Strategy for Strategy Games

Moses Lin - Linear Regression Mod 2 Project

Overview

Project goal:
Entering the Mobile Game industry is profitable. What aspects of mobile games determine success?

Process

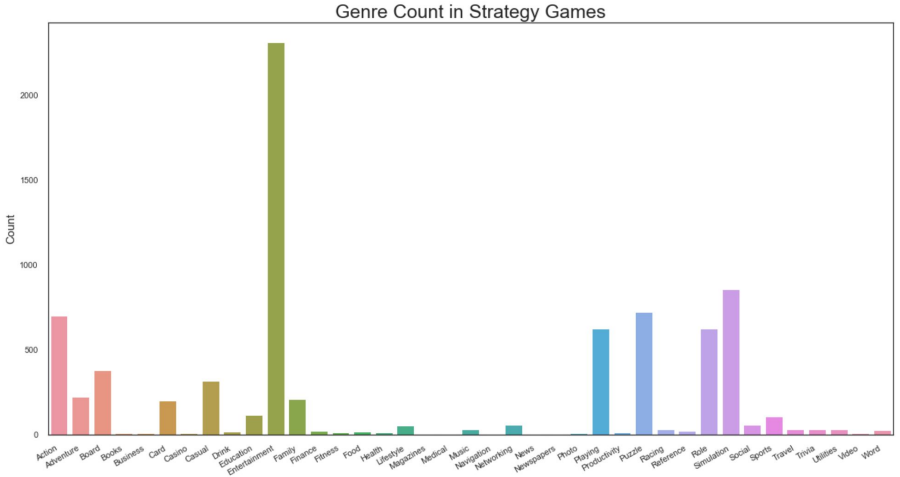
1. Data Cleaning
2. Exploratory Data Analysis
3. Feature Engineering
4. Hypothesis Testing and Correlation
5. Attempted Modeling

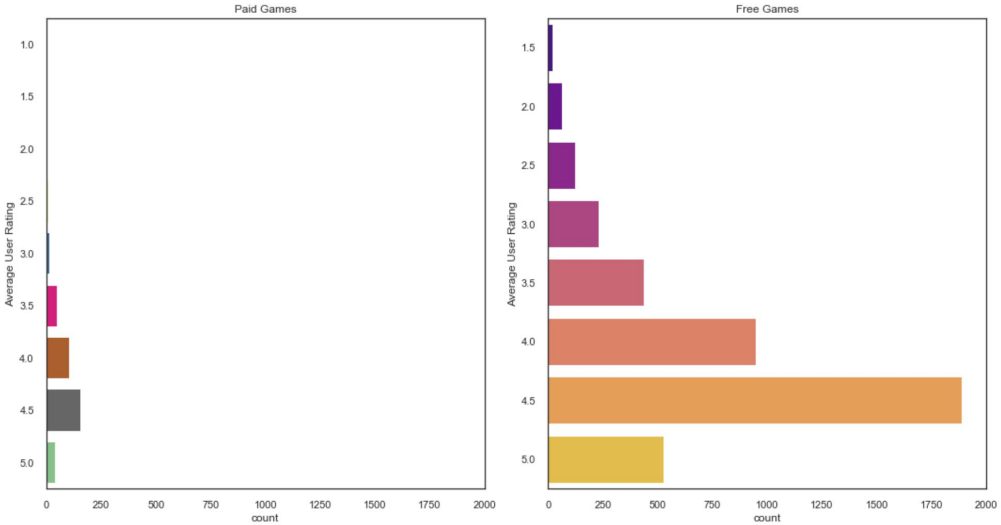
Dataset Overview

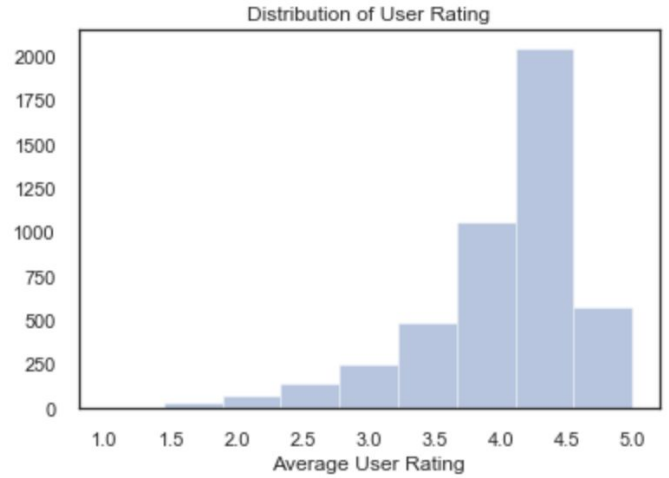
- Data of 17k Strategy Games on Apple App Store, including well-known titles like Pokemon GO, Plants Vs Zombies, and Clash of Clans.
  - August 3, 2019 snapshot of app store.
- 4618 Observations after cleaning data.
- Metric of App Success = **Average User Rating**
- 11 Features of interest
  - User Rating, Prize, Size
  - Desc Length, Name Length
  - # of Languages, Genres
  - Days from Launch, Days since Last Update
  - Age Rating
  - In-app Purchase Options

Exploratory Data Analysis

- Free-to-play is king
- People generally rate apps favorably.
- The most popular genres are:
  - simulation, puzzle, and action







Hypothesis Testing

- There is no statistical difference between Average User Rating and User Rating Count
- Additionally, there is no statistical difference between Average User Rating and Size of the game

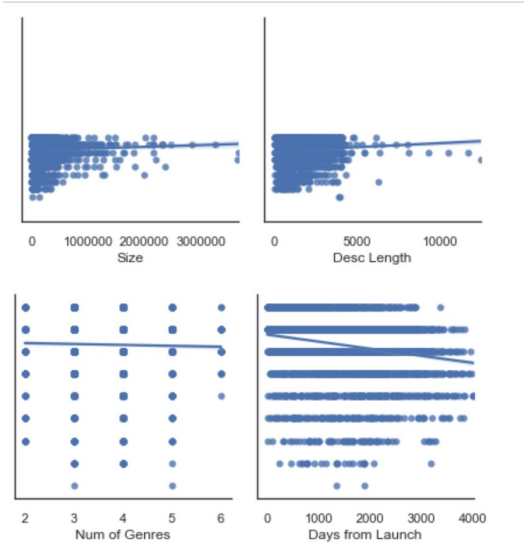
However,

- There is a statistical difference between Average User Rating and Price of the game
- There is a statistical difference between Average User Rating and Description Length.

OLS Modeling and Pairplot

This is where the dataset got very strange.

Most of the features had significance, but very very low coefficients.



OLS Regression Results

Dep. Variable:	Average_User_Rating	**R-squared:**	0.066
Model:	OLS	**Adj. R-squared:**	0.064
Method:	Least Squares	**F-statistic:**	27.29
Date:	Mon, 22 Jun 2020	**Prob (F-statistic):**	1.48e-60
Time:	09:42:55	**Log-Likelihood:**	-4549.0
No. Observations:	4618	**AIC:**	9124.
Df Residuals:	4605	**BIC:**	9208.
Df Model:	12		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	3.4878	0.047	74.808	0.000	3.396	3.579
User_Rating_Count	4.375e-07	1.8e-07	2.427	0.015	8.41e-08	7.91e-07
Price	-0.0007	0.004	-0.178	0.859	-0.009	0.007
Size	-1.031e-07	4.26e-08	-2.417	0.016	-1.87e-07	-1.95e-08
Desc_Length	2.915e-05	1.03e-05	2.819	0.005	8.88e-06	4.94e-05
Name_Length	-0.0008	0.001	-1.136	0.256	-0.002	0.001
Num_of_Lang	-0.0003	0.000	-0.958	0.338	-0.001	0.000
Num_of_Genres	0.0010	0.014	0.069	0.945	-0.027	0.029
Days_from_Launch	-9.947e-05	1.4e-05	-7.102	0.000	-0.000	-7.2e-05
Days_since_update	-0.0001	1.73e-05	-7.924	0.000	-0.000	-0.000
twelveup	0.8939	0.024	37.688	0.000	0.847	0.940
seventeenup	0.7598	0.043	17.669	0.000	0.676	0.844
fourup	0.9030	0.020	45.926	0.000	0.864	0.942
nineup	0.9311	0.023	40.271	0.000	0.886	0.976

Modeling

Model	Feature #	Training RMSE	Test RMSE	R^2 Value
Linear Reg	86	0.627	0.705	0.121
Random Forest Reg	86	N/A	0.631	-4.736
Lasso	86	0.476	0.711	N/A

Recommendations

- It is recommended to develop free-to-play apps, as they are shown to receive a lot more ratings, increasing the chances of having a well rated app.
- It is important to have a well-written description for the game, in order to both entice new users and to not overhype their expectations.
- It is better to have frequent updates to the game over slower, possibly bigger updates.
- Small sum in-app purchases, most specifically $0.99 options are likely the most profitable

Future Work

- Gather data from other sources for comparison
  - Google Play Store
  - Windows Apps - Microsoft Store
- Use iTunes API to get more information about apps
  - More recent data
  - Variables not found in dataset
  - Games outside of only Strategy Games
- Obtain revenue data/ad metrics from SensorTower to directly analyze app success.

Thank You!

Are there any questions?