of installs of applications will be key variables in the process of predictive analysis. It is expected that higher ratings of the application along with number of installs and reviews are more likely to involve with the trends of applications.

1. Introduction and Background

1.1 The problem you tried to solve

The Google Play Store is the largest app market in the world but compared to its magnitude, it makes a low amount of money. For example, while the Google Play Store produces double downloads of the Apple App Store, it only makes half the money of the Apple App Store makes. This project tries to solve if there is any issue in Google Play Store applications and investigate any application trend in the Google Play Store.

The Google Play Store apps dataset has a plenty of various applications data, which can potentially help many applications-making businesses to a success by analysing key aspects of the dataset.

1.2 Business Question

From data analysis, I would like to determine:

- The key variables responsible for the current trends of applications.
- which category of applications is currently the most potential and popular and close to current trend.
- Current Top 10 popular applications
- Potential correlations between variables.
- Applications, that should be paid, have higher rating and number of installations than application that are free.

1.3 Dataset

This dataset contains 9660 Google Play Store applications, including specific details of each application described below.

App: Application name

Category: Category the application belongs to

Rating: Overall user rating of the application

Reviews: Number of user reviews for the application

Size: Size of the application

Installs: Number of user downloads/installs for the application

Type: Paid or Free

Price: Price of the application

Content Rating: Age group the application is targeted at – Children/Mature 21+/ Adult

Genres: Specific genres the application belongs to. An application can belong to multiple genres

Last Updated: Date when the application was last updated on Play Store

Current Ver: Current version of the application available on Play Store

Android Ver: Min required Android version

2 Overview of the Data Analysis Pipeline

2.1 Data Preparation (5Pts)

1. I firstly loaded google app dataset after downloading CSV file from external resource () and then briefly view samples using head() and tail() methods.

The screenshot shows a Jupyter Notebook interface with the following code and output:

```
df_googleapp = pd.read_csv("googleplaystore.csv")
df_googleapp.head()
```

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
0	Photo Editor & Candy Camera & Grid & ScrapBook	ART_AND_DESIGN	4.1	159	19M	10,000+	Free	0	Everyone	Art & Design	January 7, 2018	1.0.0	4.0.3 and up
1	Coloring book moana	ART_AND_DESIGN	3.9	967	14M	500,000+	Free	0	Everyone	Art & Design;Pretend Play	January 15, 2018	2.0.0	4.0.3 and up
2	U Launcher Lite – FREE Live Cool Themes, Hide ...	ART_AND_DESIGN	4.7	87510	8.7M	5,000,000+	Free	0	Everyone	Art & Design	August 1, 2018	1.2.4	4.0.3 and up
3	Sketch - Draw & Paint	ART_AND_DESIGN	4.5	215644	25M	50,000,000+	Free	0	Teen	Art & Design	June 8, 2018	Varies with device	4.2 and up
4	Pixel Draw - Number Art Coloring Book	ART_AND_DESIGN	4.3	967	2.8M	100,000+	Free	0	Everyone	Art & Design;Creativity	June 20, 2018	1.1	4.4 and up

The screenshot shows a Jupyter Notebook interface with the following code and output:

```
df_googleapp.tail()
```

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres	Last Updated	Current Ver	Android Ver
10836	Sya9a Maroc - FR	FAMILY	4.5	38	53M	5,000+	Free	0	Everyone	Education	July 25, 2017	1.48	4.1 and up
10837	Fr. Mike Schmitz Audio Teachings	FAMILY	5.0	4	3.6M	100+	Free	0	Everyone	Education	July 6, 2018	1.0	4.1 and up
10838	Parkinson Exercises FR	MEDICAL	NaN	3	9.5M	1,000+	Free	0	Everyone	Medical	January 20, 2017	1.0	2.2 and up
10839	The SCP Foundation DB fr nn5n	BOOKS_AND_REFERENCE	4.5	114	Varies with device	1,000+	Free	0	Mature 17+	Books & Reference	January 19, 2015	Varies with device	Varies with device
10840	iHoroscope - 2018 Daily Horoscope & Astrology	LIFESTYLE	4.5	398307	19M	10,000,000+	Free	0	Everyone	Lifestyle	July 25, 2018	Varies with device	Varies with device

2. I have also check any null samples and data type of each column using info() and the total number of samples and columns using shape().

```
C:\Users>y>Downloads> Assignment-PartB-Notebook (1).ipynb > df_googleapp
+ Code + Markdown | Run All Clear Outputs of All Cells Restart
RangeIndex: 10841 entries, 0 to 10840
Data columns (total 13 columns):
#   Column          Non-Null Count  Dtype
---  -
0   App              10841 non-null  object
1   Category         10841 non-null  object
2   Rating           9367 non-null   float64
3   Reviews          10841 non-null  object
4   Size             10841 non-null  object
5   Installs         10841 non-null  object
6   Type             10840 non-null  object
7   Price            10841 non-null  object
8   Content Rating   10840 non-null  object
9   Genres           10841 non-null  object
10  Last Updated     10841 non-null  object
11  Current Ver      10833 non-null  object
12  Android Ver      10838 non-null  object
dtypes: float64(1), object(12)
memory usage: 1.1+ MB
```

```
#Printing the shape of the Dataframe
print(df_googleapp.shape)

[22]
... (10841, 13)
```

3. I have checked index of columns in dataset, using `print(df_columns)`.

```
print(df_googleapp.columns)

[23]
... Index(['App', 'Category', 'Rating', 'Reviews', 'Size', 'Installs', 'Type',
          'Price', 'Content Rating', 'Genres', 'Last Updated', 'Current Ver',
          'Android Ver'],
          dtype='object')
```

4. I used `nunique()` technique to check the total number of applications and `unique()` technique to confirm category samples. I found a potential error, which is '1.9' included in

the category column (This will be removed in data cleaning phase). As category is an object data type, numeric value should not be included.

```
df_googleapp['App'].nunique()
```

[24]

... 9660

```
df_googleapp['Category'].unique()
```

I predict that there should be an error with '1.9', which is not actual ca

[25]

```
array(['ART_AND_DESIGN', 'AUTO_AND_VEHICLES', 'BEAUTY',  
      'BOOKS_AND_REFERENCE', 'BUSINESS', 'COMICS', 'COMMUNICATION',  
      'DATING', 'EDUCATION', 'ENTERTAINMENT', 'EVENTS', 'FINANCE',  
      'FOOD_AND_DRINK', 'HEALTH_AND_FITNESS', 'HOUSE_AND_HOME',  
      'LIBRARIES_AND_DEMO', 'LIFESTYLE', 'GAME', 'FAMILY', 'MEDICAL',  
      'SOCIAL', 'SHOPPING', 'PHOTOGRAPHY', 'SPORTS', 'TRAVEL_AND_LOCAL',  
      'TOOLS', 'PERSONALIZATION', 'PRODUCTIVITY', 'PARENTING', 'WEATHER',  
      'VIDEO_PLAYERS', 'NEWS_AND_MAGAZINES', 'MAPS_AND_NAVIGATION',  
      '1.9'], dtype=object)
```

TERMINAL

2.2 Missing value and duplicate values exploration (5Pts)

1. First, I tried to find any missing values in the dataset, using `df.isnull().sum()`. There were 1474 missing values in Rating column.

2. Data Preparation and Data Cleaning

2.1. Handling Missing Values

```
#Finding missing values in the dataset
df_googleapp.isnull().sum()
```

[26]

...	App	0
	Category	0
	Rating	1474
	Reviews	0
	Size	0
	Installs	0
	Type	1
	Price	0
	Content Rating	1
	Genres	0
	Last Updated	0
	Current Ver	8
	Android Ver	3
	dtype:	int64

2. I dropped missing values from the dataset, using `df.dropna()` and then after removal, check again if there is any missing values left, I used `df.isnull().sum()`.


```
Android Ver      3
dtype: int64

#Dropping null values from the dataset
df_googleapp = df_googleapp.dropna()
#Sum of missing values after removal
print(df_googleapp.isnull().sum())

[27]
... App      0
Category    0
Rating      0
Reviews     0
Size        0
Installs    0
Type        0
Price       0
Content Rating 0
Genres      0
Last Updated 0
Current Ver  0
Android Ver  0
dtype: int64

2.2. Handling Duplicate Values
```

3. Confirmed that there are no more missing values.
4. I also checked if there are any duplicate rows in the dataset, as shown in the screenshot and then dropped duplicate values using `df.drop_duplicates()`.

```
2.2. Handling Duplicate Values

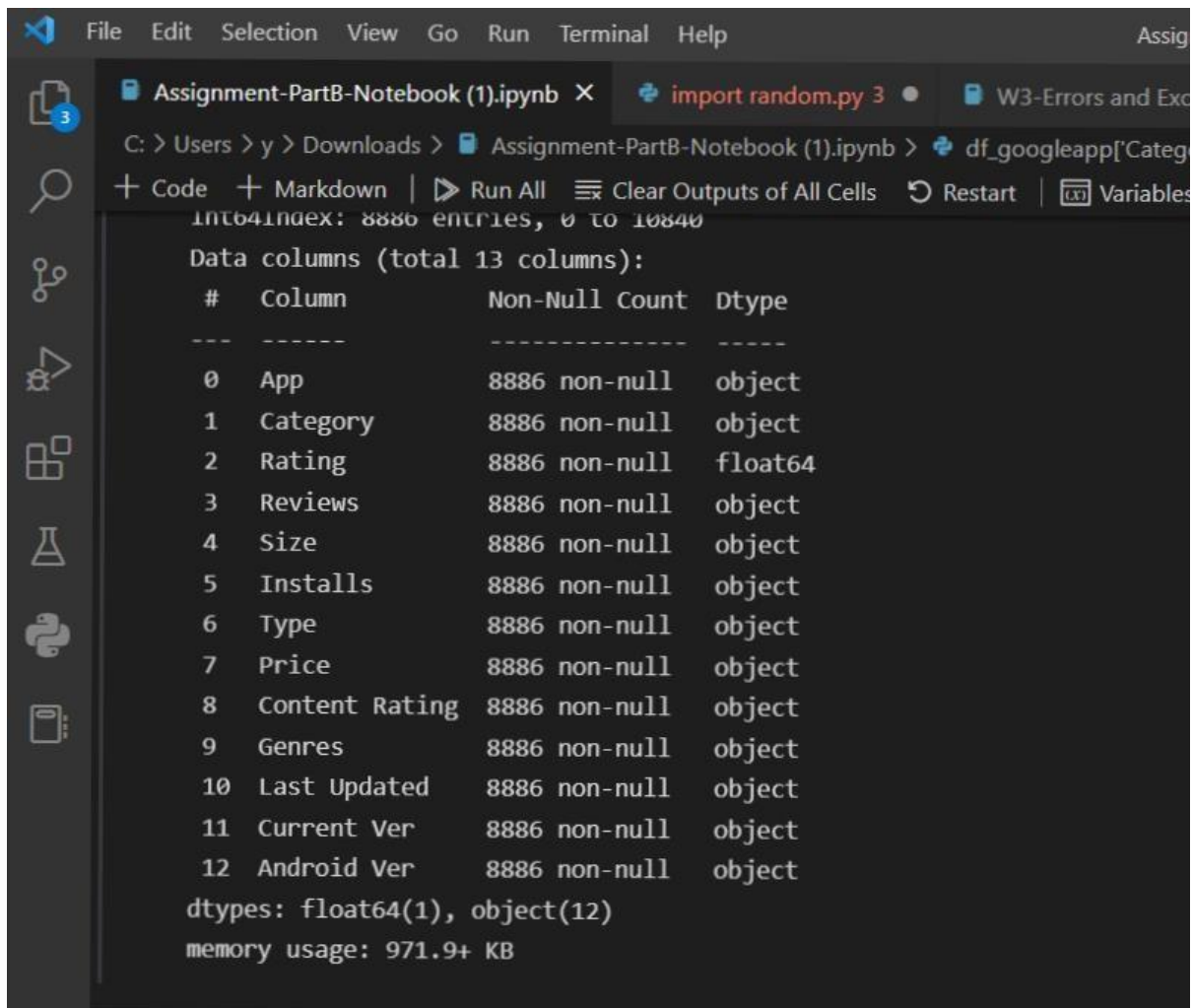
#Finding duplicate rows in the dataframe and then remove duplicate rows.
df_duplicate = df_googleapp[df_googleapp.duplicated()]
print("number of duplicate rows: {}".format(df_duplicate.shape))
df_googleapp = df_googleapp.drop_duplicates()

[28]
... number of duplicate rows: (474, 13)

df_googleapp.info()

[29]
```

5. Finally, I confirmed the samples of the dataset after removal of missing values and duplicate values (Simply used `df.info()`).

A screenshot of a Jupyter Notebook interface. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, and Help. The notebook title is 'Assignment-PartB-Notebook (1).ipynb'. The code cell shows the output of a pandas DataFrame inspection. The output indicates 8886 entries and 13 columns. A table lists the columns with their non-null counts and dtypes. The dtypes are float64 for one column and object for the others. The memory usage is 971.9+ KB.

```
INT64Index: 8886 entries, 0 to 10840
Data columns (total 13 columns):
#   Column             Non-Null Count  Dtype
---  -
0   App                 8886 non-null   object
1   Category            8886 non-null   object
2   Rating              8886 non-null   float64
3   Reviews             8886 non-null   object
4   Size                8886 non-null   object
5   Installs            8886 non-null   object
6   Type                8886 non-null   object
7   Price               8886 non-null   object
8   Content Rating      8886 non-null   object
9   Genres              8886 non-null   object
10  Last Updated        8886 non-null   object
11  Current Ver         8886 non-null   object
12  Android Ver         8886 non-null   object
dtypes: float64(1), object(12)
memory usage: 971.9+ KB
```

2.3 Outlier identification (5Pts)

[Use of Appropriate visualization techniques to identify whether there are outliers in the dataset, and take appropriate action to removed/handle them. You can include screenshots of the plots, etc.]

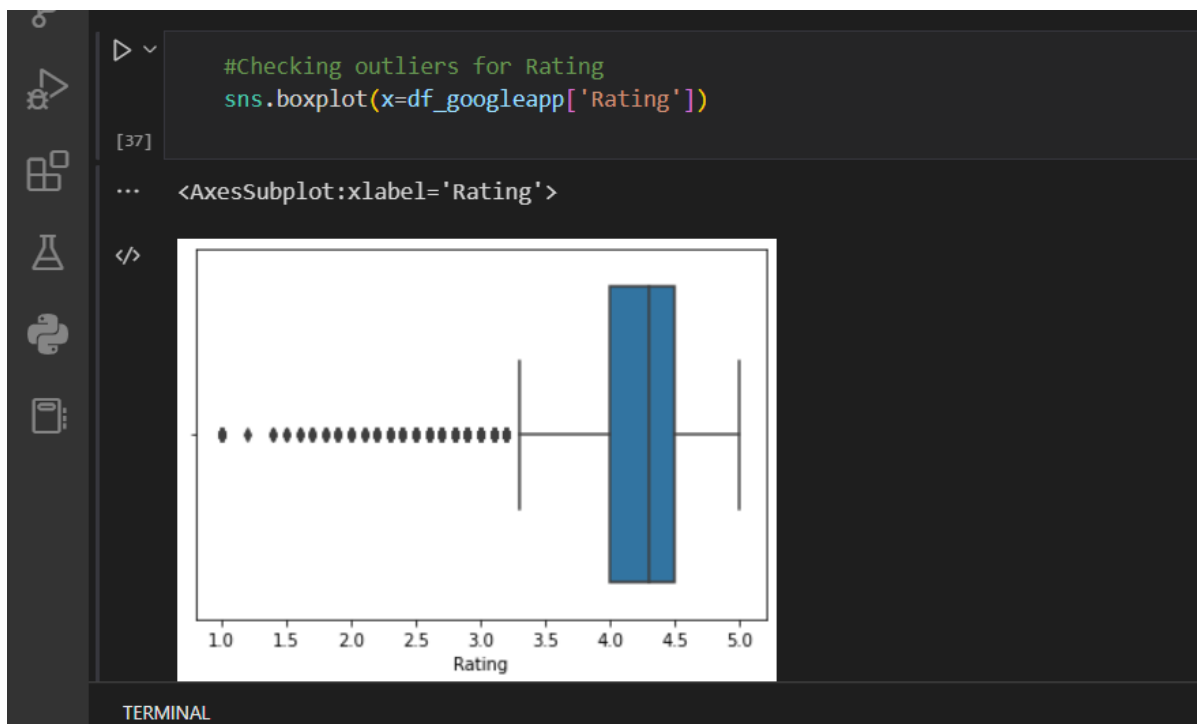
1. Firstly, I checked shape of data frame before removing outliers.

2.4. Handling Outliers

```
#Checking the final list of columns after dropping unamed and custid
print(df_googleapp.columns)
print("Shape of dataframe before removing outliers: {}".format(df_googleapp.shape))
```

```
[36] ... Index(['App', 'Category', 'Rating', 'Reviews', 'Size', 'Installs', 'Type',
          'Price', 'Content Rating', 'Genres', 'Last Updated', 'Current Ver',
          'Android Ver'],
          dtype='object')
      Shape of dataframe before removing outliers: (8886, 13)
```

2. I checked if there are outliers in Rating column and defined 25TH and 75TH percentiles and then calculated Interquartile Range (IQR) by subtracting those two percentiles. Finally, I was able to obtain the total number of outliers in Rating by obtaining the ratings that are outside of lower and upper bound.



```
#Defining Quartiles for removal of outliers
Q1 = df_googleapp['Rating'].quantile(0.25)
Q3 = df_googleapp['Rating'].quantile(0.75)
IQR = Q3 - Q1 #IQR stands for Interquartile Range(IQR): difference between the 75th and 25th percentiles.
total_outlier_num = ((df_googleapp['Rating'] < (Q1 - 1.5 * IQR)) | (df_googleapp['Rating'] > (Q3 + 1.5 * IQR))).sum()
#print(IQR)
print("Total Number of Outliers in Rating: {}".format(total_outlier_num))
```

```
[38] ... Total Number of Outliers in Rating: 494
```

3. In data preparation phase, the error in category column, which was 1.9, was shown. It is assumed that 1.9 should be the rating and 19 in rating column should be the review.

Rating of 19 was an outlier, which was caused by the error in category and shifted to rating and other columns. Therefore, I decided to drop this row and remove the false outlier. After I drop this row, I confirmed that there is no '1.9' in Category and also 19 in rating was removed.

10473	Xposed W	PERSONAL	3.5	1042	404k	100,000+	Free	0	Everyone	Personalize	#
10474	Life Made	1.9	19	3.0M	1,000+	Free	0	Everyone	#####	1	
10475	osmino W	TOOLS	4.2	134203	4.1M	10,000,000	Free	0	Everyone	Tools	#
10476	Sat-Fi Voic	COMMUN	3.4	37	14M	1,000+	Free	0	Everyone	Communic	#
10477	Wi-Fi Visu	TOOLS	3.9	132	2.6M	50,000+	Free	0	Everyone	Tools	#

The screenshot shows a Jupyter Notebook window titled 'Assignment-PartB-Notebook (1).ipynb'. The code cell contains the following Python code:

```
# Category '1.9' has been removed.
df_googleapp['Category'].unique()
```

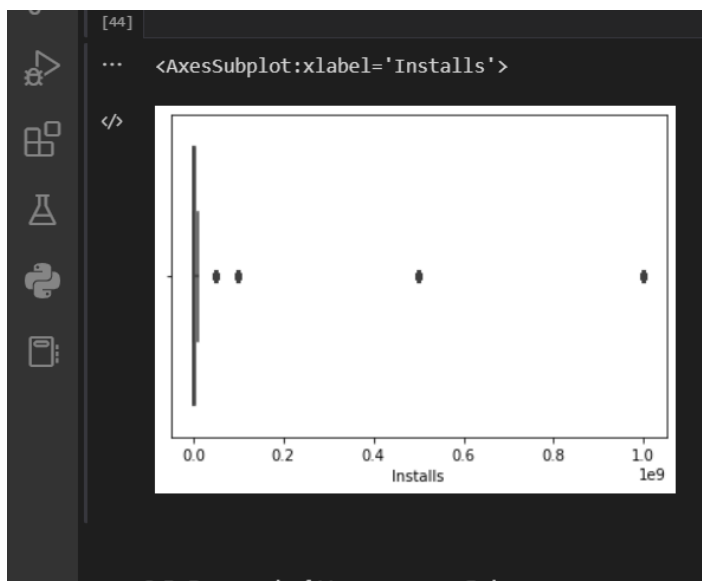
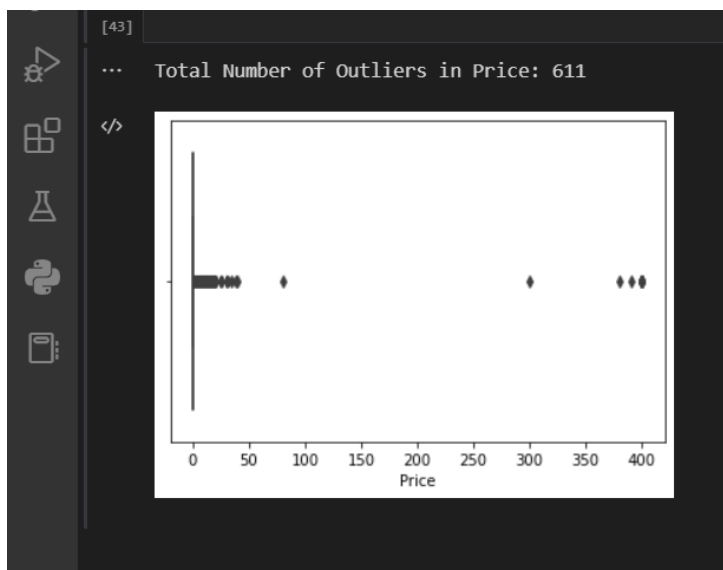
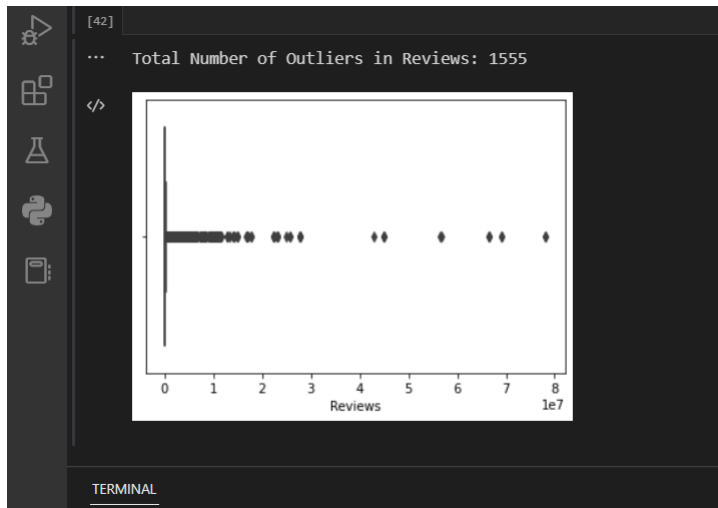
The output of the code is displayed below the cell:

```
[41]
... array(['ART_AND_DESIGN', 'AUTO_AND_VEHICLES', 'BEAUTY',
        'BOOKS_AND_REFERENCE', 'BUSINESS', 'COMICS', 'COMMUNICATION',
        'DATING', 'EDUCATION', 'ENTERTAINMENT', 'EVENTS', 'FINANCE',
        'FOOD_AND_DRINK', 'HEALTH_AND_FITNESS', 'HOUSE_AND_HOME',
        'LIBRARIES_AND_DEMO', 'LIFESTYLE', 'GAME', 'FAMILY', 'MEDICAL',
        'SOCIAL', 'SHOPPING', 'PHOTOGRAPHY', 'SPORTS', 'TRAVEL_AND_LOCAL',
        'TOOLS', 'PERSONALIZATION', 'PRODUCTIVITY', 'PARENTING', 'WEATHER',
        'VIDEO_PLAYERS', 'NEWS_AND_MAGAZINES', 'MAPS_AND_NAVIGATION'],
        dtype=object)
```

Below the output, a comment states: '*Now, there is no '1.9' category, which was an error.'

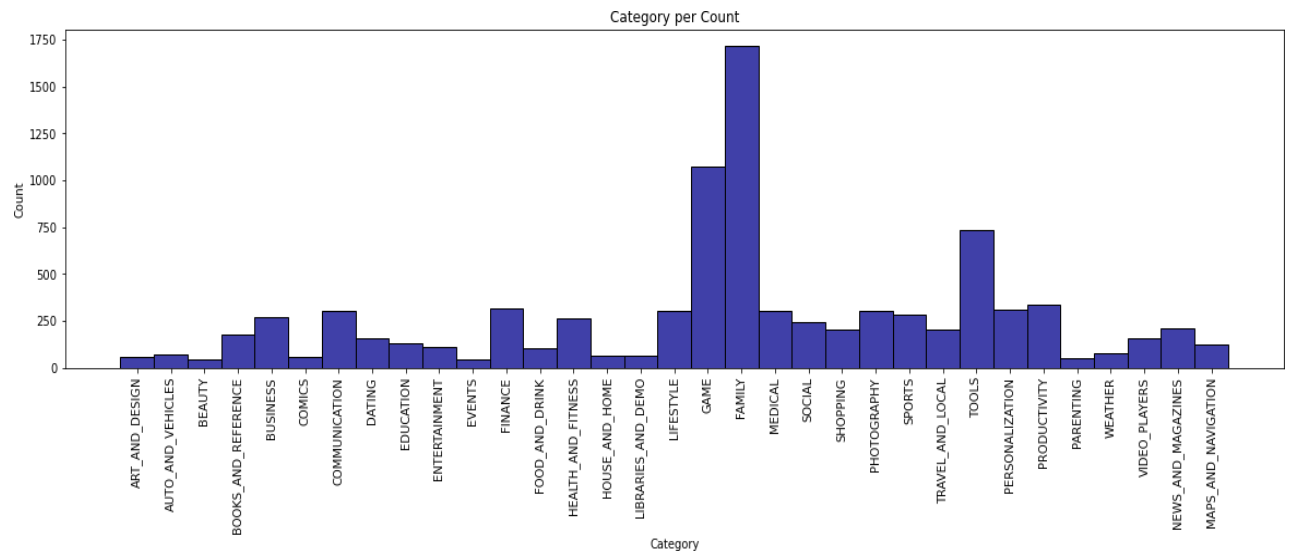
The other outliers in Rating were known as 'true outliers', which were not involved with measurement error. Therefore, I decided to include them in my data analysis.

- I checked outliers in Review, Price and Installs as well. They also have many outliers but considered that this is a huge, enormous dataset and hence a huge amount of variation should be included. Because they are not involved with any error and are true outliers, I also included all of outliers in Review, Price and Installs in data analysis.



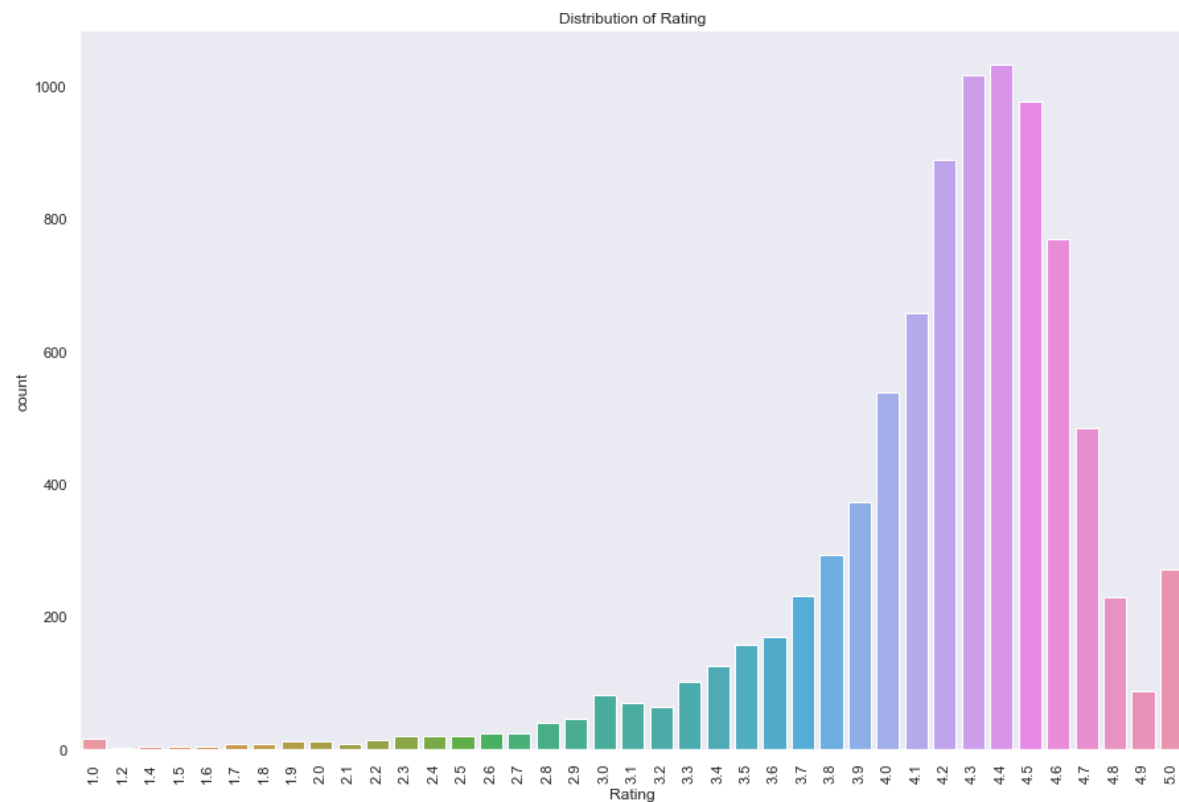
2.4 Data Visualization (5Pts)

1. The number of applications per category



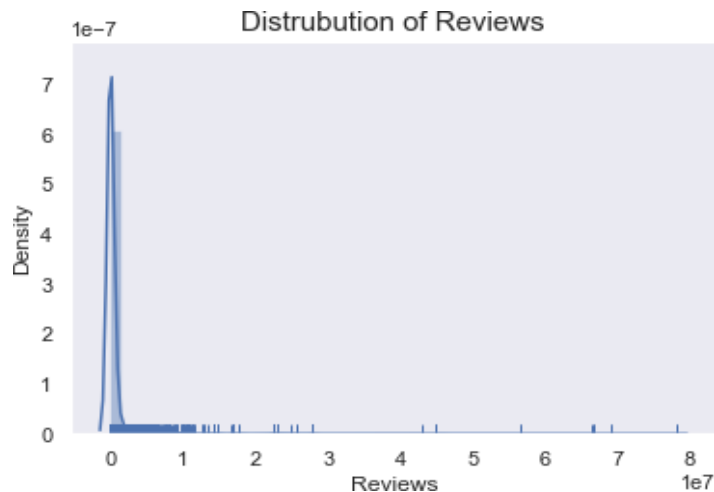
Interpretation: Family category has the most applications, followed by Game category and tool category.

2. Distribution of rating

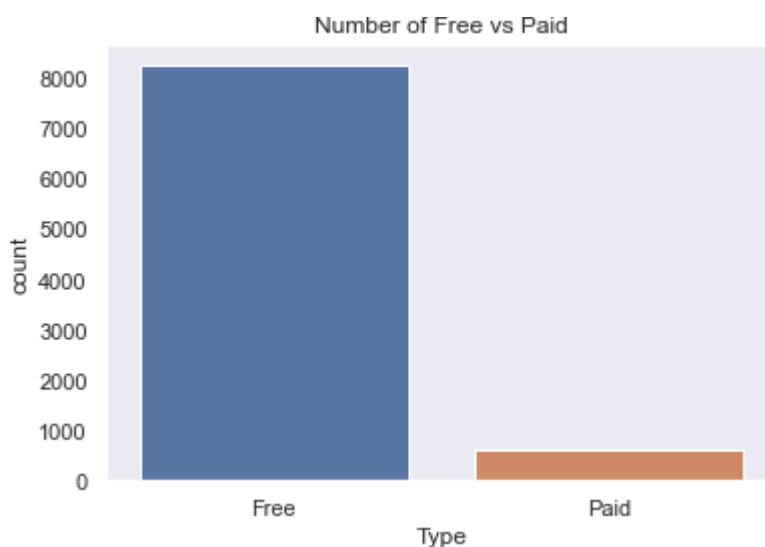


Interpretation: The highest range of ratings is between 4.1 and 4.6 with the highest in 4.4. Overall, most of applications have high ratings.

3. Distribution of reviews

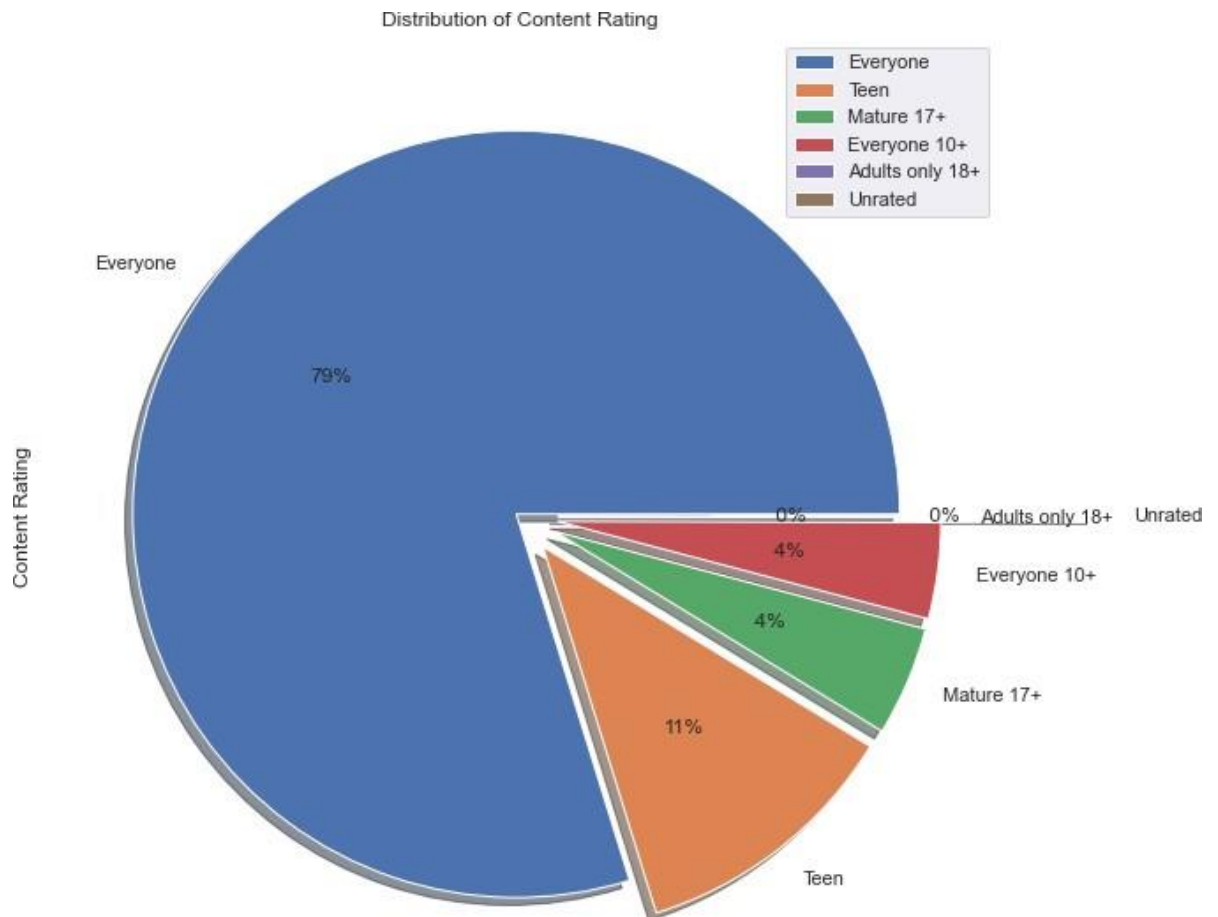


4. Distribution of applications that are free or paid



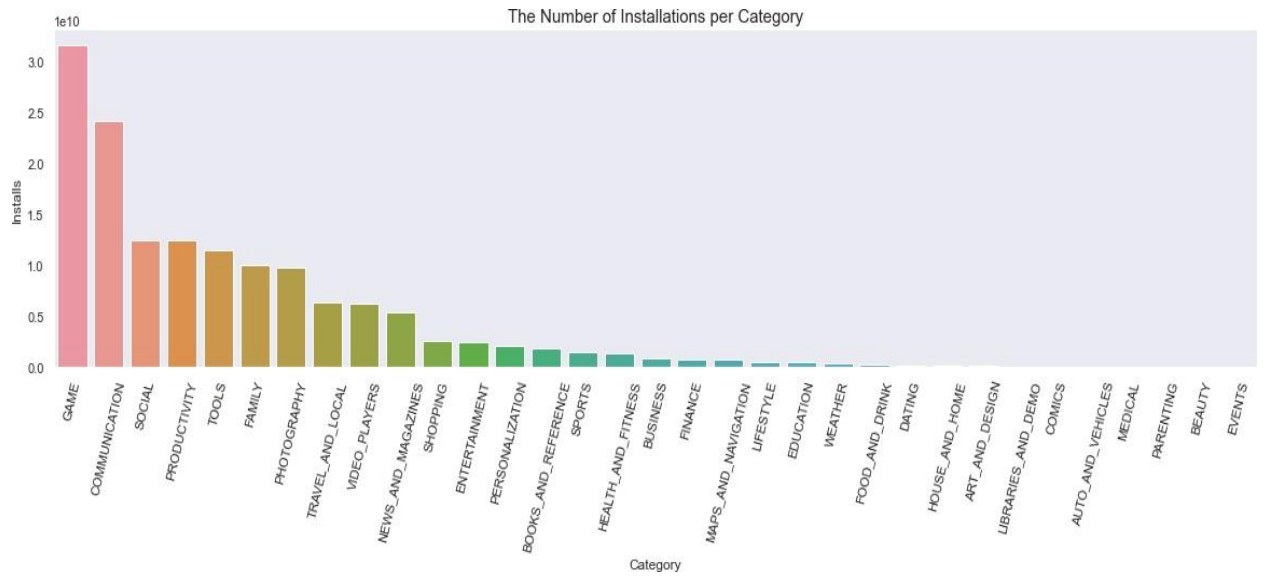
Interpretation: While most of applications are free (8274), there are some applications that should be paid (611).

5. Distribution of content rating



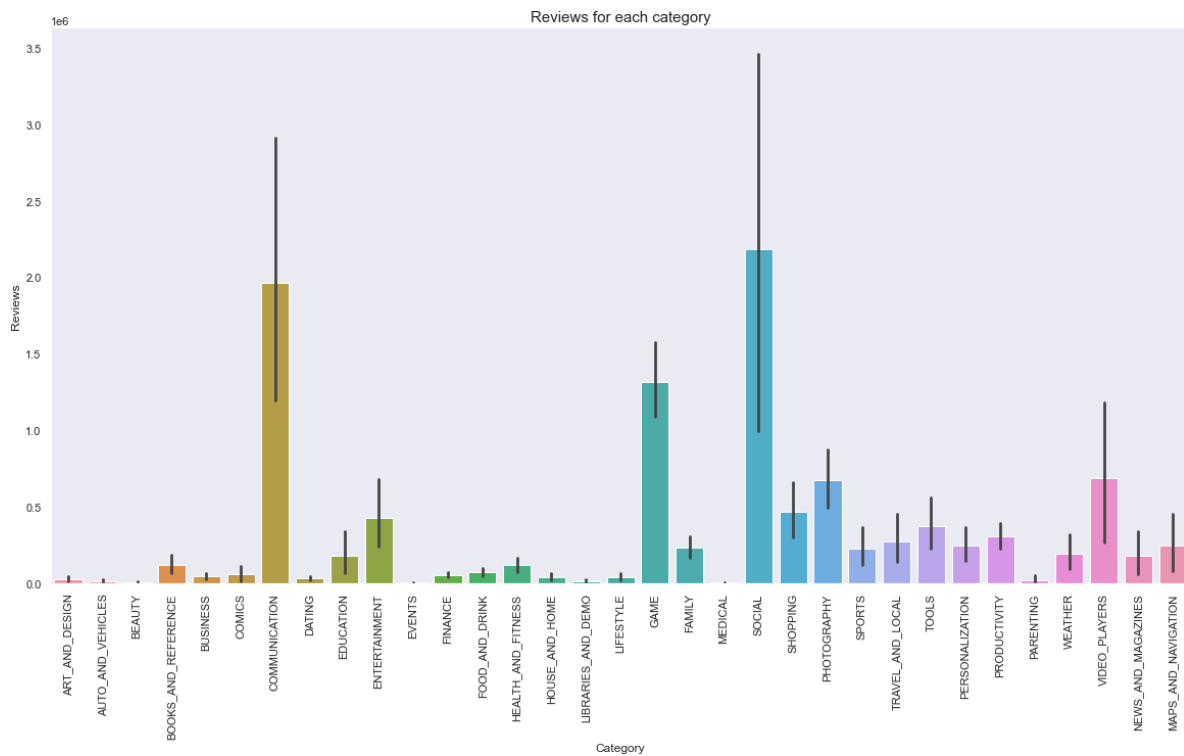
Interpretation: Application contents that are for everyone have the highest ratings, followed by teen contents, mature contents, 10+ contents and adults only contents.

6. The number of installations for categories



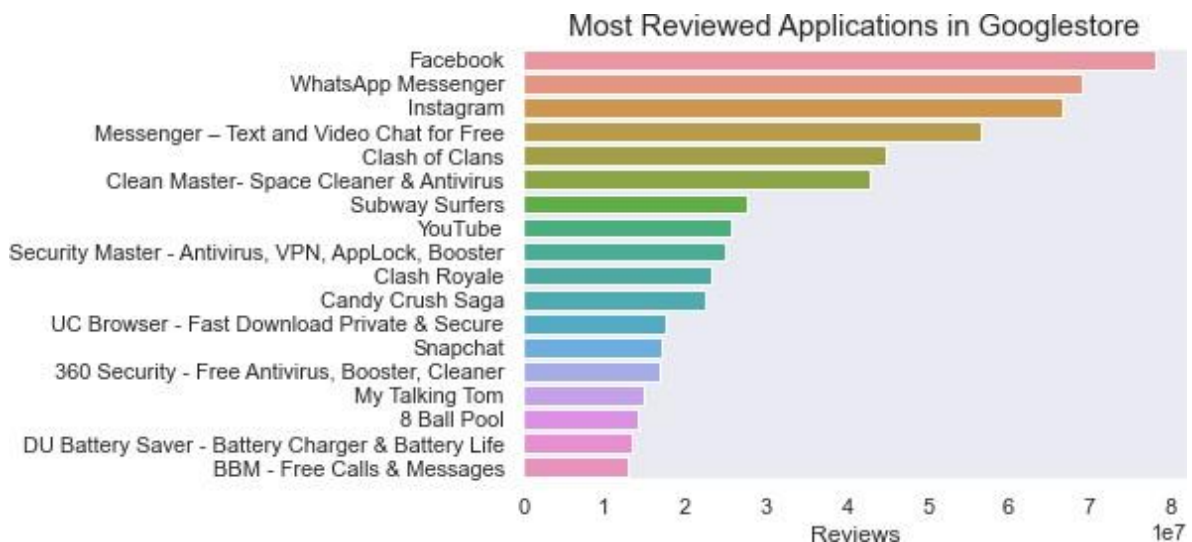
Interpretation: Game and communications are the most popular category applications based on the number of installs, but in order to confirm, need to look into other variables as well.

7. The distribution of reviews for categories



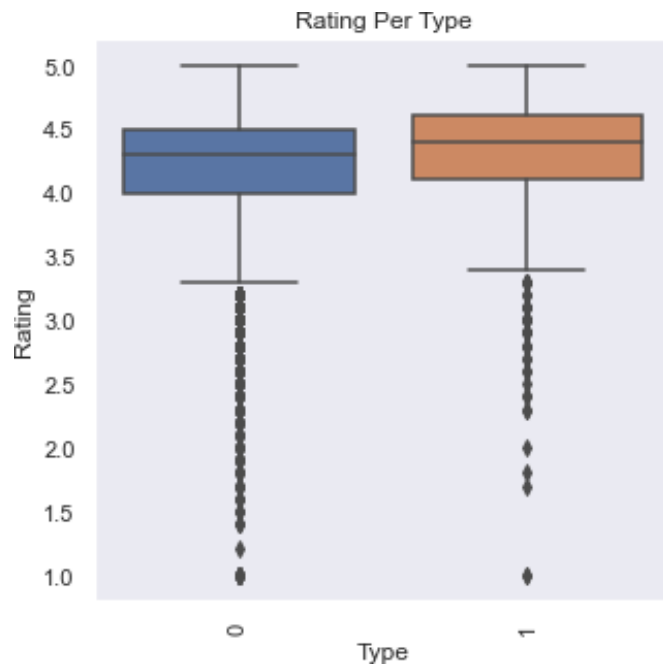
Interpretation: The top three most reviewed applications are Social, Communication and Game applications. Those three applications are thus very trendy and popular among people.

8. Top 18 most reviewed applications



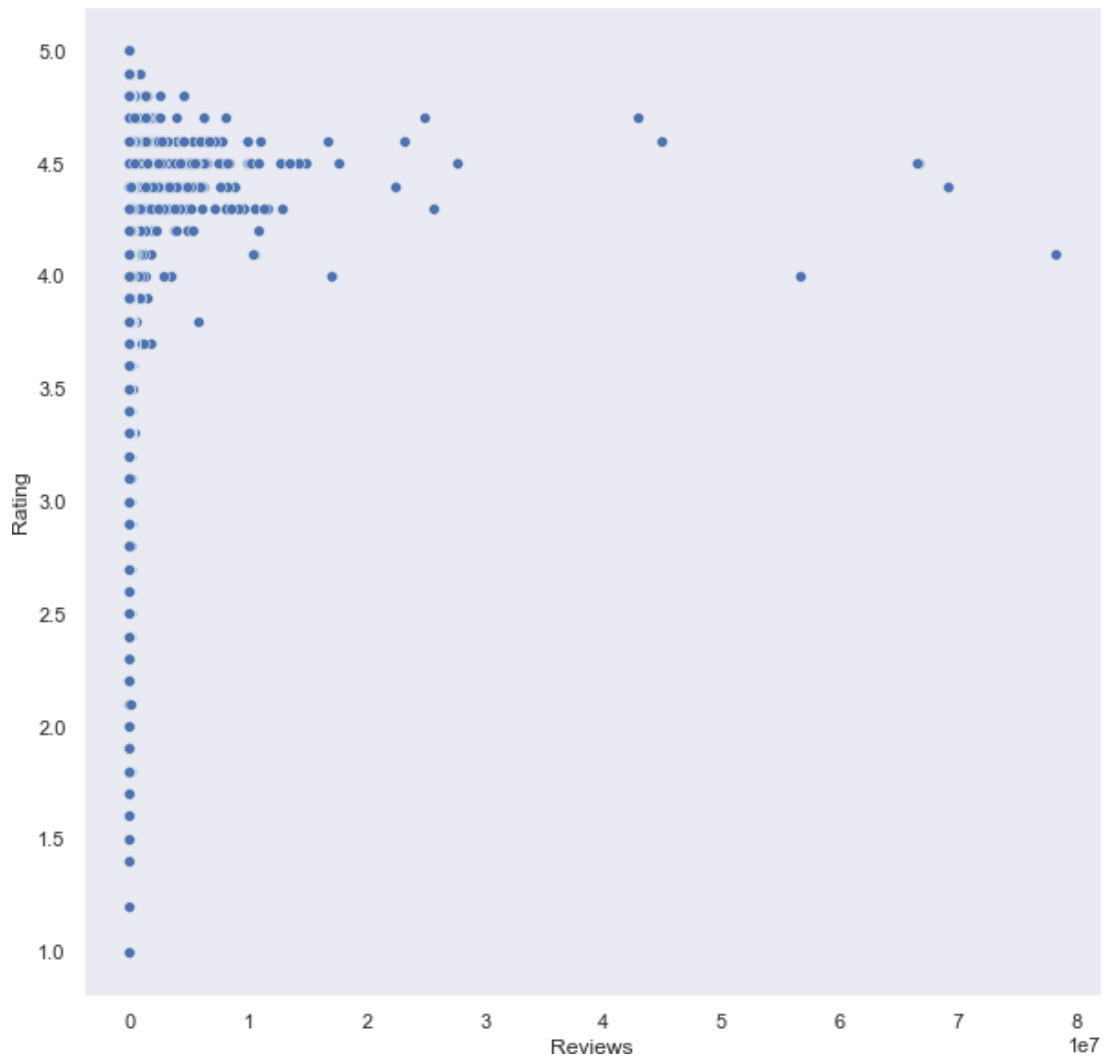
Interpretation: Most of those applications (E.g., Facebook, Instagram and Messenger) belong to either Game, Communication or Social category. This confirms that Game and communications are current trends in making applications.

9. The correlation between rating and type



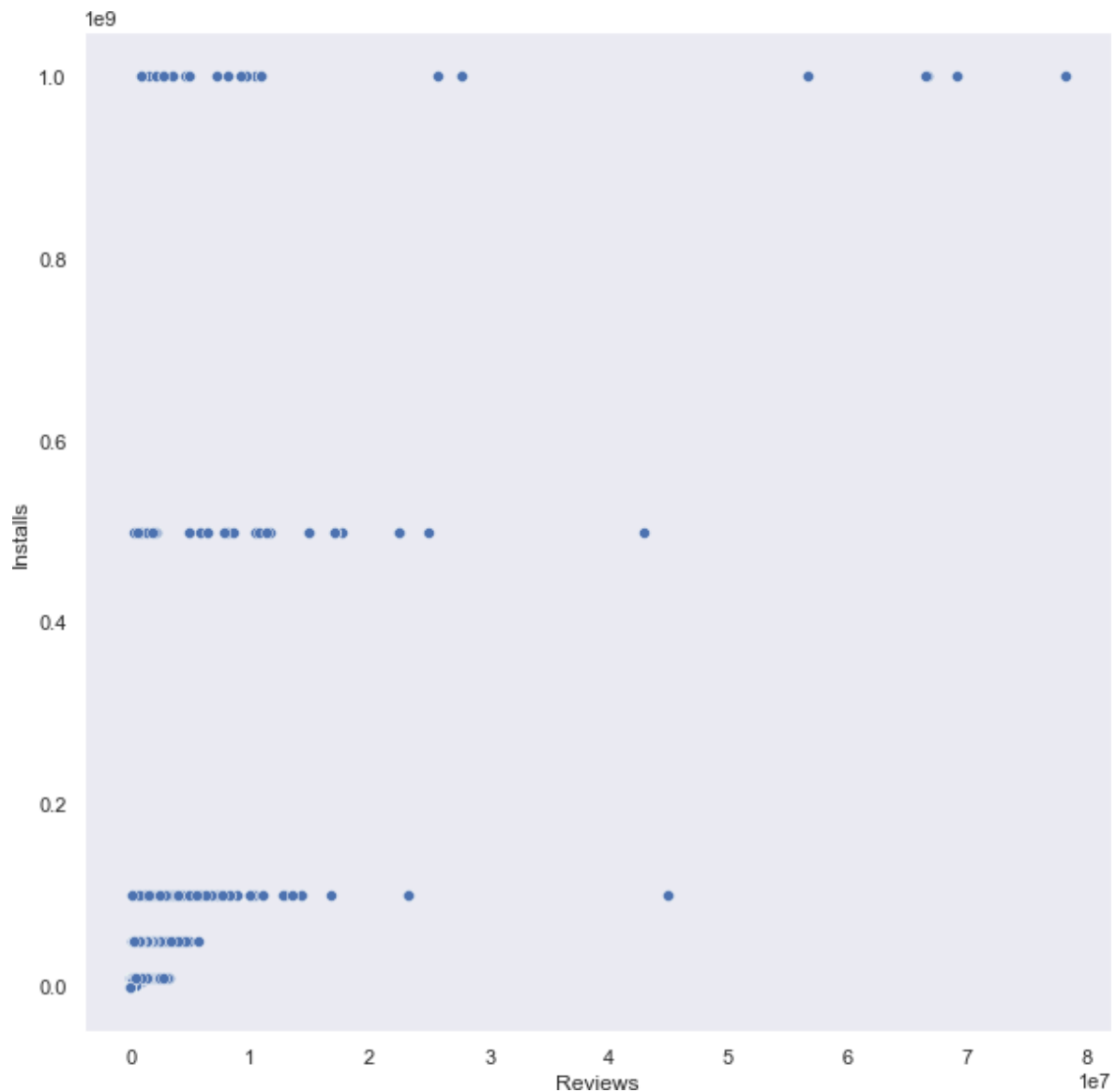
Interpretation: There is a slightly higher ratings in paid application than free applications.

10. Correlation between reviews and rating



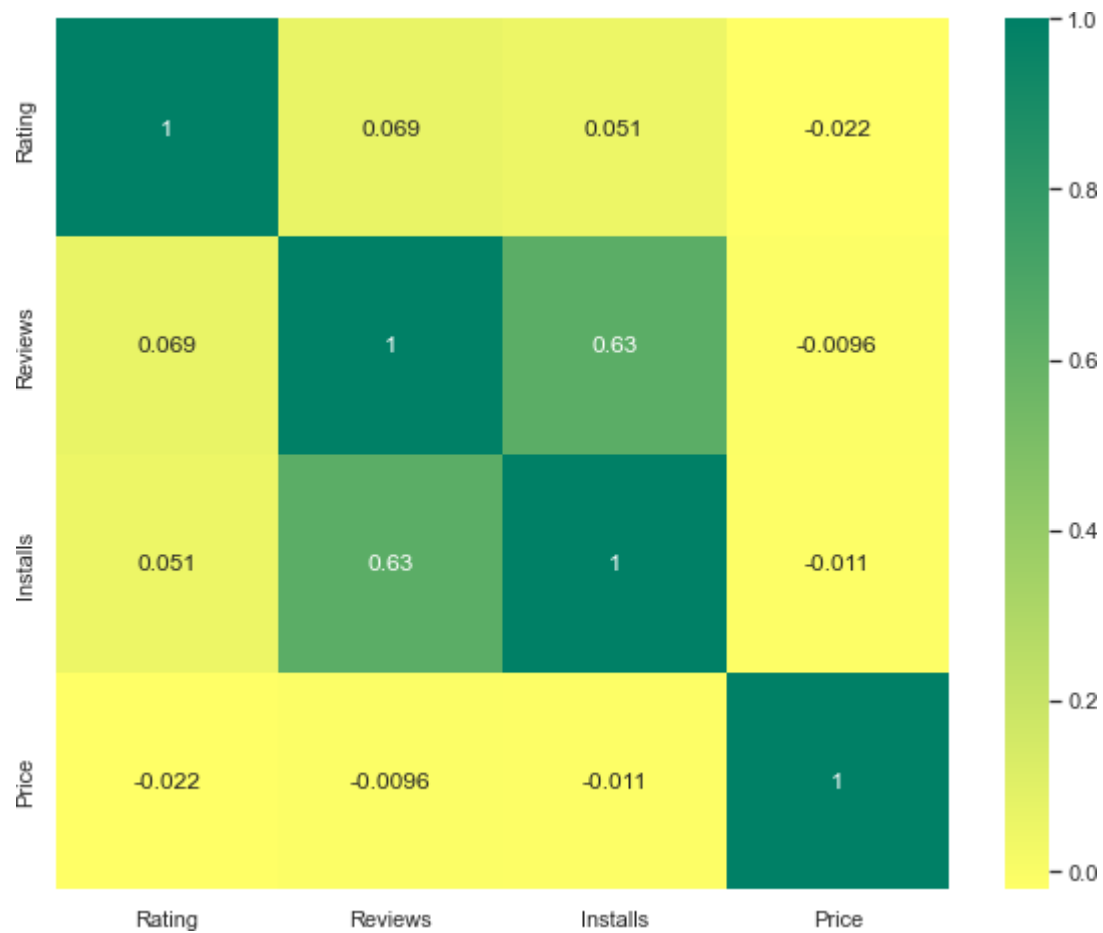
Interpretation: The scatterplot shows that there is a strong correlation between higher ratings and popular applications.

11. Correlation between reviews and installs



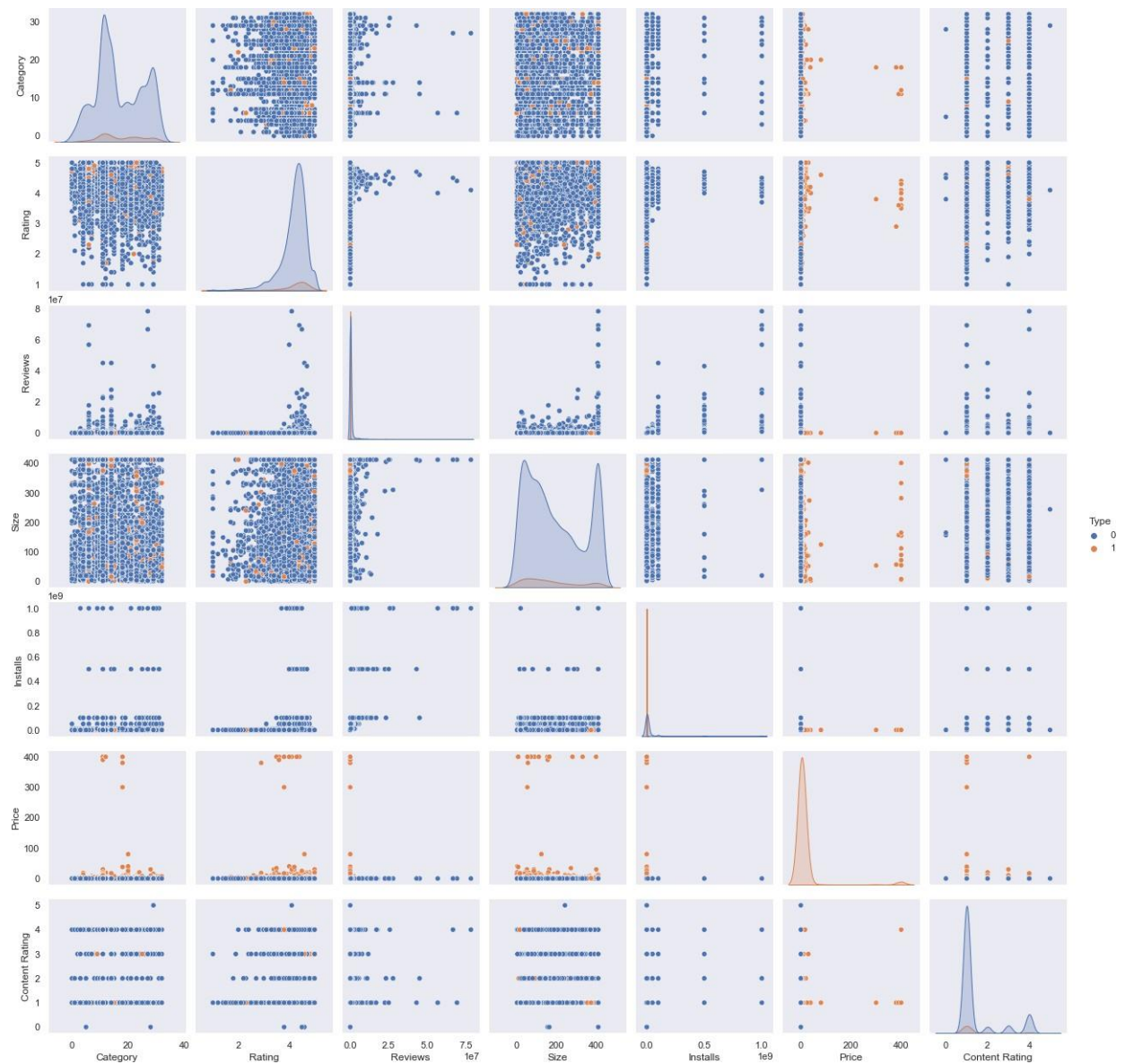
Interpretation: It shows that lower number of reviews is correlated to lower number of installs. Therefore, installs and reviews have a strong correlation.

12. Correlation between numeric variables



Interpretation: It confirms that there are moderate correlations between reviews and installs (0.63). On the other hand, reviews and rating and rating and installs are not correlated.

13. Pairwise relationship between variables



Interpretation: Applications that are paid mostly have very high ratings but less reviews and installations compared to free applications. Also, as applications are more expensive, they are less likely popular, as they have low reviews and installs.

3 Discussion and Conclusions (5pts)

This dataset represents enormous applications in the Google Play Store. First, most of applications are available for free rather than paid. We also successfully found the parameter of current trends in applications based on top reviewed applications and their categories and the number of installs and distribution of rating. According to the results, reviews and installs are correlated and therefore the higher number of reviews and install would represent the current trend in applications. High rating can also predict the trend in applications, but rating was not potentially correlated with other variables. While family and game applications have the highest number of applications in Google Play Store, family applications have the low number of installations and reviews. On the other hand, social and communication applications have very high proportion of installs and reviews but there are very few applications in Google Play Store. Consequently, game and social communication related applications would be very potential in application market.

1. The key variables responsible for the current trends of applications
 - The current trends of application can be found by the process of obtaining a range of applications and their categories based on rating, review and installs.
2. Which category of applications is currently the most potential and popular and close to current trend
 - Game and social communication applications have the most reviewed and installations and hence they are the most potential and close to current trend.
3. Current Top 10 popular applications
 - Game, Social and communication related applications. For example, Facebook, Instagram, messenger, Clash of Clans, Youtube.
4. Potential correlations between variables.
 - Reviews and installs are strongly correlated each other. Moreover, type of applications (free vs paid) and rating are somewhat correlated as well. Paid applications relatively have higher ratings than free applications.

Also, as paid applications are more expensive, they have low number of installs and reviews.

5. Applications, that should be paid, have higher rating and number of installations than application that are free.
- Applications, that are paid, have relatively higher rating but low number of installations, which decreases as the price of application increases.

4 References

Lavanya. (2019, February 3). *Google play store apps*. Kaggle. Retrieved October 28, 2022, from <https://www.kaggle.com/datasets/lava18/google-play-store-apps?select=googleplaystore.csv>