Introduction to Data Science Capstone Project

Data Analysis of Spotify Tracks

Presented by Team 404NotFound

Yuhao Liu yl5308@nyu.edu

Qingwei Meng qm351@nyu.edu

December 19th, 2022

Introduction

This semester, we have had a conversation with a manager from Spotify in class. We see that most students use Spotify frequently, and they are curious about the recommendation system used in Spotify. The recommendation system aims to help listeners find tracks they like but haven't heard before. Thus, building algorithms that recommend new and interesting music to its users is important to Spotify. The purpose of this paper is to figure out how to build the most suitable recommendation model based on data from Kaggle[1]. This is a dataset of Spotify tracks over a range of 114 different genres. Each row represents a track, and each track has some features associated with it. There are mainly two kinds of features: professional features (duration_ms, explicit, key, loudness, mode, speechiness, tempo, time_signature, track_genre), and personal features (danceability, energy, acousticness, liveness, valence). The popularity feature is our target and is calculated by algorithm based, in the most part, on the total number of plays the track has had and how recent those plays are. We try to build a model based on these features and figure out the most significant feature in our prediction model. We first preprocess the data by dropping the missing value. There is only one missing value in our dataset so we just simply drop it. Since there are many artists in one cell of the artists column, we split them into different rows and add a column num_artists to represent the number of artists in one track. We have also done a median splits on acousticness and danceability. Also, we encoded the boolean feature explicit and categorical feature track_genre to help us build the model. We also computed the correlation matrix for our features to explore the pattern of our data (Figure 1). The highly correlated features like energy and loudness (r = 0.8) or energy and acousticness (r =-0.7) can be reasonably explained and we decided to use all dimensions of the data to train our models to interpret the result for our audience better.

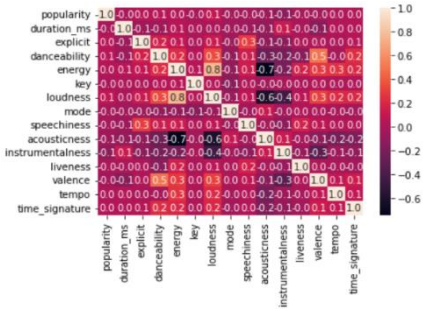


Figure 1. Correlation matrix for features

Inference question

Question: Do the tracks that have higher danceability more popular?

We want to compare the medians of more and fewer danceability groups' popularity to determine which group is popular. Our approach is to adopt a one-sided Mann-Whitney U test to solve this problem. Since normality does not hold (Figure 2), we avoided the T-Test.

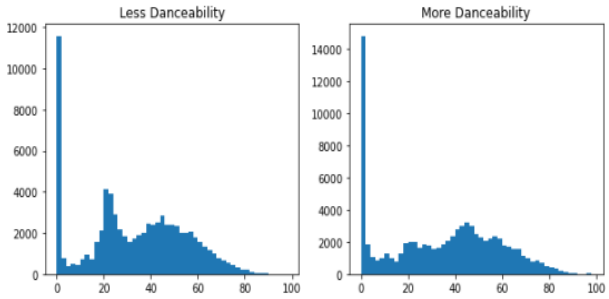


Figure 2. Danceability Distribution

Another reason to use Mann Whitney U test is the similar distribution each group has. We first used a median division on danceability: the tracks with higher than median danceability are assigned into one group, else to the other. The test statistic is 3226291622 with a corresponding p-value of 1.86^-29. The boxplots below (Figure 3) illustrate the test result:

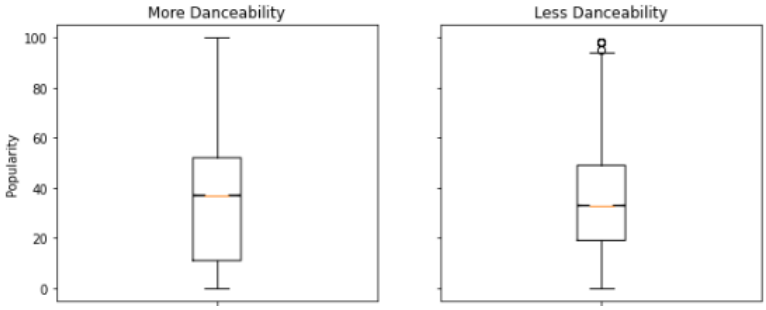


Figure 3. Danceability box plot

However, more interesting results come out when we add the "accousticness" variable into consideration. We also did a median division on accousticness. Then we are comparing the median popularity of the two groups with both having less accousticness, while one has less danceability and the other has more danceability. It is appropriate to use Mann Whitney U test due to similar distributions of the data.

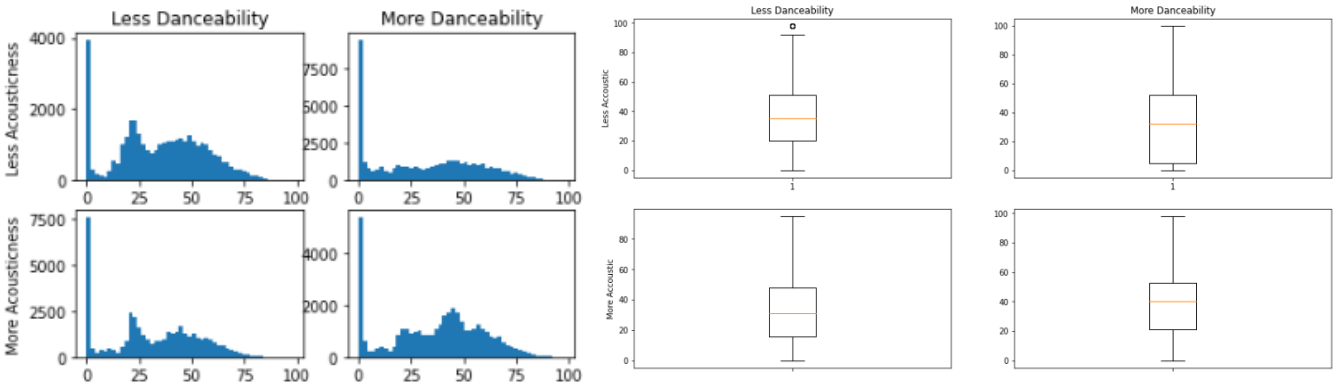


Figure 4. Danceability result with/without acousticness

Given both groups have less accousticness, the group that has more danceability is surprisingly significantly less popular than those with less danceability, with the Mann-Whitney U test statistic 705173009 and corresponding p-value 1.31^-77. For the two groups with more accousticness, the group that has more danceability is significantly more popular, with the test statistic 901041025 and p-value 1^-264 (Figure 4).

As a result, the tracks with higher danceability are more popular. However, given that the tracks are less acoustic, those with higher danceability are actually less popular. With those tracks that are more acoustic, those with higher danceability are more popular.

Prediction question

Question: Do the track's danceability predicts its popularity if we control acousticness?

To answer this question, we first averaged the danceability and popularity by each genre, and then we did a median split for acousticness. Then we divide popularity by 100 so that it represents the percentage of popularity. Then we did a linear regression with the dependent variable average percentage popularity and the independent variable average danceability. We obtained two scatter plots with their best-fit lines (Figure 5).

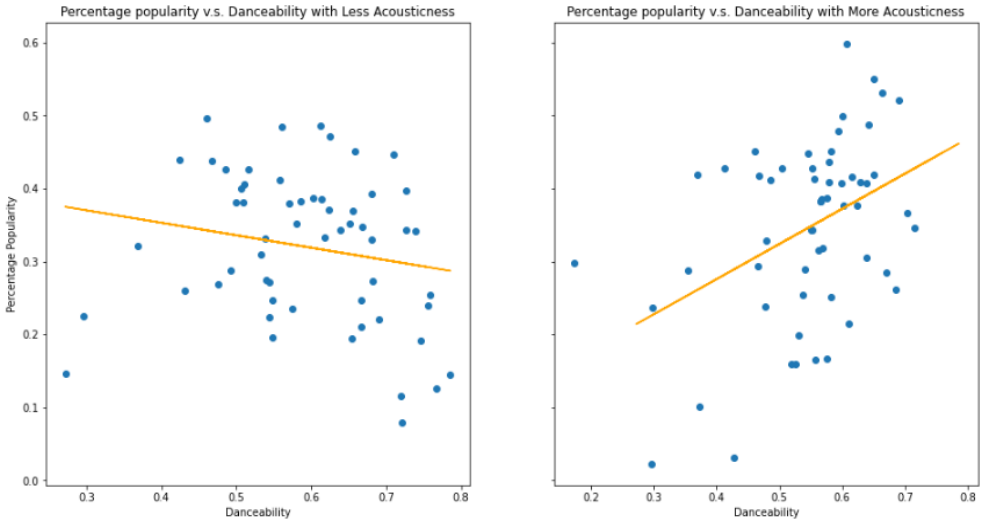


Figure 5. Popularity vs. Danceability scatter plot

As a result, for less acoustic tracks, the R^2 is around 0.036. The 95% confidence interval for the coefficient is (-0.407,0.065), which includes 0, and thus, we do not have strong evidence that the track's danceability can predict its popularity given the accousticness is low. On the other hand, the more acoustic ones have R^2 around 0.18. The 95% confidence interval for the coefficient is (0.2,0.763), which excludes 0. Thus, we do have evidence that the track's danceability can predict its popularity given the acousticness is high. However, due to its low R^2, the power of this linear regression model is low.

ML question

Question: Figure out the three most important features affecting the popularity, separate the tracks into different clusters using these features and classify the popularity in different clusters.

We want to figure out what feature has the highest importance in different models. Initially, we choose data with popularity larger than 0 to prevent extreme values and normalize our

dataset for better prediction. We compare the effectiveness of the XGB regressor, Lasso regression, Ridge regression, and Decision Tree Regressor. We use grid search to do hyperparameter tuning and split the available data into train and test with 80/20 train/test split and then see different models' RMSE (root-mean-squared-error) and COD (coefficient of

	model	RMSE	R-Squared
0	XGBRegressor	12.5751	0.5820
1	Ridge	18.7567	0.0700
2	Lasso	18.7567	0.0700
3	Decision Tree	15.0055	0.4048

Figure 6. Different models' RMSE and COD

determinant). XGB regressor and Decision Tree Regressor perform much better with RMSE lower than 15 and COD around 0.5 (Figure 6).

Then we see the feature importance of these two models and find that track_genre is much more important than other features in both models (Figure 7). In particular, the most professional features like key, mode, and track signature are not that important. Most audience doesn't care much about professionalism in music and the popularity of a track doesn't depend on how professional the artist is.

From this observation, we decide to choose the three most significant features: track_genre, acousticness, and instrumentalness. Then we use these three features to cluster the popularity of our tracks. Since we have categorical data in our feature, we cannot use k-means clustering. Then we decide to use a median split of feature acousticness and instrumentalness and change these two features into categorical data. Then we decide to use k-modes clustering to separate categorical data. We compare the cost of different k values and choose k=3 as the number of clusters using the "elbow" method (Figure 8). Then we plot the 3D clustering graph and try to do logistic regression to classify popularity in different clusters (Figure 9).

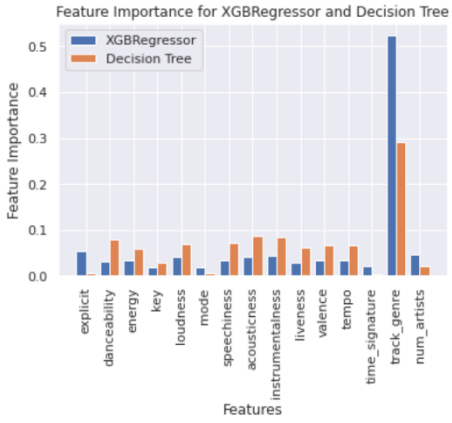


Figure 7. Different models' feature importance

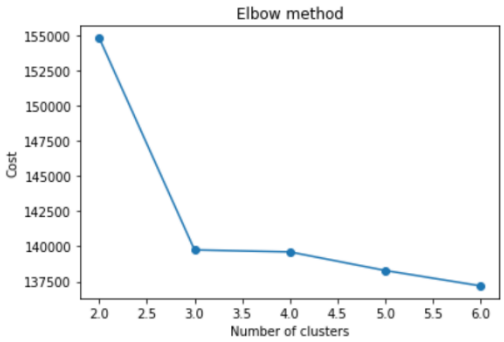


Figure 8. K-modes elbow method

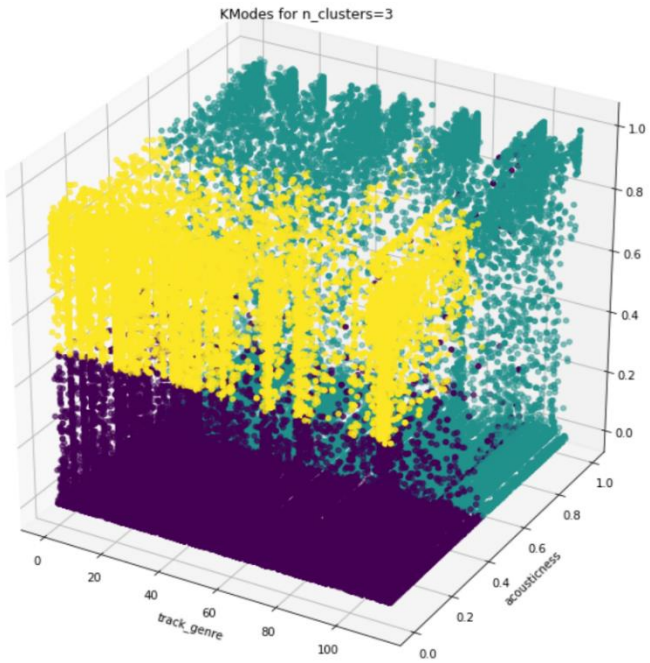
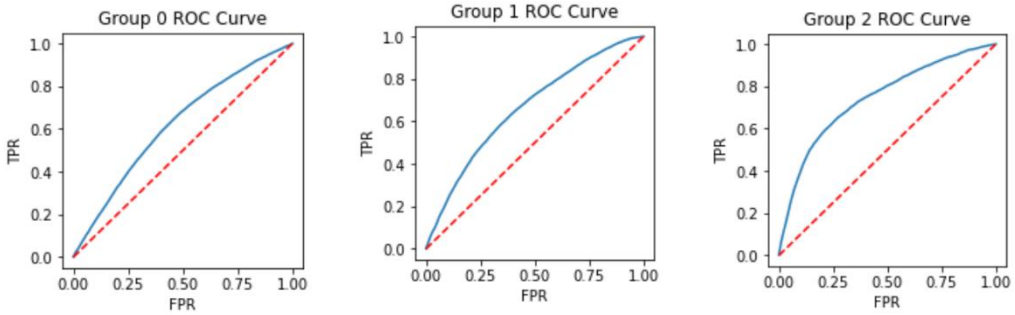


Figure 9. K-modes clustering 3D graph

In each cluster, we still do a median split of popularity and use all the features to perform logistic regression to classify popularity. The AUC score is around 0.7, which is good (Figure 10). The weighted AUC for the whole dataset is 0.64. This does imply that in different genres of tracks, we could somehow predict the potential popularity of this track based on the features and we could recommend new tracks in the same cluster to our audience.



	cluster	AUC	explicit	danceability	energy	key	loudness	mode	speechiness	acousticness	instrumentalness	liveness	valence	tempo	time_signature	track_genre	num_artists
0	0	0.618852	0.355588	0.479770	-1.089421	0.005845	0.078325	-0.047279	-0.495527	0.150211	-0.389529	-0.531383	-0.845806	-0.001953	0.405223	-0.000820	0.079938
1	1	0.662869	0.109708	1.082444	0.061522	0.006284	0.016168	0.105658	-1.318561	-1.075591	-0.563634	-0.635630	-1.244627	0.003197	0.259905	-0.001754	-0.014453
2	2	0.743454	-0.134124	0.669705	-0.670830	0.003425	0.047091	0.028525	-0.389848	0.593919	-0.388884	-0.236604	-1.511665	-0.006937	0.178175	0.020795	0.208051

Figure 10. ROC curve for different clusters

Conclusion

As we can see from our inference question, the tracks with higher danceability are more popular. However, given that the tracks are less acoustic, those with higher danceability are actually less popular. With those tracks that are more acoustic, those with higher danceability are more popular. For the prediction problem, we do not have strong evidence that the track's danceability can predict its popularity given the accousticness is low. On the other hand, we do have evidence that the track's danceability can predict its popularity given the accousticness is high. The analysis indicates that popularity can be affected by personal feeling factors if the

track's professionalism is kept.

In our ML results, track genre is important when we build the model. This means some genre indicates the popularity of the track. The result could somehow explain why lots of artists play similar types of music like hip pop and R&B in the market. These songs are more likely to be popular. However, an engineer at Spotify should be cautious to recommend the specific track genre of music to their audience. Some tracks are not suitable to be categorized in a specific genre and some tracks could be categorized into different genres. This could be one direction to improve the recommendation model for Spotify.

However, the study has limitations. In linear regression, even though the coefficient is significant, the R^2 is low, indicating that the power of this linear regression is not high enough. This is due to the low significance of our features and the large proportion of zero values. In the machine learning part, we may use dimension-reduction methods to train our model. We decide not to use it to keep the interpretability of our results.

One remarkable thing we have investigated is whether pop music is really more popular. We have divided tracks into pop music and non-pop music by the criterion of "pop" in the name of the genres. We used a one-sided Mann-Whitney U Test, and the p-value is 0.006, which shows that pop music is significantly more popular than non-pop music. Then we create a boxplot (Figure 11), and it is easy to see that pop music is more popular, which fits our normal definition of pop music.

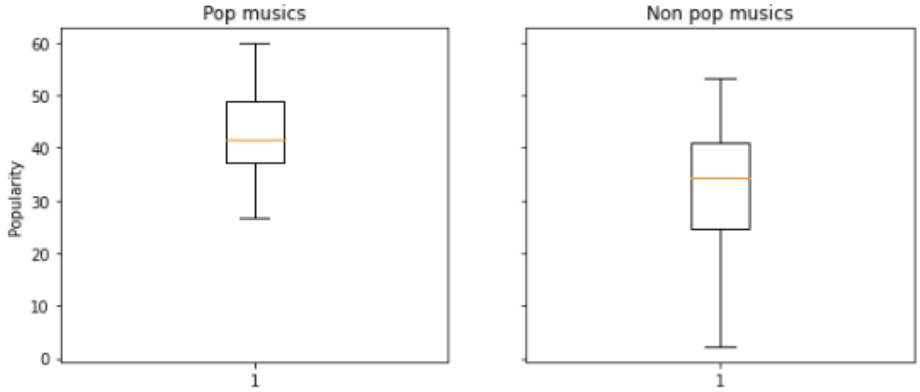


Figure 11. pop/non-pop music box plot

In an ideal world, we need data about how people really feel about the tracks, including emotions and audience ratings, etc. If we have better data, we can figure out which genre is more popular in the market, and we can add the potential influence of the artist's name and more non-numerical features. These new features can help us to control confounding variables in a more meaningful and powerful way, according to our research on others' related work. To gather this type of data, we might want to call Spotify to see if they can reveal them since our dataset is just a tiny subset of all Spotify's data. Further research related to the popularity and recommendation system could be investigated with better datasets.

Reference

[1] Spotify Tracks Dataset. https://www.kaggle.com/datasets/maharshipandya/-spotify-tracks-dataset