PREDICTING SPOTIFY'S SONGS POPULARITY, USING SONG'S FEATURES

A. INTRODUCTION.

1.1 ABOUT SPOTIFY.

Spotify is a music streaming app used worldwide. It provides extensive song data on it's API. I will be using this data to create a model that can be used to predict the popularity of new music.

1.2 OBJECTIVES.

I am aiming to create a user-friendly system to help young artists, producers, Djs and music promoters understand what features contribute to high song popularity. The end product am targeting creating an app that can predict the popularity of a new song before it is even released as long as you have the features.

1.3 REFRENCES.

Some have done this study before. It is worth mentioning that the music industry has grown so much. It is not a surprise as music is a critical tool in every day life. I hope to build on what most have done and improve on some that was not covered.

B. ANALYSIS.

2.1 DATA CLEANING

The data obtained from spotify api is not clean, thus not ready for analysis.

First, I check the characteristics of the data; is it numeric? Does it have missing values? Are there repeated songs?

After making observations on the data, I need to act on them. Before we remove the non-numeric columns check weather any contain information that can be useful.

Some of the columns you can work on before removing include artists and country, here I created new columns from these; ie, artist count, market count and other countries featured.

Then some of the columns, is explicit, is Boolean. I turned it into integer. Then I removed duplicates, systematically. I grouped the songs by their spotify's distinct id, then look for the country it had the highest popularity in, pick that and put all other distinct countries in the other countries featured column.

The album release date and snapshot date needed action too, so I created a column, days since release, this takes the difference in days between the two. There after I checked for any missing values, the days since release column had a few missing values, so I replaced them with the column median as it interferes less with the data's skewness.

There after I removed all the columns I needed not to get a clean data for analysis.

2.2 EXPLORATORY ANALYSIS.

Having the clean Data set I went ahead and checked its properties, how the columns data is distributed. I pulled the data's column; mean, median, mode and quadrant information.

Afterwards I identified popularity as the dependent variable and all other numeric features as the independent variables.

I went ahead and checked the features linear correlation with popularity. I found out from both the heat map and scatter plot matrix that popularity has very weak linear correlation with the features.

I also performed both anova and linear regression of the features and popularity to check for feature importance, I discovered, all features had a p-value less than 0.05 bar key and danceability. But since they may have other non-linear relationship to popularity I kept the features.

There I decided to explore advanced machine learning techniques to check for complex relations.

2.3 MACHINE LEARNING RERESSION

Here I sort to employ some of the models that could capture my data best, ie Random Forest(R.F) Xgboost(XGB) and Gradient boosting machines (GBM). These models use the concepts of decision trees to make conclusions and predictions.

To ensure the models are trained and tested on new unseen data I split the data identically into 0.8(training) and 0.2(testing) sets on the training data. I created 5 folds on the training set that would increase the robustness of the models while learning traits of the data.

Results: from the models after running the test set, Xgboost (XGB) model had the highest Rsquared of 0.61, followed closely by Random Forest with 0.60 and GBM with 0.59. Similarly,

XGB had the least root mean squared error (RMSE), followed closely by RF and GBM with the most.

I plotted graphs showing the relationship between the actual popularity score and the predicted popularity scores for all the three models.

C. CONCLUSIONS

What I learned is even though these features account for a substantial part of popularity (0.61), there are other features not included in my dataset that contribute to popularity too.

There could be ways better employed to further reduce the RMSE. This will likely tighten the grip on the relationship between features and popularity, therefore rise the Rsquared.

All the results and plots in; *"J-K-spotify_analysis_all_models_plots"*