Predicting A Song's Popularity Using Spotify Data

Mohd Arifullah
School of Science & Technology
City University of London
London, England, United Kingdom
Mohd.Arifullah@city.ac.uk

ABSTRACT: In this paper, we explore how machine learning models can be used to predict the popularity of music songs. For our study, we use Spotify data, in which two sets are included: one containing music features, the other external factors concerning artists. Based on this general approach, we sought to identify the subtle interactions between different track shares and relative listener interest. We use these two data sets to analyze patterns and relationships with the highest effect on track success from the machine learning model, in addition to gaining a new insight into the inner workings of this popular music. This study will provide additional information for musicians, producers, or anyone else connected in some way to that ever-changing music industry.

I. INTRODUCTION

Spotify is a leader in digital music streaming, changing how we enjoy music. It offers valuable data to predict song's popularity. With a huge library of over 70 million songs across all styles, Spotify is a treasure trove for machine learning models. They want to find out what songs will charm listeners the most. The music platform's vast library and clever features, like tailored playlists, are made by cutting-edge algorithms. They watch what users are doing and how they listen to music. These algorithms look closely at many song details - track length, how good it is to dance to, its energy, key, volume, mode, and more. All these play a key role in guessing if a song will be a hit.

In addition to these musical elements, artist-related factors like followers, artists name, and popularity play a significant role in a song's success. Spotify's platform, supporting both renowned artists and independent musicians, provides a diverse range of data reflecting both emerging and established musical trends.

By harnessing these detailed features within machine learning models, we can predict the likelihood of a song becoming popular. The issue is crucial for up-and-coming and independent artists who wish to increase their chances of success by comprehending the essential elements that lead to a song's popularity [1]. This approach not only benefits Spotify and artists in strategizing their releases but also enhances the user experience by refining music recommendations and highlighting potential hits. This is an interesting field for machine learning research since it combines artist-related data from Spotify with specific song attributes to create a comprehensive framework for accurately forecasting track popularity.

II. DATA ANALYSIS, QUESTIONS, DOMAIN & PLAN

A. Data and Domain

We will be using data from the Kaggle repository in this study. The information was extracted from Spotify and is divided into two datasets: one for tracks, which includes attributes like loudness, duration, speechiness, mode, energy, key, instrumentality, liveliness, etc., and another for artists, which includes information like name, popularity, and followers. This data is originally from Spotify and is used by several researchers for making various predictions. Both the datasets contain some missing and NaN values which would be removed later. Artists id was the common attribute we have in the datasets, so we used it to merge them. After merging the datasets, we have around 300000 rows with some missing values and outliers. There were few categorical data such as artist's name, genres which were important features while predicting the song's popularity. The Fig. 1 Pie chart below shows the distribution of songs popularity. The data seemed to be imbalance. We will handle this further.

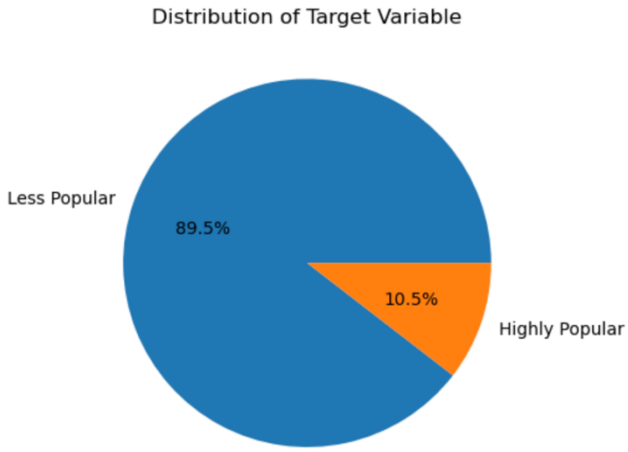

Figure. 1 Distribution of Highly Popular and Less Popular

B. Research Questions

The aim of this study is to predict the success of a song. To analyze this, we will explore the existing songs data and observe the different factors that are affecting the popularity of songs. Below are the few questions we will explore in our scope.

- How do audio features influence the popularity of a song?
- What role do artist-related factors play in predicting a song's popularity?
- How do temporal factors (like release time or season) influence song popularity?

- What is the relationship between energy levels in a song and its popularity?

- How the features of highly popular songs are different from the less popular ones?

The dataset sourced from Spotify is exceptionally apt for the analysis and prediction of song popularity, encompassing a diverse array of audio features such as danceability, energy, loudness, instrumentalness, speechiness, acousticness, liveness, time_signature, valence, and tempo. These variables are integral in assessing the inherent qualities of a song, including rhythm, mood, and style, directly pertinent to our research inquiry. Additionally, artist-centric factors like artist's followers, artists_name, and artists_popularity are included, offering critical insights into the influence of an artist's recognition and market presence on the success of their music, thereby addressing key aspects of our research on the determinants of song popularity.

C. Analysis Plan

The main aim of this study is to explore the songs features and find the success factors in the song's industry by analyzing it visually and creating a predictive model to extract insights for decision making, for this the following approach will be used.

1. Importing both the datasets from Kaggle repository
2. Merging the datasets on the common attribute, cleaning, formatting, removing outliers.
3. Understanding the different insights from visual analytics.
4. Doing Feature engineering to select the relevant features.
5. Develop the predictive model.
6. Comparing the results and conclusion.

III. Analysis

A. Data Preparation

We did some initial changes to the datasets by renaming few columns like id, name as these were common in both datasets. Then we further merged the datasets.

1) Merging the datasets

Initially the datasets contains some missing and nan values. The two datasets used were tracks.csv which contains tracks related data and the other one was artists which was related to details of a specific artist. We merged the datasets with the artists id as it was the common attribute in both the datasets, and we observed that the merged dataset now contains some missing values, nan, and duplicates rows too.

2) Data Processing & Handling Missing Values

The merged datasets contain around 5170 duplicate rows and some Nan values around 6,000. For numeric features such as artists followers, artists popularity, we fill the missing values with mean of the columns. The artists name columns contain some missing values which we will remove them. Moving further there were some outliers as well as they can be visualize using the boxplot (Fig 2), so we remove them as well. For removing outliers, we have the IQR method for removing the outliers. A quartile, in turn, is any of the three values that divides a sorted data set into four parts of equal size. The first quartile Q1 marks the end of the first 25% of data, the third quartile Q3 the beginning of the last 25%. Every value being smaller than Q1–1.5 IQR or larger than Q3+ 1.5 IQR is regarded as outlier [2].

As from the Fig 1 used above depicts that the data is imbalance with 89.5 % negative class and only 10.5% positive, to counter this we will be using SMOTE (Synthetic Minority Oversampling Technique). In general, there are two methods for addressing imbalanced data: oversampling and under-sampling. An oversampling algorithm with great classification accuracy is SMOTE [3].

Moving further, our dataset is splitted into 80:20 ratio for training and testing purpose. Our model will be evaluated on the 20% dataset which will unseen data during the model training.

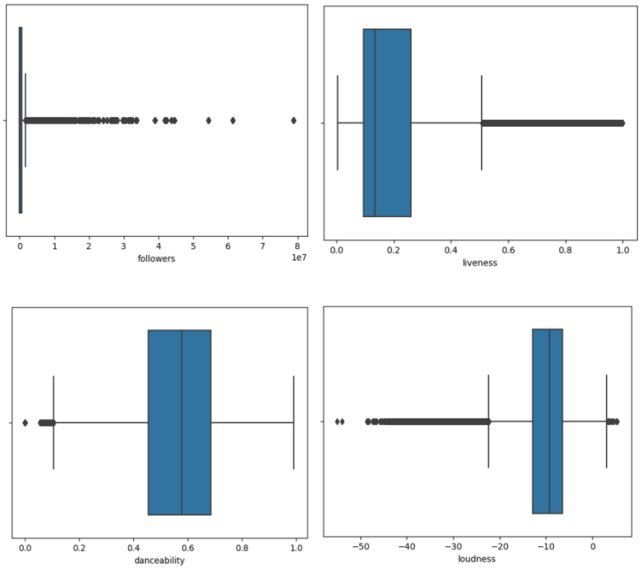

Figure. 2 Boxplot for outliers in some numeric features

3) Feature Engineering

We did some feature engineering by selecting the required columns which were relevant to our prediction such as picking the audio related features such as loudness, key, mode, energy, duration, name of artist, artist followers and popularity. We removed few columns as well such as id, artist's id as they were irrelevant to our objective.

Moving on further, the target variable (track_popularity) contains continuous values in range [0-100], we categorized it by setting a threshold value of 50, that means if the song's popularity is below 50 means it is less popular and if it is equal to or greater than 50 it is highly popular. As we will be using classifier algorithms for the predicting the song's popularity.

B. Data Derivation (Label Encoding)

Our dataset contains one categorical column which is artist. Name, as it is an important feature for our prediction. To encode this, we have used Label Encoding. Label encoding simply converts each value in a textual column into a number [4].

C. Predictive Model Construction

For predicting the success of a song, we have used two supervised machine learning algorithms.

A) Random Forest

We have used Random Forest algorithm for predicting determining the song's success. Random Forest is a robust ML algorithm that can be used for a variety of tasks including regression and classification. It is an ensemble method, meaning that this model is made up of a large number of small decision trees, called estimators, which each produce their own predictions. These results are then combined to produce a more accurate prediction [5].

SMOTE pipelining was used to balance the dataset after splitting the data into train and test and then the model was trained with the RFX_train and RFY_train data. The trained model is extracted from the pipeline for the predictions. Initially we have used around 100 trees (n_estimators) with the gini's splitting criteria and using default min_samples_split.

B) Logistic Regression

Logistic Regression model was also used to predict the song's popularity using the same SMOTE pipeline technique. Linear Regression is a commonly used classification algorithm when the target variable is categorical. This method aims to create a correlation between the features and the desirable output. Logistic Regression model uses a logistic function to determine a binary dependent variable in its main form [6]. We have used the "**lasso**" parameters which is a regularization technique with the model.

D. Validation of results

The Table 1 describes the summary of the model's performance in terms of accuracy, RUC, F1 Score and Precision. In comparing the performance metrics of Logistic Regression and Random Forest models on a binary classification task, it is evident that Random Forest outperforms Logistic Regression across multiple criteria. Random Forest achieves a higher accuracy of 0.85 compared to Logistic Regression's accuracy of 0.74, indicating better overall predictive performance. Fig 3 shows that the area under the ROC curve (AUC-ROC) is slightly higher for Random Forest (0.6) than for Logistic Regression (0.64) one in Fig 4, suggesting improved discrimination ability. However, Logistic Regression exhibits higher recall (0.51) compared to Random Forest (0.36), indicating better sensitivity in capturing positive instances. The F1 score, which balances precision and recall, is also higher for Random Forest (0.40) compared to Logistic Regression (0.29), highlighting better overall model effectiveness. Precision, measuring the accuracy of positive predictions, is notably higher for Random Forest (0.36) compared to Logistic Regression (0.21). In summary, Random Forest demonstrates superior performance across key metrics, making it a more favorable choice for this specific classification task. Consideration should be given to the specific goals and trade-offs between precision and recall based on the application's context. Further investigation, including hyperparameter tuning and exploration of alternative algorithms, may enhance model performance.

Feature	Logistic Regression	Random Forest
Accuracy	0.74	0.85
RUC	0.64	0.69
Recall	0.51	0.47
F1 Score	0.29	0.40
Precision	0.21	0.36

Table 1. Model Results

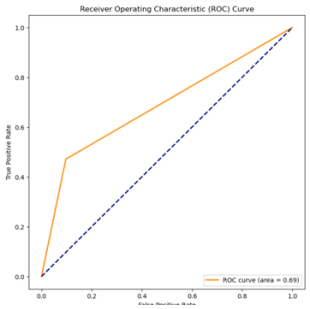

Fig 3. ROC RF Model

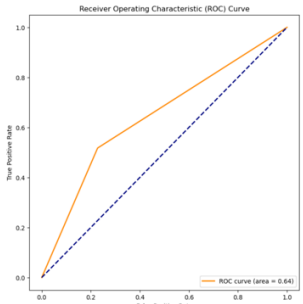

Fig 4. ROC LR Model

IV. Findings, Reflections & Further Work

A. Findings

The Fig 5 below shows the correlation heatmap of the merged dataset which shows that the correlation between the numerical features like loudness, speechiness, artist's follower, popularity and other audio features which are highly correlated to the target variable.

We can observe that the artist's popularity and artist's followers are highly correlated for predicting the success of a song followed by loudness, danceability etc.

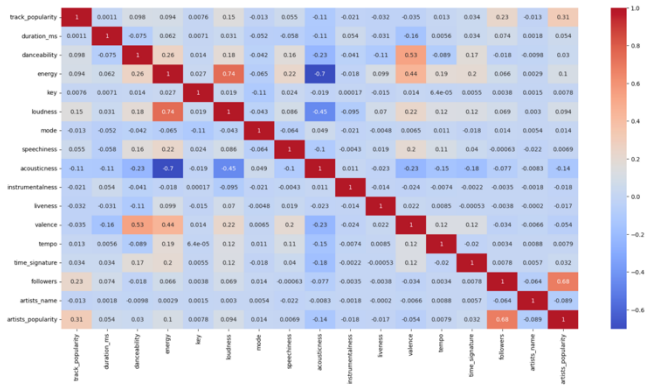

Fig 5. Correlation Heatmap

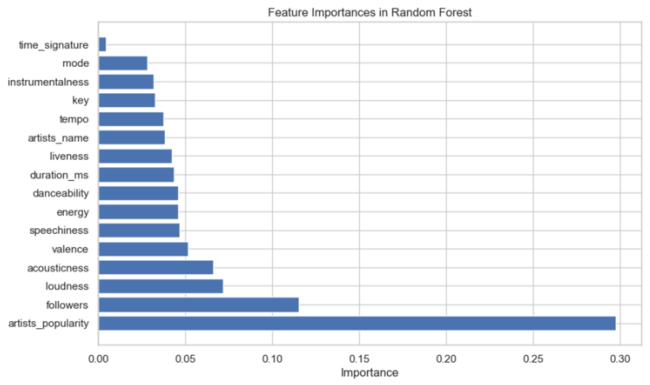

Fig 6. Important Features Graph of RF Model

The Fig 6 shows the important features graph generated by Random Forest Model, which shows 'artists_popularity' is the most significant feature, followed by 'followers' and 'loudness', indicating these have a more substantial influence on the model's predictions. 'acousticness', 'valence', 'speechiness', and 'energy' are in the middle range of importance.

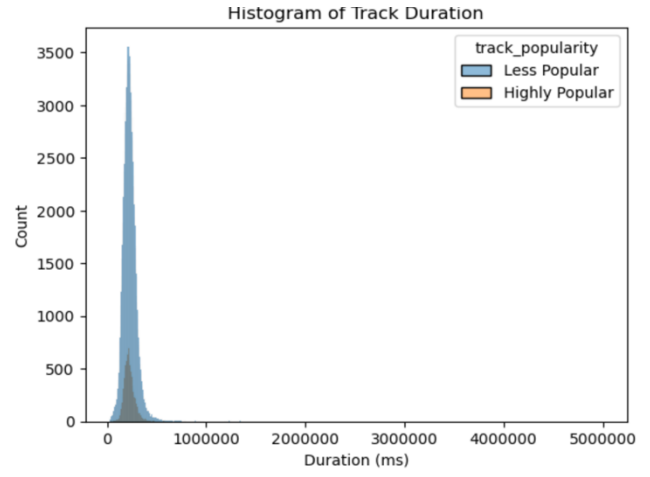

Fig 7. Histogram of Track Duration by Popularity

The Fig 7 shows the histogram for the distribution of track durations for songs as Less Popular and Highly Popular. For both categories of track popularity, the duration distribution is heavily skewed towards the left, indicating that most tracks have shorter durations. The peak for both categories is quite sharp, indicating a strong concentration of tracks around a specific duration which could be around 200,000 to 300,000 milliseconds (3 to 5 minutes), commonly seen in typical song lengths. The overlap of the two categories indicates that the duration alone may not be a distinguishing factor between less popular and highly popular tracks since their distributions appear quite similar.

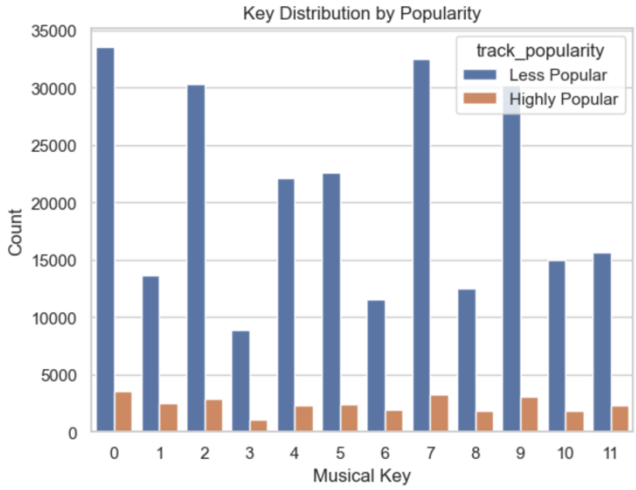

Fig 8. Bar Graph of Key Distribution by Song Popularity

The bar chart Fig 8 depicts the key distribution among tracks classified as Less Popular and Highly Popular. It reveals that certain musical keys are favored across all tracks, with keys 0, 5, 7, and 10 having notably higher counts. The pattern of key distribution is similar between the two popularity categories, suggesting that the choice of key may not be a pivotal factor in determining a track's popularity.

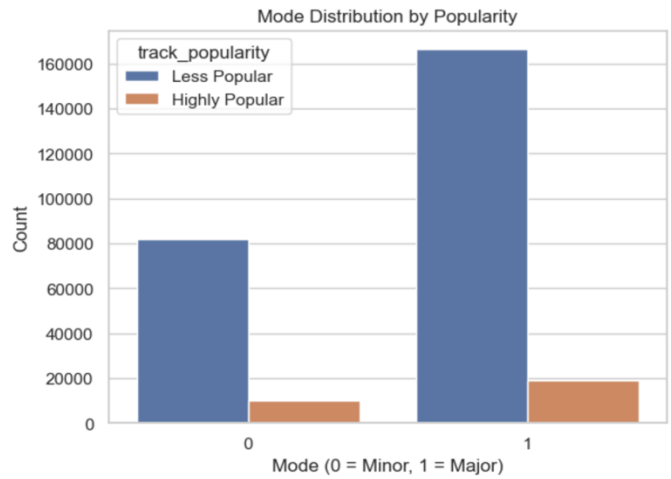

Fig 9. Bar Graph of Mode Distribution by Song Popularity

The bar chart Fig 9 illustrates the mode distribution of tracks categorized by popularity. It shows two modes: Minor (0) and Major (1). The data indicates a significant preference for the Major mode over the Minor mode in the composition of tracks, which is consistent across both "Less Popular" and

"Highly Popular" categories. Major mode tracks are far more prevalent in the dataset for both popularity levels, but this preference is even more pronounced among the "Highly Popular" tracks, this could be due to the imbalance dataset.

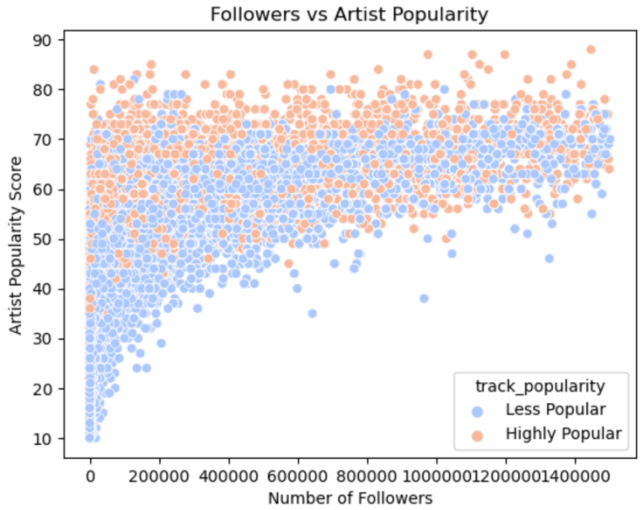

Fig 10. Scatter Plot of Artist's follower by Song Popularity

The scatter plot Fig 10 helps in analyzing the relationship between the number of followers and the artist popularity score with songs categorized as "Less Popular" and "Highly Popular." Both categories are spread across the full range of followers, showing no clear distinction between the number of followers and song's popularity.

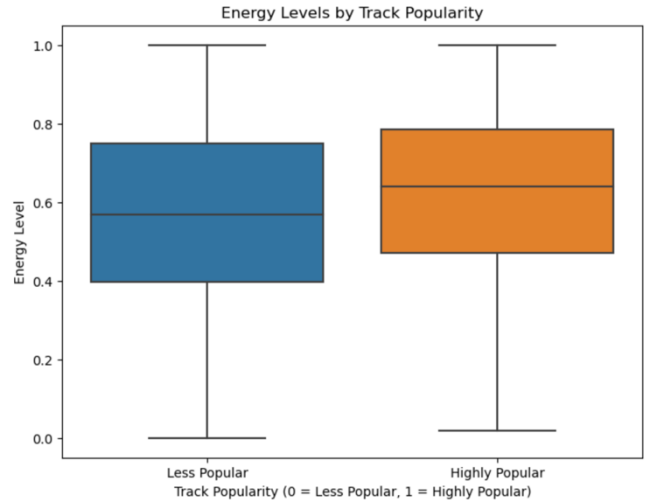

Fig 11. Box Plot of Energy Level by Song Popularity

From the above boxplot Fig 11, we can infer the central tendency and variability of the energy level within each popularity category. The median lines indicate the typical energy level within each category, and the span of the boxes shows how much variation there is around that median.

The study's initial questions are now successfully addressed by the overall analysis results, which also emphasize the significance of several audio-related variables and artist traits that are critical in forecasting a song's popularity.

B. Further Work

The analysis of this study may be improved in the future by adding more data to the feature set as the dataset we used was imbalance so focusing on gathering more accurate data can help in improving the model performance. Also adding additional data features such as genre, user behavior, and social media attitude can be useful in improving the predictive models. More in-depth understanding can be gained by examining user interaction, the influence of societal and industry-specific trends, and the function of Spotify's recommendation engines. The impact of fame on an artist's career path and the financial consequences for the music industry could be monitored through longitudinal research. Another way of improving the model's performance can be done by using Grid Search technique on both the models by using certain hyperparameters tuning.

V. References

[1] Sivasai Bhavanasi, Sahil Malla, V Manichetan , CVNJ Dhanush, Dr B Prakash, "Spotify Data Analysis and Song Popularity Prediction" Volume 5, Issue 5 May 2023, pp: 296-304, IJAEM

[2] K. Benkert, E. Gabriel and M. M. Resch, "Outlier detection in performance data of parallel applications," 2008 IEEE International Symposium on Parallel and Distributed Processing, Miami, FL, USA, 2008, pp. 1-8, doi: 10.1109/IPDPS.2008.4536463.

[3] Yakshit, G. Kaur, V. Kaur, Y. Sharma and V. Bansal, "Analyzing various Machine Learning Algorithms with SMOTE and ADASYN for Image Classification having Imbalanced Data," 2022 IEEE International Conference on Current Development in Engineering and Technology (CCET), Bhopal, India, 2022, pp. 1-7, doi: 10.1109/CCET56606.2022.10080783.

[4] E. Jackson and R. Agrawal, "Performance Evaluation of Different Feature Encoding Schemes on Cybersecurity Logs," 2019 SoutheastCon, Huntsville, AL, USA, 2019, pp. 1-9, doi: 10.1109/SoutheastCon42311.2019.9020560.

[5] Lejla Vardo, Jana Jerkic, Emir Zunic, "Predicting Song Success: Understanding Track Features and Predicting Popularity Using Spotify Data" 2023 IEEE 22nd International Symposium INFOTEH-JAHORINA

[6] Dimolitsas, Ioannis & Kantarelis, Spyridon & Fouka, Afroditi. (2023). SpotHitPy: A Study For ML-Based Song Hit Prediction Using Spotify.

Word Count

Content	Words
Abstract	**135**
Introduction	**269**
Data	**149**
Analytical Questions	**207**
Analysis	**950**
Findings & Future Work	**588**