Predicting Song Popularity

Sam Xu *samx@stanford.edu*, Joyce Xu *jexu@stanford.edu*, and Eric Tang *etang21@stanford.edu*

***Abstract*—This project aims to predict the popularity of songs by analysing acoustic features such as tempo, duration, mode, loudness, key; artist information such as location and popularity; and metadata such as release title and year. The codebase for the project is located at `https://github.com/inSam/SoundScorer221`**

I. Introduction

The goal of this project is to predict the popularity of a song by analyzing its audio features and metadata. Such a tool would be valuable for record labels, streaming services, and average consumers; it would also help the research community understand what acoustic features are currently popular with the public. In this progress report, we would like to expand on (1) our overall strategy for dataset and feature extraction (2) details about our regression algorithms (3) next steps we plan to take to finish up our project.

II. Methods

A. Dataset

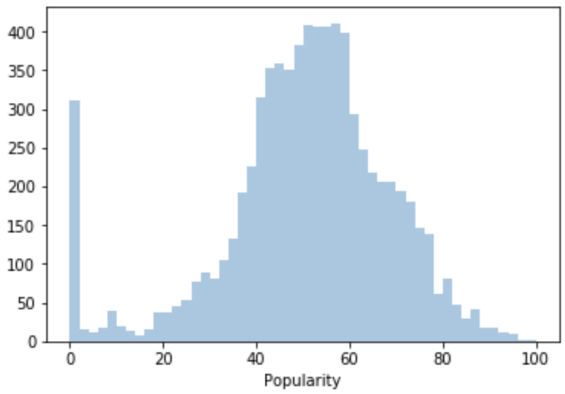
Our dataset comprises 7,473 songs extracted from the Spotify API, which provides acoustic features, author information, song metadata, and a popularity score. Spotify's popularity score ranges from 0 (least popular) to 100 (most popular), and is determined as a function of number of plays on Spotify and the recency of those plays.

We pulled these songs from Spotify's 1,000 top albums released in 2018, which gives us a range of songs across popularity, genre and acoustic features. In addition, limiting our analysis to songs released in 2018 partially controls for the effect of age on popularity.

Note to the reader: Our original sampling of songs, described in the project proposal, turned out to be highly imbalanced, skewing heavily towards popular songs over unpopular songs. To remedy this, we changed our data sample from the top 10,000 songs to all songs from the top 1,000 albums. This results in a much more balanced dataset. We've re-run our original baseline tests on our new dataset, and we report these figures throughout the paper.

B. Data Exploration

Below, we plot the distribution of song popularity in our dataset.



C. Feature Selection

1) Baseline Features: We begin with seven audio features extracted by The Echo Nest API, which is integrated into the Spotify API. The Echo Nest uses proprietary audio analysis algorithms to extract high-level features from audio files. The seven features we used are "acousticness", "danceability", "energy", "loudness", "speechiness", "tempo", "valence". These audio features are extracted directly from the audio files of each song. Unfortunately, The Echo Nest's algorithms are propietary, and we are unable to access how they extracted these features.

2) Additional Features: We plan to add two more categories of features: further audio features, and song/artist metadata.

D. State Vector Regression

As our first algorithm, we chose to to leverage non-linear Support Vector Regression (SVR). SVR applies hinge loss and non-linear kernels to regression problems, performing regressions on a dataset and generating non-linear contours. We feed our baseline features and additional features into our SVR, with results reported below.

E. Random Forest

For our second classical algorithm, we will train a Random Forest Regressor on our features. Random Forest Regressors are extremely similar to Random Forest Classifiers, also using a forest of decision trees. In our case, the Random Forest classifier should give us a reasonably strong and interpretable model for use on our features.

III. Next Steps

A. Improve Features

We will extract more features from the Spotify database such as song age, location, audio sections, and signature. We

will also try to construct more advanced features from historical data such as musical trends. Using Spotify's provided links to song previews, we may be able to extract more advanced features from the audio recordings of each song. This step will involve a lot more feature tuning and experimentation.

B. Improve Models

For future models, we will try to tackle this problem by using ensemble methods such as gradient boosting regressors and random forests, support vector regressors, and neural networks to predict popularity. In particular, neural networks may be able to extract more meaningful features from the audio tracks of each song.

IV. Challenges

One challenge is to consider the effects of time on the popularity on the song. We may be able to tackle this by simply adding the age of the song as a additional feature.

Another challenge is that there is plenty of noise that influences a song's popularity, beyond its mere auditory features. While we have assumed a 100% success rate with our Oracle, this may be unrealistic to achieve with a classifier.

References

[1] Myra Interiano, Kamyar Kazemi, Lijia Wang, Jienian Yang, Zhaoxia Yu, Natalia L. Komarova, *Musical trends and predictability of success in contemporary songs in and out of the top charts* Royal Society Open Sciences, 2006.

[2] James Pham, Edrick Kyauk, Edwin Park, *Predicting Song Popularity* Stanford University CS 229, 2015.

[3] `https://developer.spotify.com/documentation/web-api/`

[4] Li-Chia Yang, Szu-Yu Chou, Jen-Yu Liu, Yi-Hsuan Yang, Yi-An Chen, *REVISITING THE PROBLEM OF AUDIO-BASED HIT SONG PREDICTION USING CONVOLUTIONAL NEURAL NETWORKS* Academia Sinica, Taiwan, 2017.