Play Store Apps Review Analysis

Ajay Pedhekar, Uday Firake,
Narendra Ghodese, Saurabh Kumar,
Sandhya Sah
Data science trainees,
AlmaBetter, Bangalore

Abstract-

Google play store is engulfed with a few thousand new applications regularly with a progressively huge number of designers working freely or on the other hand in a group to make them successful, with enormous challenges from everywhere throughout the globe. Since most Play Store applications are free, the income model is very obscure and inaccessible regarding how the in-application buys, adverts, and memberships add to the achievement of an application. In this way, an application's prosperity is normally dictated by the quantity of installation of the application, and the client appraisals that it has gotten over its lifetime instead of the income created. Application (App) ratings are feedback users provide and function as important evaluation criteria for apps. attributes present in my datasets such as which application is free or paid the user reviews, and the rating of the application.

Key Words: Google Play Store Apps, Ratings Prediction, Exploratory Data Analysis, Machine Learning

1. Problem Statement:

The Play Store app data has enormous potential to drive app-making businesses to success. Actionable insights can be drawn for developers to work on and capture the Android market.

Each app (row) has values for the category, rating, size, and more. Another dataset contains customer reviews of the android apps.

Explore and analyze the data to discover key factors responsible for app engagement and success.

2. Introduction

Machine learning approaches are essential for us to take care of numerous issues. In this paper, we present machine learning models and structures in detail. Machine learning has numerous applications in numerous perspectives and has incredible advancement potential.

In the future, it is predictable that machine learning could set up ideal speculations to clarify its exhibitions. In the meantime, its capacities for unsupervised learning will be improved since there is much information on the planet however it isn't relevant to add names to every one of them. It is additionally anticipated that neural system structures will turn out to be increasingly unpredictable with the goal that they can separate all the more semantically important highlights. In addition, profound learning will consolidate with support adapting better

and we can utilize these points of interest to achieve more assignments.

2.1 Google Play store and User Review Analysis

In today's scenario, we can see that mobile apps play an important role in any individual's life. It has been seen that the development of mobile application advertising has an incredible effect on advanced innovation. Having said that, with the consistently developing versatile application showcase there is additionally an eminent ascent of portable application designers inevitably bringing about high as can be income by the worldwide portable application industry.

With the enormous challenge from everywhere throughout the globe, it is basic for a designer to realize that he is continuing in the right heading. To hold this income and their place in the market the application designers may need to figure out how to stick to their present position. The Google Play Store is observed to be the biggest application platform. It has been seen that although it creates more than two-fold the downloads as the Apple App Store yet makes just a large portion of the cash contrasted with the App Store. In this way, I scratched information from the Play Store to direct our examination on it.

With the fast development of advanced cells, portable applications (Mobile Apps) have turned out to be basic pieces of our lives. Be that as it may, it is troublesome for us to follow along with the fact and understand everything about the apps as new applications are entering the market each day. It is accounted for that the Android market achieved a large portion of a million applications in September 2011. Starting now, 0.675 million Android applications are accessible on Google Play App Store. Such a lot of applications are by all accounts an extraordinary open door for clients to purchase from a wide determination extent. We trust versatile application clients consider online application surveys as a significant impact for paid applications. It is trying for a potential client to peruse all the literary remarks and ratings to settle on a choice. Additionally, application engineers experience issues in discovering how to improve the application execution dependent on generally speaking evaluations alone and would profit by understanding a huge number of printed remarks.

We develop Android apps & release them on Play Store. From a Developer or say Business Perspective it's very important to know whether users are enjoying the app or facing any issues. To know this Play Store has a Ratings & reviews section for each app released on the play store. Users can submit the ratings and has the freedom to write a review for a particular app. This approach is quite lengthy to rate & review the app i.e. navigate to the Play store to submit feedback or redirect leaving a current app workflow to open the Play Store App link using URI. We never wanted our customers to leave our application, but with this flow, we are forced to redirect the control to the Play store app.

2.2 Google Play store Dataset

The dataset consists of the Google play store application and is taken from Almabetter, which is the world's largest community for data scientists to explore, analyze and share data.

This dataset is for Web scratched information of 10k Play Store applications to analyze the market of android. Here is a downloaded dataset that a user can use to examine the Android market of different use of classifications music, camera, etc. With the assistance of this, the client can predict see whether any given application will get a lower or higher rating level. This dataset can be moreover used for future references for the proposal of any application. Additionally, the disconnected dataset is picked to choose the estimate exactly as online data gets revived all around a great part of the time. With the assistance of this dataset, I will examine various qualities like rating, free or paid and so forth utilizing Hive and after that, I will likewise do a forecast of various traits like client surveys, ratings, etc.

The data set contains the following columns:

- **App:** This Column contains the name of the app
- **Category:** This contains the category to which the app belongs. The category column contains 33 unique values.
- **Rating:** This column contains the average value of the individual rating the app has received on the play store. Individual rating values can vary between 0 to 5.
- **Reviews:** This column contains the number of people that have given their feedback on the app.
- **Size:** This column contains the size of the app i.e. The memory space that the app occupies on the device after installation.
- **Installs:** This column indicates the number of times that the app has been downloaded from the play store, these are approximate values and not absolute values.
- **Type:** This column contains only two values- free and paid. They indicate whether the user must pay money to install the app on their device or not.
- **Price:** For paid apps, this column contains the price of the app, for free apps, it contains the value 0.
- **Content Rating:** It indicates the targeted audience of the app and their age group.
- **Genre:** This column contains which genre the app belongs to, genre can be considered as a sub-division of Category.
- **Last updated:** This column contains the info about the date on which the last update for the app was launched.
- **Current version:** Contains information about the current version of the app available on the play store.
- **Android version:** Contains information about the version of the Android OS on which the app can be installed.

2.2 User Review Dataset

- The user reviews data frame has 64295 rows and 5 columns. The 5 columns are identified as follows:
- **App:** Contains the name of the app with a short description (optional).
- **Translated Review:** It contains the English translation of the review dropped by the user of the app.
- **Sentiment:** It gives the attitude/emotion of the writer. It can be 'Positive', 'Negative', or 'Neutral'.
- **Sentiment Polarity:** It gives the polarity of the review. Its range is [-1,1], where 1 means 'Positive statement' and -1 means a 'Negative statement'.
- **Sentiment Subjectivity:** This value gives how close a reviewer's opinion is to the opinion of the general public. Its range is [0,1]. The higher the subjectivity, the closer the reviewer's opinion to the opinion of the general public, and lower subjectivity indicates the review is more factual information.

2.3 Python

Most of the info scientists use python due to the good built-in library functions and therefore the decent community. Python now has 70,000 libraries. Python is the simplest programing language to select compared to another language. That is the most reason data scientists use python more often, for machine learning and data processing data analyst wants to use some language that is straightforward to use. That is one of the most reasons to use python. Specifically, for data scientists, the foremost popular data inbuilt open-source library is named panda. As we have seen earlier in our previous assignment once we got to plot scatterplots, heat maps, graphs, and 3-dimensional data python built-in library comes very helpful.

2.4 Data Cleaning and Preparation

Preprocessing is important in transitioning raw data into a more desirable format. Undergoing the preprocessing process can help with completeness and comparability. For instance, you'll see if certain values were recorded or not. Also, you'll see how trustable the info is. It could also help with finding how consistent the values are. We need preprocessing because most real-world data are dirty. Data can be noisy i.e. the data can contain outliers or simple errors generally. Data can also be incomplete i.e. there can be some missing values.

The available data is raw and unusable for Exploratory data analysis, so before we do anything with the data we will have to explore and clean it to prepare it for data analysis.

3. EXPLORATORY DATA ANALYSIS

Exploratory Data Analysis, or EDA, is an important step in any Data Analysis or Data Science project. EDA is the process of investigating the dataset to discover patterns, and anomalies (outliers), and form hypotheses based on our understanding of the dataset. EDA involves generating summary statistics for numerical data in the dataset and creating various graphical representations to understand the data better. In this article, we will understand EDA with the help of an example dataset. We will use **Python** language (**Pandas** library) for this purpose.

3.1 Free vs Paid

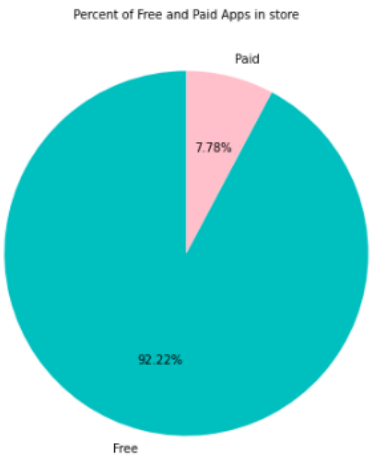
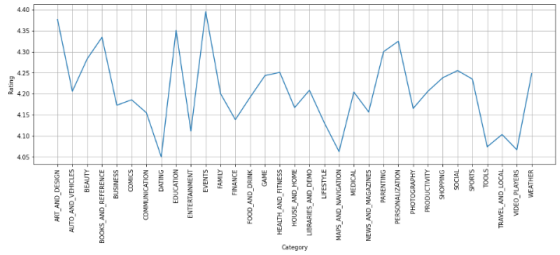


Fig. Free Vs Paid

Here we can see that 92.2% of apps are free, and 7.80% of apps are paid on Google Play Store, so we can say that Most of the apps are free on Google Play Store.

3.2 Rating

In the below plot, we plotted the apps Rating



- **<u>Feature Selection</u>**
  In these steps, we used algorithms like the extra tree classifier to check the results of each feature i.e which feature is more important compared to our model and which is of less importance.
  Next, we used Chi2 for categorical features and ANOVA for numerical features to select the best feature which we will be using further in our model.

- **<u>Standardization of features</u>**
  Our main motive through this step was to scale our data into a uniform format that would allow us to utilize the data in a better way while performing fitting and applying different algorithms to it.
  The basic goal was to enforce consistency or uniformity to certain practices or operations within the selected environment.

- **Fitting different models**
  For modelling we tried various classification algorithms like:

1. **Logistic Regression**
2. **SVM Classifier**
3. **Random Forest Classifier**
4. **XGBoost classifier**

- **Tuning the hyperparameters for better accuracy**
  Tuning the hyperparameters of respective algorithms is necessary for getting better accuracy and to avoid overfitting in case of tree based models like Random Forest Classifier and XGBoost classifier.

- **SHAP Values for features**
  We have applied SHAP value plots on the Random Forest model to determine the features that were most important while model building and the features that didn't put much weight on the performance of our model.

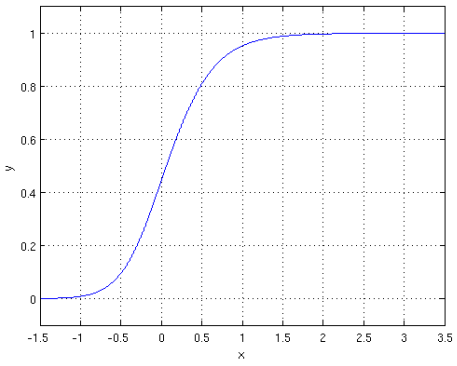
7.1. Algorithms:

1. Logistic Regression:

Logistic Regression is a classification algorithm that was given the name regression because the mathematical formulation is very similar to linear regression.

The function used in Logistic Regression is sigmoid function or the logistic function given by:

f(x)= 1/1+e ^(-x)



The optimization algorithm used is Maximum Log Likelihood. We mostly take log likelihood in Logistic:

$$\ln L(\mathbf{y},\beta)=\ln\prod_{i=1}^{n}f_i(y_i)=\sum_{i=1}^{n}\left[y_i\ln\left(\frac{\pi_i}{1-\pi_i}\right)\right]+\sum_{i=1}^{n}\ln(1-\pi_i)$$

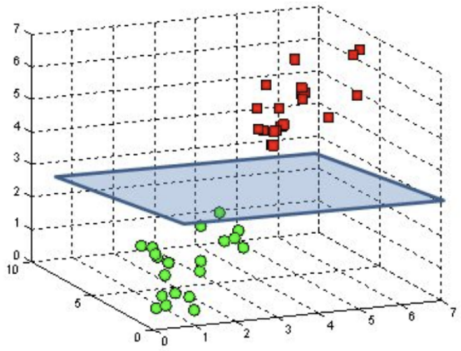
.

2. Support Vector Machine Classifier:

SVM is used mostly when the data cannot be linearly separated by logistic regression and the data has noise. This can be done by separating the data with a hyperplane at a higher-order dimension.
In SVM we use the optimization algorithm as:

$$\begin{aligned}\min_{\xi,w,b}\quad & \frac{1}{2}||w||^2+C\sum_{i=1}^{m}\xi_i\\ \text{s.t}\quad & y^{(i)}\left(w^Tx^{(i)}+b\right)\geq 1-\xi_i,\\ & \xi_i\geq 0;\ i=1,\ldots,m.\end{aligned}$$

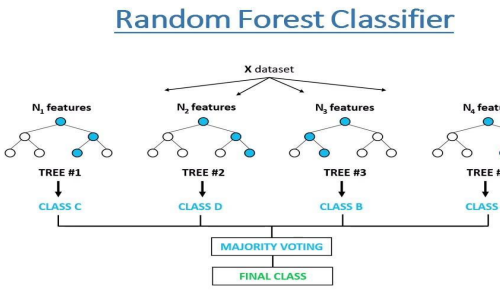
where C is a cost parameter and ξ_i's are slack variables.



We use hinge loss to deal with the noise when the data isn't linearly separable.
Kernel functions can be used to map data to higher dimensions when there is inherent non-linearity.

3. Random Forest Classifier:

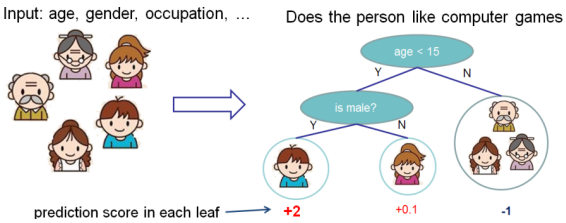
Random Forest is a bagging type of Decision Tree Algorithm that creates several decision trees from a randomly selected subset of the training set, collects the labels from these subsets, and then averages the final prediction depending on the number of times a label has been predicted out of all.



4. XGBoost-

To understand XGBoost we have to know gradient boosting beforehand.

- **Gradient Boosting-**
  Gradient boosted trees consider the special case where the simple model is a decision tree



In this case, there are going to be 2 kinds of parameters P: the weights at each leaf, w, and the number of leaves T in each tree (so that in the above example, T=3 and w=[2, 0.1, -1]).
When building a decision tree, a challenge is to decide how to split a current leaf. For instance, in the above image, how could I add another layer to the (age > 15) leaf? A 'greedy' way to do this is to consider every possible split on the remaining features (so, gender and occupation), and calculate the new loss for each split; you could then pick the tree that most reduces your loss.

XGBoost is one of the fastest implementations of gradient boosting. trees. It does this by

tackling one of the major inefficiencies of gradient boosted trees: considering the potential loss for all possible splits to create a new branch (especially if you consider the case where there are thousands of features, and therefore thousands of possible splits). XGBoost tackles this inefficiency by looking at the distribution of features across all data points in a leaf and using this information to reduce the search space of possible feature splits.

7.2. Model performance:

Model can be evaluated by various metrics such as:

1. **Confusion Matrix-**
The confusion matrix is a table that summarizes how successful the classification model is at predicting examples belonging to various classes. One axis of the confusion matrix is the label that the model predicted, and the other axis is the actual label.

2. **Precision/Recall-**
Precision is the ratio of correct positive predictions to the overall number of positive predictions : TP/TP+FP
Recall is the ratio of correct positive predictions to the overall number of positive examples in the set: TP/FN+TP

3. **Accuracy-**
Accuracy is given by the number of correctly classified examples divided by the total number
of classified examples. In terms of the confusion matrix, it is given by: TP+TN/TP+TN+FP+FN

4. **Area under ROC Curve(AUC)-**
ROC curves use a combination of the true positive rate (the proportion of positive examples predicted correctly, defined exactly as recall) and the false positive rate (the proportion of negative examples predicted incorrectly) to build up a summary picture of the classification performance.

7.3. Hyperparameter tuning:

Hyperparameters are sets of information that are used to control the way of learning an algorithm. Their definitions impact parameters of the models, seen as a way of learning, change from the new hyperparameters. This set of values affects performance, stability and interpretation of a model. Each algorithm requires a specific hyperparameters grid that can be adjusted according to the business problem. Hyperparameters alter the way a model learns to trigger this training algorithm after parameters to generate outputs.

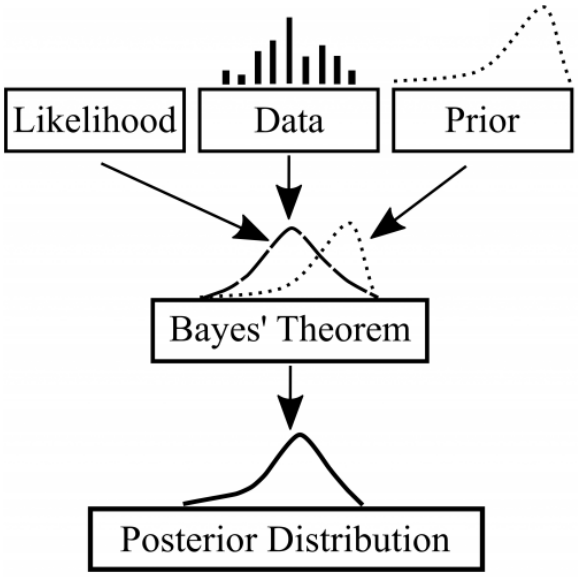
We used Grid Search CV, Randomized Search CV, and Bayesian Optimization for hyperparameter tuning. This also results in cross-validation and in our case we divided the dataset into different folds. The best

performance improvement among the three was by Bayesian Optimization.

1. **Grid Search CV-**Grid Search combines a selection of hyperparameters established by the scientist and runs through all of them to evaluate the model's performance. Its advantage is that it is a simple technique that will go through all the programmed combinations. The biggest disadvantage is that it traverses a specific region of the parameter space and cannot understand which movement or which region of the space is important to optimize the model.

2. **Randomized Search CV-** In Random Search, the hyperparameters are chosen at random within a range of values that they can assume. The advantage of this method is that there is a greater chance of finding regions of the cost minimization space with more suitable hyperparameters since the choice for each iteration is random. The disadvantage of this method is that the combination of hyperparameters is beyond the scientist's control

3. **Bayesian Optimization-** Bayesian Hyperparameter optimization is a very efficient and interesting way to find good hyperparameters. In this approach, in naive interpretation way is to use a support model to find the best hyperparameters.A hyperparameter optimization process based on a probabilistic model, often the Gaussian Process, will be used to find data from data observed in the later distribution of the performance of the given models or set of tested hyperparameters.



As it is a Bayesian process at each iteration, the distribution of the model's performance in the relation to the hyperparameters used is evaluated and a new probability distribution is generated. With this distribution, it is possible to make a more appropriate choice of the set of values that we will use so that our algorithm learns in the best possible way.

8. Conclusion:

That's it! We reached the end of our exercise.
Starting with loading the data so far we have done EDA , null values treatment, encoding of categorical columns, feature selection and then model building.
In all of these models, our accuracy revolves in the range of 70 to 74%.
And there is no such improvement in accuracy score even after hyperparameter tuning.
So the accuracy of our best model is 73% which can be said to be good for this large dataset. This performance could be due to various reasons like no proper pattern of data, too much data, and not enough relevant features.

References-

1. MachineLearningMastery
2. GeeksforGeeks
3. Analytics Vidhya