Introduction:

To identify those who are likely to spend money in their game after finishing the tutorial, I decided to develop a 'profile' of a likely spender. It shows common characteristics of a potential spender. To determine this, I use clustering in PCA, XGboost and Visualization.

Approach:

Data_investigation.ipynb

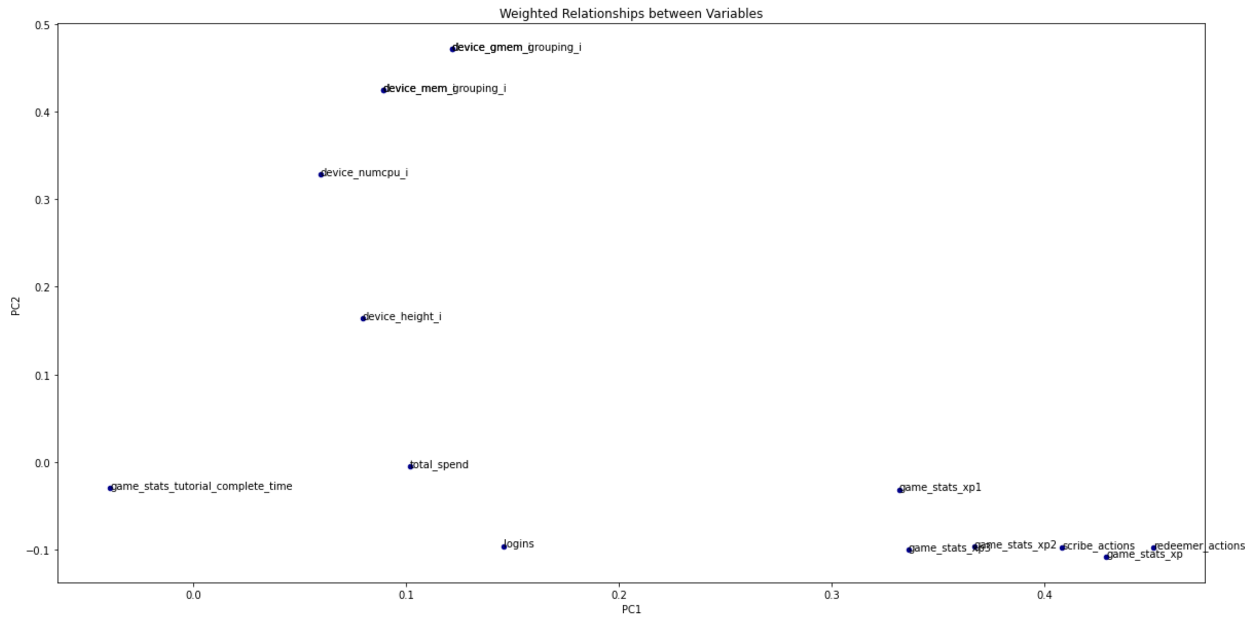
Data is preprocessed. **load_and_combine** joins data from three separate sources on the user_id. We then filter by those who completed the tutorial. This is the population of interest. Next is data investigation.

filter_data checks for and counts nulls. If the count is negligible in relation to data size then rows with nulls are removed from the dataset. In **split_column_types,** we also check for all datatypes within the dataframe. Objects, floats and integer types are grouped. This breakdown is important because the numerical type is used for PCA while the objects will be "one-hot-encoded" for XGboost. Recall that One-hot-encoding creates additional columns, so those with large amounts of unique values drastically increase the dataframe dimension. In **show_unique_values** and **count_unique**, I investigate how many distinct values are present. Due to time constraints, it is easier to remove these columns. Another feasible solution will be to group said distinct values. For example, all those whose text contain Intel are grouped as Intel. Regex and the nlp module nltk will accomplish the groupings.

ML_PCA.ipynb

The purpose of PCA here is to identify which variables capture the most similar information to the numerical response variable **total_spend**. PCA is more than just a dimension reduction algorithm; by using the loading score we can cluster the variables in relation to the two largest dimensions (PC1, PC2). Plotting the loading scores in these two dimensions, we can see which variables are most similar to Total_spend. Granted that this only captures linear relationships and uses numerical variables. We will make use of a robust algorithm to validate this.

PCA shows that '**tutorial_complete_time'** and '**logins'** are closest to our variable of interest '**Total_spend'**. I will consider this candidate set further in Xgboost.



ML.ipynb

Decision tree algorithms are useful for tabular data; they are robust. Of the various ensemble tree based algorithms, I believe that XGboost works best. A great boosting algorithm that handles complex intricacies of large datasets. We can also use the variable importance feature of the model. I use 5 fold cross validated Xgboost to optimize the training model and select the best hyperparameters. I am not building a prediction model.

Although grid search is a plausible method for tuning the model, I used trial and error to find suitable hyper-parameters due to its minimal memory cost. The model performs well with an RMSE of 0.976046.

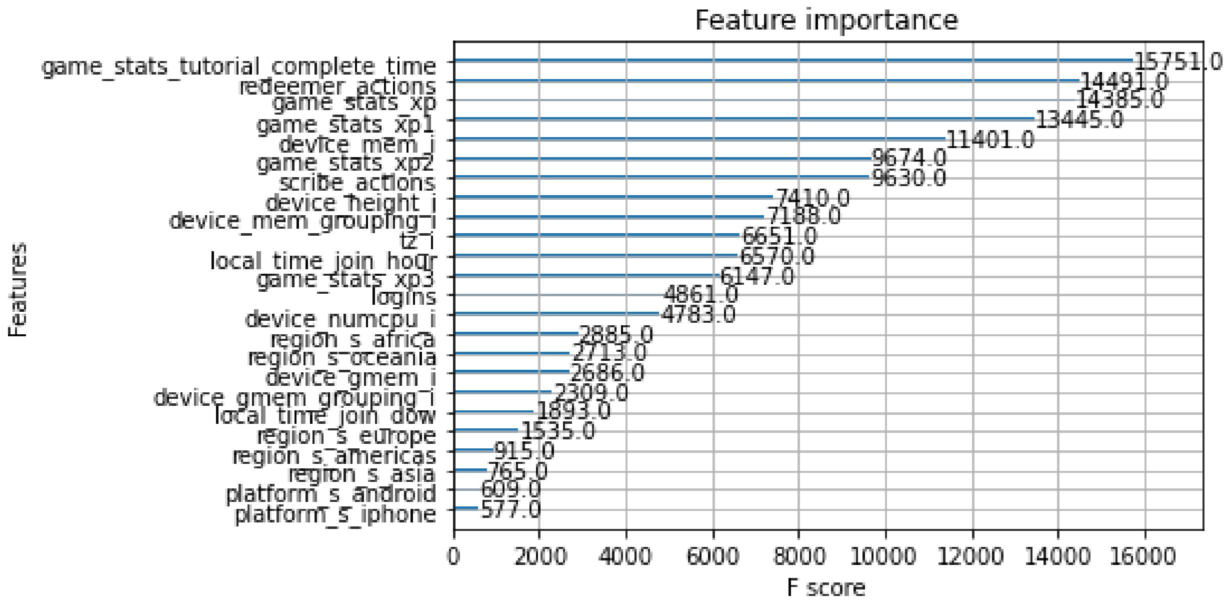
The important variables found here and in PCA are then further investigated using visualization. XGboost confirms the PCA results; the only difference is the importance of '**logins'**. I can attribute this change to the addition of categorical features.

Variables found to be important are:

- 'redeemer_actions',
- 'device_mem_i',

- 'game_stats_xp',
- 'scribe_actions',
- 'game_stats_xp1 and
- 'game_stats_tutorial_complete_time

I did not select game_stat_xp2 because it is likely useful in combination with experience points accumu lated by first and second time interval from install



Data_Profile.ipynb

I converted the numerical response total_spend to categorical variables called 'Spend_Likelihood'. This new variable classifies amount spent based on percentiles. My assumption is that those likely to spend will be in the 99th percentile. I use conditional plots with the new category and important variables to see where the likelihood of spending drastically differs from the likelihood of not spending.

Using conditional plots and the most useful variables from the ML models, a profile is determined.

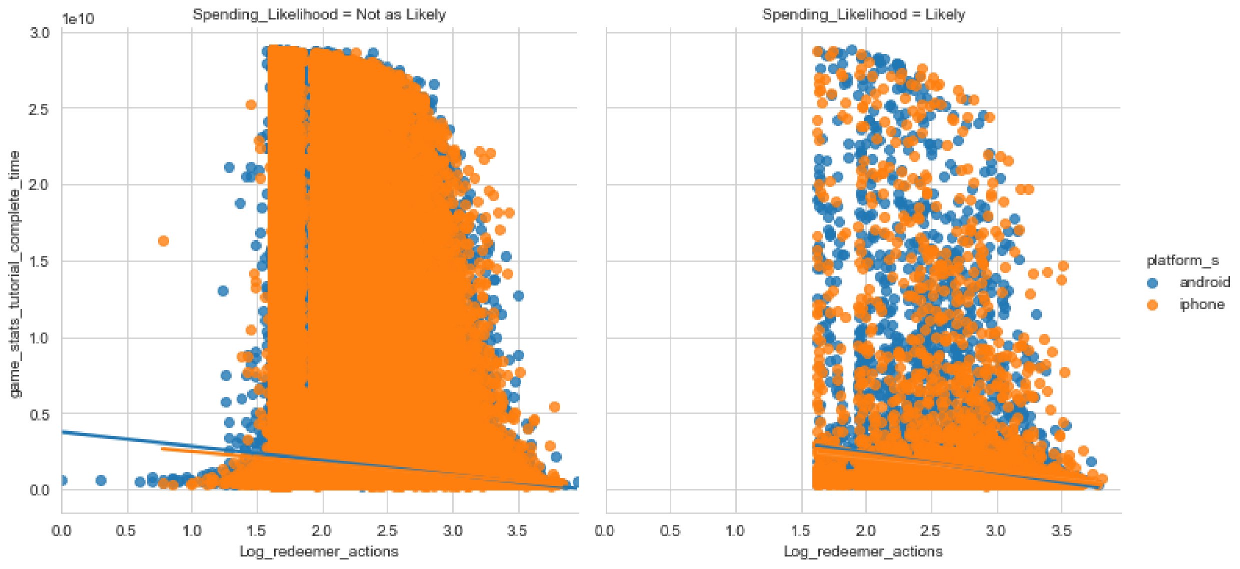
A User likely to spend will:

- Have redeemer actions above 31
    - Extremely true for those in Oceania region
- 11.7% faster at completing the tutorial upon install than those who are not likely (Median value)
    - Specifically from Oceania Region a median complete time of under 527 seconds
- Complete the tutorial under a median value of 571 seconds.
    - They are 20% faster if they use an IPhone
- Have at least 80 experience points accumulated by first time interval from install of
- Device memory of at least 794 for both apple and android
    - Android users have 8% more memory than iPhone so longer tutorials may be pushed for them
- Have at least 31.6 experience points accumulated by second time interval from install
- Have over 115 scribe actions taken since install

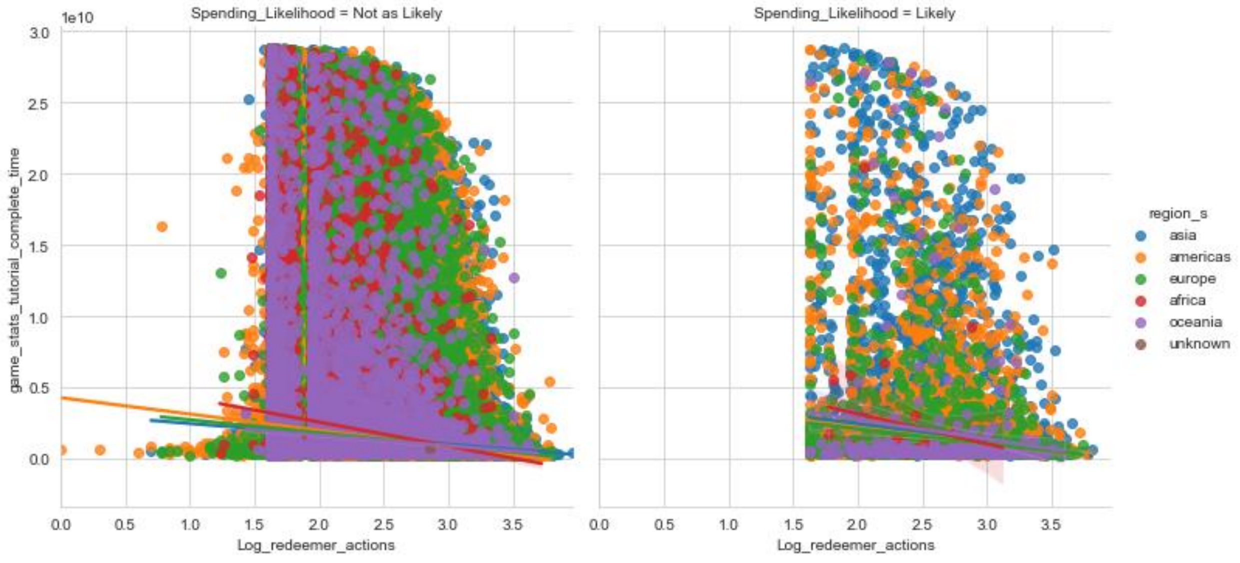
Summary

Prices can be increased for users who fit these criteria, as they are more likely to spend. Key metrics are time taken to finish tutorials, memory available on device, amount of experience points accumulated within several period after install, platform, region, redeemer and scribe actions.

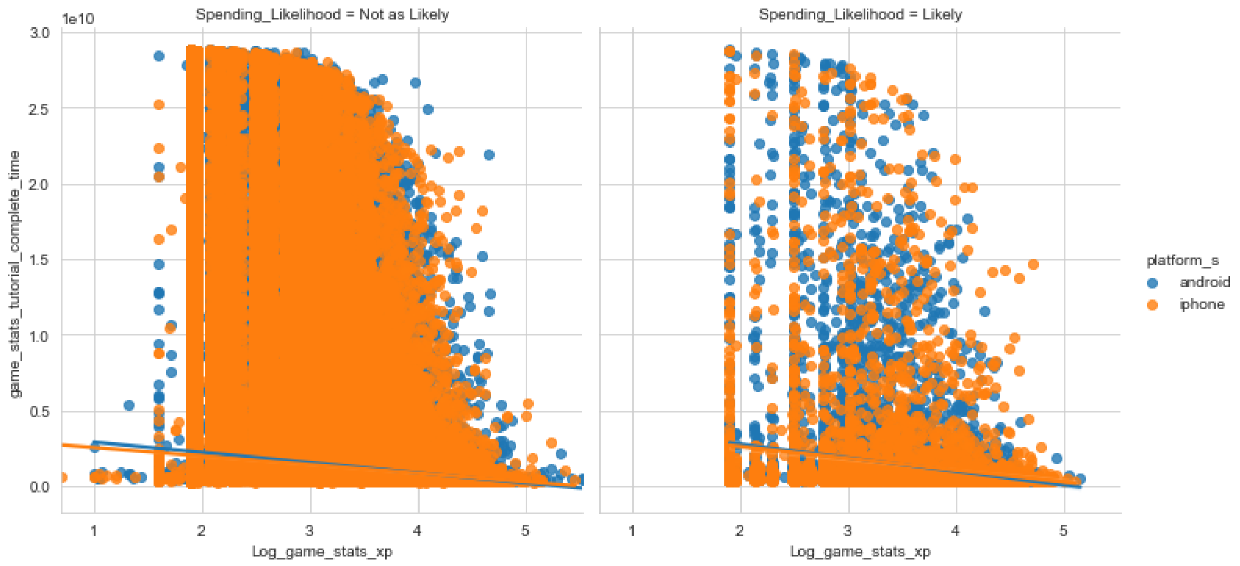
Attached below are outputs to generate the profile



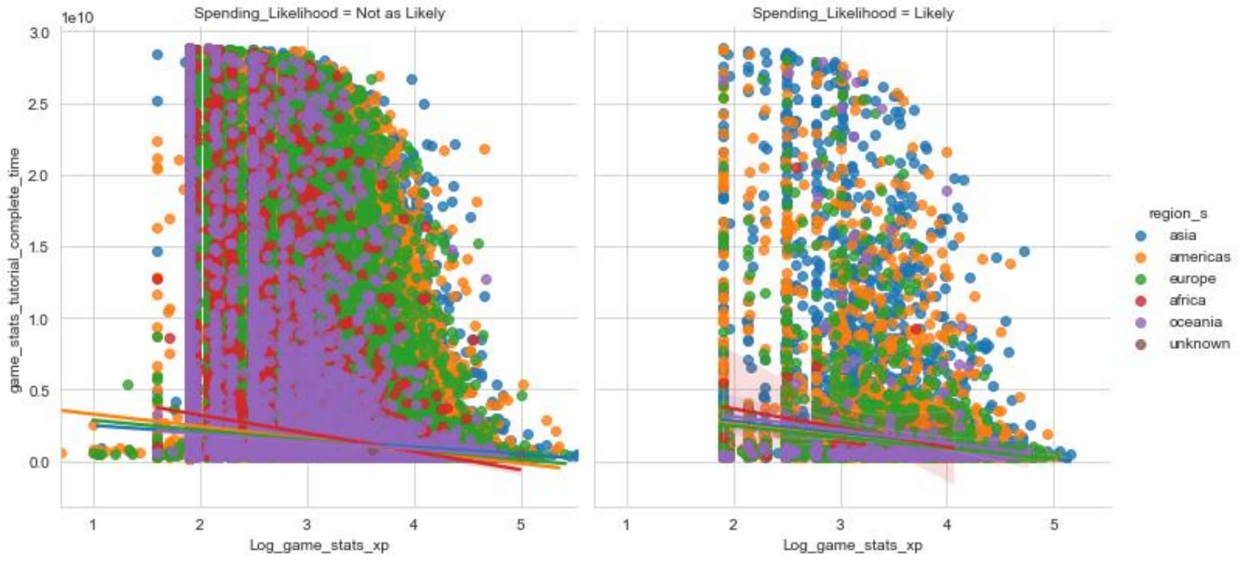
Spending_Likelihood = Not as Likely
Spending_Likelihood = Likely
1e10
game_stats_tutorial_complete_time
Log_redeemer_actions
platform_s
android
iphone



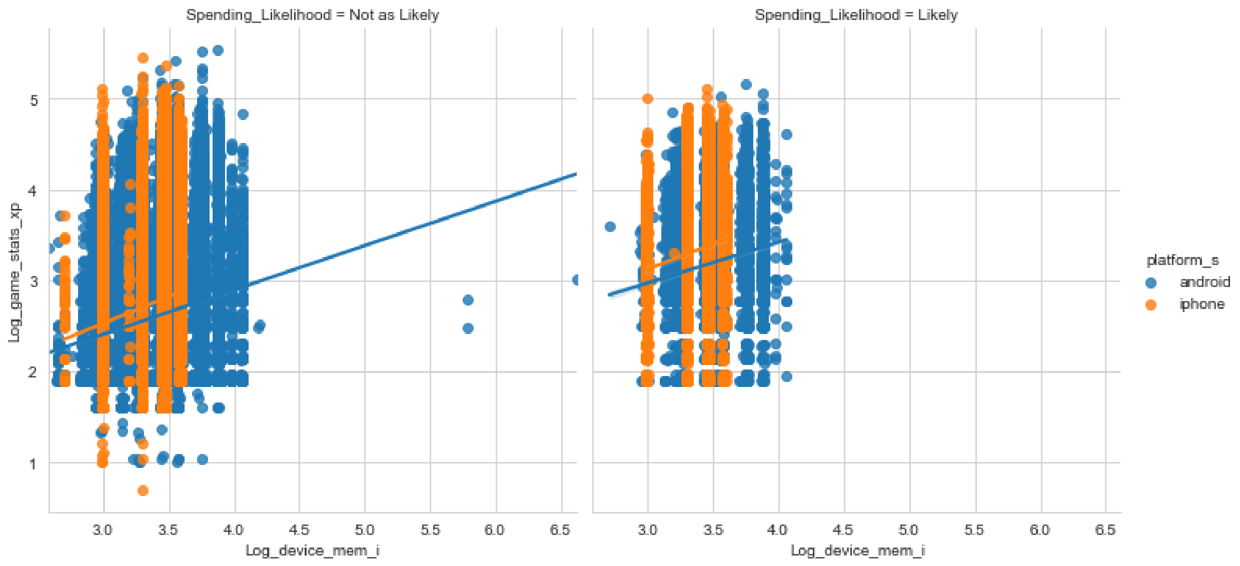
Spending_Likelihood = Not as Likely
Spending_Likelihood = Likely
1e10
game_stats_tutorial_complete_time
Log_redeemer_actions
region_s
asia
americas
europe
africa
oceania
unknown



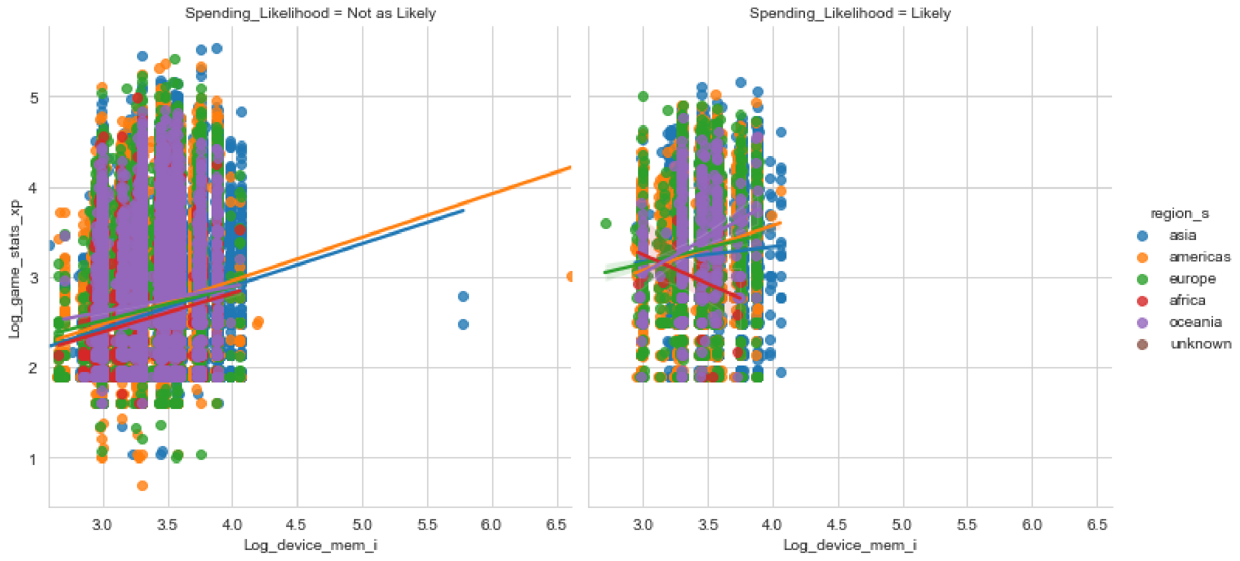
Spending_Likelihood = Not as Likely
Spending_Likelihood = Likely
1e10
game_stats_tutorial_complete_time
Log_game_stats_xp
platform_s
android
iphone



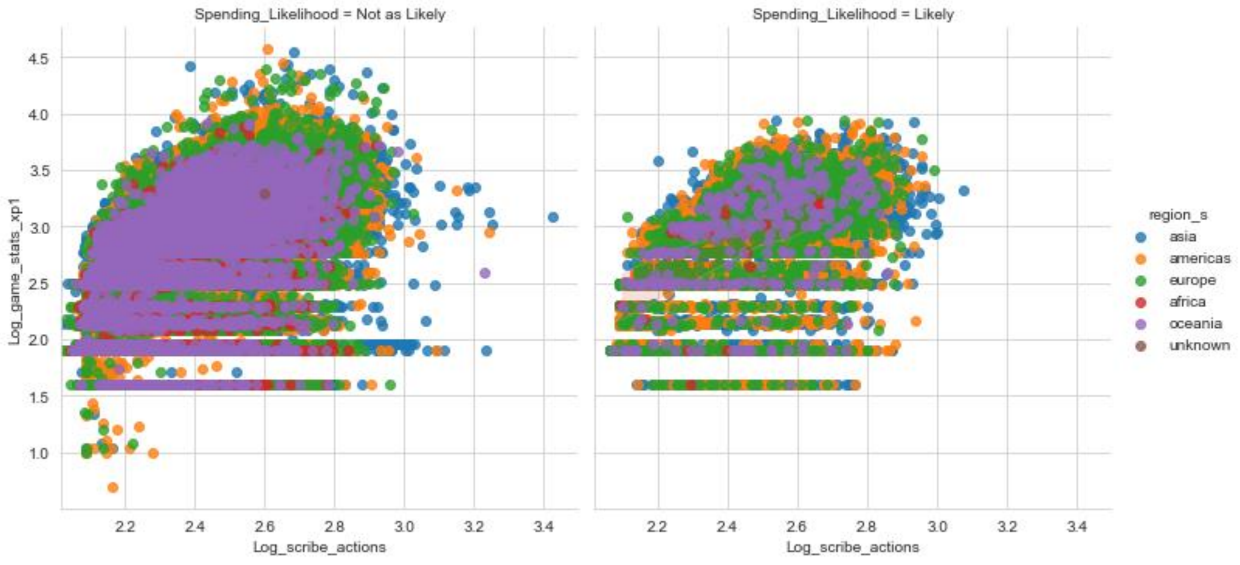
Spending_Likelihood = Not as Likely
Spending_Likelihood = Likely
1e10
game_stats_tutorial_complete_time
Log_game_stats_xp
region_s
asia
americas
europe
africa
oceania
unknown



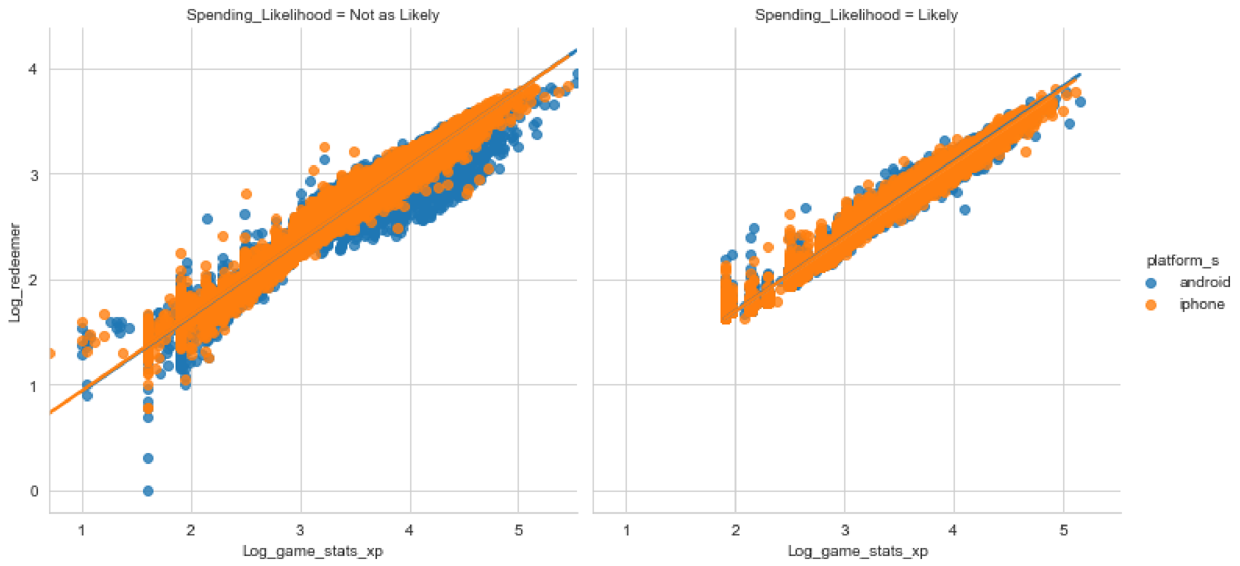
Spending_Likelihood = Not as Likely
Spending_Likelihood = Likely
Log_game_stats_xp
Log_device_mem_i
platform_s
android
iphone
1
2
3
4
5
3.0
3.5
4.0
4.5
5.0
5.5
6.0
6.5



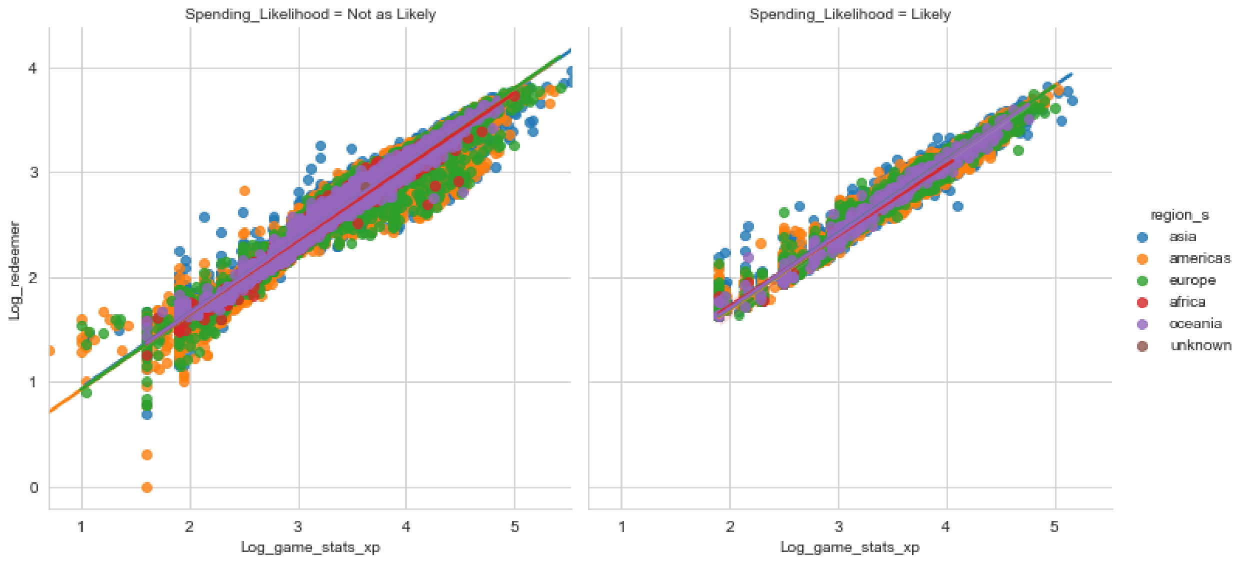
Spending_Likelihood = Not as Likely
Spending_Likelihood = Likely
Log_game_stats_xp
Log_device_mem_i
region_s
asia
americas
europe
africa
oceania
unknown
1
2
3
4
5
3.0
3.5
4.0
4.5
5.0
5.5
6.0
6.5



Spending_Likelihood = Not as Likely
Spending_Likelihood = Likely
Log_game_stats_xp1
Log_scribe_actions
region_s
asia
americas
europe
africa
oceania
unknown



Spending_Likelihood = Not as Likely
Spending_Likelihood = Likely
Log_redeemer
Log_game_stats_xp
platform_s
android
iphone


Spending_Likelihood = Not as Likely
Spending_Likelihood = Likely
Log_redeemer
Log_game_stats_xp
Log_game_stats_xp
0
1
2
3
4
1
2
3
4
5
region_s
asia
americas
europe
africa
oceania
unknown