Analyzing and Predicting Ratings for Apps on Google Play Store

Nishtha

29th May, 2020

1. Introduction

1.1 Background

In this ever-changing era of technology and development, the ratio of mobile to desktop is increasing day-by-day. There is a huge outburst in the number of mobile companies launching new phones every single day and this, along with it, has brought an even larger boon in the mobile app market. As of 2019, the users had around 2.6 million Android Apps to choose from and this number is ever-increasing. The Apple App Store and the Google Play Store dominate this landscape and are the two most important marketplaces in the world. Android holds around 53% of the smartphone market.

With the large amount of scope this market possesses, it is important for a developer to know which apps are preferred by the users. This is because most apps are free on these stores, so the major source of revenue is the in-app adds. So, the larger the audience for the app, the larger will be the income for the app. Therefore, we want to analyze the data so as to find out the preferences of users in this domain and to build apps that will bring the best revenue and audience and Mobile App Analytics is a great way to understand the existing strategy to drive growth and retention of future user.

1.2 Problem

User ratings and reviews are huge factor that help in determining the most popular apps among users. Out of the millions of apps in the market, we take around 10k of them and use information such as the genre, reviews, size, price, installs etc. and try to predict the user ratings for the apps so that when we develop a new app with certain features, we can predict beforehand how well it will do among the users.

1.3 Interest

Any developer who will be putting a lot of effort in making a new app would want to know beforehand how his app will do in the market. So this project would interest all developers. Apart from that, it would interest the advertising and marketing companies who would want to know the most favored apps so that they can advertise to reach a larger audience. It may also interest the users who want to maybe find the best apps in the field of their interest.

2. Data Acquisition and Cleaning

2.1 Data Sources

The data regarding the details of Google Play Store Apps can be found at the kaggle dataset here. The information for the reviews of customers for Google Play Store Apps can be found at kaggle dataset here. All the data sets have been picked from kaggle.

2.2 Data Cleaning

Loading and Basic Description: The Google Play Store data file had 10,841 rows and 13 columns which contained information regarding App name, category, last updated, current version, genres, review, size, type, installs etc. and the Google User reviews dataset had 64,295 rows and 5 columns originally with app name, numeric sentiment analysis and app description. But the translated_review (description) column had huge textual lines which were not going to be used either in analysis or training so the column was dropped before adding to the project.

Detecting Anomaly: Next we went through the discussions section here and the summary of the dataset on kaggle and found in one of the discussions here that there is a row with rating greater than 5 i.e. 19 which is an anomaly as Google rates apps between 0 and 5 only. So we removed this entry from the dataset as this was the only outlier and removing the entry won't cause any harm.

Duplicate Entries: Another thing discovered on going through the data was that there were a lot of entries that were duplicates. While some of them had just same values for all the columns more than once, some had different values for the reviews column which may be due to different times at which it was recorded. But there was not much difference in the values and more than one value could confuse our model and would train for the same model with different values. There were 1181 duplicate entries in total and 9659 unique apps (after dropping the anomaly). So, we dropped these duplicates and now our dataset has 9659 rows and 13 columns.

Handling missing values and unnecessary columns: Analyzing the data on kaggle we see that there are not many missing values in most columns but since we dropped the duplicates, so, we don't know the exact number of the missing values. So, we run checks and find out that there is one missing values in the Type column and we see that this column has only two values Free and Paid. So, on checking the corresponding Price column we find that the app is actually free and thus we fill in the value Free. Also, there are some missing values found in the Current Ver and Android Ver columns but since we will not be using these columns in analysis and modeling, so dropping their corresponding rows is not beneficial. So, we remove these columns along with the Last Updated column.

Handling data types: Lastly, we see that all columns except the ratings had object data type. Columns such as App, Category, Content Rating and Genre are supposed to be objects as they are strings but columns such as Reviews, Size, Installs and Price are clearly not object data types. So, we changed the data type of the reviews column; stripped off the '+' and ',' in Installs column and then changed the data type to integer. Next, we stripped the dollar sign off the Price values for the paid

apps and then changed the type to float and finally, for he size column firstly we replaced the string values 'Varies With device' to 0 and then replace the 'k' and 'M' for 1,000 and 1,000,000 respectively for the size of the apps. Also for the apps with Varies with device, we don't want the size to be zero as this is not the case for any app. So we replace that with the average values in the data frame.

3. Exploratory Data Analysis

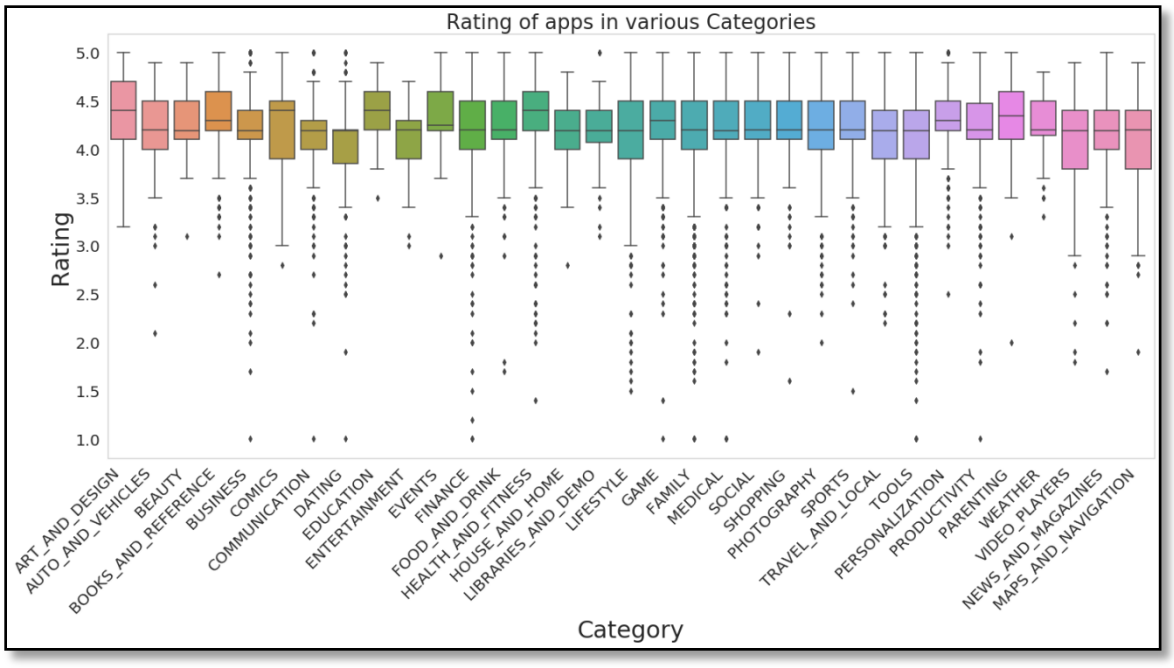
3.1 Basic Descriptive Analysis

	App	Category	Rating	Reviews	Size	Installs	Type	Price	Content Rating	Genres
count	9659	9659	9659.000000	9.659000e+03	9.659000e+03	9.659000e+03	9659	9659.000000	9659	9659
unique	9659	33	NaN	NaN	NaN	NaN	2	NaN	6	118
top	Sticky Note + : Sync Notes	FAMILY	NaN	NaN	NaN	NaN	Free	NaN	Everyone	Tools
freq	1	1832	NaN	NaN	NaN	NaN	8903	NaN	7903	826
mean	NaN	NaN	4.176287	2.165926e+05	2.006433e+07	7.777507e+06	NaN	1.099299	NaN	NaN
std	NaN	NaN	0.494365	1.831320e+06	2.041318e+07	5.375828e+07	NaN	16.852152	NaN	NaN
min	NaN	NaN	1.000000	0.000000e+00	0.000000e+00	0.000000e+00	NaN	0.000000	NaN	NaN
25%	NaN	NaN	4.000000	2.500000e+01	5.300000e+06	1.000000e+03	NaN	0.000000	NaN	NaN
50%	NaN	NaN	4.200000	9.670000e+02	1.600000e+07	1.000000e+05	NaN	0.000000	NaN	NaN
75%	NaN	NaN	4.500000	2.940100e+04	2.500000e+07	1.000000e+06	NaN	0.000000	NaN	NaN
max	NaN	NaN	5.000000	7.815831e+07	1.000000e+08	1.000000e+09	NaN	400.000000	NaN	NaN

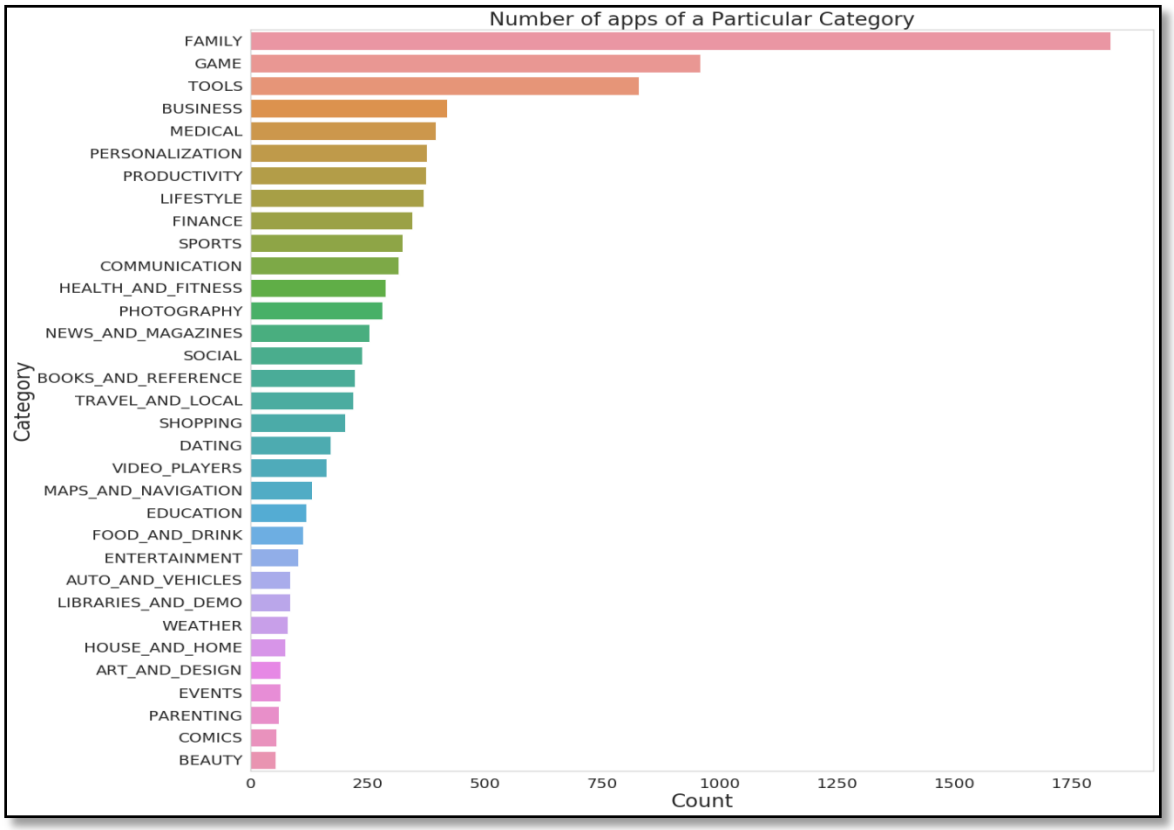
We, therefore see that most apps are for Family (1829) and almost all of the apps on the Google Play Store are free (~92%). Also, most of them have 'Everyone' content rating i.e. they can be used by all including children. Also for the integer columns, we can get the statistical values like mean, standard deviation, maximum and minimum for further insights. The column Category has 33 unique values and family apps are most frequent out of them. So, a lot of developers seem to have preferred this domain maybe due to large number of accessible viewership.

3.2 Visual Analysis

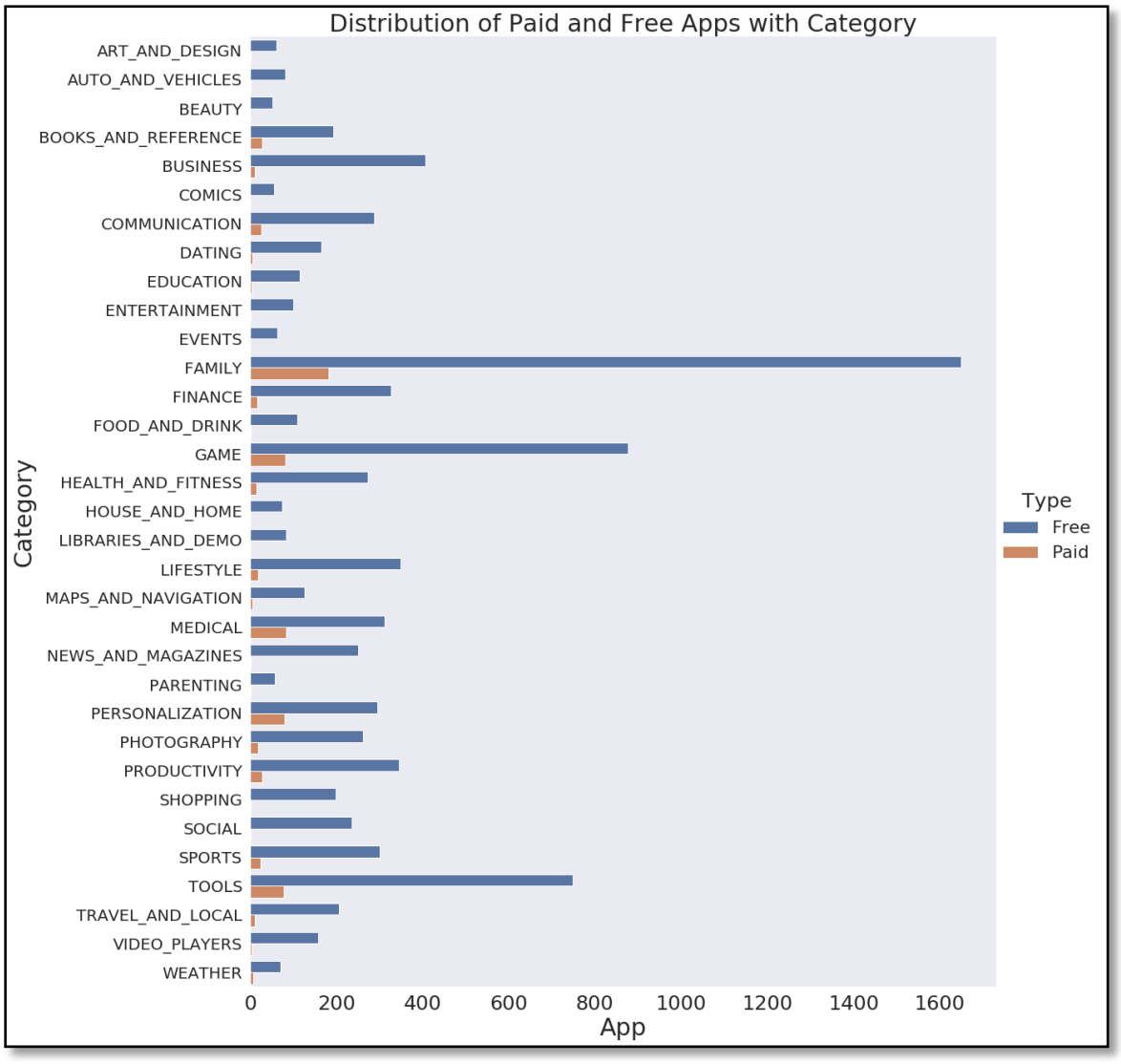
- **Influence of Categories on Ratings:** With the help of the grid, we see that all the apps have their mean ratings between 4 and 4.5. We also see that while some categories like weather have no outliers, maps and navigation, house and home, events, education have one outlier on the other hand, categories like finance, tools, games, family have a lot of outliers. The reason can be due to the presence of large number of apps in these categories thus, increasing the chances of outliers while those with less outlier are also less in number as seen from the graph before.



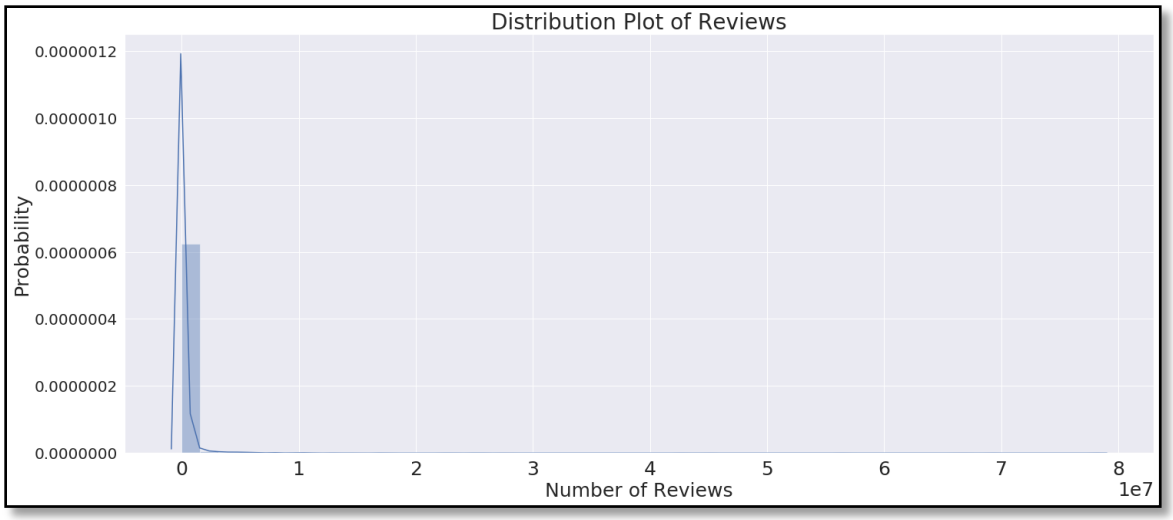
- **Most Important Categories:** The top five most important categories include Family with 1,832 apps, Games with 959 apps, Tools with 827 apps, Business with 420 apps and medical with 395 apps. The categories such as beauty, comics and parenting are the least common in the market of apps.



- **Relationship of type of app and category:** Here, we see that in each category the count of free apps is way more than that of paid apps. The category with most number of paid apps is family and categories like games, personalization, tools and medical also have quite a few paid apps. Rest of the categories has a very small number of paid apps. Similarly, for free apps, family category is at the top followed by games and tools category.

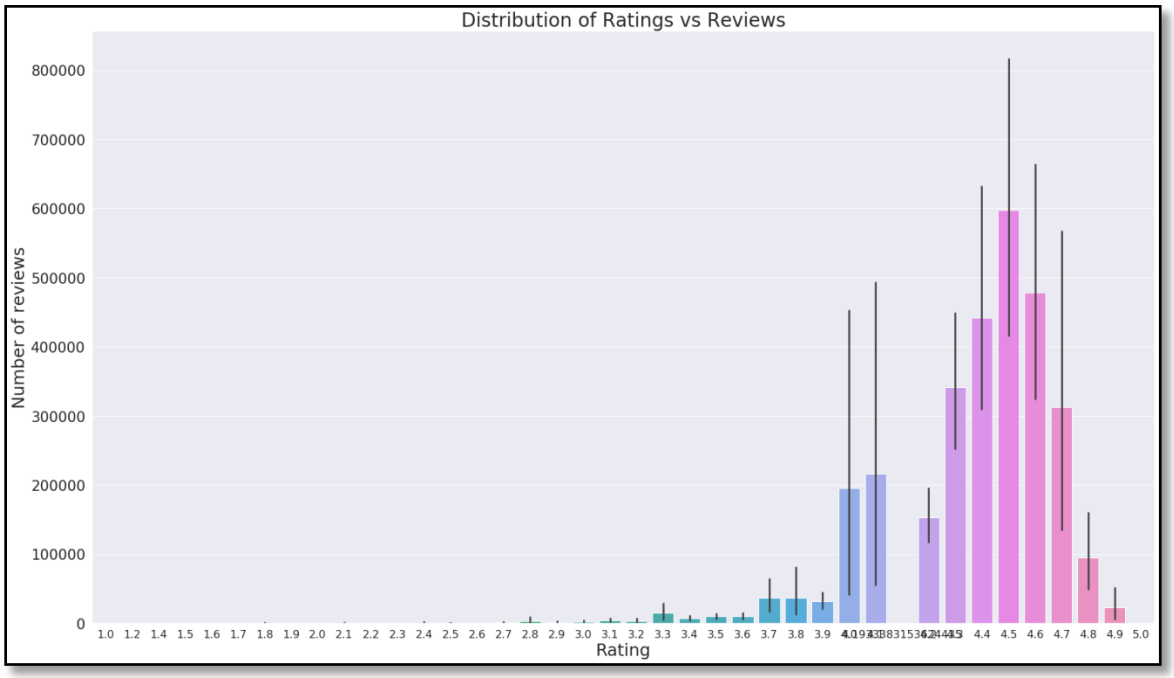


- **Getting to know more about reviews:** The above distribution is clearly skewed. Apps with very few reviews easily managed to get 5.0 ratings which can be misleading. With this plot, we see that as the probability of having a very high number of reviews is very less. Most apps have reviews near 1,000,000 from the above plot and this can be confirmed by the fact that the mean of reviews column is 1.310014e+06. Thus, most apps have high chance of having reviews around that value only.

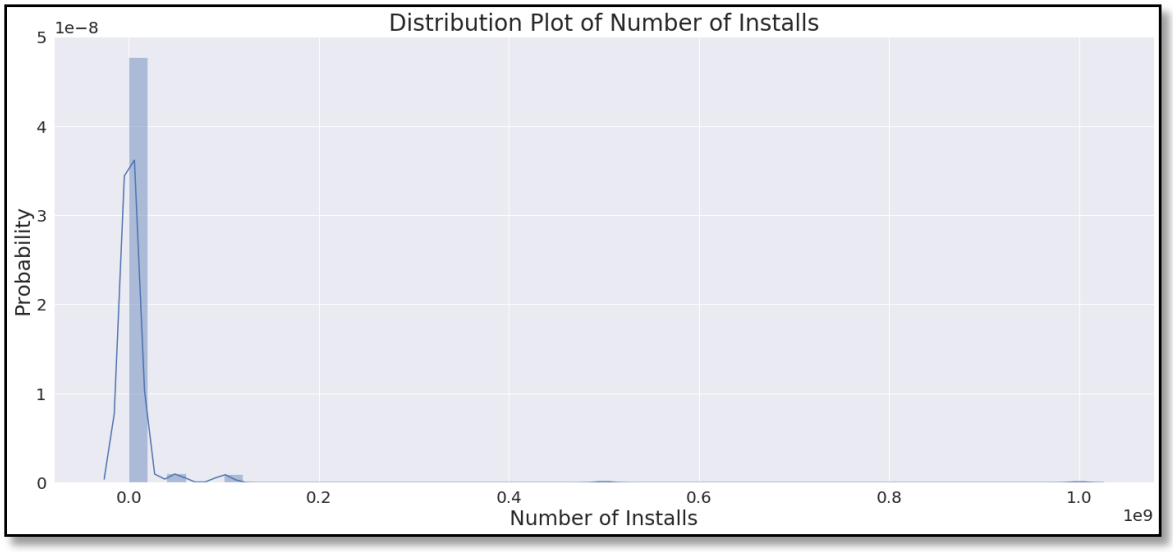


- **Most Reviewed Apps:** We get an interesting insight from this and see that the most viewed apps are the ones related to social media and communication. And this should not be a surprise seeing the popularity and reach these apps have got these days. Facebook is the most reviewed app today with 1 Billion+ installs. We also see that although the rating is not very good, yet it is very popular among the users and is most reviewed. In fact all the top reviewed apps don't have the best rating still they are popular among users.

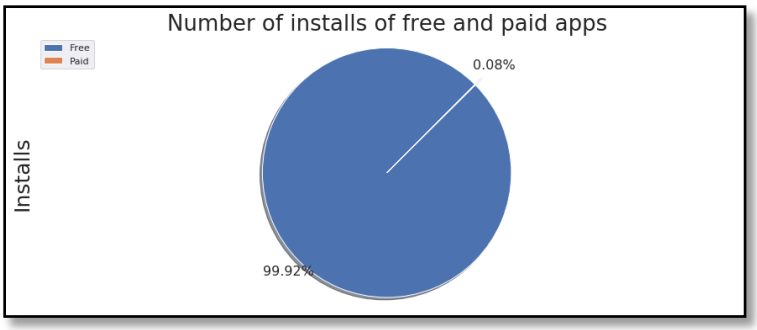
App	Rating	Category	Reviews
Facebook	4.1	Social	78,158,306
WhatsApp Messenger	4.4	Communication	69,119,316
Instagram	4.5	Social	66,577,313
Messenger-Text and Video Chat for Free	4.0	Communication	56,642,847



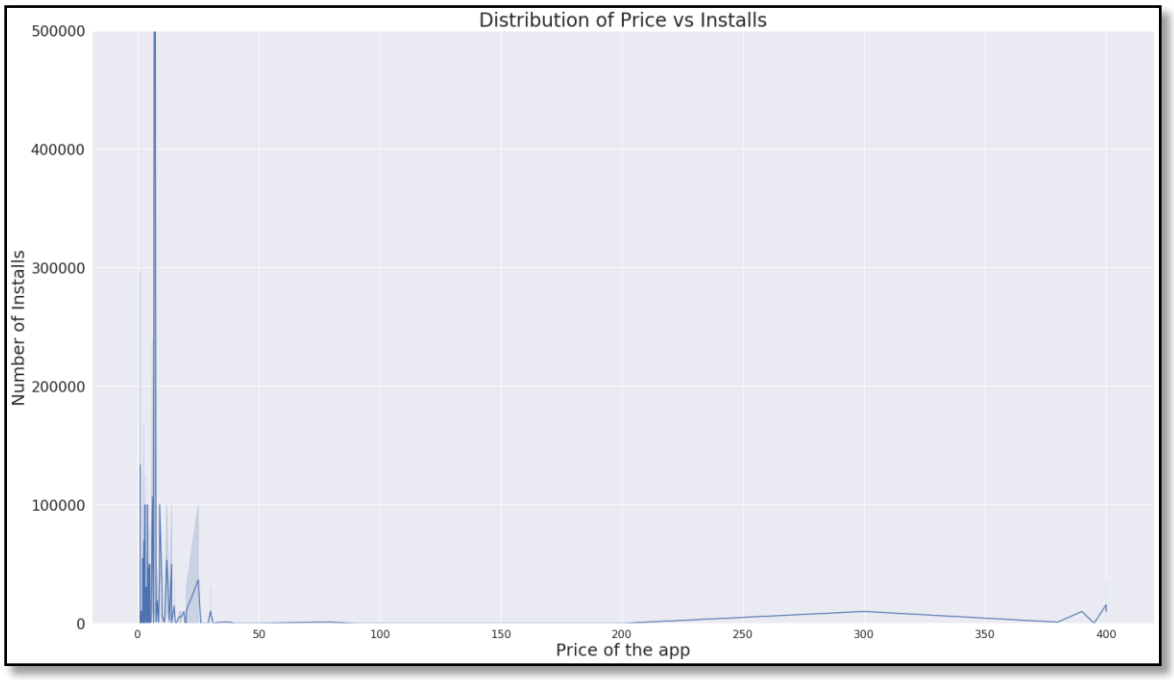
- **Reviews and Ratings:** Here, we see that the apps with rating around 4.5 are the most rated. And the highest rated apps are not the most rated. In fact, close to 5 rating, number of reviews is very less. It can be due to the fact that the more the number of people rating and reviewing, the more is the variation in the reviews and thus the rating average comes a bit down as not everyone would like the same app the same way.
- **Distribution plot of Installs:** Again we see to have maximum probability of having around a 100 Million downloads. Only few have more than a billion downloads.



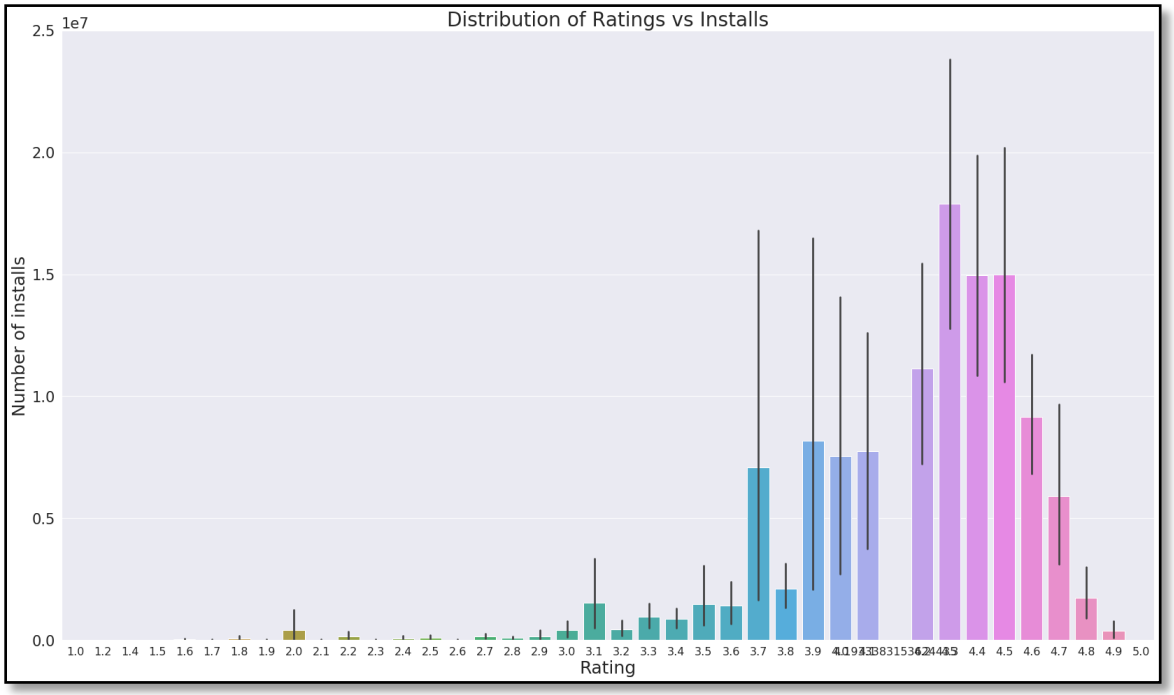
- **Most Installed Apps:** We thus see that a lot of have more than 1 Billion installs. These include Facebook, Instagram, messenger as well (they were the most rated apps). Also we have Google Play Books, Google Play Movies & Tv, YouTube, Google Drive, Google, Google Street View, Google Photos, Google+, and Subway Servers. In fact a lot of Google apps o the list due to the fact that Android has a majority of smartphone market and these come installed in android phones. All these apps are free which is obvious because users would prefer to download free apps. Also we see that none of these apps are rated adult and all are either 'Teen' or 'Everyone'.
- **Installs and Type:** We can see from the number and plot that people prefer to install free apps way more than paid apps. And so, any developer should aim to make his app free. From analysis, we found that 75,065,572,646 installs were of free apps and only 57,364,881 installs were of paid apps.



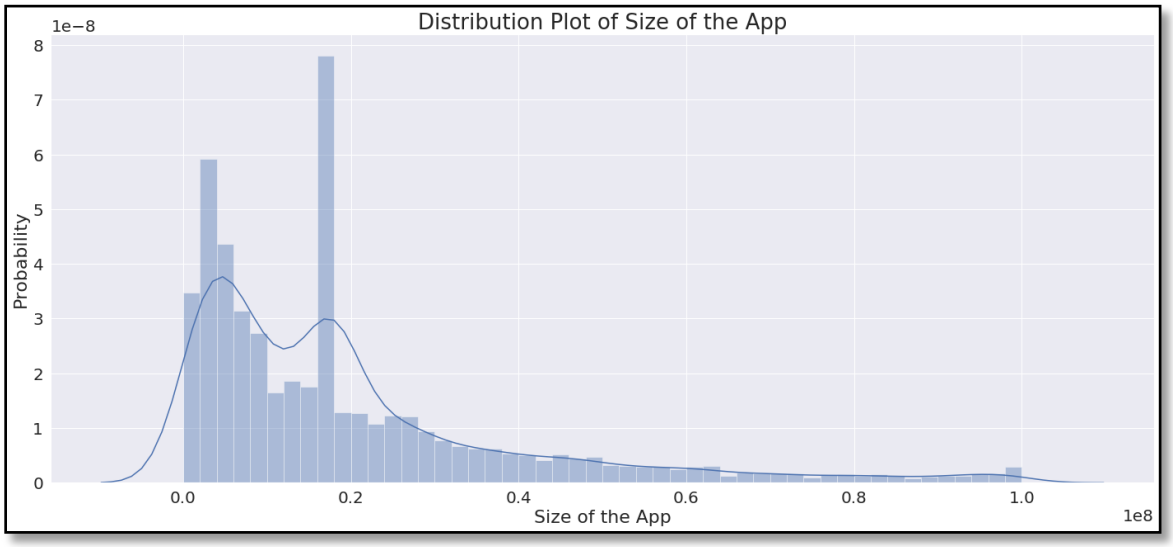
- **Paid Apps and Installs:** We can see that although with there is some demand of apps with minimal prices around 0 to 25$ but the higher paid apps are not usually preferred by the people. Of course, who would want to spend a ton of money on an app. So even if a developer aims to make his app paid, he may get customers if his rates are minimal and the content has to be very good for him to succeed.



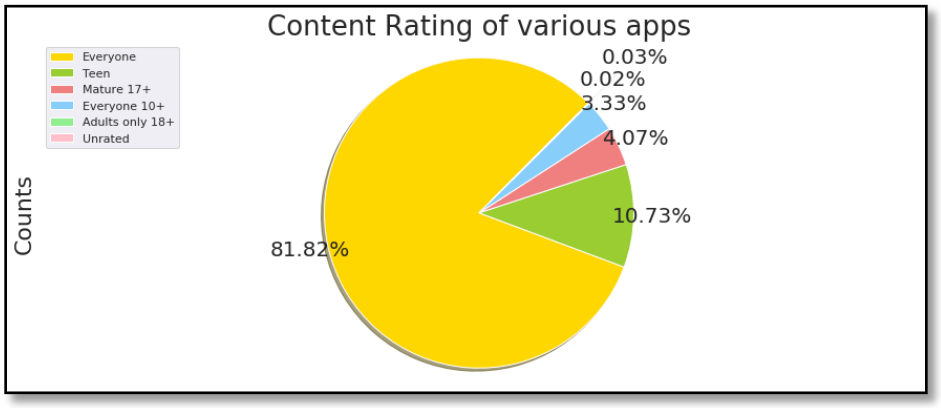
- **Installs and Ratings:** We find quite a few number of ups and downs in this graph but overall again, the number of installs is highest around the mean rating of 4.3 ~ 4.5.



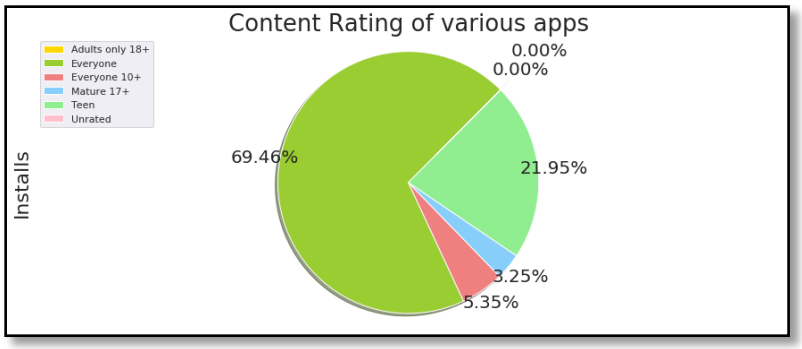
- **Distribution Plot of Size of App:** From the plot, it is clear to see that apps around size 0 to 2 M are mostly found in the dataset. This is also a major factor as the developer would not want to take his app too heavy so that it takes up too much space. Small sized apps are always a preference for users as they may have space issues in their mobiles. Although, in today's era, the capacity of a mobile phone has increased a lot in terms of space, still it's always preferable to make small sized apps.



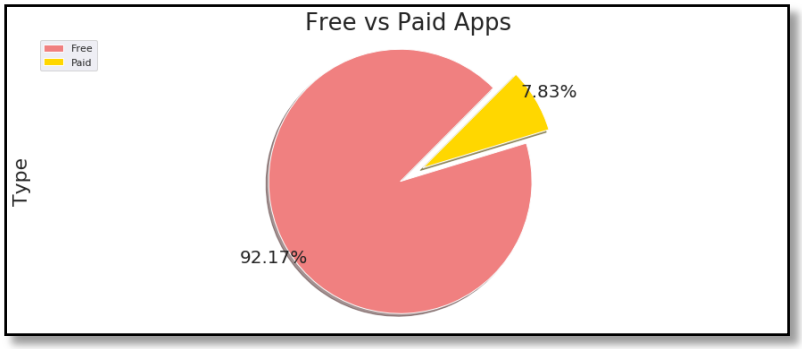
- **Content Rating of various Apps:** Here, we see ~82% of the apps i.e. 7,903 have 'Everyone' rating. Next most common category is 'Teen' (1,036) which has around 11% of the apps. Unrated and Adult only apps are only 2 and 3 in number respectively. Thus, most developers don't want to make adult apps as it limits their viewership.



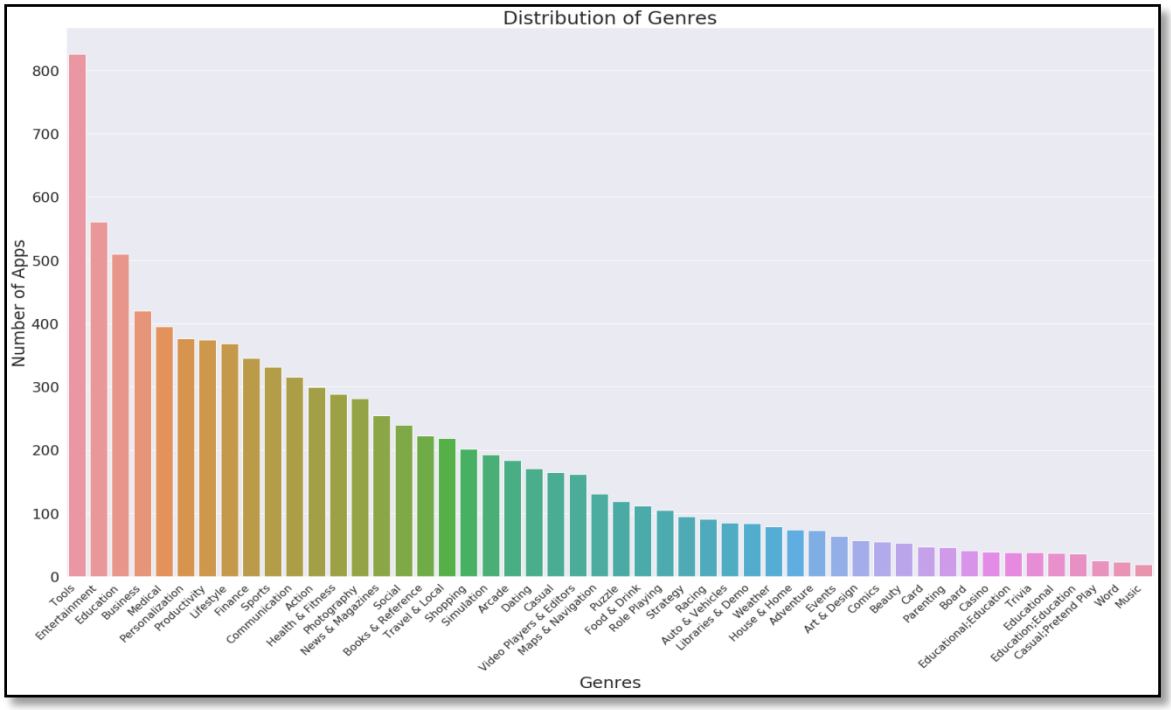
- **Content Rating and Installs:** With no surprises, 'Everyone' content rating has the most number of installs i.e. 52,179,352,961 followed by 'Teen' 16,487,275,393 and 'Everyone 10+' 4,016,271,795. Adult rated apps have 2,000,000 installs and unrated apps are least preferred with 50,500 installs only.



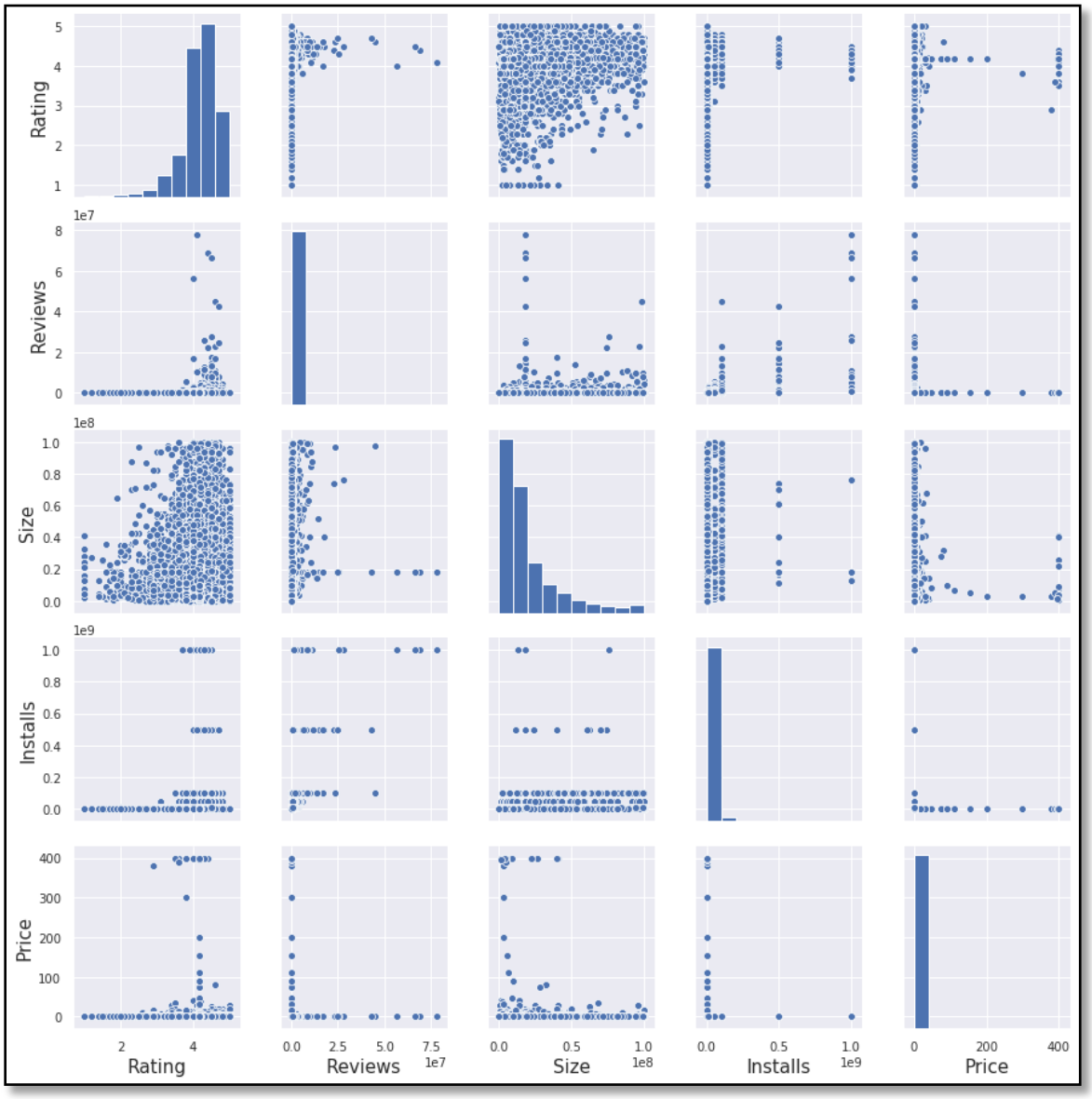
- **Free vs. Paid Apps:** With no surprises, maximum number of apps i.e. 8,903 are free on Google Play Store and only 756 apps are paid. We saw this earlier also with the categories plot where almost all categories had majority of free apps and also when we counted installs of free and paid apps with ~99.9% installs comprising of free apps.



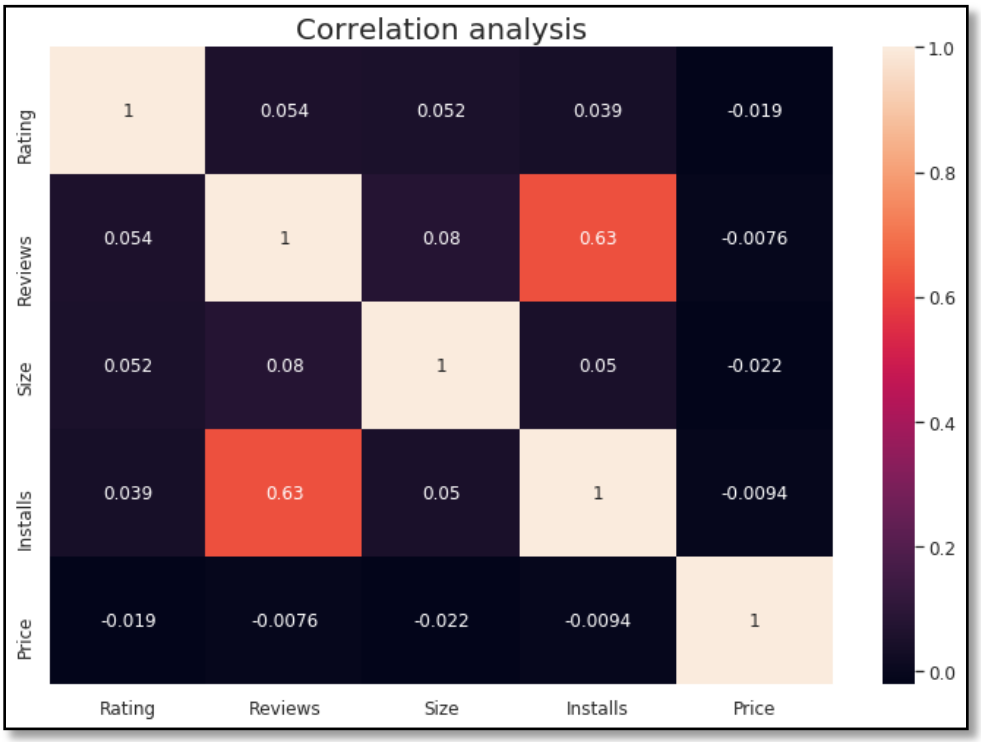
- **Exploring Genres:** Here, we see that Tools is the most preferred genre followed by Entertainment and Education. We have a graph of top 50 genres vs. number of apps.



- **Pairwise plot of features:** With the help of pairwise plot, we can see how the various features plot against each other and see the relation they have with each other. If, we may miss any combination of graph, it would cover it all and give the plot of each numeric feature against the other.



- **Correlation Map of features:** From the correlation map of the features in our data frame, we see that there is not linear relation for any two features except for Reviews and Installs which have a possibility of some sort of linear relationship with each other. Apart from that and the diagonals all values are pretty close to zero which indicates that these features have weak linear dependence on each other. Therefore, linear regression won't give good results on this data and so we would chose other forms of regression.



- **Looking at the Reviews Data frame:** In this, we find that we have reviews for 1074 different apps in this data frame and we see the count of reviews for each app. And we find out that Bowmasters, Angry Birds and CBS Sports Apps have the most number of reviews in this data frame. But we also see that this has reviews for only around 1074 apps while there are 9569 apps in the other data set. So this does not have information regarding all the apps and so we will not rely on this data set.

App Name	Reviews Count
Bowmasters	320
Angry Birds Classic	320
CBS Sports App	320

- **Analyzing Sentiment Polarity:** Polarity, also known as orientation is the emotion expressed in the sentence. It can be positive, negative or neutral. Subjectivity is when text is an explanatory article which must be analyzed in context of the textual review. We see that Home Work App has the highest sentiment polarity of 1 i.e. its reviews have positive emotions followed by the Google Slides App with sentiment polarity 0.934 which is also very high.

4. Data Modeling

4.1 Preprocessing and Feature Engineering

Dependent and Independent Variable: First, we will split the data frame to independent and dependent variables. Here, we will be predicting Rating in our model so it will be our dependent variable.

Label Encoding: Next, we dropped the names of the apps and also label encoded the features category, content rating, genres and type as we can't use string in modeling process.

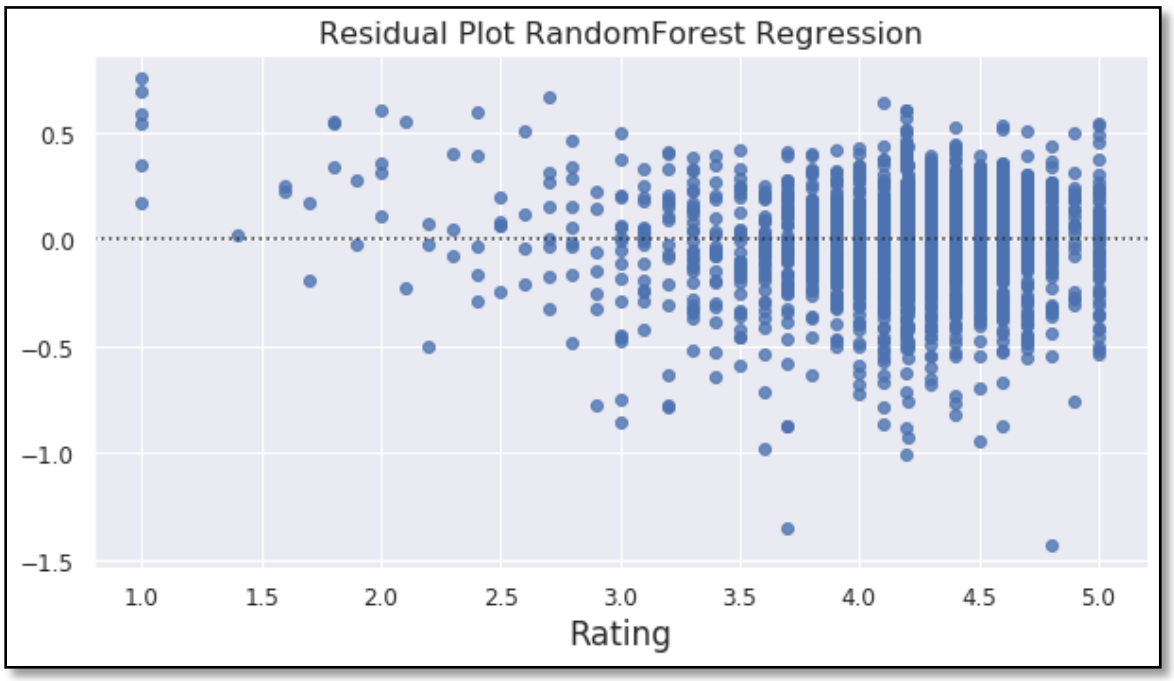
Showing Dependency of features on Target Variable: We calculate this in the form of an array which shows the dependency of the eight columns Category, Reviews, Size, Installs, Type, Price, Content Rating and Genres with the target variable. Rating which measures the dependency between the variables. It is equal to zero if and only if two random variables are independent, and higher values mean higher dependency. We see that Reviews and Installs seem to have a higher dependency than others.

Scaling and Train/Test Split: Using sklearn Standard Scaler and fit transform features; we scale the variable for faster training. And then we split the dataset into train and test data sets with 7,244 entries in the train set and 2,415 in the test set.

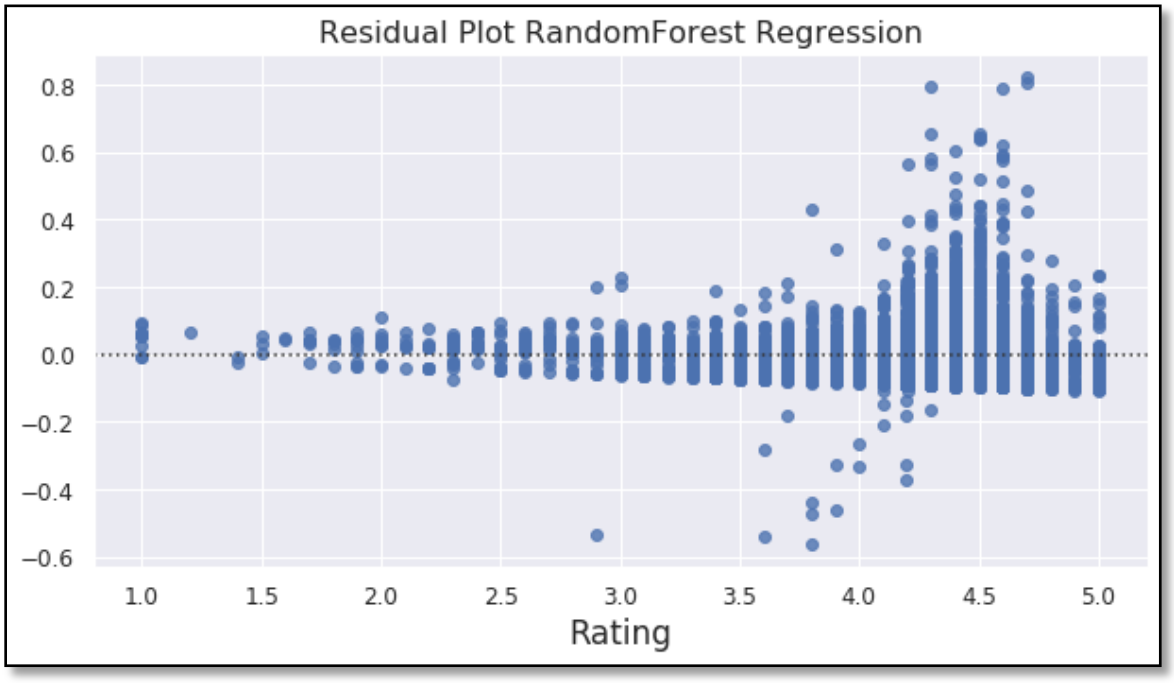
4.2 Regression and Training

Random Forest Regression: We used this model with n_estimators value set to 200 and criterion mean square error as default with max_features set to auto and max_depth as none. We saw that it had a very low train set mean square error of 0.03139 and test set error of 0.2299. It turned out to be our best model. We can see the residual plot for the train and the test set, and find the difference.

Test Set

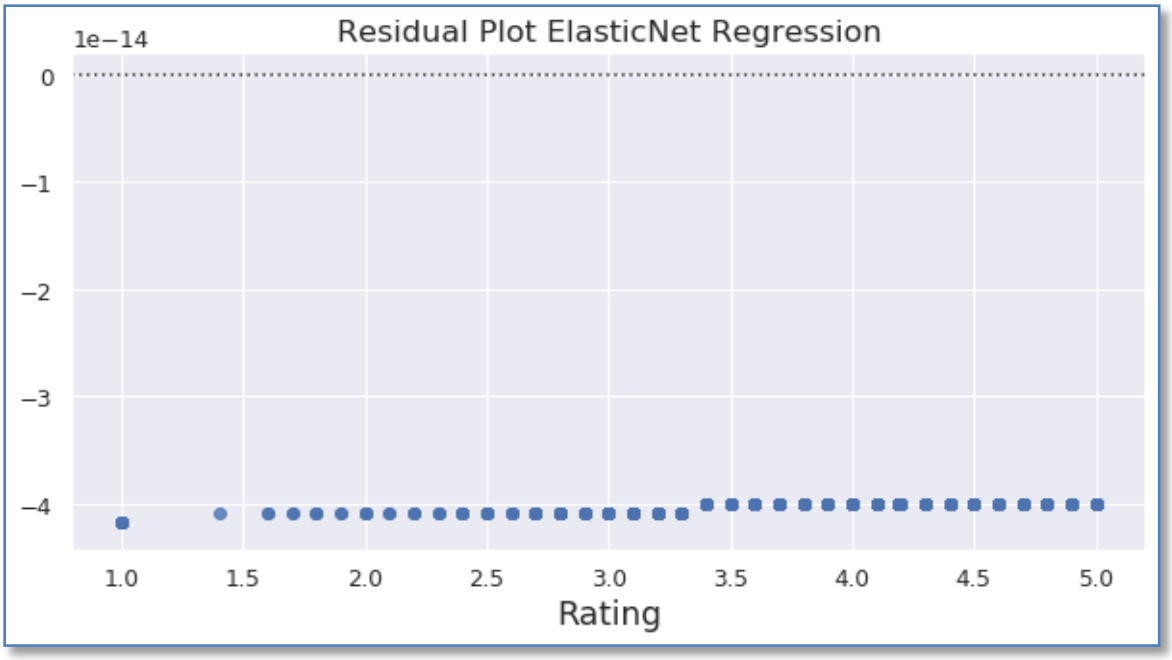


Train Set



Elastic Net Regression: In this, we used maximum iteration to be 5000 and transformed our dataset to polynomial features of degree 2. We found that it had almost similar results on the train and the test set. Again seeing the residual plot we can see that it is not a good estimate for the model.

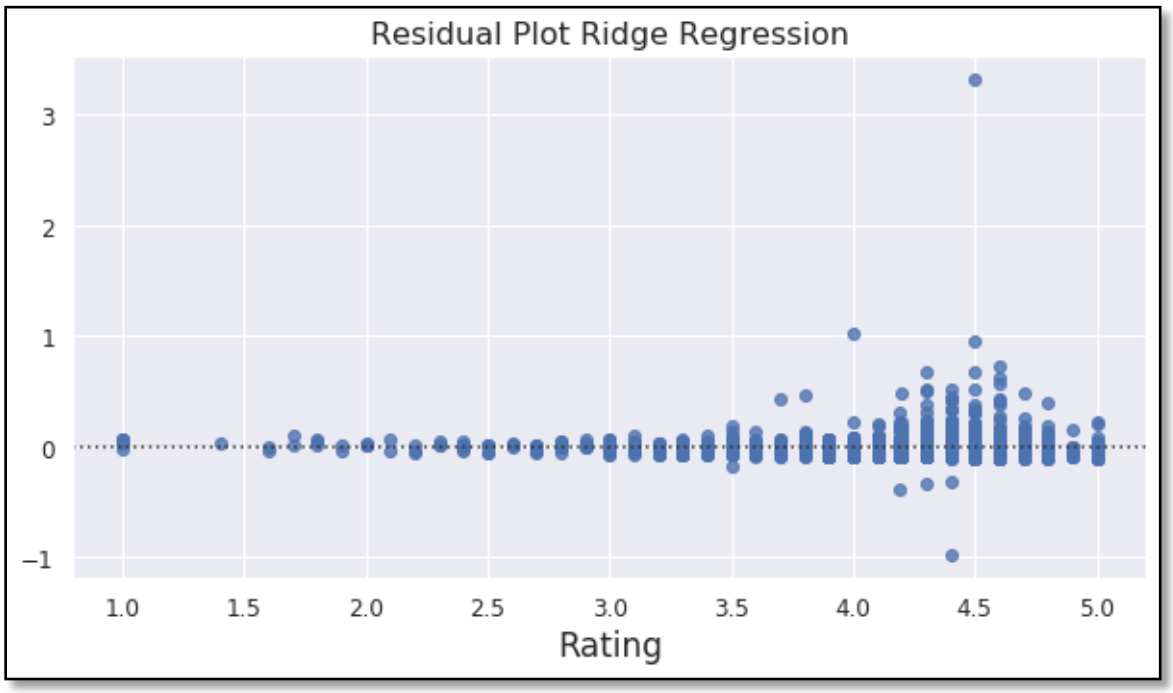
Test Set



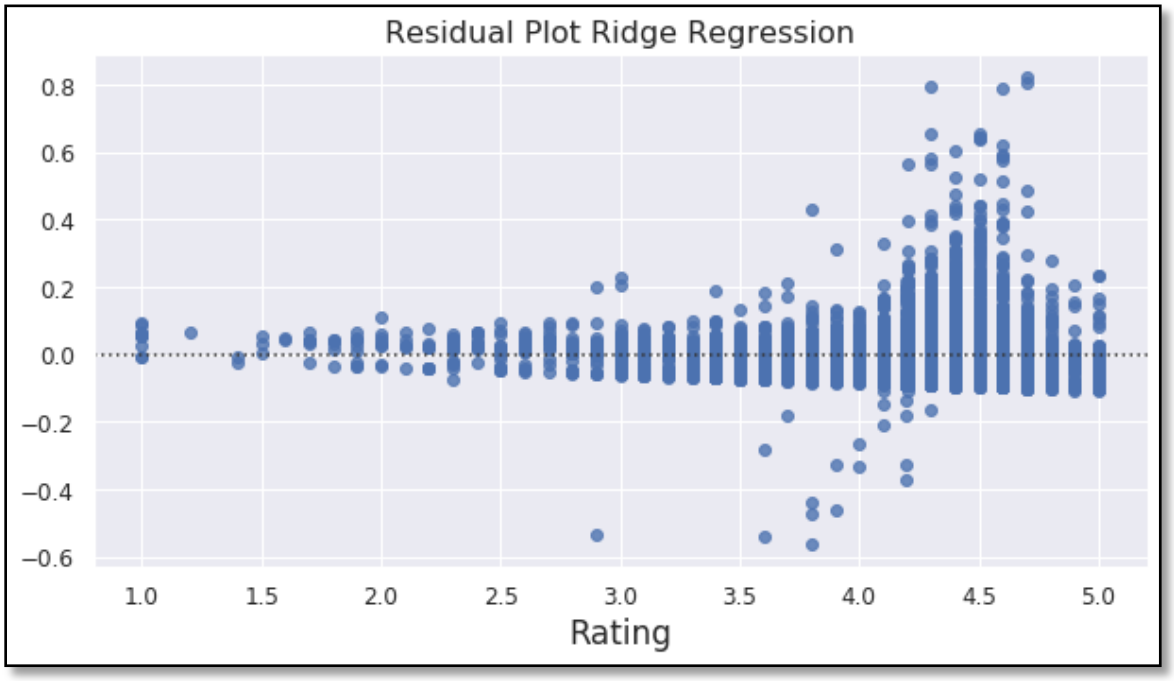
Train Set

Ridge Regression: In this, we again transformed our dataset to polynomial features and tried to find out the mean square error with maximum iteration set to 1000. Seeing the residual plots, we can conclude that it fits somewhat but random forest regressor has the best results.

Test Set



Train Set



4.3 Result

Thus, we run a few different regressor but we see that random forest regressor has the least error for the train and the test set and so it is our best model to predict the rating. Although it does seem to over-fit the train set, but it still has best accuracy for the test set.

Model	Mean Squared Error (Train)	Mean Squared Error (Test)
Random Forest Regression	0.03139	0.22994
Elastic Net Regression	0.24119	0.25395
Ridge Regression	0.23563	0.25423

5. Conclusion

From the above analysis of the dataset, we draw some important conclusions:

- The Developers should build some quality apps in the categories like Weather, Art and Design, Beauty etc. as there are less number of apps in that category so there are chances of getting more likes if the content is good. Also the famous categories like family, tools should have apps the best content if a new app is made in the category as there already exits so many choices to the user.

- The advertising companies should advertise on the top 40 most installed apps as the will have a large viewership and they can reach to maximum number of people through that. Also the apps with high reviews can be considered as they will have a high audience that's why they have high number of reviews. But these to features usually go side by side so just look into the top installed apps and you will have a way.
- Everyone building apps should consider that the Category and Genre of an app may strongly dictate if an app will be popular or not. For example: Maybe there won't be much audience for beauty apps apart from women but still if the quality of app is very good, then your app can do well. However, the Size, Type, Price, Content Rating, and Genre features should all be used to most accurately determine if an app will gain maximum installs as these features also affect the audience.