



GOOGLE PLAY STORE APPS RATE PREDICTION

Team 3
Yuchen Xing
Jiayi Luo
Yiming Jia



Motivations

As smart phone becomes more popular, people have started to focus on user experience of different features besides basic services like making phone calls and sending text messages. The most direct way to review these experiences is through the App Store. The rating system reflects what people think of an app. In this project, our goal is to use a good model to forecast the rating of a certain app, after reviewing it's basic information.

Methodology

First, we perform data cleaning on our dataset to remove some invalid entries and only preserve the features which from our guess is relevant to our prediction.

Second, we visualize our data based on different features and also visualize the possible correlation between features.

Finally, a model is presented to predict the rating of an App with given information such as 'Installs' and 'Genres'

Dataset

We are use the dataset scraped from Google Play Store in Android devices from Kaggle. With this advantage, this dataset is more representative of the general user demographics. This dataset possesses a total of 8190 rows (data points) and 13 columns (features).

App: Application names

Category: Category the apps belong to

Rating: Overall user rating of the app

Reviews: Number of user reviews for the app

Size: Size of the app

Installs: Number of user downloads/installs

Type: Paid or Free

Price: Price of the app

Content Rating: Age group the app is targeted at - Children / Mature 21+ / Adult

Genres: An app can belong to multiple genres

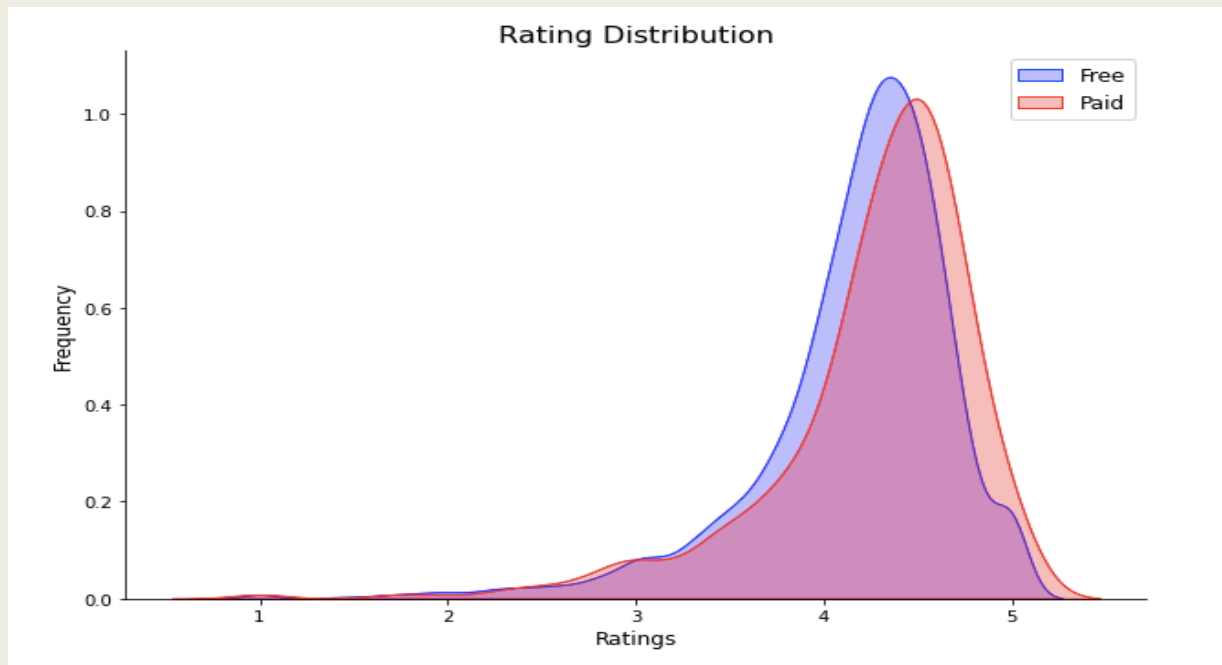
Last Updated: Date when the app was last updated on Play Store

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

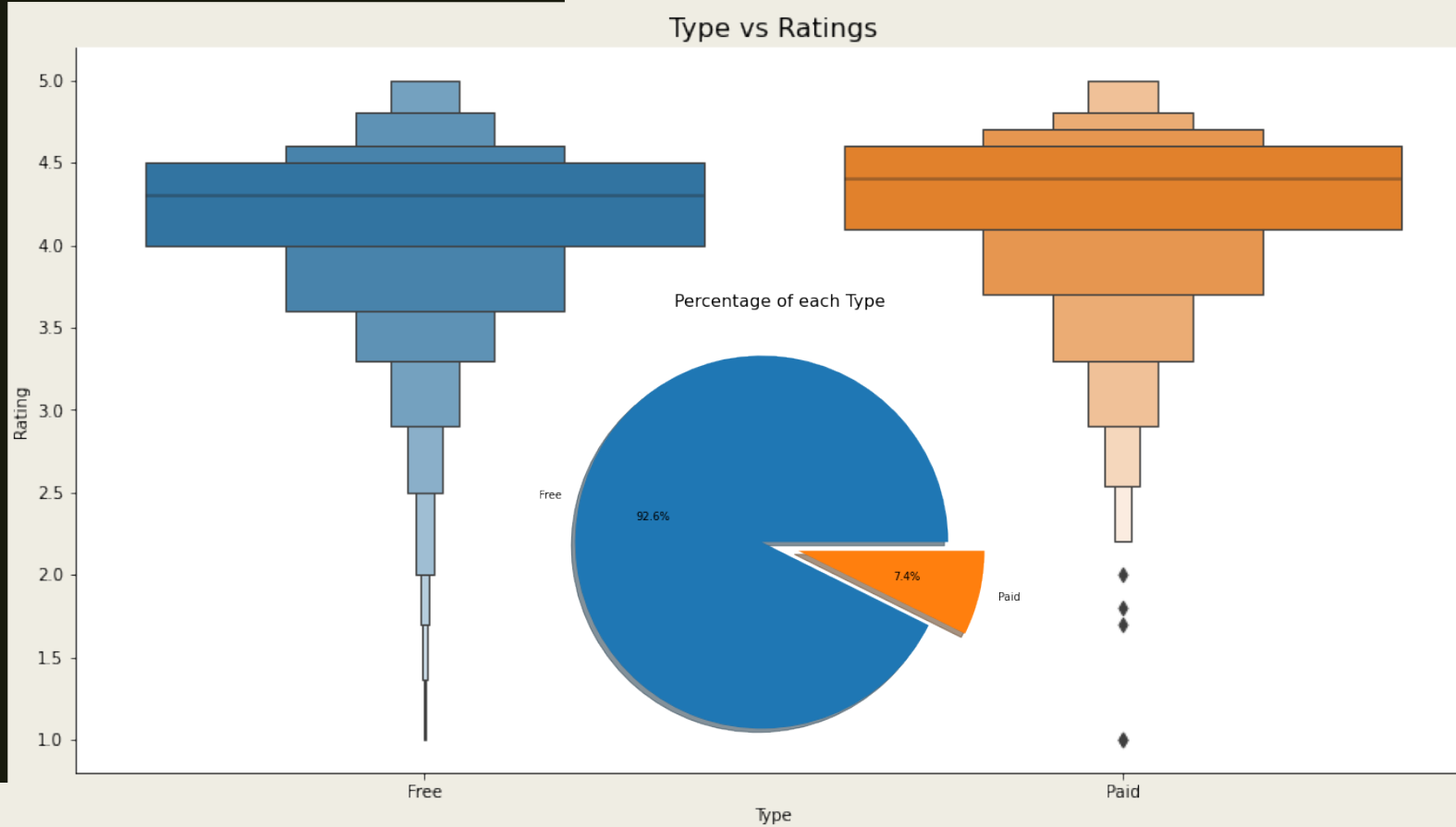
Android Ver: Min required Android version

Data Cleaning and Visualization

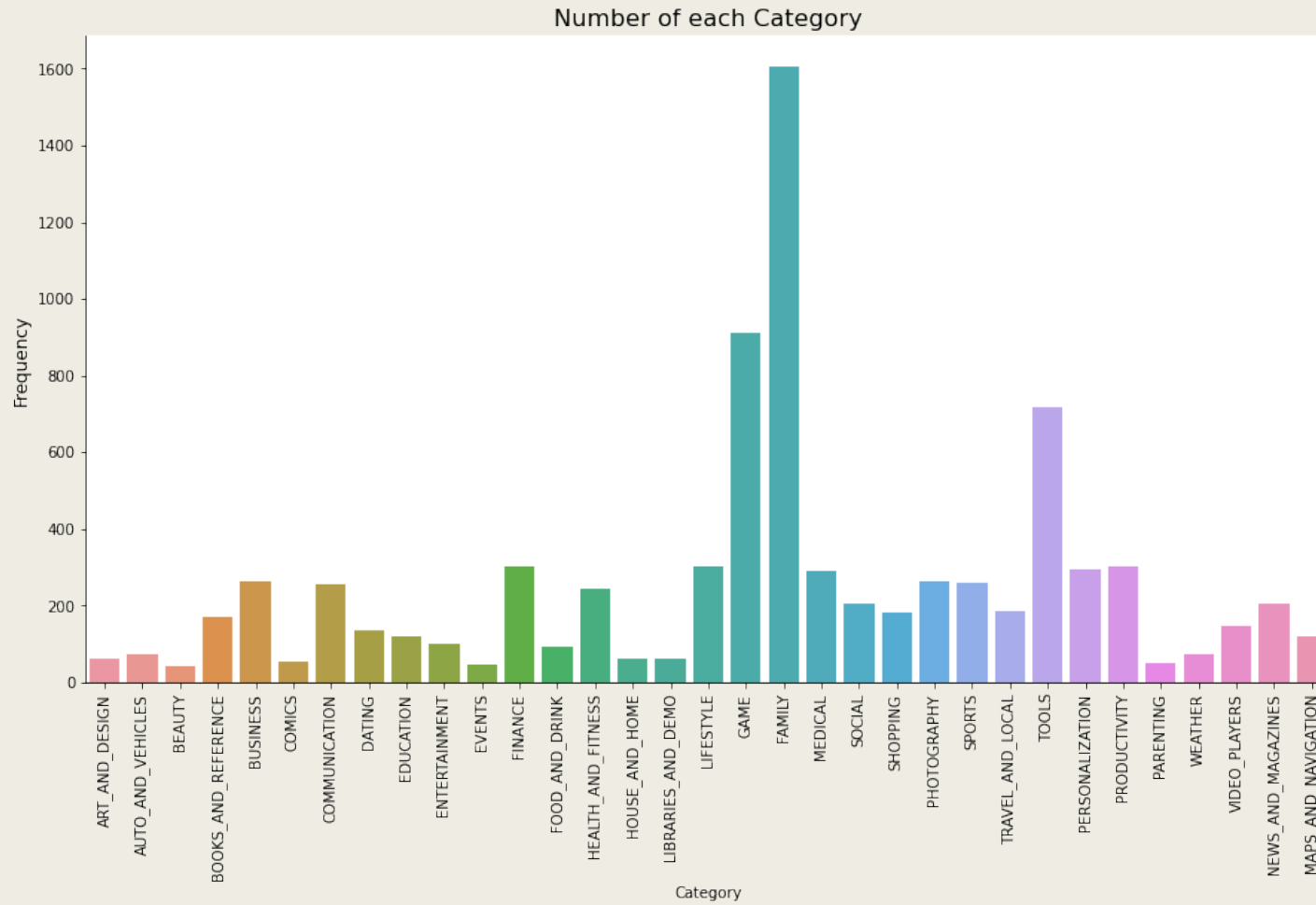
We first show show the distributions on rating with two types: Free & Paid



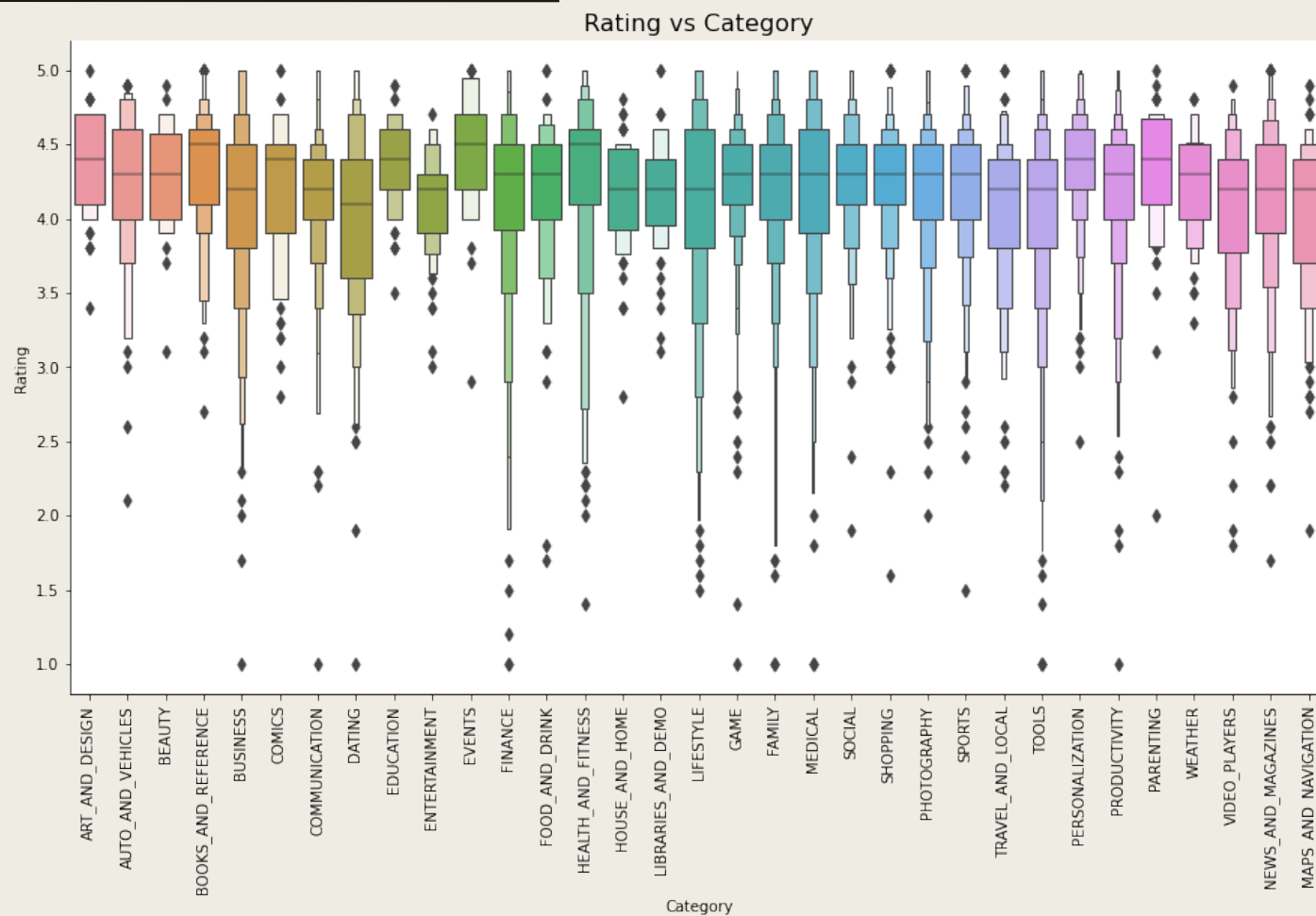
The graph shows that the average rating for paid Apps is slightly higher than that of free ones, which is consistent with our intuition that paid Apps need to be good enough so that people are willing to pay for them.



However, we can see that most of the data come from Free and the average rating for Paid is slightly higher than that of Free.

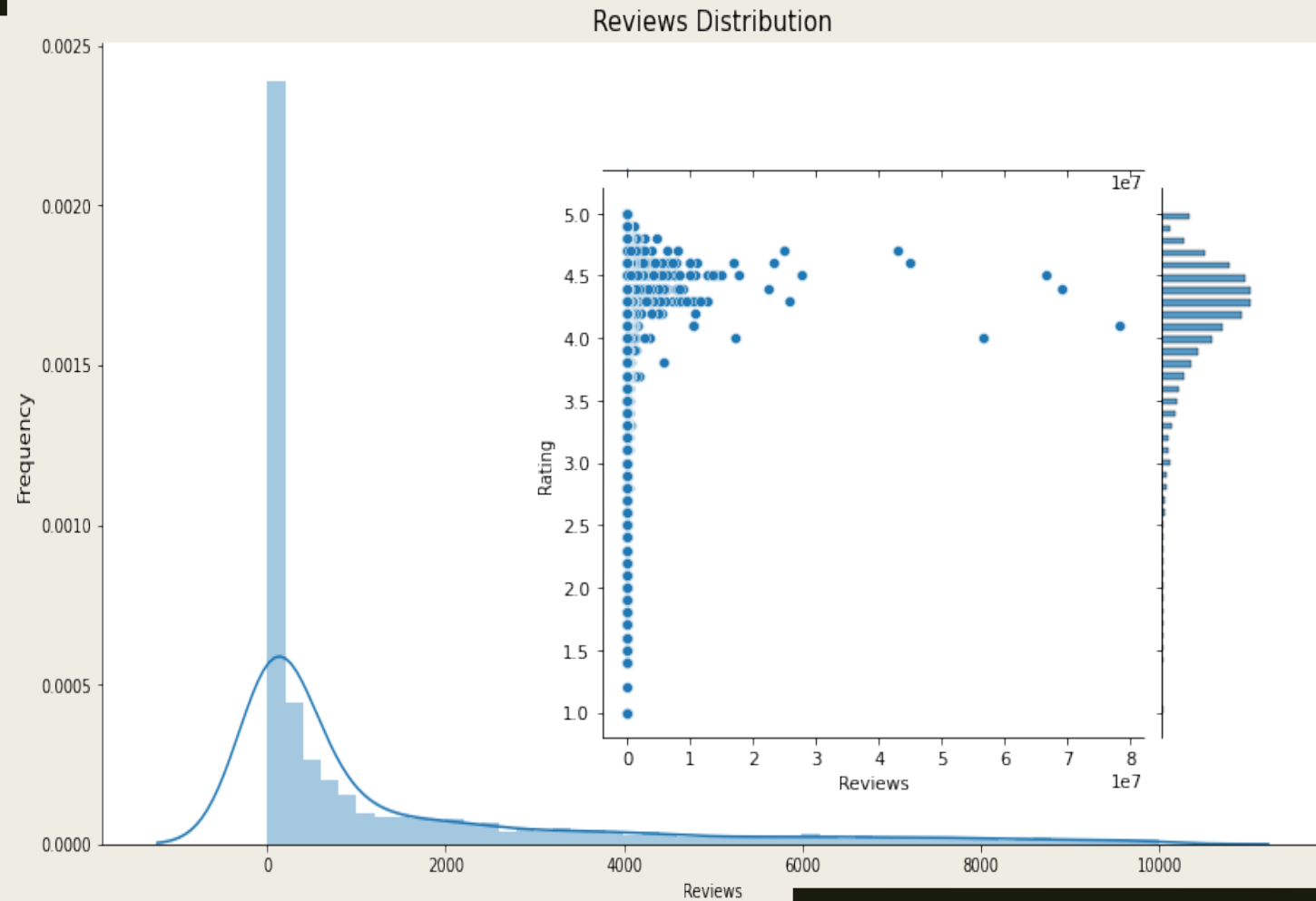


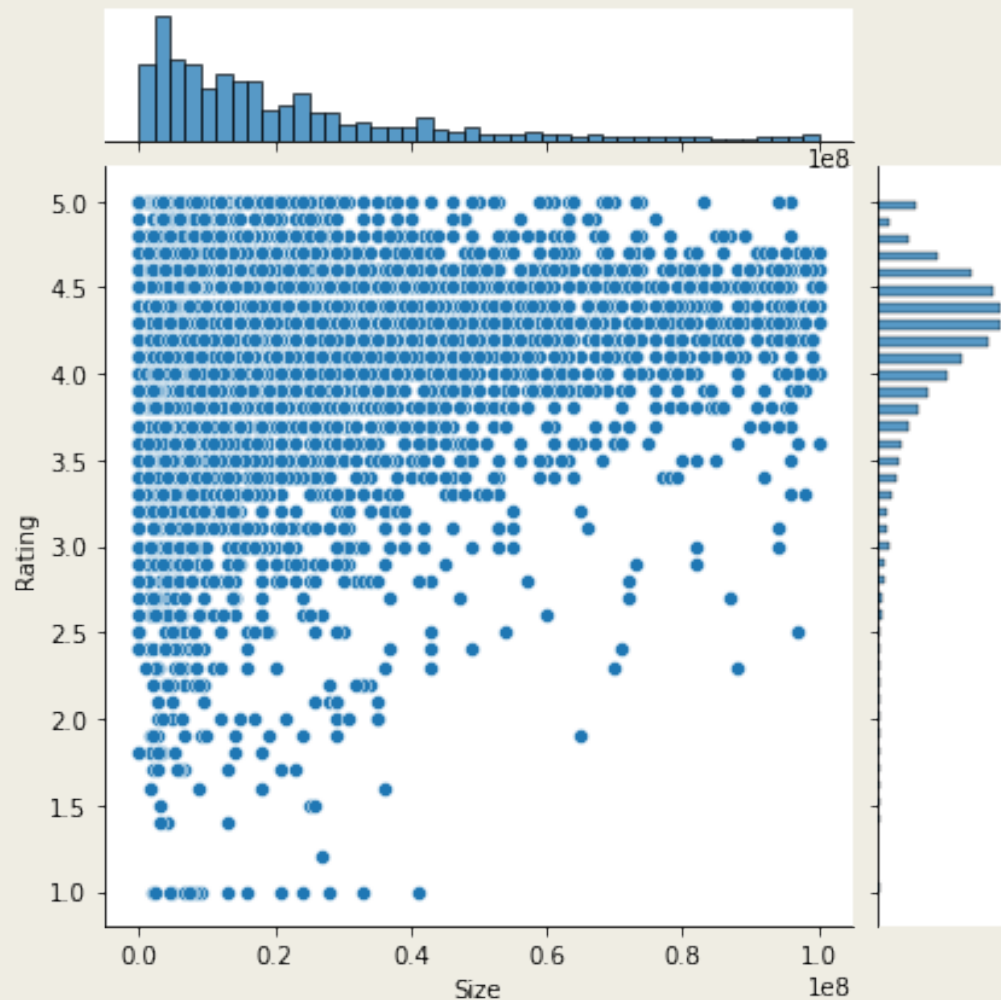
Next, we show the number of Apps each category contains and we can easily see that most data comes from 'Family' and 'Game.'



Next, we visualize the rating with various categories. In this graph, the average rating doesn't vary much from category to category. Almost all the average ratings are in the range from 4.0 to 4.5.

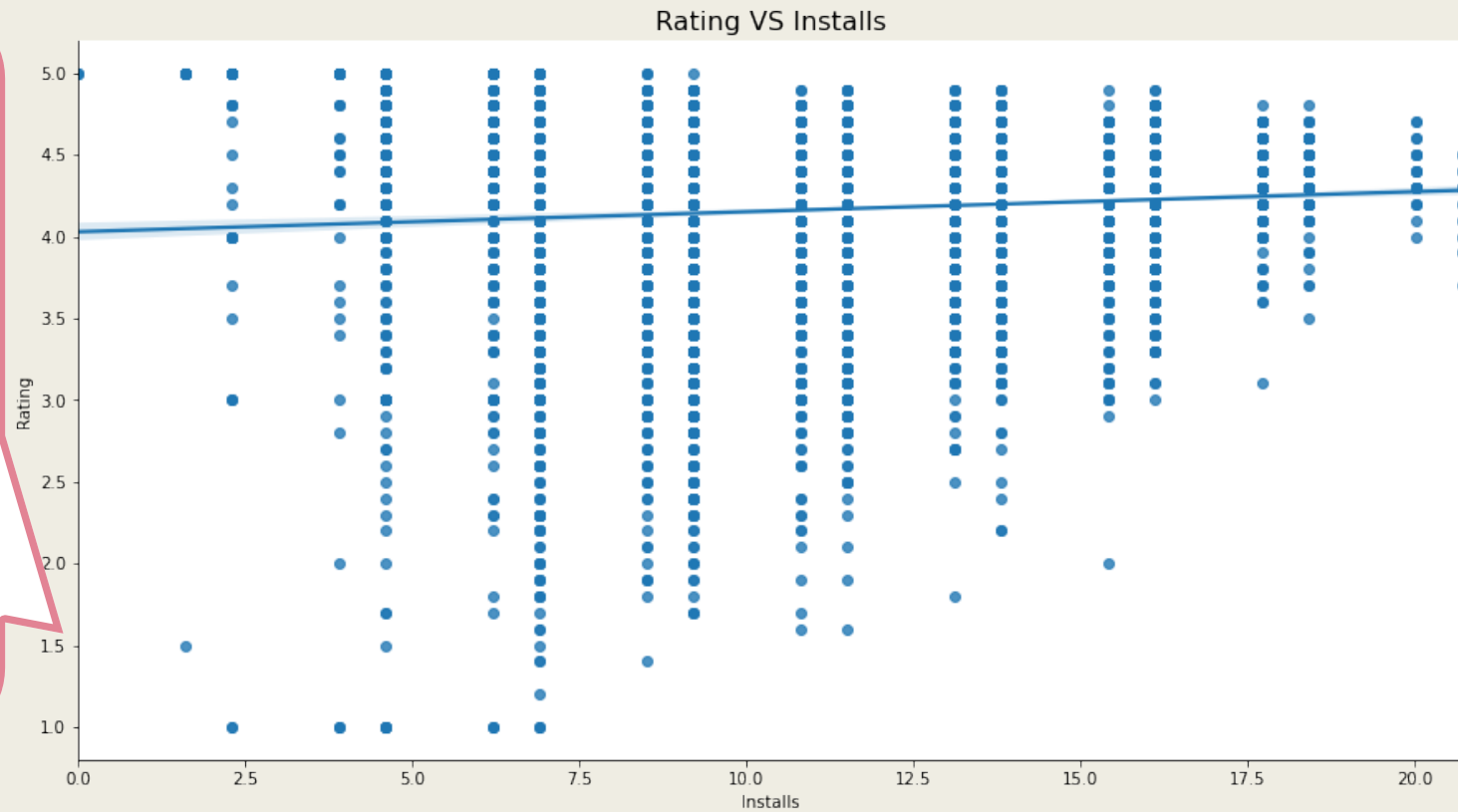
The next feature we care about is 'Reviews'. We show its distribution and then visualize its possible relationship with 'Ratings'. From these graphs, it seems that more reviews make an App higher rating.

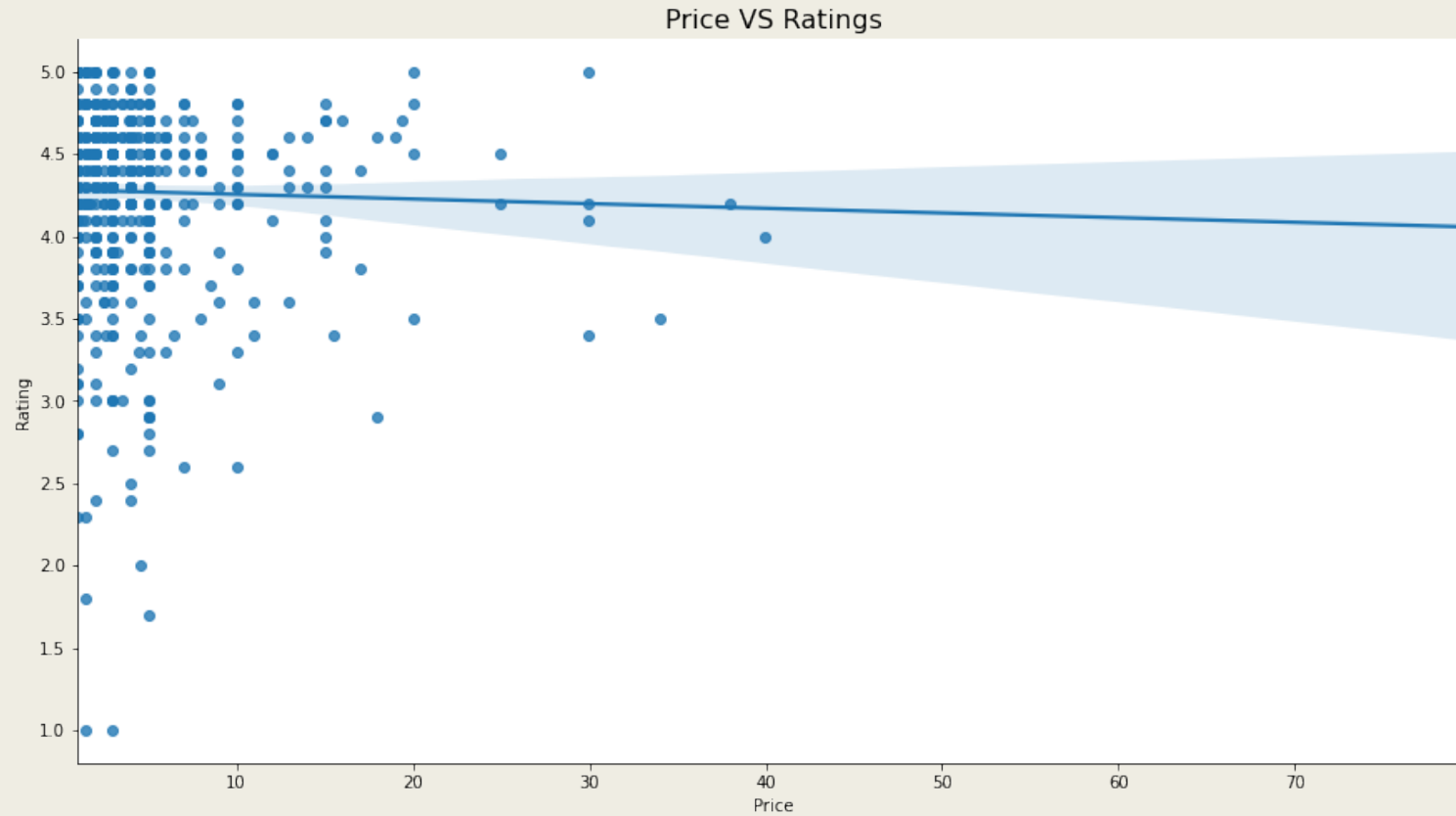




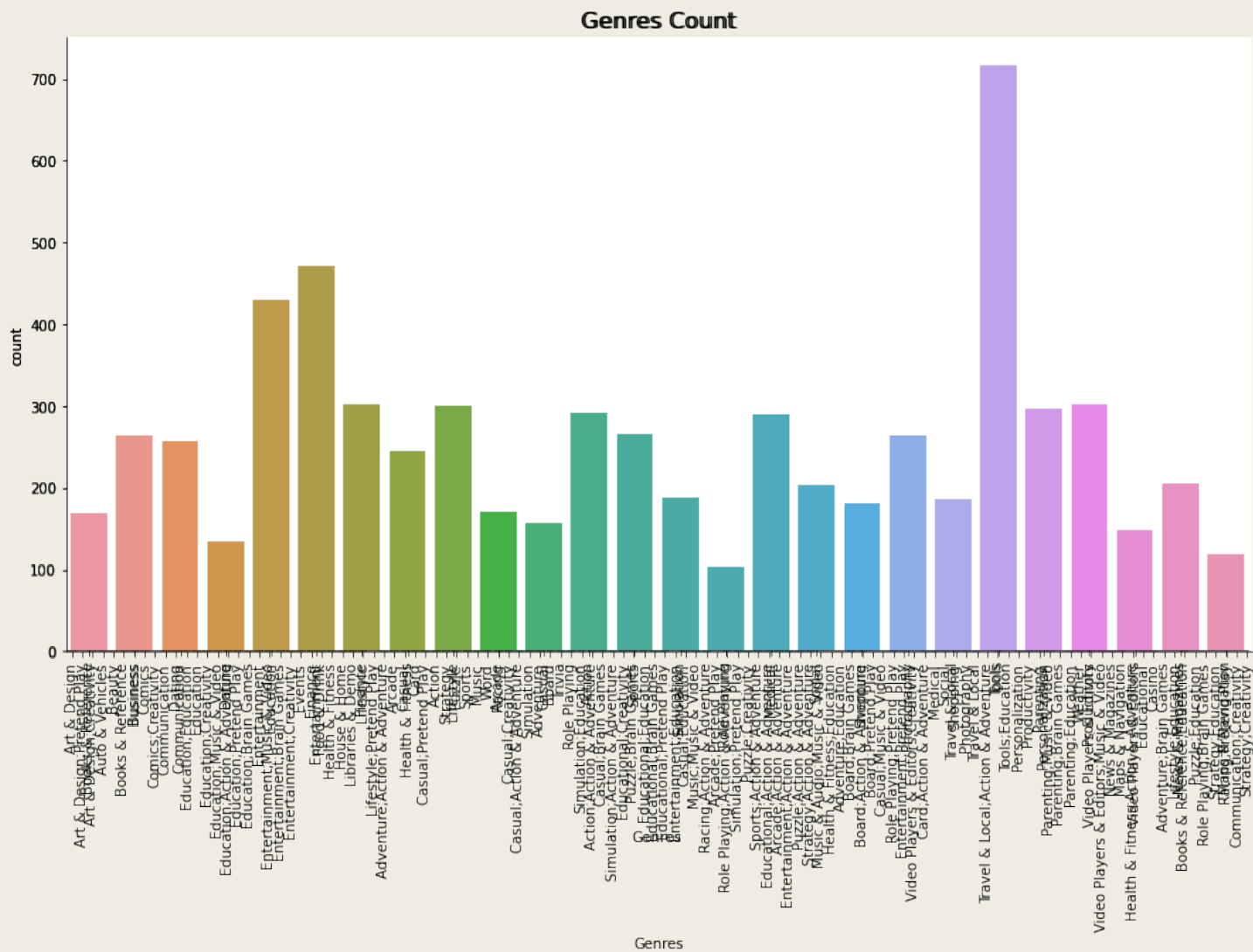
The next feature we want to analyze is 'Size'. We show all the data points based on 'Size' and 'Rating'. Similar to the conclusion from 'Reviews', Apps with larger size tend to have higher rating.

Same conclusion with 'Installs' is achieved that more installs make an App higher rating.

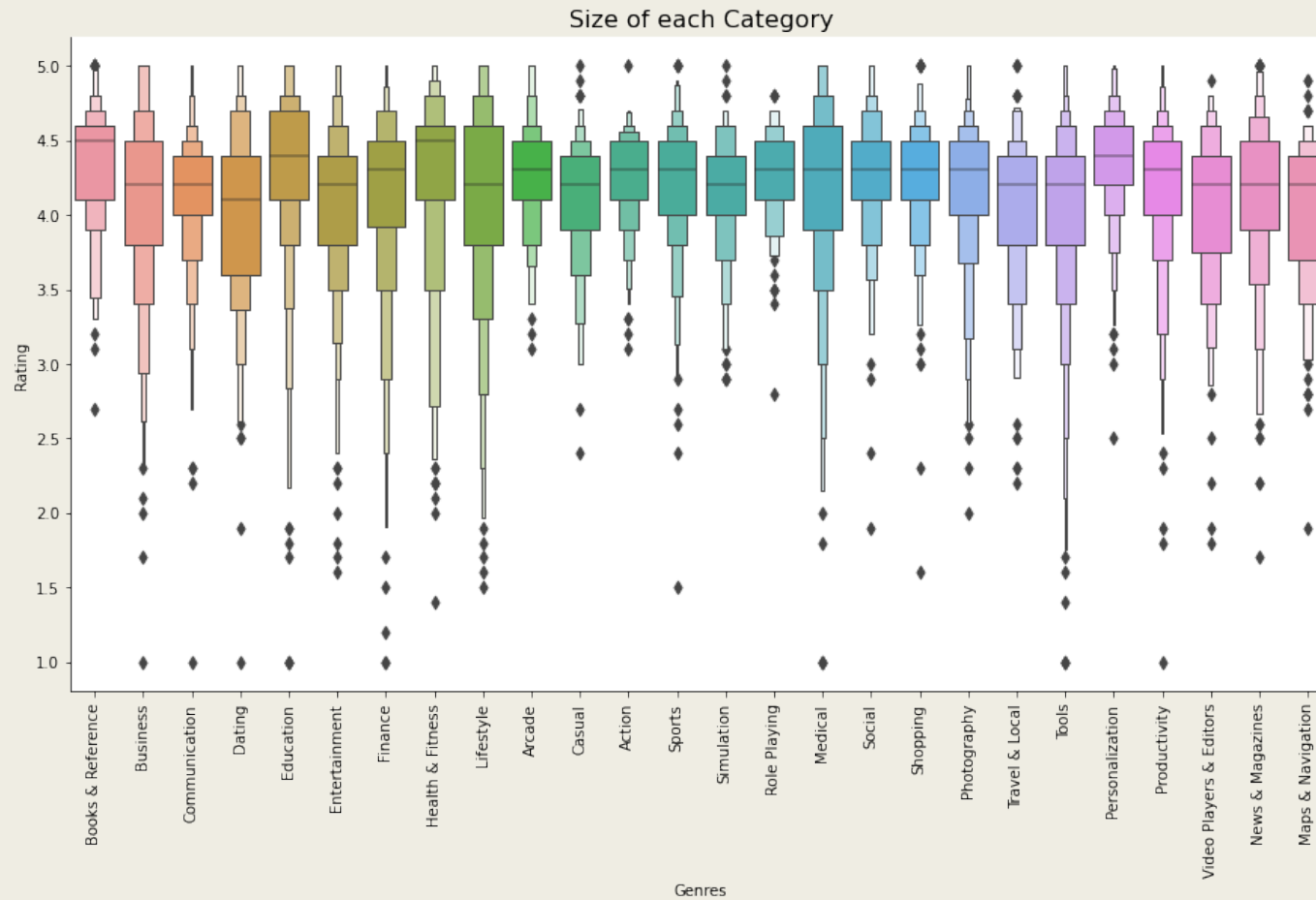




For 'Price', the conclusion is consistent with our intuition that if the price of an App is too high, users will have much higher expectation on this App. Usually, this App cannot satisfy users, which will disappoint the users and make the rating relatively lower.



There are over a hundred genres in this dataset, but the number of data of many of them are small. Therefore, we remove genres with less than 100 data and get a new distribution.



We also visualize the distribution for rating for each genre. This graph is almost the same with the one with 'Rating vs Category', since feature 'Genres' is very correlated to 'Category'.

Rating Prediction

Features: Reviews, Size, Installs, Type, Price, Last Updated, Category

Label: Rating

Data: 75% train, 25% test

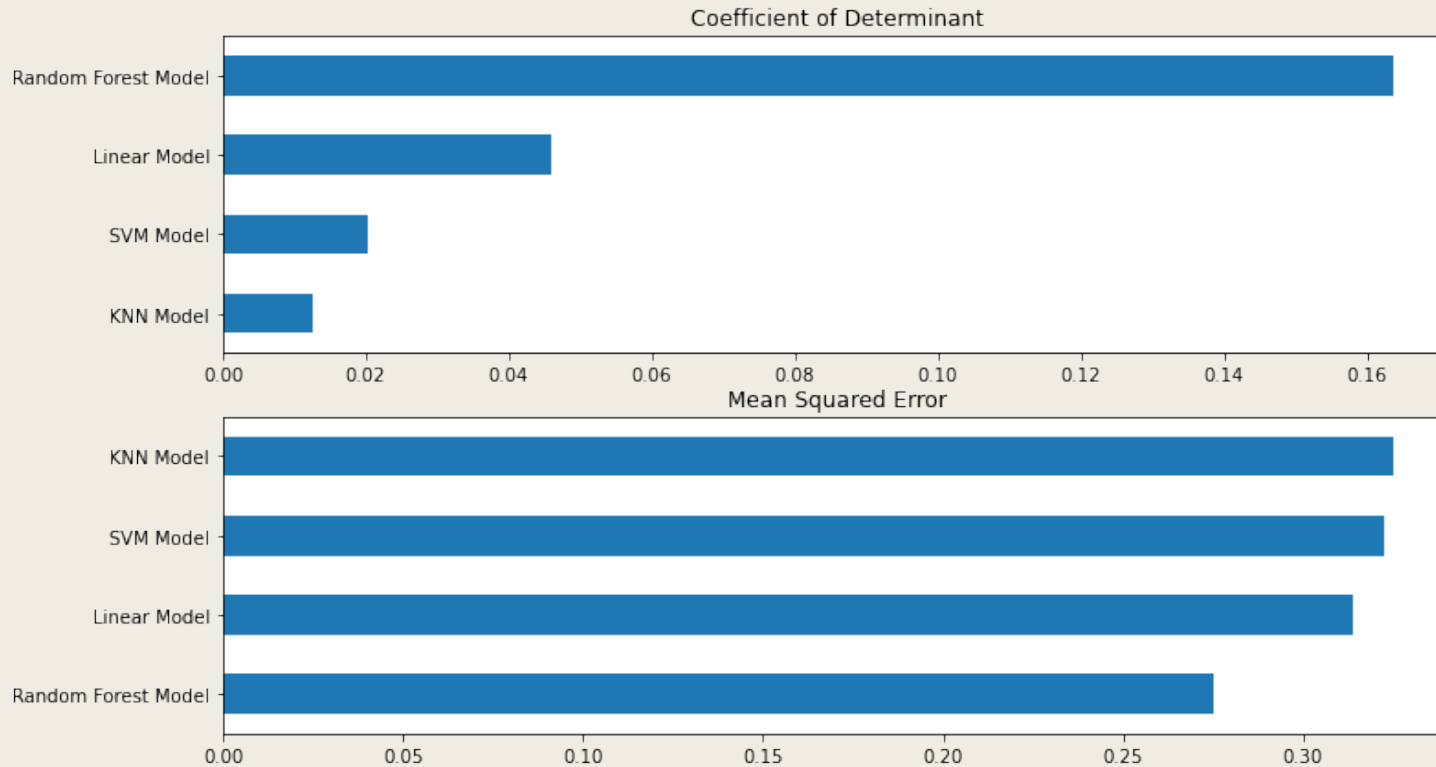
Models:

- K-Nearest Neighbors Model
- Linear Regression Model
- SVM Model
- Random Forest Model

Metrics:

- Coefficient of Determinant: $R^2 = 1 - \frac{(y - \hat{y})^2}{(y - \bar{y})^2}$
- Mean Squared Error: $MSE = \frac{(y - \hat{y})^2}{n}$

Models Comparison



R^2 is close to 0

All outcomes are
the mean of the
train data

Models don't work

References

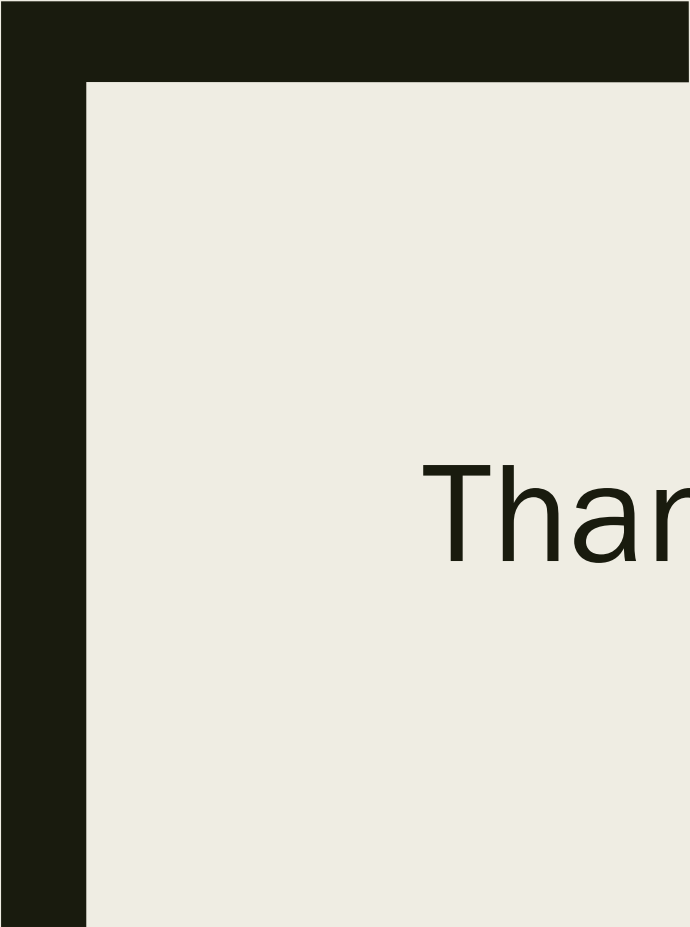
<https://www.kaggle.com/jemseow/machine-learning-to-predict-app-ratings/notebook> Improper evaluation

<https://www.kaggle.com/data13/machine-learning-model-to-predict-app-rating-94/notebook#K-Nearest-Neighbors-Model> Fit on garbage data

<https://www.kaggle.com/rajeshjnv/ml-to-visualization-prediction-of-app-ratings>
Improper evaluation

None of these solid work...

Variables irrelevant? More complicated model?



Thanks for your listening!

