Google Play Store Analytics

Final Report

Sumadhuri Damerla,Shaoor Jan,Ashjan Khan

Contents

1 Introduction to the dataset

1.1 Data source

The data source used for this analysis is the *2018 google play store*(https://www.kaggle.com/lava18/google-play-store-apps) collected from Kaggle.

1.2 Description of the dataset

The dataset is a collection of web-scraped data of 10,000 apps from Google Play Store. Google Play Store originally referred as the Android Market, is Google's official store and portal for Android apps, games and other content for Android-powered phone, tablet or Android TV device. As of May 2017, it has over two billion monthly active users, the largest installed base of any operating system, and as of January 2020, the Google Play Store features over 2.9 million apps[13].

The variables of the dataset are as follows:

1) App (Name) – Name/Title of the application
2) Category (App)- Category/Domain to which the app belongs to
3) Rating (App)- Overall user rating of the app
4) Reviews (User)- Number of user reviews for the app
5) Size (App)- Space or memory that the app takes up
6) Installs (App)- Number of user downloads/installs
7) Type (Free/Paid)-Apps may be free or paid depending on the developer's choice
8) Price (App)-Price of the app if not free
9) Content Rating - Age group the app is based off at - Children / Mature 21+ / Adult
10) Genres (Detailed Category)- An app can belong to multiple genres, For eg, a musical family game will belong to Music, Game, Family genres.
11) Last Updated (App)- Date when the app was last updated on Play Store
12) Current Version (App)- Current version of the app available on Play Store
13) Android Version (Support) – minimum version of android it takes to have the app on the device

2 Purpose of the project

- The aim of our project is to find out if we can predict ratings of an app based on different variables and we intend to summarise the different factors that influence the success of an app.These analysis might also help the developer community to build more successful apps by taking accurate data-based decisions, and focusing on those aspects of applications that matters most.
- Also,since this is the first time we are doing data analysis using R, it is a fun way to learn and to strengthen the concepts learned during the course by taking a hands-on approach.

3 Intended audience of the project

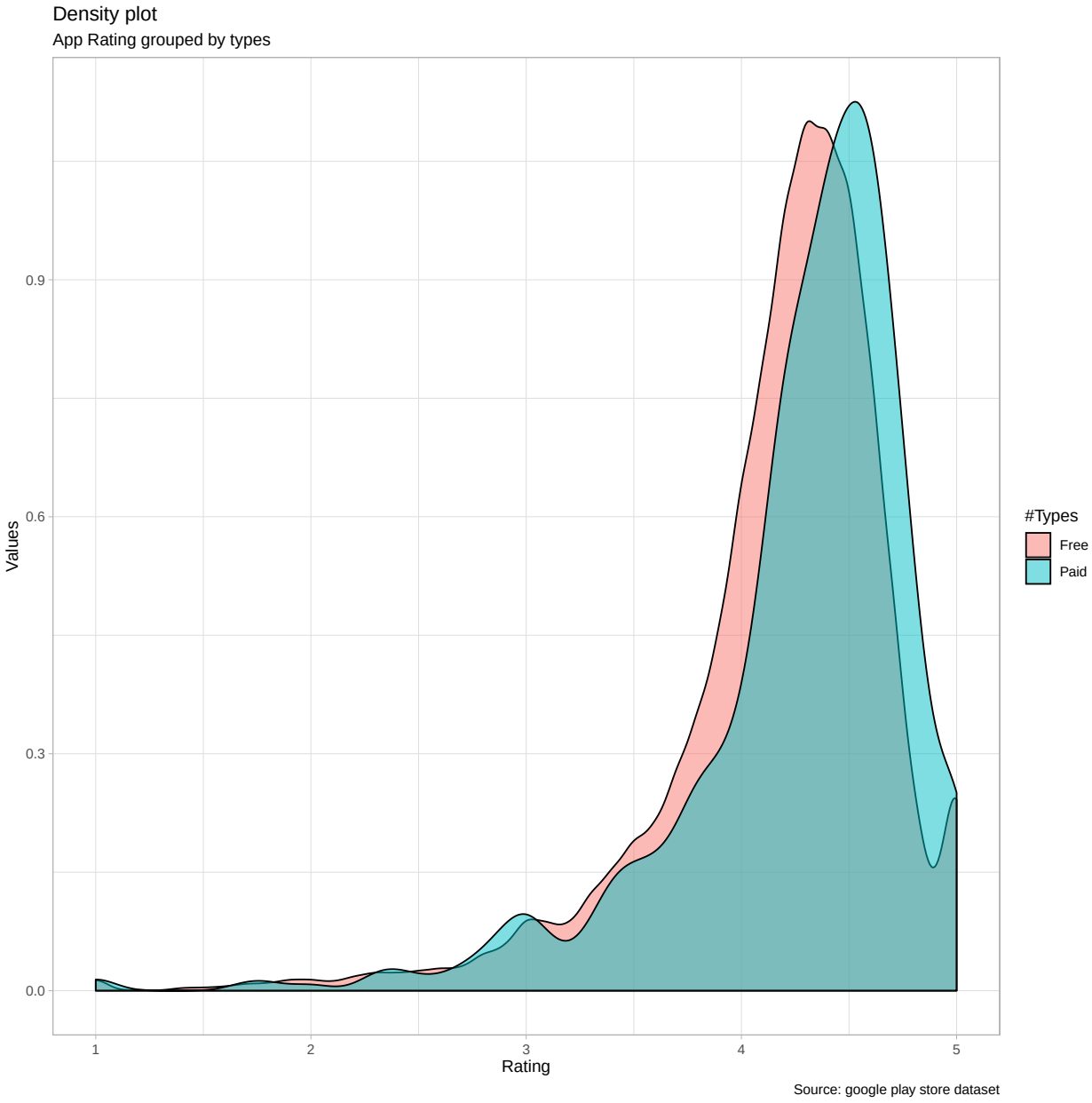
- We believe there's a diverse set of audience who might be interested in our project. As of Febraury 2020, 73.3% of the mobile operating system market share belongs to Android devices[11]. This large community consists of the general public who use android devices and appstore,the developer community and anyone who wants to understand how the app market works.
- This project is primarily intended for the growing developer community. It will help them make data backed decisions before launching their application. Besides developers, it is also helpful for tech journalists, Google Play Store users or any other interested party.

4 Exploratory Data Analysis to find factors for success of an app

Generally, the most successful apps have high ratings and high installs.To look at which app makes it to the top, we consider ratings and installs, so we explore these to find any relationship or trends

4.1 Rating column in depth

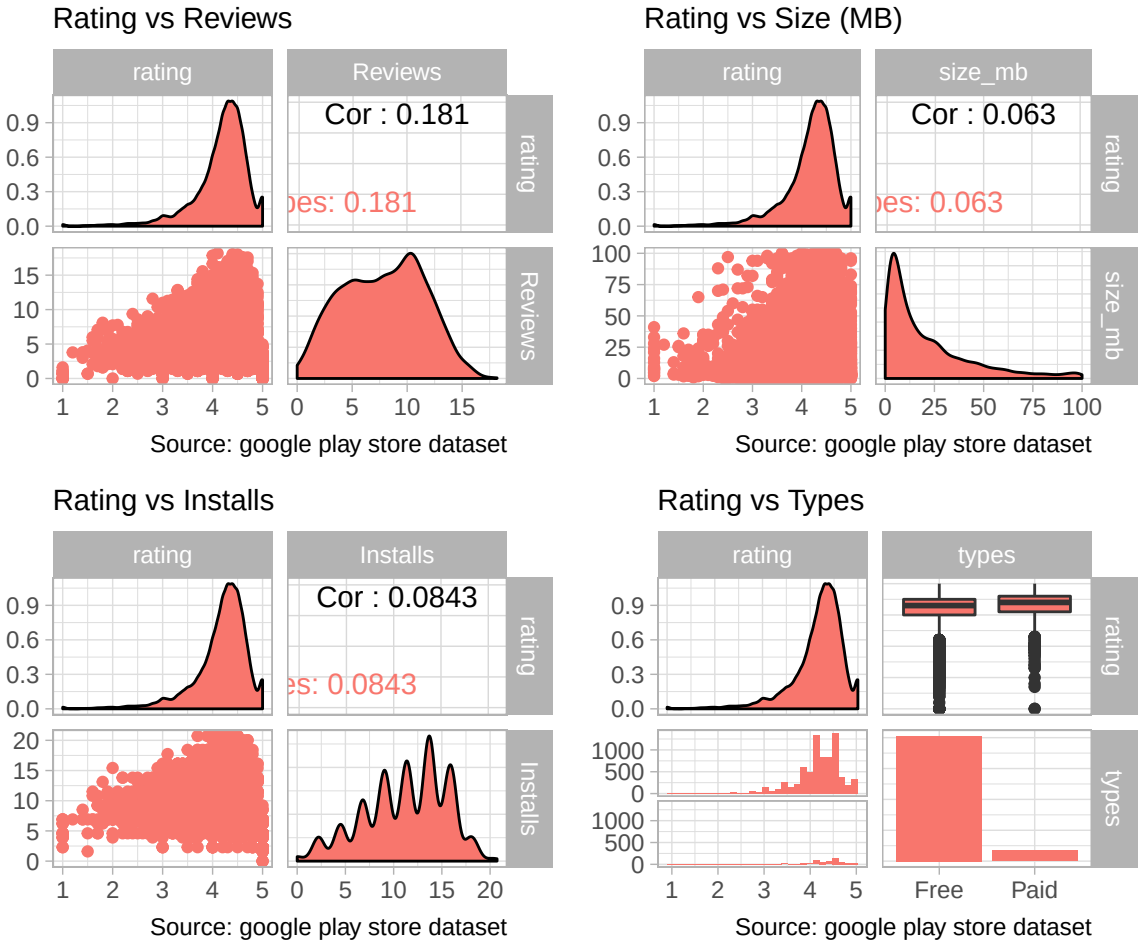
4.1.1 Distribution of rating



Finding: We observed that there are minimal number of apps with low ratings whereas most of the apps have ratings between 3-5.

4.2 Correlogram plots

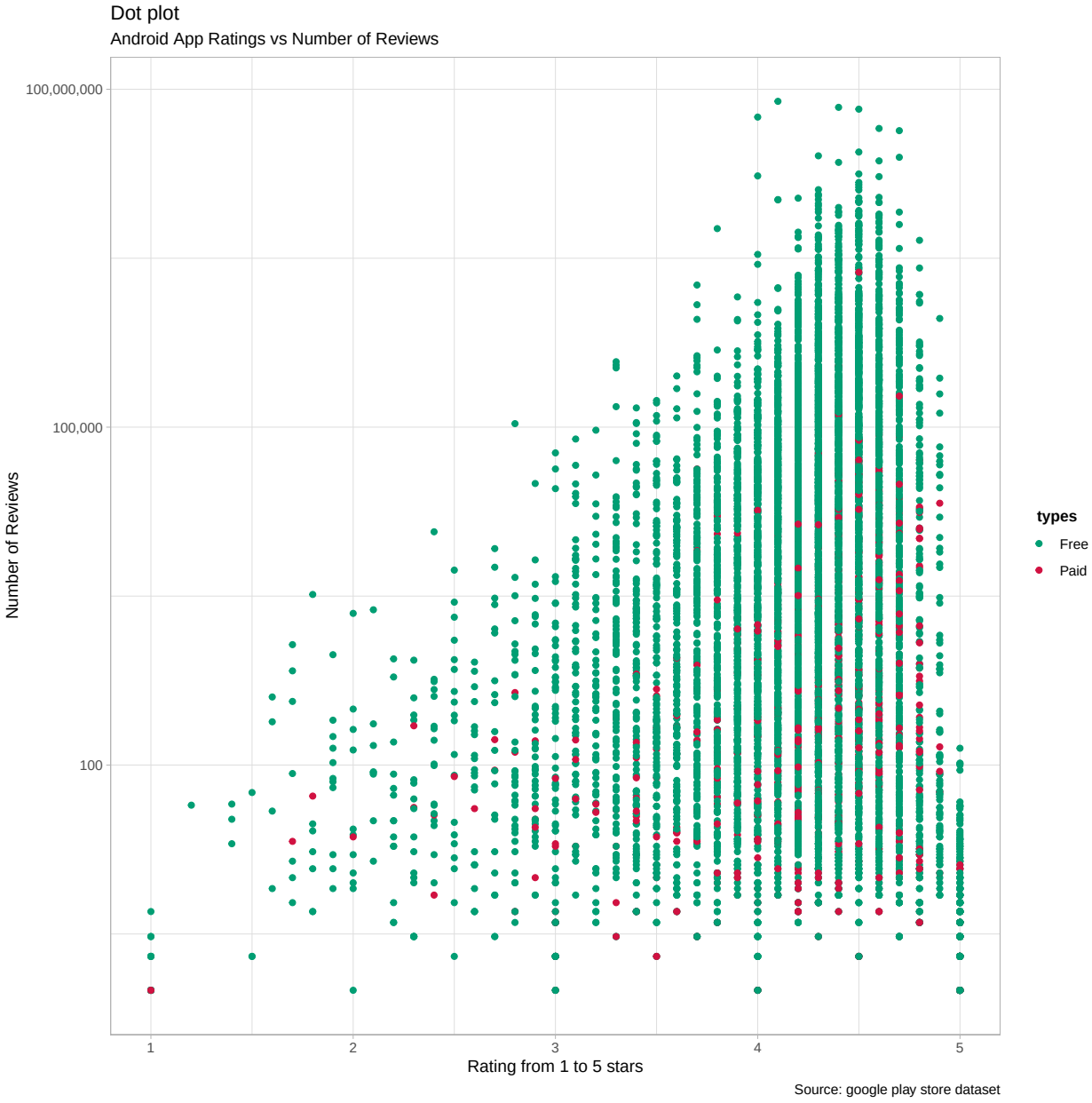
Firstly, we plot correlograms of rating column versus different columns to find relationships between the variables. Correlograms are useful to understand the relationship between different numerical variables. If the correlogram index is 1, it means that the variables are directly proportional to each other.



Finding:

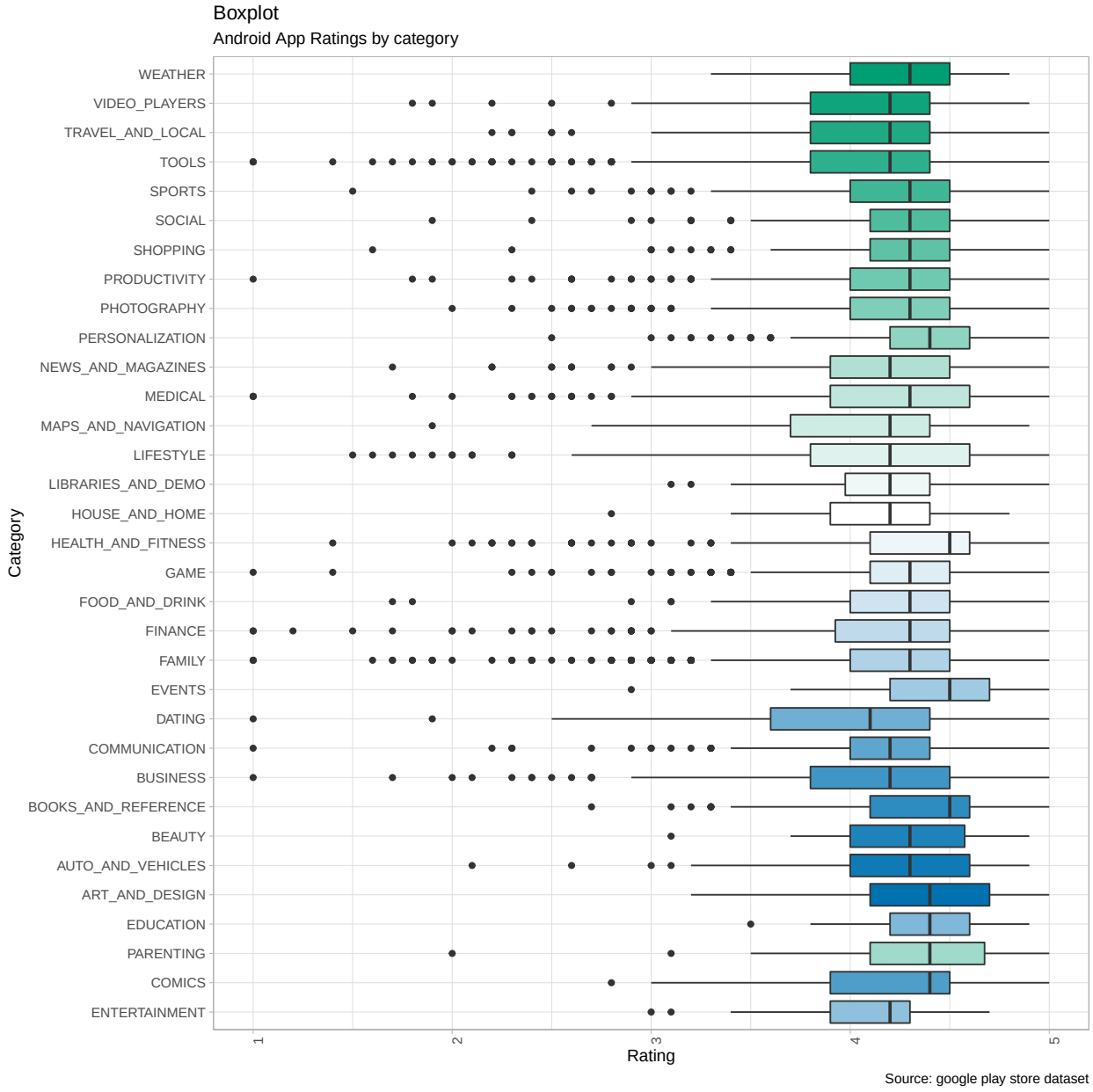
* Each correlation index talks about the relationship between plotted columns. If the index is 1, it means they have linear relationship
* Each plot on the diagonal refers to the density plot of the respective column
* We can observe that there is no significant linear relationship between rating and the plotted numerical variables.

4.2.1 Plot of reviews vs app ratings



Finding: We can observe that the number of reviews influence the ratings. Generally, as the number of reviews increase, the rating is higher.

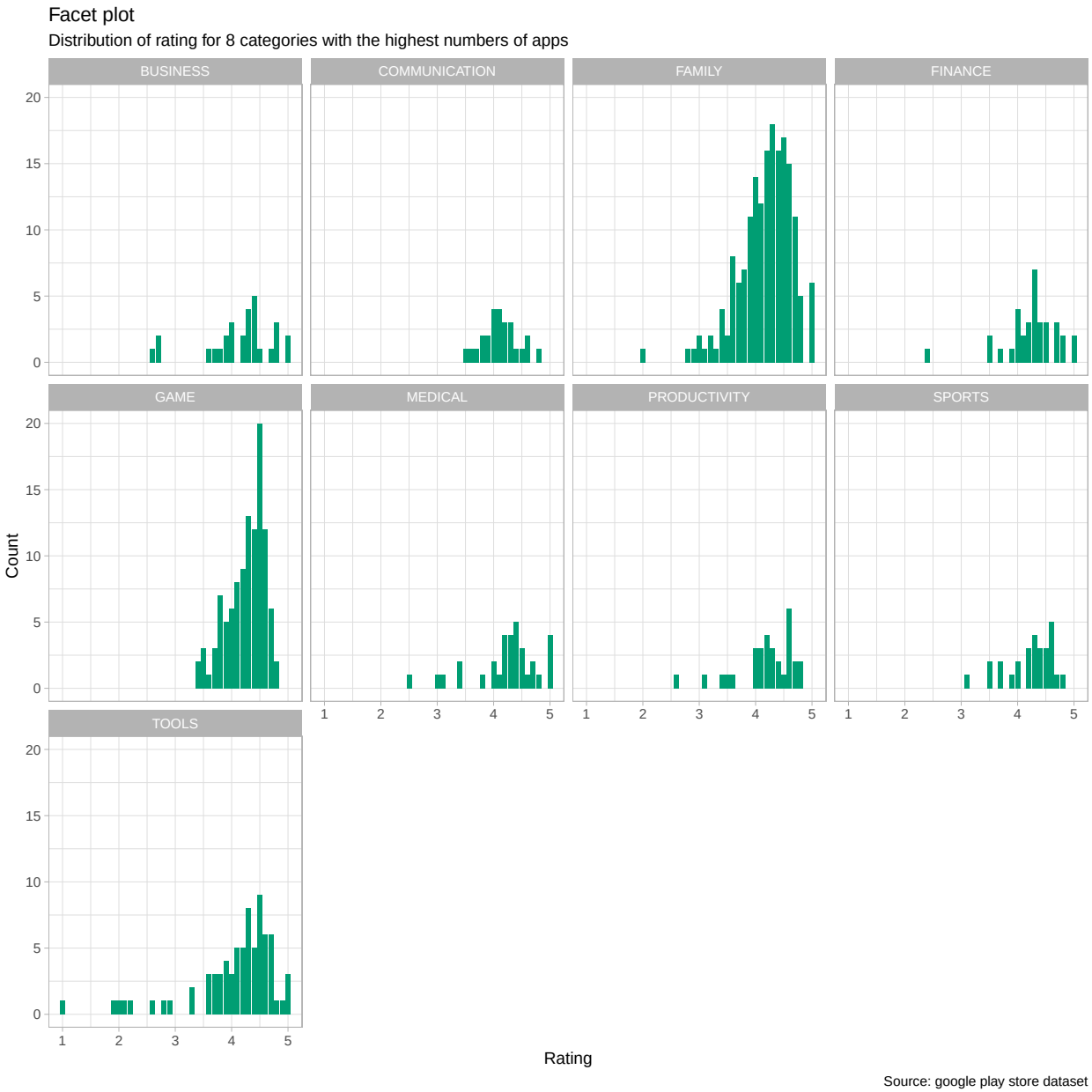
4.2.2 App rating vs category



Finding: We observed that some categories like TOOLS, FAMILY, FINANCE and LIFESTYLE and a great majority of applications fall below first quartile. Thus, even though the median rating is high, the deviation from median is significant.

4.2.3 Distribution of rating for 8 categories with the largest numbers of apps

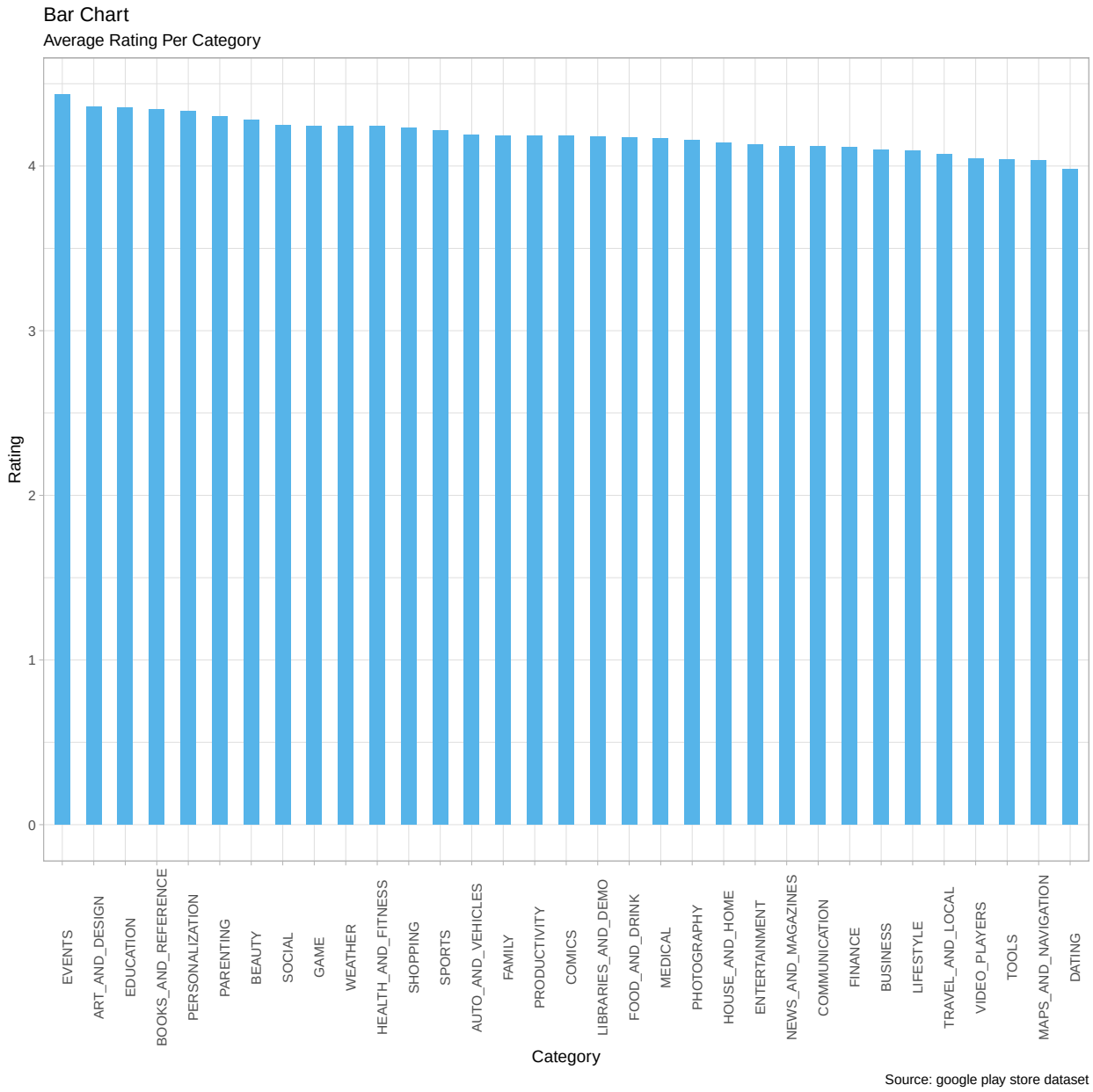
We look at the distribution of rating across different categories. We chose 8 categories with the largest number of applications.



Finding: The distribution of rating varies significantly across each categories.

4.2.4 Average rating per category

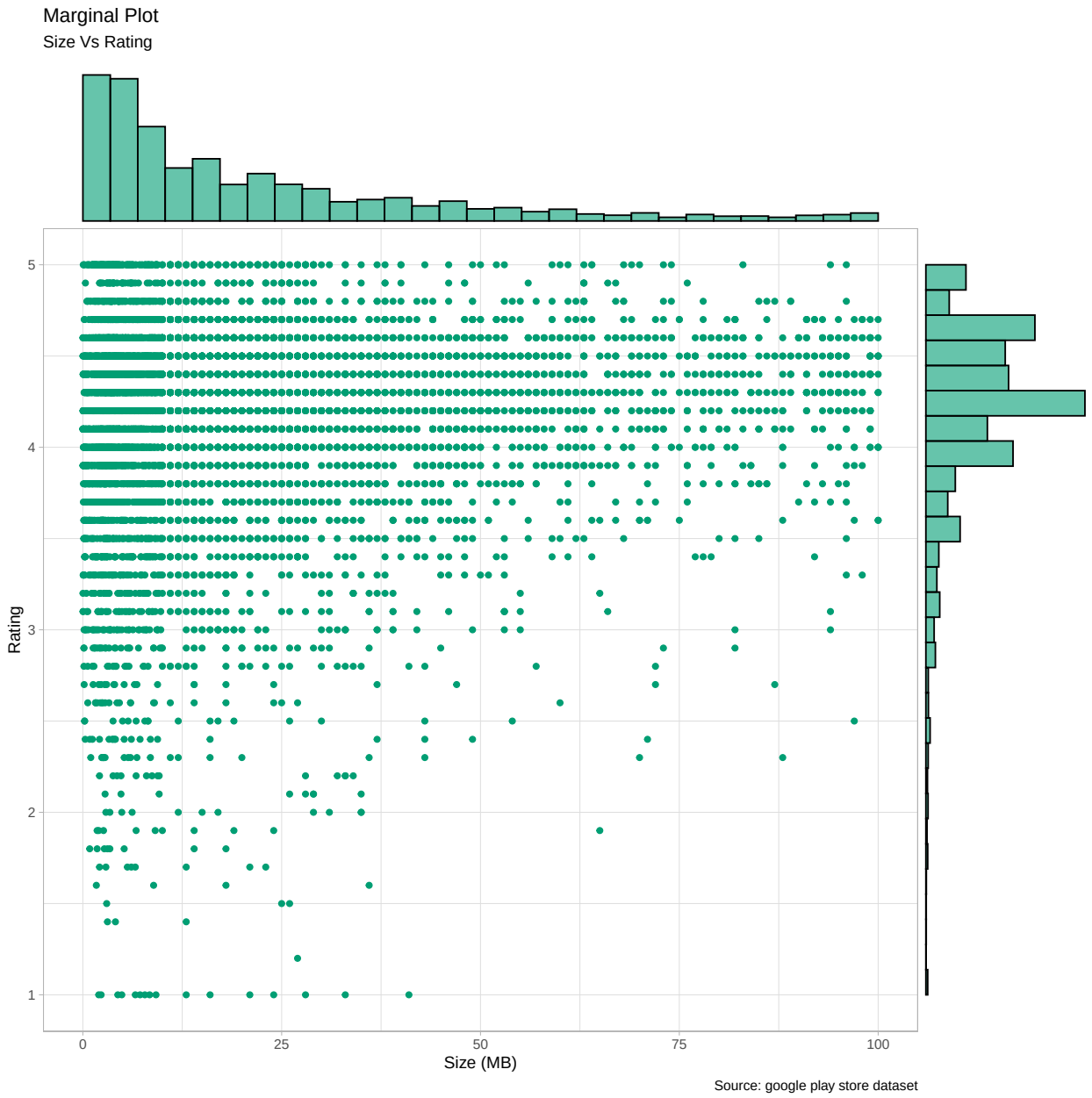
In the previous graph, we observed that the distribution of rating varies significantly with different categories. This plot is to find the average rating per category.



Finding: We observed that the average rating per category is not very different for each category. The "EVENTS" category still has the highest average rating, and "DATING" category has the least average rating.

4.2.5 Size and rating

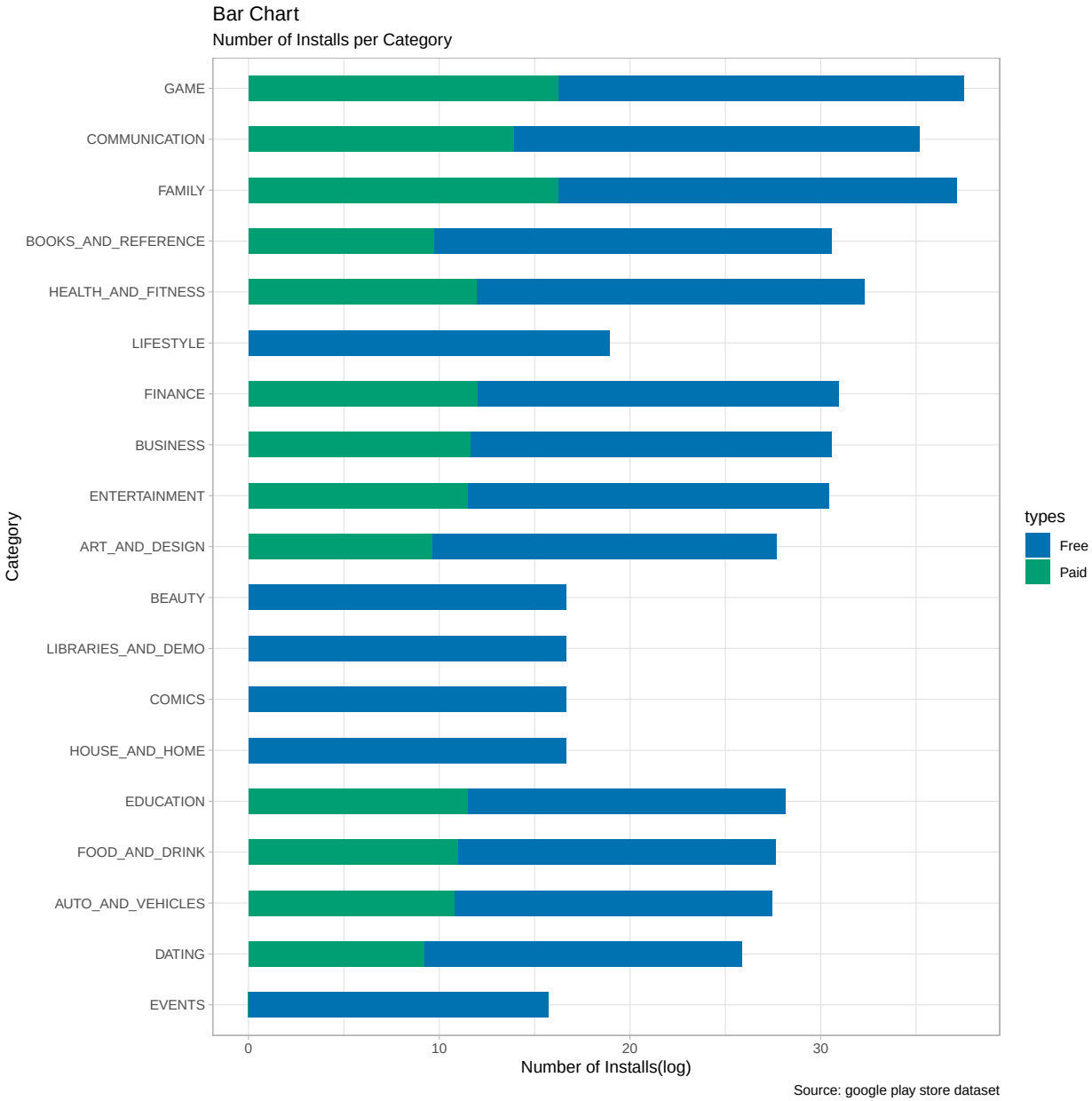
The below plot is to verify if application sizes influence the rating. A marginal plot is a scatterplot that has histograms, boxplots, or dotplots in the margins of the x- and y-axes.



Finding: We observed that majority of applications with sizes under 25 MB have a good rating (above 4).

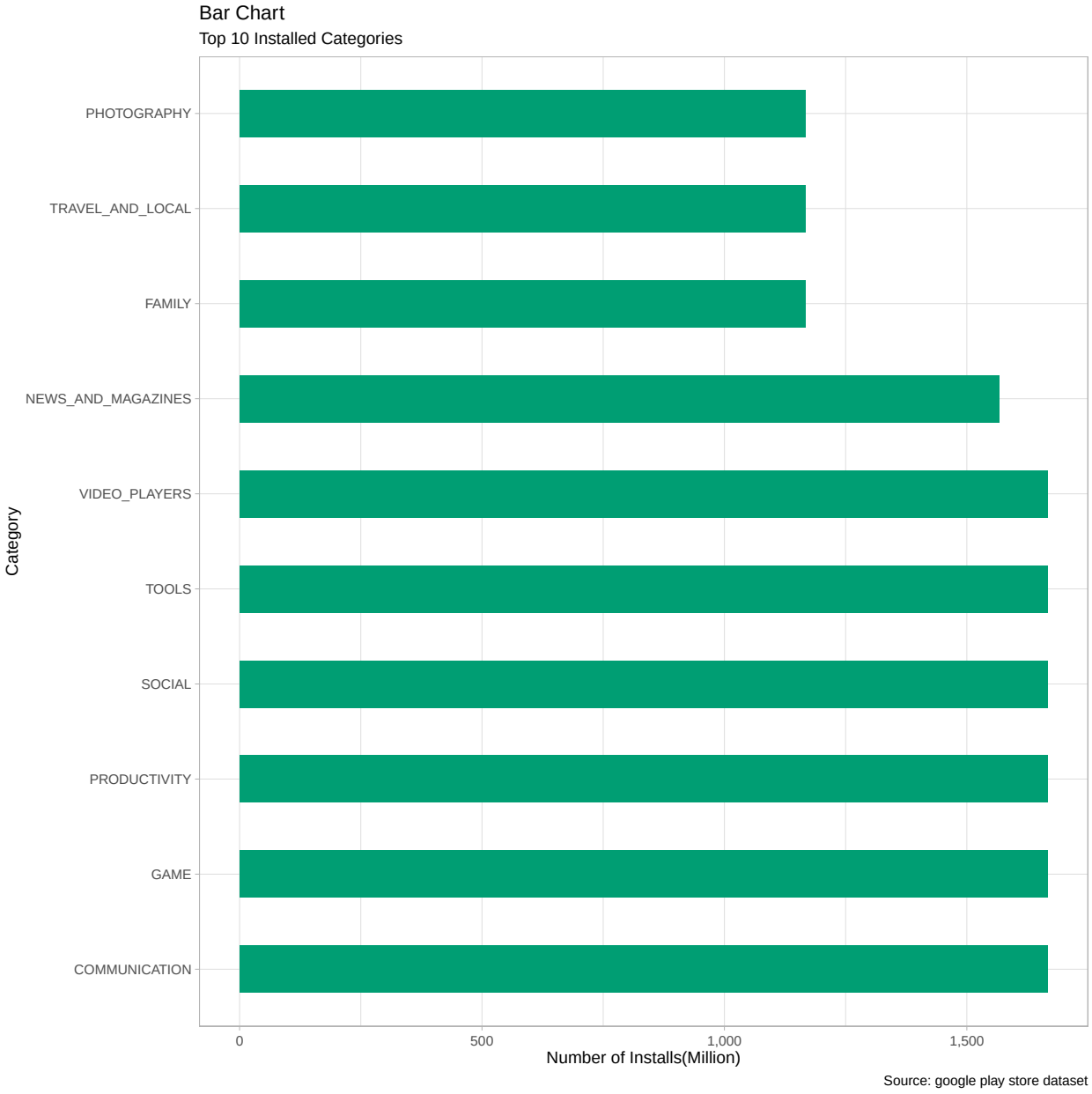
4.3 Exploring installs column

4.3.1 Number of installs per category



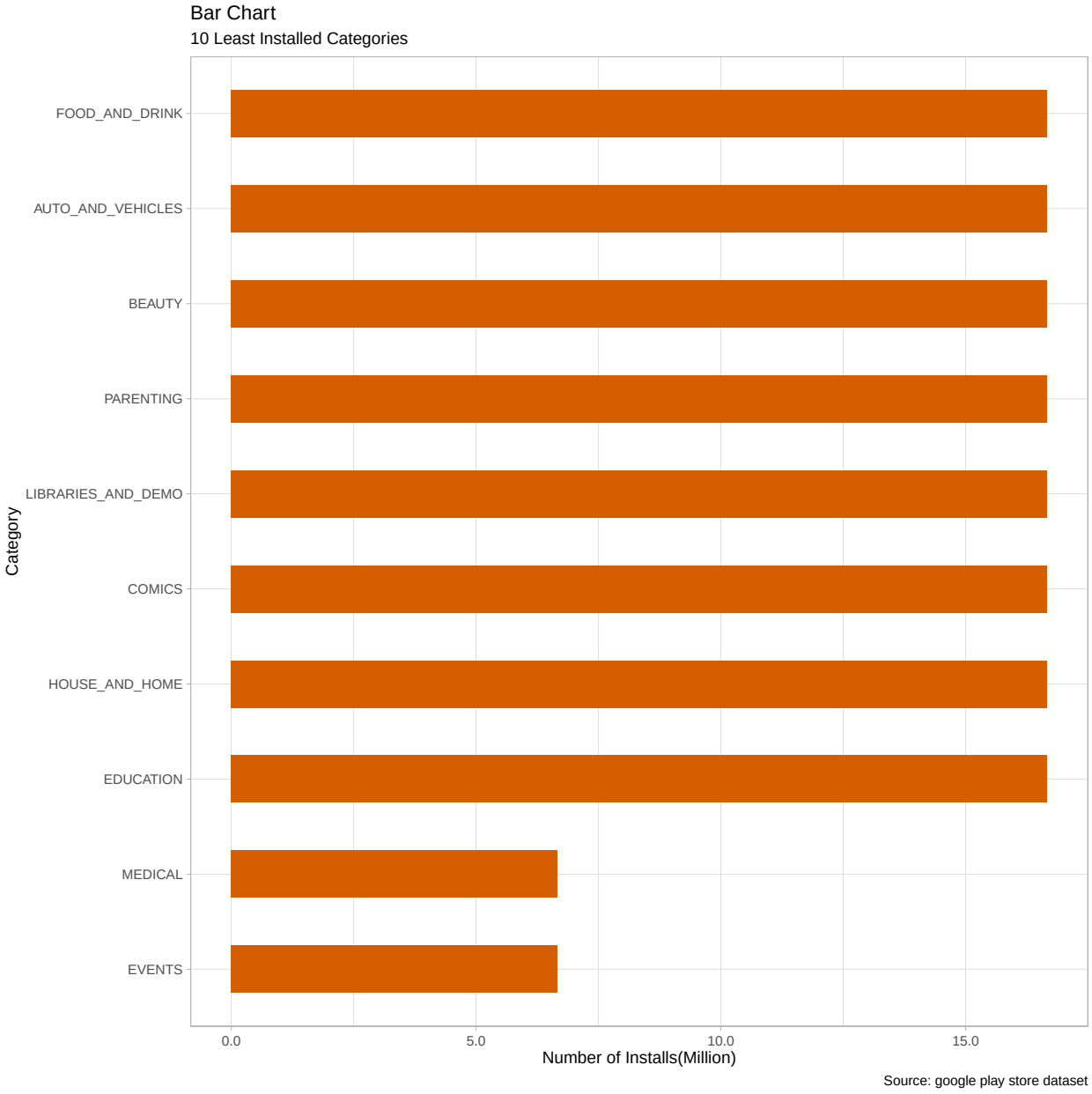
Finding: The graph shows the log of number of installs (the values of installs varied from 0 to 1 billion) vs CATEGORY. FAMILY and GAME have the highest number of installs. EVENTS, HOUSE_AND_HOME, COMICS, LIBRARIES_AND_DESIGN and BEAUTY have the least number of installs.

4.3.2 Top 10 installed categories



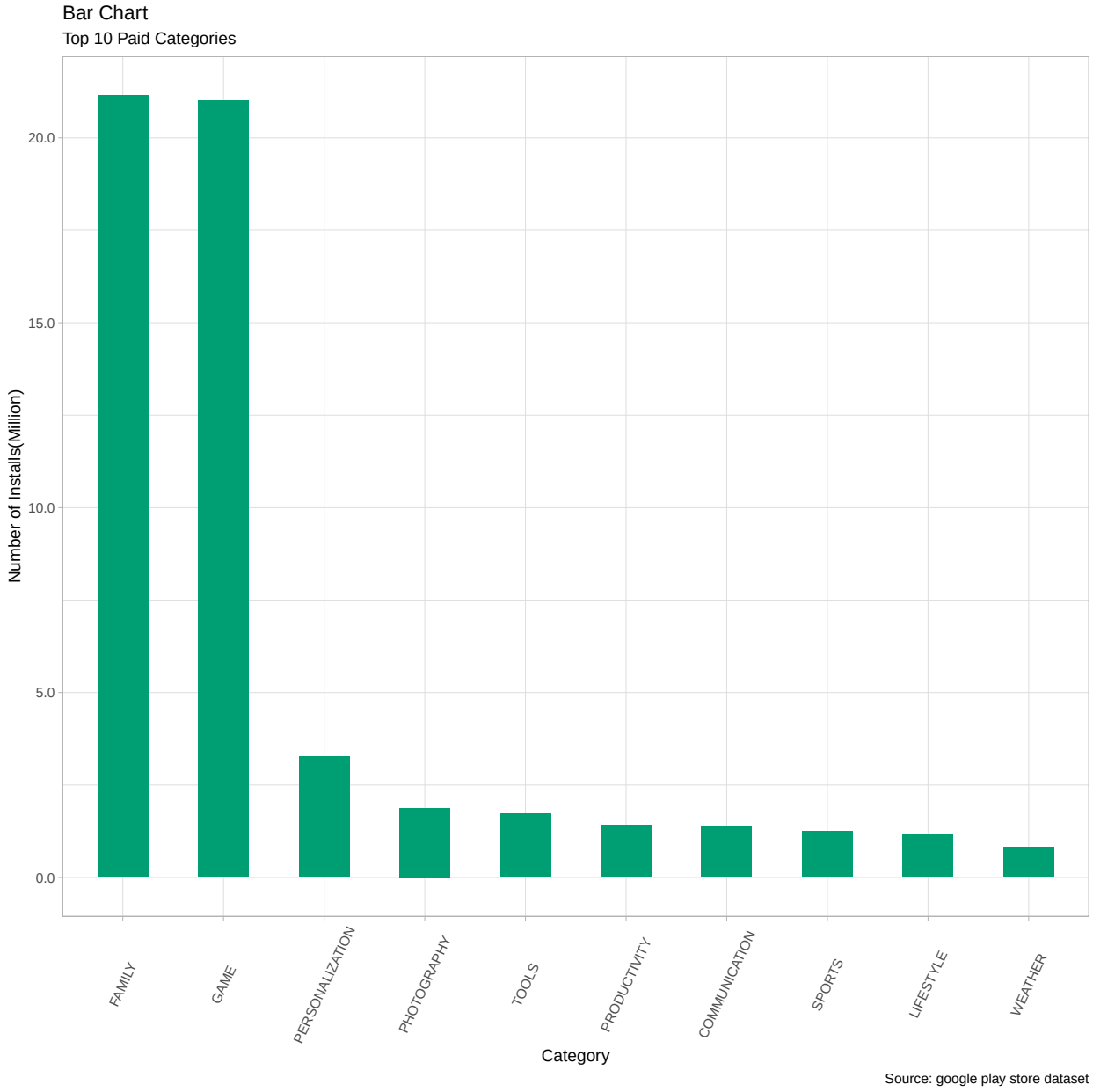
Finding: The apps under Communicationcategory have more than 1500M number of installs.

4.3.3 10 least installed categories



Finding: We observed that applications under "Events" category have the least number of installed applications followed by medical and education categories.

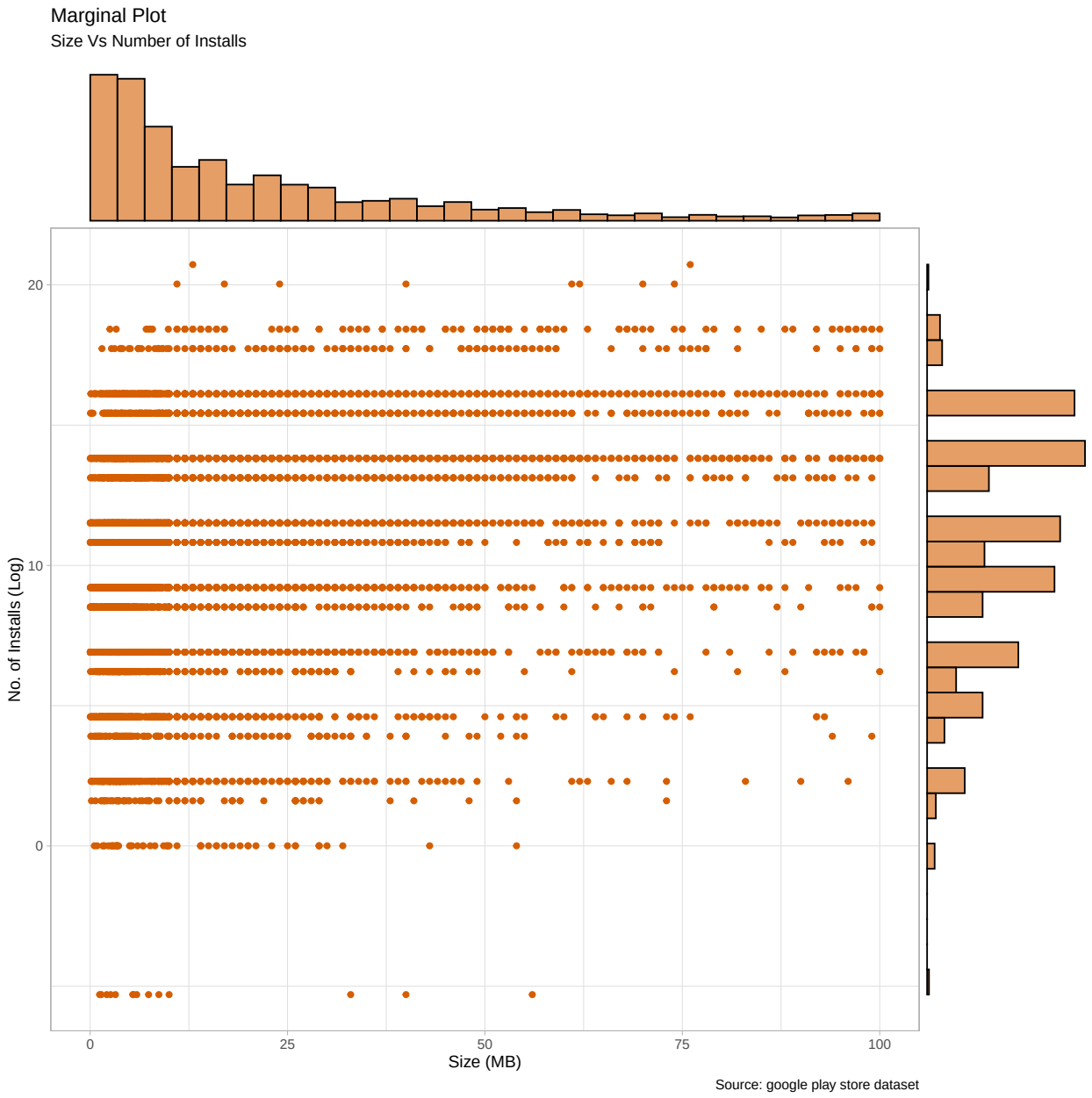
4.3.4 Top 10 paid Categories



Finding: Applications with family and game category with 20M installs have the highest number of installs.

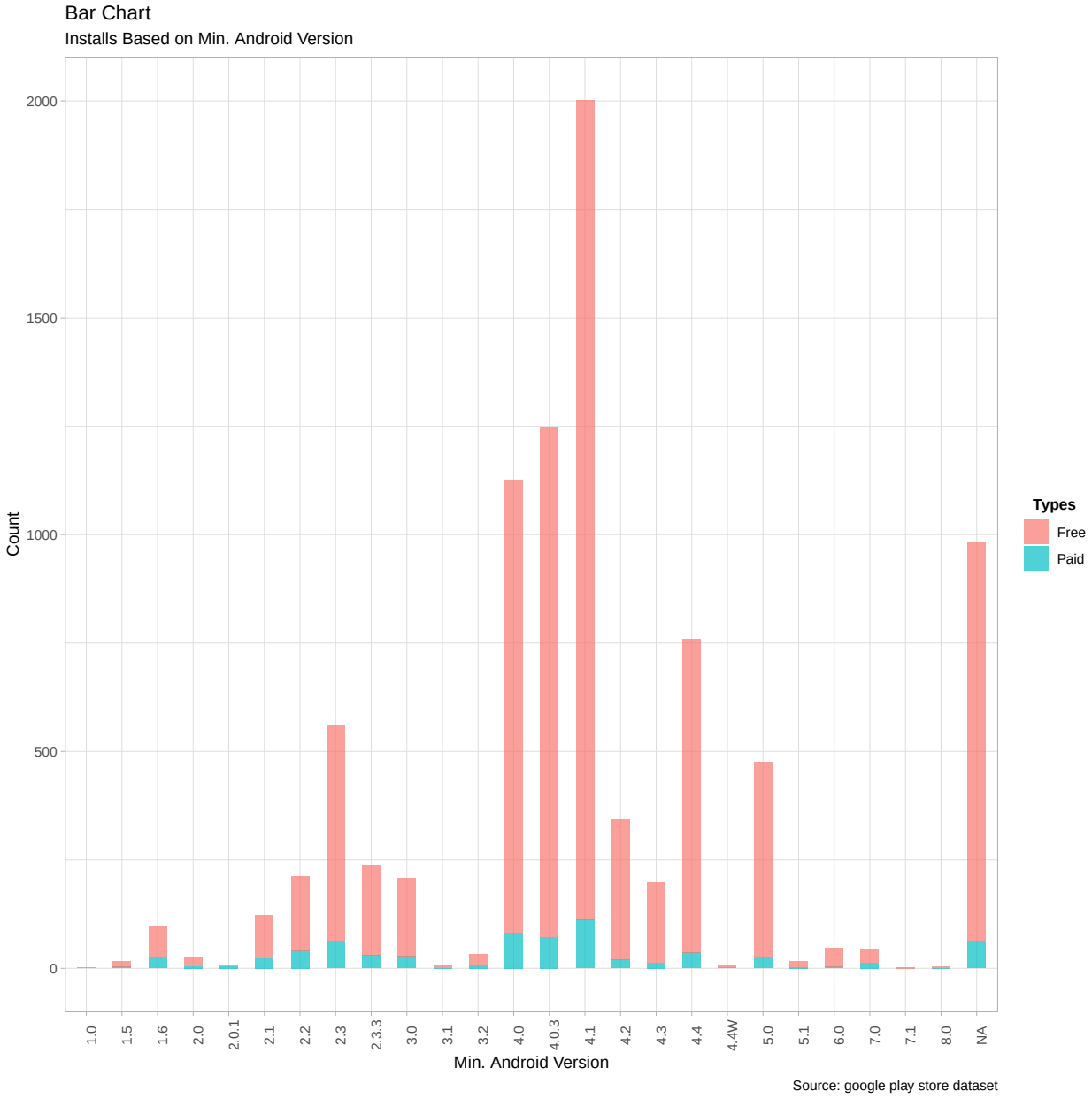
4.3.5 Size Vs. number of installs

Size is an important characteristic of an application. Large applications might reduce the number of installs, as it reduces to targeted audience. We can test this by plotting size against installs.



Finding: Optimally sized applications with sizes between 5MB and 30MB have the greatest number of installs.

4.3.6 Number of installs based on support by minimum android version



Finding: 4.1 android version has the maximum number of installs. This implies that most google play store users use a minimum of 4.1 android version.

5 Summary of EDA

- To get most success from an app, it has to have maximum number of reviews and maximum number of ratings. These are the other trends we found:
  1) Apps should atleast support 4.1 android version or more to succeed.This is expected because the majority of users have smart phones with constant android updates.

  2) They have high chances of success if the genre is family, game, communication or productivity apps whereas food_and_drink, auto_vehicles categories have very low probability to succeed.

  3) The apps which are be free have huge success rates however there are a few exceptions to this.

  4) Apps should be optimally sized between 5MB and 30MB.

6 Future work (models)

6.1 What are we trying to achieve (thesis/hypothesis)

The dataset contains information about ten thousand apps dated till the year 2018. We plan on using Linear Regression Model, Recursive Partitioning Model and Random Forest Model to perform predictive analysis. We want to answer the following hypothesis:

1) Find similarities in apps that make it to the top of Play Store. Factors contributing to the success of applications.
2) Can we predict rating of apps based on other parameters such as number of reviews or the size of an app?

To answer these questions, we will be exploring all the variables of this dataset to find if there's any relationship between rating and other variables. We intend to find out which of these variables will play the most important role in predicting rating.

6.2 Why is this important/interesting?

Before we started this project, we took part in a competition called game-jam [12] where we built a game,we had an idea of launching it on Google Play Store.We then thought it would be interesting to see currents trends in the market and to do a detailed analysis to get more insights to the following questions:
1.Factors that influence the success of an app,
2.Which categories are highly installed
3.What are the most famous applications and do they have any trends in common like number of installs, number of reviews, size or android-version? and etc

This analysis inturn will aid the developer community to build successful apps targeting a specific audience.

6.3 How are we going to test the hypothesis?

Since the dataset is dated till 2018. We are thinking to test it by comparing the predictions of our model with the 2019 dataset if possible or we will test our model with dummy application

6.4 Any challenges that we might encounter

We cannot ensure complete accuracy of the model because there is a lot of useful information missing that could have given us more insight into the Google Play Store market e.g. Demographic data could have offered insights into the rating and number of installs of apps, with respect to different regions, different cultures and different trends popular to specific age groups. Also, it would have been interesting to see how different global trends affect the usage of the app, for instance, the current pandemic "covid-19" has called for quarantine

across the globe and many people, markets and other companies are relying on smart phones and virtual connections, this will heavily increase the use of many applications, thus deviating from the general trend.

7 Models

- The goal is to predict the rating of an application based on different parameters. Our EDA confirms that variables like number of installs, number of reviews, category of the app and the type of app (paid/free) can affect the rating of an app.
- In this section,firstly, we tried to fit our dataset for both explanatory and predictive models. We used the cross-validation technique to select the best fitting model for our dataset and lastly, we test the models by predicting ratings for a dummy app.

7.1 Correlation using Pearson method

- The bivariate Pearson correlation indicates whether a statistically significant linear relationship exists between two continuous variables or it indicates the strength of a linear relationship (i.e., how close the relationship is to being a perfectly straight line)
- Before creating a model, we calculated the correlation between different variables to help us select appropriate exploratory variables.

```
## Rating and Installs
##          0.04024461
```

```
## Rating and Installs(transformed)
##                         0.1138791
```

- **Correlation coefficient between rating and installs**: The correlation coefficient is 0.0402. This indicates a slight positive relation between these two variables. Using square of log of installs, increased the correlation coefficient to 0.1138.

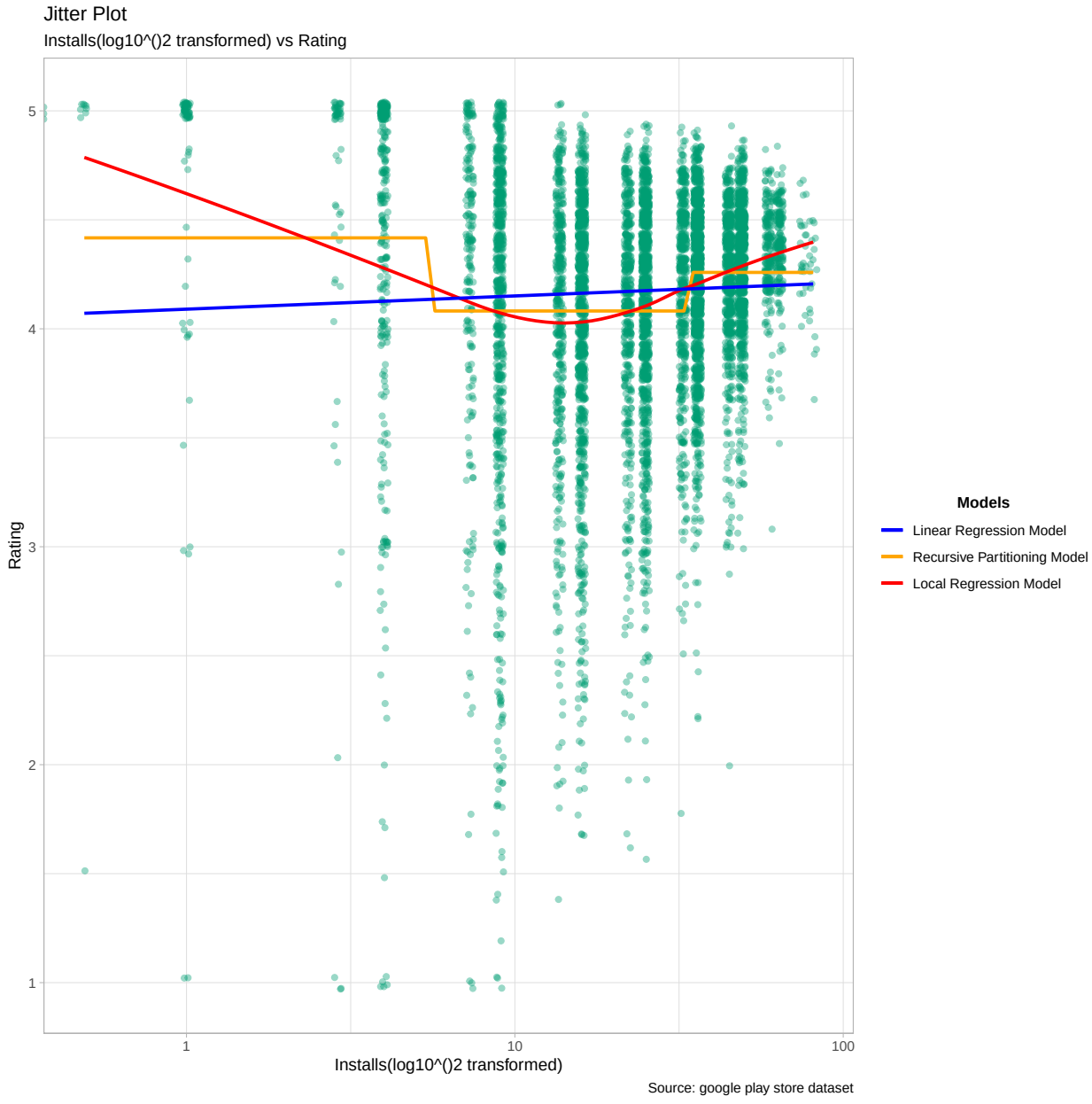
```
## Rating and Reviews
##         0.05515293
```

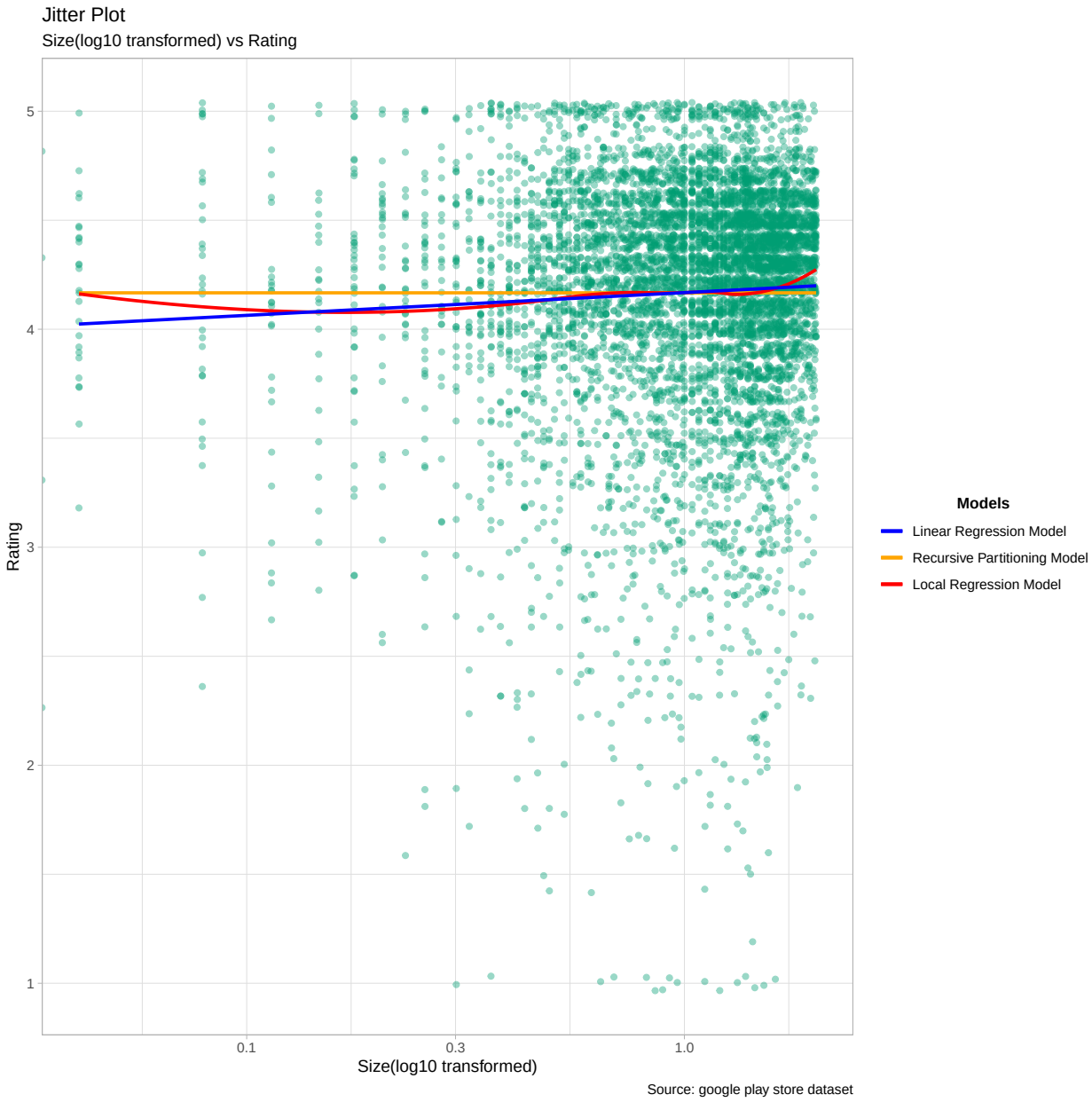
```
## Rating and Reviews(transformed)
##                        0.2030894
```

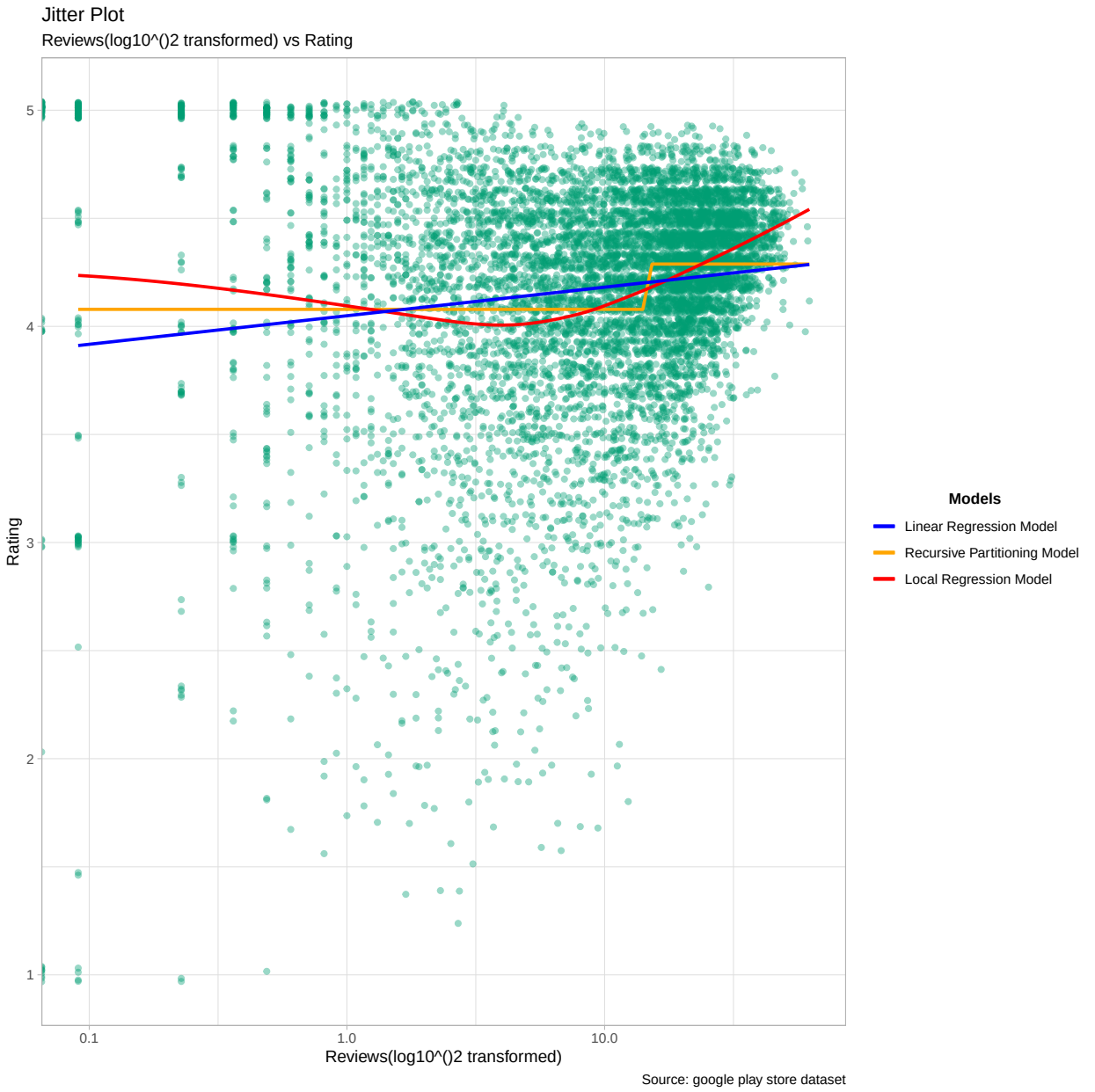
- **Correlation coefficient between rating and reviews**: The correlation coefficient is 0.05515, indicating a slight positive relationship between the two variables. Using square of log of installs, increased the correlation coefficient to 0.20308.

7.2 Explanatory/Descriptive modeling

- The following models are used:
- Linear Regression Model
- Local Regression Model
- Recursive Partitioning Model



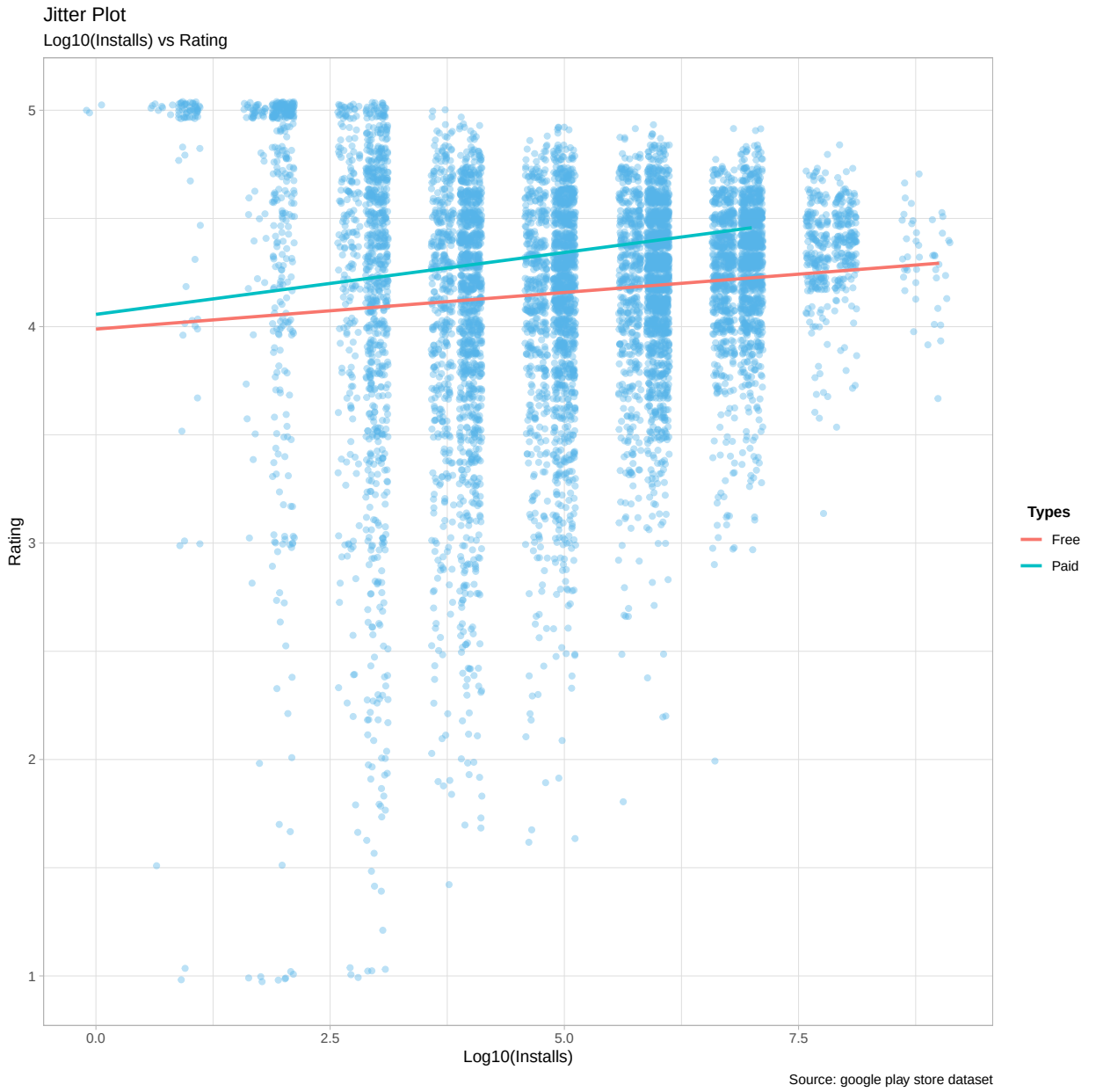

Jitter Plot
Size(log10 transformed) vs Rating
Rating
5
4
3
2
1
0.1
0.3
1.0
Size(log10 transformed)
Source: google play store dataset
Models
Linear Regression Model
Recursive Partitioning Model
Local Regression Model



The graphs above show the fitted Linear Regression model, Local Regression model and Recursive Partitioning model for predicting rating using number of installs, size of the app and number of reviews variables.

7.3 Explanatory/Descriptive modeling (cont.)

Continuing with our explanatory modeling, we will fit a model for rating vs installs given taking into consideration whether the app is Free or Paid. We have also calculated sum of square residuals, R-Square value and Root Mean Square Error for the model.



Observations/Findings: We observe that the rating of paid apps is higher than the free apps indicating better quality of paid apps. On the other hand, free apps have a higher number of installs, thus covering a wide range of audience.

7.4 Cross Validation of models

This technique helps us identify the best fit model for our dataset i.e. model with the least root mean square error

- We sliced the dataset into training and test datasets. Training dataset contain 6,951 observations (75% of total) and test data set contains 2,198 observations (25% of total).
  - We trained both Linear Regression model (lm) and Recursive Partitioning model (rpart) with the training dataset.
  - After training our model, we predicted the rating for test dataset. In order to find which model fits better, we calculated the root mean square error (RMSE) for both models. RMSE values for Linear

Table 1: Comparision of models

model	RMSE
Linear regression model	0.5277467
Recursive partitioning model	0.5183104

Regression model and Recursive Partitioning model are 0.5277467 and 0.5183104 respectively. We can see that RMSE value for Recursive Partitioning model is less than RMSE value for Linear Regression model implying that Recursive Partitioning model can make a better prediction than Linear Regression model.

– Along with performing a comparison between Linear Regression model and Recursive Partitioning model, we experimented with different explanatory variables for each model. We got the least RMSE by using installs, types, categories and reviews as explanatory variables.

7.5 Making Predictions

We can predict ratings for an application with the help of parameters like installs, types, category and reviews. We choose applications from Google Play Store and used their number of installs, reviews, type and category to calculate their rating. We will compare this rating with the existing rating of the application. We will perform this experiment for three applications, to see how close the calculated values are to the original values.

```
##        1
## 4.285381
```

```
##        1
## 4.131302
```

- We chose the "Nike Training Club - Workouts & Fitness Guidance" application for our first experiment. It has 10 million downloads. It belongs to HEALTH_AND_FITNESS category and is reviewed by 272000 users. The actual value of it's rating is 4.2. We used this information to calculate the rating of this app. The Recursive Partitioning Model predicted the rating to be 4.28 which is close to the actual value.
- The second application we chose is "Wedding Planner & Organizer, Guest Checklists". It has 10000 installs with 220 reviews. It is Free app and, belongs to EVENTS category. Its actual rating is 4.5. Our model calculated its rating to be 2.13. This is a considerable difference. We observed that for values that are on either extreme i.e. too low or too high, the model does not perform well. In this case, the number of reviews is very less.

8 Conclusion

- We have found that recursive partitioning model predicts ratings better than linear regression model with RMSE 0.51 and 0.52 respectively.
- We found that our model's accuracy is better if the rating is 4 or more. This is because the majority of the dataset has ratings more than 4 which is evident from the density plot of rating which was highly left-skewed.
- The recursive partitioning model does not perform well with extreme values.

9 References

[1] Dataset;
[2] R markdown ;
[3] Stackoverflow ;
[4] GGally;
[5] Custom color pallete;
[6] [App details] (https://play.google.com/store/apps); [7] Colors;
[8] TidyVerse;
[9] ggplot2;
[10] Google playstore Kernel by Danilodiogo;
[11] market share ;
[12] Global game jam ;
[13] Play Store Statistics ;