Final Project: Song Popularity Prediction

Ashwini Marathe

12/8/2019

Summary

This project aims to understand the features that *hit* songs have in common. In particular, I was interested in knowing if a song's artist, the genre of the song and, its other musical features like danceability, energy, loudness among others can help distinguish *hit* and *non-hit* songs. To achieve this, I trained a logitistic regression model and compared the results of this model with the non-parametric classification method, random forests. Since the logistic regression model is more interpretable, I have drawn inferences from it. From the model, it can inferred that the popularity of a song's artist and the song genre does highly influence the odds of a song getting *hit*. Another influential music feature is danceability. Songs that are more danceable have higher odds of becoming *hit*.

Introduction

The Music industry is a multi-Billion dollar market. In 2018, the collections of the music industry were around $9.8 Billion while the top 10 artist contributed a whooping $886 Million. As with many cash rich markets, some previous work has already been done in predicting popularity of songs based on the song lyrics using Natural Language Processing techniques. However, I was interested to know the infuence of musical features on the success of songs. Additionally, I wanted to answer questions like are some song genres more popular than others? Do songs of famous artists tend to be more popular? To answer these questions, I used data from one of the leading media service platforms: Spotify. Spotify scores the popularity of songs based on the total number of plays of a track. In particular, I have tried to predict if a song will make it to the top 20% of the popularity bracket (*hit* songs). The predictors used in the prediction include 11 music related features (described in Table 1), artist popularity, the song genre, month of release. Such inferences might be of interest for the artists as it gives deeeper analysis into the effect of song loudness, danceability and other features and can help artists emphasize on certain music features to increase odds of commercial popularity.

Data

In order to do an analysis, I needed music data with music related features, artist popularity, song_genre and, other metadata of the songs. Spotify's music data that it provides via its developer APIs seemed like the right place to start to create this dataset.

1. Data Collection

Spotify provides access to a subset of its database for the developer community using web APIs. Songs can be downloaded based on particular artist, albums or song ids. Since, I wanted the data to be as diverse as possible, I selected 14 genres and collected song ids from these genres for the year 2018. I chose the year 2018 as the popularity of songs released recently (2019) might not be very accurate given that they didn't get enough time to gain popularity. Even for the songs that came at the end of 2018 got around 11 months to gain popularity. A total of 47,094 song ids were collected. For all these song ids, song features were downloaded, a description of which is given in table below.

	Feature	Description
1	acousticness	A confidence measure from 0.0 to 1.0 of whether the track is acoustic. 1.0 represents high confidence the track is acoustic.
2	danceability	Danceability describes how suitable a track is for dancing based on a combination of musical elements including tempo, rhythm stability, beat strength, and overall regularity.
3	duration_ms	The duration of the track in milliseconds.
4	energy	represents a perceptual measure of intensity and activity.
5	instrumentalness	whether a track contains no vocals. Ooh and aah sounds are treated as instrumental in this context. Rap or spoken word tracks are clearly vocal
6	key	The key the track is in. Integers map to pitches using standard Pitch Class notation.
7	liveness	Detects the presence of an audience in the recording.
8	loudness	The overall loudness of a track in decibels (dB).
9	mode	Mode indicates the modality (major or minor) of a track, the type of scale from which its melodic content is derived.
10	speechiness	Speechiness detects the presence of spoken words in a track.
11	tempo	The overall estimated tempo of a track in beats per minute (BPM).

In addition to the above variables, the *artist*, *artist popularity*, *song_popularity* and, *song_genre* was appended to the dataset.

Using the song features, the aim of the project is to predict if a song will make it to the top 20% (hit category). The *song_popularity* score from Spotify is a number between 0 to 100. Only 19% songs in the data had a popularity score of above 50. As a result, I used 50 as a threshold and labelled songs with popularity greater than or equal to 50 as *Hit* and those with popularity less than 50 as *not hit*.

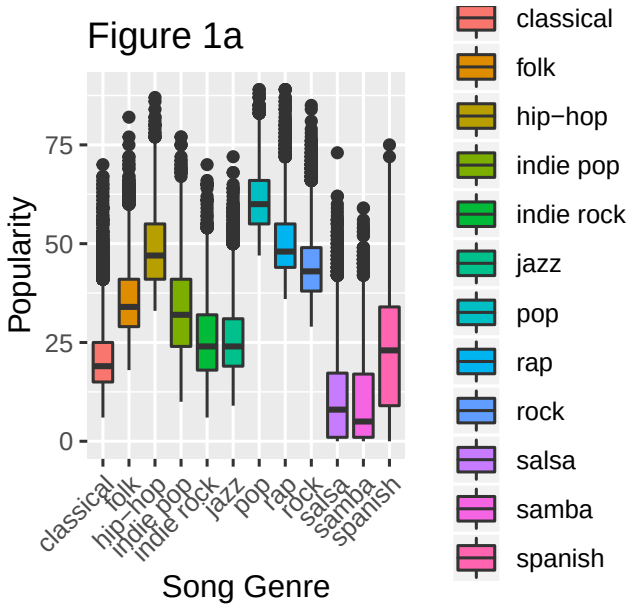
2. Data Cleaning

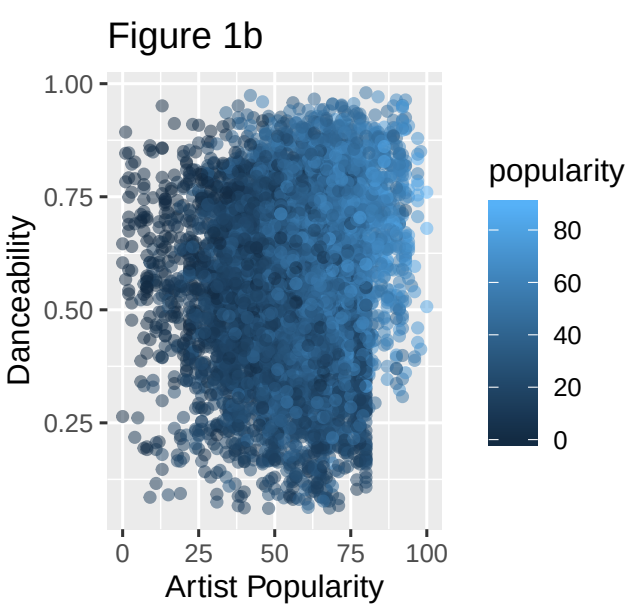
- Around 6000 rows in the data had a non-unique song name and artist name pair, yet with a unique Spotify song id. The song features for such songs were similar but, the popularity score was different. Since these rows looked ambiguous, they were discarded from the data resulting into data with 40744 rows.
- FOr the *song_genre romance* and *progressive metal* none of the songs had *song popularity* greater than 50 and hence no *hit* songs. There might exist songs with higher *song popularity*, but were not downloaded due to Spotify's download limit. But having these genrs in data led to high standard error and songs from these genres were dropped
- To analyze the effect of month and day of release of a song on its popularity, I created a month and day variable. However, 14 entries had release date precision in years. For these entries, the month and day column were marked NA. All the continous variables (artist popularity, acousticness, danceability, energy, instrumentalness, loudness, valence, speechiness, tempo) were standardized so that all the variables are on a comaparable scale.

3. Exploratory Data Analysis

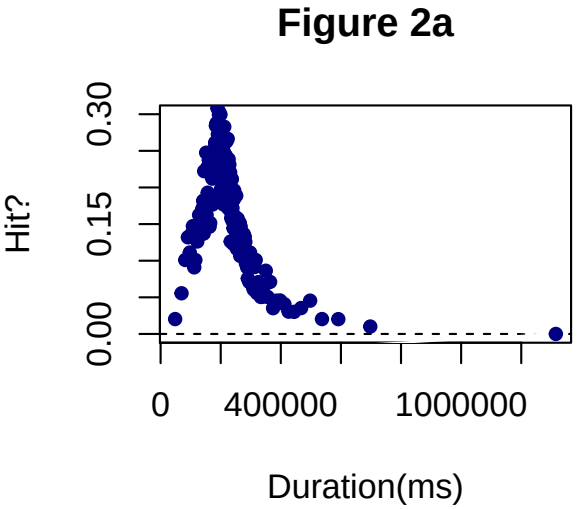
Various graphs were plotted to visually analyze the effect of different predictors on the odds of becoming *hit*.

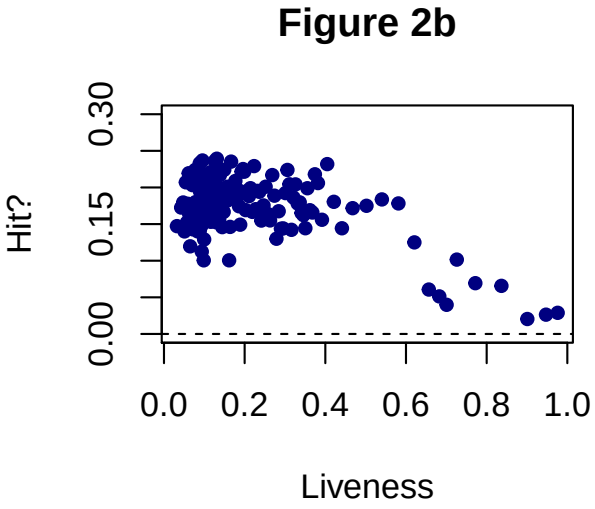
- From the figure 1a, we can see that the *song_popularity* (0-100) of songs varies a lot by its *song_genre*. Pop songs have a much higher popularity than songs from progressive metal genre. There are few outliers with high values of popularity for almost all genres, since all genres have a atleast a few hit songs.
- In figure 1b, we can see that as *artist popularity* and *danceability* increases, the popularity of the song increases. As a result we create a new variable which is the product of danceability of the song and the artist popularity.





- In figure 2a, we can see that the average value of *hit* variable has an increasing trend followed by decreasing trend for the *duration* of the song. This indicates that very short and very long songs tend to be less popular. Hence, this continous variable was binned into three classes: [0-200s), [200s,400s), [400s,Inf).
- From figure 2b, it can be seen that for *liveness* value less than 0.5, the chances of being *hit* are higher than the points with *liveness* greater than 0.5. This indicates that people don't prefer live audience in songs. Hence, liveness variable was also binned into two classes: [0,0.5), [0.5,1].





- There didn't seem to be any influence of *key* on the outcome variable. The Chisquare tests also confirmed this observation.
- Songs released on Friday seemed to have higher popularity overall from visual analysis. (plot in Appendix). I had expected a variation in popularity based on *month*, for example higher popularity for songs released during holidays or summer. However, popularity varies very less by *month*.

- Also, there seemed to be interaction between *song_genre* and *loudness* fom the EDA. Overall trend for *loudness* was increasing, indicating louder songs tend to be more popular. However, the contribution of *loudness* can vary based on the *song_genre*. For example, classical songs can be popular without being very loud, which might not be the case for rock music.
- For the variable *instrumentalness*, the values were concentrated either near zero or 0.9. Hence the variable was split into two categories, instrumental, non-instrumental. Visual analysis indicated that instrumental songs were less popular.

Model Selection

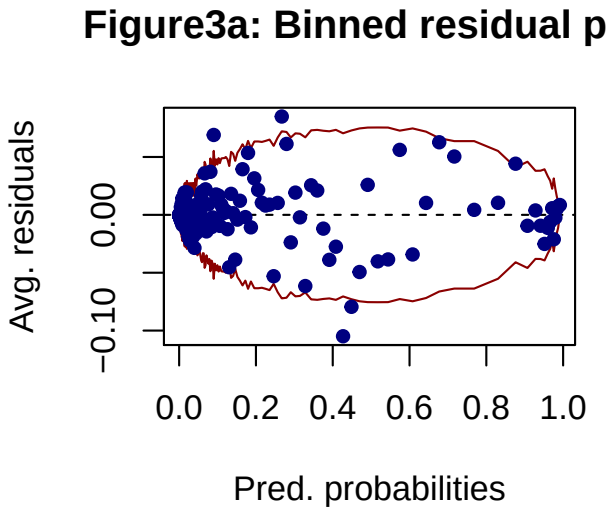
I trained two models for the binary classification task and compared the classification results.

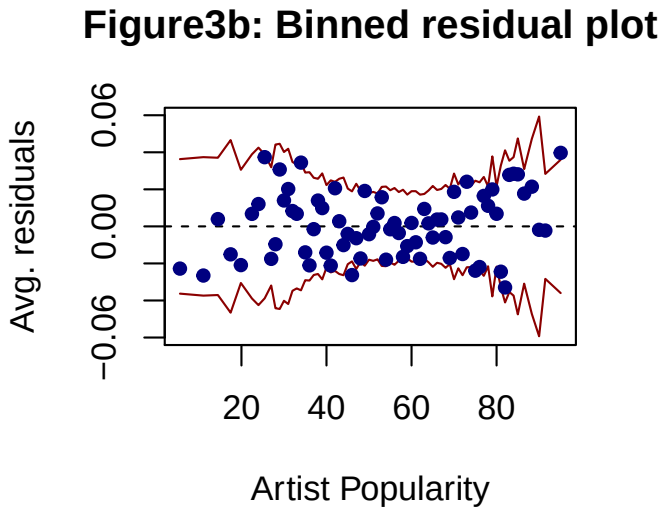
Logistic Regression

I used Logistic Regression method as it is a binary classification problem. For both the models, the data was split into train and test sets (80%-20% split).

For the logistic regression model, I included all the variables described in the data section as well a the interaction term and used the step wise AIC method to select the relevant variables. As observed in the EDA, *key* and *month* were dropped from the model. *instrumentalness* and *valence* were also dropped in the step-wise selection process. The final model used is described below:

$$hit \sim song\ genre + acousticness + danceability + energy + liveness + loudness + speechiness + mode + tempo + artist\ popularity + duration\ ms + song\ genre : loudness + day$$





Figures 3a and 3b show the residual plots for the logistic regression models. In figure 3a, a few points lie outside the error bounds but majority of the points lie within the error bounds. For the residuals vs artist_popularity plot only three points lie outside the error bounds. There seems to be no discrenable pattern in the residual vs. fitted plot. However, for the residuals vs. Artist Popularity plot there seems to be some pattern in the residuals. For higher values of artist popularity the residuals average tends to be higher. Thus, there exists some relationship not captured by the model. Residual plots against other variables did not have any discernible pattern.

Since the data was imbalanced (81% negatives, 19% positives) and there was no relative emphasis on False positives and False negatives I optimized the threshold parameter to maximize the F1-score.

Table 1: Train Dataset

	Model	Acc.	AUC	Sensitivity	Specificity	PPV	F1
1	Logistic Regression	90.5%	0.938	0.0.70	0.94	0.71	0.70
2	Random Forest	95%	0.986	0.87	0.96	0.83	0.85

Table 2: Test Dataset

	Model	Acc	AUC	Sensitivity	Specificity	PPV	F1
1	Logistic Regression	89.83%	0.93	0.69	0.94	0.71	0.71
2	Random Forest	94.9%	0.985	0.86	0.96	0.83	0.84

I also trained a random forest model as it is a non-paramteric method and can capture relationships not included by the logistic regression model. For the random forest, I used 1000 trees and the other parameters were selected using grid seach in the paramter space. The maximum depth upto which the trees were allowed to grow was 20 and split criterion used was Gini Index. The number of features for the best split were restricted to log2(n) which is 4 in this case. The table below summarizes the accuracy, AUC, sensitivity, specificity, Positive Predictive Value and F1-score for the train and test data.

Conclusions

The purpose of the project was to infer which features influnece wether a song will be *hit* or not. According to the Logistic Regression model:

- Artist popularity is the most important factor influencing the odds of song being *hit*. Increase in *artist popularity* by 1 level (on a scale of 0 to 100) increases the odds of being a *hit* by 7.6%
- Genre too influences the odds of a song being *hit*. For example hip-hop songs have 48% higher odds of being *hit* than classical songs, whereas pop music has 16% higher odds of becoming *hit*
- If the danceability of the song increases by 0.1 (on a sclae of 0 to 1) the odds of being *hit* increase by 10%
- Also, odd sof being *hit* decrease if speechiness increases and surprisingly, songs released on Wednesday have highest odds of being *hit*

Limitations

- Spotify has a maximum cap of 10k songs download in one category. The downloaded data might be ordered by some variable but is not mentioned in the API documentation. As a result the data might be biased.
- Downloading data is a very time consuming process. The API tokens have small expiry and maximum of 50 requests can be done at once. This led to smaller dataset
- Multiple Spotify Id for the same songs exist leading to erronous data

Future Work

- Better and accurate models can be built using larger dataset
- Analysis of song lyrics can be included