Spotify Music Popularity Prediction

Spotlight: Xinyu Mei, Yiran Lin, Yuxiao Zhou, Yuxin Shang
xm2256, yl4600, yz3913, ys3386

1. Introduction

Spotify is a Swedish-based music streaming platform. It now has 286M active users, and 45% of them have premium memberships, which contribute to 91% of revenues and 100% of gross profits in 2019. [1] Spotify, therefore, prioritizes user experience optimization to enlarge the premium user group and enhance user loyalty.

This project evaluates key factors to users' decisions on music selection, such as music audio features and text information accessible during the user journey. We use these factors and machine learning methods to predict music popularity. The dataset is from Kaggle with more than 170,000 music data. We believe our analysis will help Spotify make informed decisions regarding new album procurement, music copyright renewal, and music recommendations for users.

2. Data

2.1 Data Preprocessing

Our dataset is from Kaggle and contains data of over 170,000 songs.

Variable Selection

We keep variables that may have impacts on music popularities and are favorable to our analysis. String variables such as music id and artists information are removed from our dataset. The variable that indicates whether the song is child-friendly is removed given the demographic feature of Spotify users.

Missing Values

Rows with missing values are removed because they consist of a relatively small proportion of our dataset and may harm our analysis.

Unit conversion

We convert music duration from milliseconds to minutes for the convenience of our following analysis.

Correlation checking

We plot the heatmap below to illustrate the correlation between all the selected variables. Most correlation coefficients are within reasonable ranges.

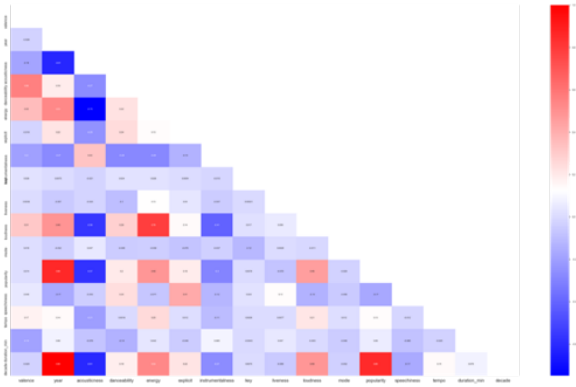

Figure I. Correlation Heatmap

2.2 Exploratory Data Analysis

We first probe into the distribution of music popularity. It is a discrete variable and ranges from 0 to 100. The majority of music has popularity ranging from 0 to 10. We classify music popularity by its median value of 33. Music with popularity above 33 is labeled as *Popular* and vice versa.

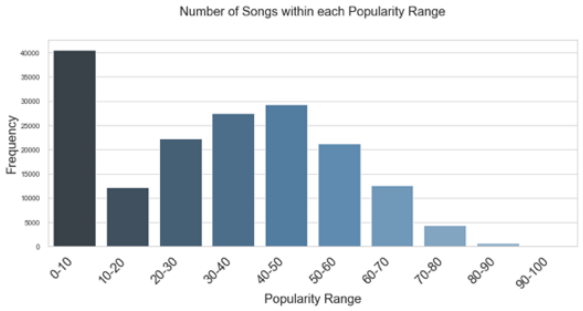

Figure II. Music Popularity Distribution

Next, we find that audio features have different impacts on music popularity. *Danceability, energy,* and *explicit* are positively correlated to music popularity but *acousticness* and *speechiness* are negatively related to it. *Loudness* has a neutral effect on popularity based on the boxplot. In Figure III, we classify these audio features by their mean levels.

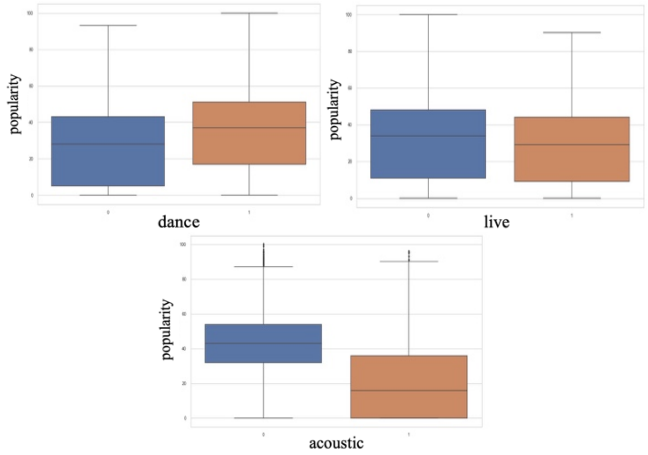

Figure III. Music Popularity of Music Audio Features

We find some interesting trends in music history. In the word cloud for each decade (see Appendix 5), we notice that *love* is the major music topic since the 1950s; there is a shift in popular music types from classical music to pop songs. However, a significant number of classical music were remastered during the second half of the 20th century; in the recent 20 years, music producers derive their creative passion mainly from their current life, and you will find many words such as *young, time, single,* and *future* in their works. Trivial words such as feat and remix, along with words with less than three letters, are removed because they are irrelevant.

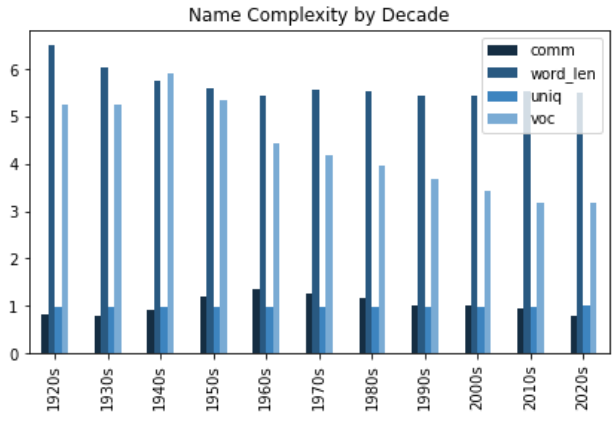

Figure IV. Music Name Complexity Changes

Figure IV presents the word complexity of music names by decade. We find that the common word frequency peaks in the 1960s; music producers prefer shorter, fewer words for music names since the average word length and the average word number are both decreasing.

3. Music Feature

3.1 Audio feature

We use two methods to identify the audio features of Spotify music. First, we focus on musical traits such as *key, mode, tempo, and duration*. Second, we follow features (*acousticness, danceability, energy, liveness, loudness, instrumentalness, speechiness, valence*).

3.2 Song name feature & Text Mining

Music names are the first and one of the main text information that is accessible to Spotify users. To evaluate users' perceptions of music names, we use text mining methods to identify music names' natural language features and emotion & sentiments features.

3.2.1 Natural Language Feature

We evaluate the word complexity of music names to see if it prevents users from searching or clicking in. We focus on both the words and the natural language used in music names.

Word

(1) The number of vocabularies
(2) The average length of vocabularies
(3) The percentage of unique words
(4) The percentage of common words:

Common words are defined as the noun, verb, adjective, or adverb words from the word high-frequency list produced by Professor Adam Kilgarriff and the British National Corpus (BNC). The list contains 6,318 words with more than 800 occurrences in the whole 100M-word BNC.

Natural Language: English vs non-English

The English language is one of the most widely-speaking languages in today's world. As Spotify operates mainly in English-speaking countries, we want to see if music names' natural language would affect users' music decisions.

Since music names are mostly short strings, it is challenging to detect language types with limited inputs. We compare multiple NLP tools and decide on *fasttext* as it delivers high computational efficiency and high accuracy. The result indicates that approximately 70% of Spotify music names are in English.

3.2.2 Emotion and Sentiments Feature

We use the NRC Emotion Lexicon to extract words with specific types of emotions or sentiments in music names. The NRC Emotion Lexicon is a list of English words that are classified by eight basic emotions (anger, fear, anticipation, trust, surprise, sadness, joy, and disgust) and two sentiments (negative and positive). The classification is conducted manually by crowdsourcing.

4. Predictive Models and Results

We implement multiple machine learning models to predict music popularity. The outcome variable is a dummy variable that examines whether the popularity of the music is beyond the median level. The input variables are the music release year, 12 audio features, and 13 music name features (see Appendix 1). We classify our feature inputs into three sets and test them with each model. The results show that models with all the feature variables have the best prediction performance.

4.1 Regression

Linear regression

Assume the relationship between the inputs and the outcome is linear. We adopt a binary classification model that requires a two-class dependent variable and a classification threshold. We test the threshold with 0.4, 0.5, and 0.6. We choose the model with a 0.5 threshold as it delivers the highest accuracy rate, which is 0.85 (~0.84 for the other two). The accuracy remains the same with normalized inputs.

Logistic regression

The logistic regression gives prediction accuracy of ~0.73. The model with normalized inputs delivers a better predicting performance with an accuracy of 0.85.

4.2 Decision Trees

Assume the relationship between the inputs and outputs are non-linear. We implement three different classification tree models to achieve a better prediction performance.

Simple Tree

We start with simple tree models. The tree models are the most computationally efficient and do not require additional feature scaling. However, tree models have high variance and are sensitive to small changes in training data.

Random Forest

We implement random forest models for a lower variance. Random forest models build several trees on bootstrapped training data and force each split to cover a randomly selected subset of predictors only, reducing model variance. However, our result shows that there exists potential overfitting given the huge difference between training and testing accuracy scores.

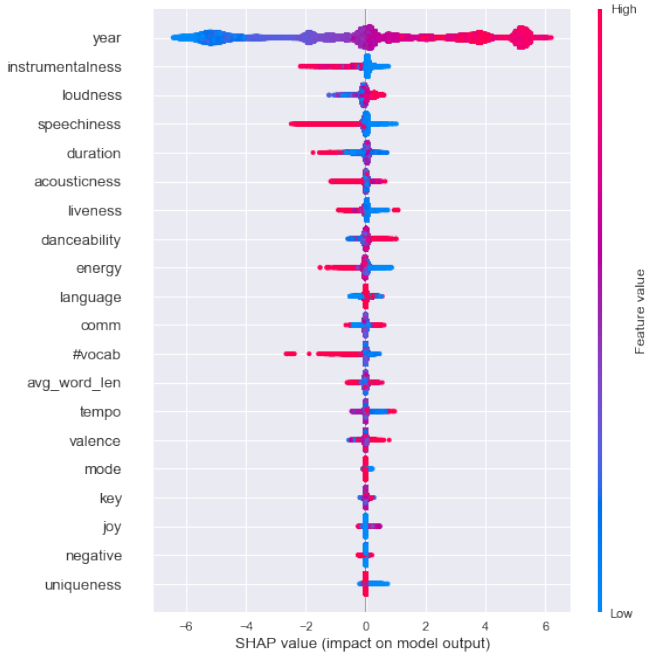

Figure V: SHAP plot of tuned model

XGBoost

XGBoost is an implementation of gradient boosted decision trees. The model sequentially fits trees to the residuals from the previous tree. Each tree is fitted to a modified version of the data, improving model performance in areas where it does not do well. [2]

We implement *GridSearchCV* to tune the model parameters including the learning rate, the number of trees and the maximum depth of XGBoost model using 5-fold cross-validation. After tuning the hyperparameters, the test accuracy score of the new model reaches 0.860.

Since gradient boosting models are like black-box, it is hard to understand their inner dynamics and estimate the impact of each feature on models. We thus use the SHAP Algorithm to interpret our tuned XGBoost model. Figure V shows that *year* is the

dominant feature in popularity prediction. In general, audio features have greater impacts on predictions than music name features.

4.3 KNN

KNN classifier is a simple and easy-to-implement non-parametric algorithm used in machine learning. This model clusters data points based on their probabilities of being similar to their top K-nearest points. [3]

We use item-based KNN model to cluster Spotify songs based on their similarities in music features. We use a 10-folds cross validation to find the optimized number of neighbors and implement KNN classifiers to make predictions. Figure VI presents the accuracy score changes across the number of neighbors from 1 to 30. The optimal number of neighbors is 30, leading to the highest accuracy score of 0.845.

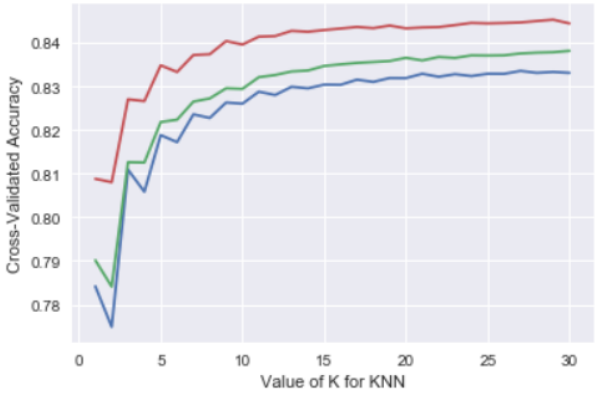

Figure VI: Neighbors and Accuracy

4.4 Neural Network

We implement the neural network model and use *GridSearchCV* to tune the parameters including the solver, the learning rate, the activation function, the number of hidden layers, and the number of training passes. The result shows that there is no overfitting as the R-square scores of the training and test group are very close. However, the model gives a negative R-squared score which indicates that it does not fit our dataset.

4.5 Discussion

We use multiple classification models to make predictions. The table below shows the accuracy scores of each model.

We recommend the XGBoost model with tuned parameters as it has the highest accuracy score, followed by two regression models. The Neural Network fits our data the worst and thus delivers the lowest accuracy score.

Models	Accuracy Score
Linear Regression	0.854
Logistic Regression	0.852
Simple Tree	0.849
Random Forest	0.848
XGBoost	0.860
KNN	0.845
Neural Network	0.730

5. Conclusions

We predict music popularity based on the names and audio features of music. Spotify can use our prediction to select and categorize its music portfolio. In addition, we recommend Spotify to increase the percentage of studio recordings and electronic music. The liveness and acoustic components of music result in lower music popularities based on our linear regression analysis (see Appendix 2). As the release year is a dominant player in predicting music popularity, Spotify may prioritize new albums than time-honored ones.

There is still much left for improvements. First, the models can include longer text information, such as *behind the lyrics* on Spotify. Longer text information is favorable for text mining analysis and immediately affects users' perceptions of music. Second, we can include user data in our models to refine our predictions and give tailored music recommendations. Lastly, models can be further customized and improved for higher computational efficiency and predicting accuracy. We can also keep exploring potential options for predicting methods.

Reference

[1] Spotify Technology S.A. (2019). 2019 20-F Form. Retrieved December 18, 2020, from https://s22.q4cdn.com/540910603/files/doc_financials/2019/ar/Spotify-2020-AGM-Annual-Report-on-Form-20-F.pdf

[2] James, G., Witten, D., Hastie, T., & Tibshirani, R. (2017). An introduction to statistical learning with applications in R. New York, NY: Springer.

[3] Harrison, O. (2019, July 14). Machine Learning Basics with the K-Nearest Neighbors Algorithm. Retrieved December 18, 2020, from https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761

Appendix

1. Definition of music features

Category	Variable Name	Definition
	Year	The release year of each song.
Audio Feature	Duration	The duration of the track in minutes.
	Key	The estimated overall key of the track, converting into integers using standard Pitch Class notation.
	Mode	Modality of a track, and 1 represents major and 0 represents minor. The melodic content is derived from the type of scale.
	Tempo	Shown in beats per minute (BPM). It is the pace or speed of a track and is directly calculated from the average beat duration.
	Acousticness	A measure from 0 to 1 of whether the track is acoustic, and 1 represents high confidence whereas 0 represents low confidence that the track is acoustic.
	Danceability	A measure of how good a track is for dancing based on a combination of musical elements including beat strength, tempo, and so on. 1 represents most danceable whereas 0 represents least danceable.
	Energy	A measure from 0 to 1 of intensity and activity. Usually, energetic tracks feel noisy and loud. It is decided by dynamic range, onset rate, timbre, and some other perceptual features.
	Instrumentalness	Tracks whether a track contains vocals. Some modal particles do not count vocals but rap or spoken word tracks are “vocal”. 1 represents that track contains no vocal content but instrumental part.
	Liveness	Catches the appearance of audiences’ sound in the recording. It is very likely that the track is live for a value above 0.8.
	Loudness	Shown in average decibels (dB) of a track, ranging between -60 and 0 db. It is the primary psychological correlate of amplitude.
	Speechiness	Tracks whether a track contain spoken words. Values close to 1 represent the recording is speech-like (e.g. talk show). Value close to 0 means the track is prone to music or other non-speech-like piece.
	Valence	A measure from 0 to 1 of the musical positiveness. Value close to 1 represents more positive feeling (e.g. euphoric, excited) in a track whereas value close to 0 represents more negative feeling (e.g. upset, unhappy).
Song Name Feature	Name	The name of each song.
	Vocab	The number of unique vocabularies in the name of a song.
	Avg_word_len	The average word length for the name of a song.

	Uniqueness	The percentage of unique vocabularies used in the name of a song.
	Language	Detect the language feature of song names. Two categories are English and other languages.
	Comm	The frequency of common words used in each song name. Higher value means more frequent words are used.
	Emotion	An overview of 8 emotions and 2 sentiments of each song name, using NRC Emotion Lexicon.
	Negative	A percentage that shows negative words used in the name of a song.
	Anticipation	A percentage that shows anticipation in the name of a song.
	Anger	A percentage that shows anger in the name of a song.
	Disgust	A percentage that shows disgust in the name of a song.
	Fear	A percentage that shows fear in the name of a song.
	Joy	A percentage that shows joy in the name of a song.
	Positive	A percentage that shows positive words used in the name of a song.
	Sadness	A percentage that shows sadness in the name of a song.
	Surprise	A percentage that shows surprise in the name of a song.
	Trust	A percentage that shows trust in the name of a song.

2. Summary of regression coefficient

Variable Name	Coefficient
Year	0.0137
Duration	0.0031
Key	0.0006**
Mode	-0.0008
Tempo	-0.00008**
Acousticness	-0.0841
Danceability	0.0167**
Energy	-0.031
Instrumentalness	-0.0321
Liveness	-0.0928
Loudness	0.0016
Speechiness	-0.0229
Valence	0.0197
vocab	-0.0026
avg_word_len	0.0022
uniqueness	0.0231
language	-0.0438
comm	0.0002
negative	-0.0197*
anticipation	-0.0344
anger	0.0169
disgust	-0.0138
fear	-0.0059
joy	0.0102
positive	0.0069
sadness	0.0202*
surprise	-0.0244*
trust	-0.0025
n	170,653
R^2	0.571

Note: Linear regression formula is $y = \alpha + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n + \epsilon$, where x_i is our model input; y is the dependent variable; α is a constant intercept term; β_i is the feature weight (coefficient) of each independent variable, and ϵ is an error term. ** represents p-values between 0.001 and 0.01 whereas * represents p-values between 0.01 and 0.5.

3. Regression model performance comparison

Model	Methods	Indicators	Feature Set 1	Feature Set 2	Feature Set 3
Linear	Numerical	Train R^2	0.753	0.745	0.742
		Test R^2	0.759	0.751	0.749
		RMSE	10.727	10.891	10.949
	Binary Classification	Train R^2	0.570	0.565	0.563
		Test R^2	0.576	0.572	0.569
		RMSE	0.326	0.327	0.328
Logistic	Binary Classification	Train R^2	0.850	0.848	0.846
		Test R^2	0.852	0.852	0.851
		RMSE	0.385	0.385	0.387

Note: Feature Set 1 contains both audio features and song name features, excluding non-numerical ones such as name and emotion; Feature Set 2 excludes the Spotify-defined features; Feature set 3 only includes factual features like key, mode, etc. The numerical linear regression model predicts music popularity with numerical values of music popularity. It has relatively better R-squares of ~0.75 but higher RMSEs because the mismatch between continuous variables are more likely than that between binary variables. The logistic regression model performs better than linear binary classification. Though they share RMSE around 0.3~0.4, the logistic model has a much higher R-squares which is about 0.85. Prediction with feature set 1 performs slightly better than predictions with feature set 2 & 3.

4. Tree model performance comparison with feature set 1

	Simple Tree	Random Forest	XGBoost
Train RMSE	10.4816	4.3285	8.7355
Test RMSE	10.3690	10.0026	9.5333
Train R^2	0.7693	0.9606	0.8397
Test R^2	0.7745	0.7903	0.8094

Note: the simple tree model has a R-square of 0.76 for both train and test dataset. Random Forest model has an extremely high train R-square while a low test R square, indicating a potential overfitting. XGBoost model has a higher R-square of 0.8 for both train and test data, and a relatively lower RMSE. XGBoost model thus gives the best prediction performance.

5. Classical to pop music trend & perpetual love theme