IEOR 142 Final Report - Predicting Spotify Track Popularity

Mina Baghai, Abhigyan Biswas, Toufiq Hossain, Naya Lee, Luna Ragot

Project Motivation

This music industry is consistently growing and changing every year, and is an industry that nearly everyone utilizes and connects with. With the rise of music streaming services such as Spotify, the way that people consume and enjoy various genres of music, different artists, and songs has changed. Spotify users are trying to find new artists of interest, and artists themselves are trying various strategies to increase their following and streams on digital platforms. There are countless factors that go into how popular an artist is, and many artists are hoping to get a better understanding of not only how popular their tracks are going to be, but how to increase their track popularity to increase their number of streams and become more successful musicians. With information on what an artist can expect their track or album popularity to be, they can better plan their marketing, live concerts, merchandising plan, future releases and much more. Therefore with Spotify being one of the largest digital music streaming platforms, we are aiming to generate a model that can predict the track popularity an artist has on Spotify's platform based on features relating to the song characteristics and Spotify statistics.

There is a large amount of data readily available that Spotify collects about each artist which gives information about their streaming statistics, different characteristics about various songs an artist has released, the genres of music released by an artist, etc. We hope to take advantage of this valuable information available to develop a prediction model that can provide useful insights into the popularity of specific tracks that artists have released. It will be beneficial not only to provide predictions on track popularity, but also to observe which factors are contributing to an artist's success in gaining popularity specifically on a digital streaming platform such as Spotify. Overall this model will help artists build up their fanbase and guide their decision making on future song releases in hopes to generate a larger following. We are aiming to create both regression and classification models, with regression models focused on predicting the specific track popularity number, and classification models having a more general prediction of whether a track an artist has released is considered popular or not.

Data

We used two datasets containing Spotify data: one had track information and the other contained artist information. These datasets were drawn from information released by Spotify. The first step in the project was data preprocessing. We first joined the two sets together on artist id to have artist data alongside the track data; this would give our model more features about the artists of the tracks to work with. We set the index to be the track id and dropped some columns that were not useful such as track name, artist name, artist id, etc. We also converted the date column into the pandas date time format. Finally, we noticed that the genre column contained long strings of genres and it would be helpful to one hot encode these. There were nearly 4000 different genres that appeared throughout the dataset, so we decided to only keep the ones that

appeared over 8000 times and this helped us find the top 23 genres. Our dataset ultimately contained 23 genre columns and totaled 46 columns.

The dataset was then split into training and testing sets, and different models were created to predict the track popularity which was on a scale from 0 to 100 with 0 being not popular and 100 being very popular. We decided to use 2 regression models, Ordinary Least Squares (OLS) and regression tree, and 2 classification models, logistic regression and classification tree. The classification models were predicting whether the track popularity would be popular, which we determined would have a score greater than 40, or not popular, less than or equal to 40. The regression models' output included information on the coefficients and their standard errors, t-values, p-values, and confidence intervals. The classification models were compared to the baseline model which was the majority class and assessed using their confusion matrix.

Analytic Models

After creating a preliminary OLS model, we examined the p-values listed in the summary and noticed that none of them exceeded 0.05, so we decided to stick with this model for our final OLS model. We calculated the out-of-sample R-squared value to be 0.42, which indicates that this model is not great (but not terrible) at predicting track popularity in the test set. In examining the coefficients given by the model, we found that the feature 'danceability' had the greatest absolute coefficient of all the features, with a value of 12.9. This indicates that danceability most greatly influences track popularity in the positive direction. 'Valence' and 'acousticness' have the greatest negative influence on track popularity with coefficients of -8.7 and -7.8, respectively. This information can be valuable to music producers and record labels because they can prioritize danceability and steer away from acousticness and high valence in the production of future tracks if they want to increase their popularity and consequently increase their profits. One surprising observation was that the coefficient for 'followers' was given to be -1.28e^-7, which implies that the number of followers the artist only slightly influences track popularity. This could be an important insight for record labels when making decisions about signing artists as well as matching artists to producers within the label.

We then created a regression tree and trained on our training set. To avoid overfitting while still maximizing accuracy, it was determined that a ccp of 0.02 and minimum sample split of 10 led to the best regression tree model. This created a tree with 843 nodes which is normal considering the many features used by the model and the massive quantity of data trained and tested on.

To compare our two regression models (see figure 1 in Appendix), we looked at the out of sample R2 scores and the residual sum of squares (RSS). We found that the regression tree model performs much better compared to the OLS model. The regression tree model has a significantly higher OSR2 score at above 0.6 compared to the OLS model having an OSR2 score much lower around 0.4. Similarly, when looking at the testing RSS scores, the regression tree has a score of around 9000000 lower than that of the OLS model. Overall, when looking at

regression models for our problem, the regression tree is much better thus suggesting our data is probably not linear.

Now moving onto our classification problem where we were predicting whether a track was popular or not. We set the threshold to a popularity score of 40 to make a split between whether a track was popular or not, and trained two types of models, a logistic regression model and a classification tree. We also built a baseline model for comparison, which simply predicted every track as not popular since that was the majority class of all tracks in the training dataset. The evaluation of the models included calculating accuracy, true positive rate (TPR), false positive rate (FPR), and creating an ROC curve to determine the area under the curve (AUC) for each model (see figures 2 and 3 in Appendix). The logistic regression model produced an accuracy of 75.9%, only 1% higher than what the baseline of 74.9% would have produced. However, we can also see through some coefficient analysis that, like the OLS model, we find very similar results in terms of which aspects of a track make it popular and we find parallels in the logic across the two models. We find that factors like the danceability score and explicitness greatly affect how popular a track is. We see that, with danceability for instance, that the odds of a track being popular goes up by almost 900% when danceability goes up by 1. When building out an ROC Curve for this model, we get an AUC of 0.76.

The next classification model we built was a classification tree. We wanted to try a classification tree to see if this model would be more accurate as tree models are better at capturing non-linear trends in data. For this model, we set the minimum_samples_split to 20 and ccp_alpha to 0.00, as we did not observe large amounts of overfitting in our regression models so we did not set a large ccp_alpha value to handle overfitting. This model performed significantly better than the logistic regression model, with a higher accuracy of 0.809, a higher TPR of 0.614, and a lower FPR of 0.126. Additionally after building an ROC curve, the AUC for this model is 0.84. The results showed that the classification tree model performed the best in every evaluation metric. These results suggest that the classification tree model is the best performing model for classifying whether a music track is popular or not.

Impact

Overall, the project provided valuable insights into the factors that drive music popularity. The findings can be used to develop strategies for music artists, record labels, and music streaming platforms to maximize music consumption and engagement. This analysis can be extended in the future by incorporating additional features as ticket/concert data, lyrics sentiment analysis and social media engagement data.

In this project, VIF was used to check for multicollinearity between predictor variables in both the logistic regression and OLS regression models. The VIF values were calculated for each predictor variable, with a high VIF value indicating high multicollinearity with other predictor variables. In the logistic regression model, the 'energy' variable was found to have a VIF value close to 5, which indicates moderate multicollinearity with other predictor variables. This variable was subsequently removed from the model to reduce multicollinearity and improve the

model's predictive power. In the OLS regression model, no variables were found to have high VIF values, indicating low multicollinearity between predictor variables in the model. This is an important finding as multicollinearity can negatively impact the interpretability and predictive power of regression models. It's worth noting that while VIF is a useful tool for detecting multicollinearity, it's not without limitations. VIF assumes that the relationship between predictor variables is linear, which may not always be the case. Additionally, VIF can sometimes fail to detect multicollinearity between variables that have a nonlinear relationship. Therefore, it's important to use VIF in conjunction with other diagnostic tools to assess the presence of multicollinearity and its impact on regression models.

One of the biggest challenges we faced during the modeling process was large computation times. Since our dataset was extremely large with a large number of rows, we tried building other models such as random forest and boosting models that used cross-validation to select hyperparameters. However, even with a lower number of folds in the cross validation and a small number of hyperparameters to select from, the computational times were still extremely large taking a significant amount of time to run. Therefore, since we were essentially forced to only tune a very small number of hyperparameters on a very small number of folds (less than 5), we felt that the benefits of running these models did not outweigh the large computation times. This is something that in the future we would want to spend more time on generating these more complex models that require cross-validation to see if we could build even more accurate models.

For music artists, knowing which track features and artist information impact track popularity the most can help them create and promote music that is more likely to resonate with their audience. They can also use this to identify areas of improvements and adjust their music production accordingly. For record labels, understanding the factors that drive the music popularity can help them select which artists and tracks to invest in. They can also use this information to identify trends and patterns in the industry and use it to inform their marketing and promotional strategies. Finally for music streaming platforms, this information can be used to develop algorithms that can recommend music to users based on their preferences and listening history. It can also help them identify emerging artists and listening trends among particular age groups and demographics, allowing them to create more personalized music recommendations for their users.

It is important to note that while predicting and promoting popular tracks can be beneficial, it can also lead to a lack of diversity and representation in the music industry. Therefore, it is crucial to consider the potential negative consequences of relying too heavily on predicting and promoting popular tracks. This can lead to a focus on creating music that fits a particular mold rather than allowing for creative expression and experimentation, which has for ages been the driver of cultural expansion and discovery of new genres. The music industry should strive for diversity and representation while also maximizing engagement and consumption.

Appendix

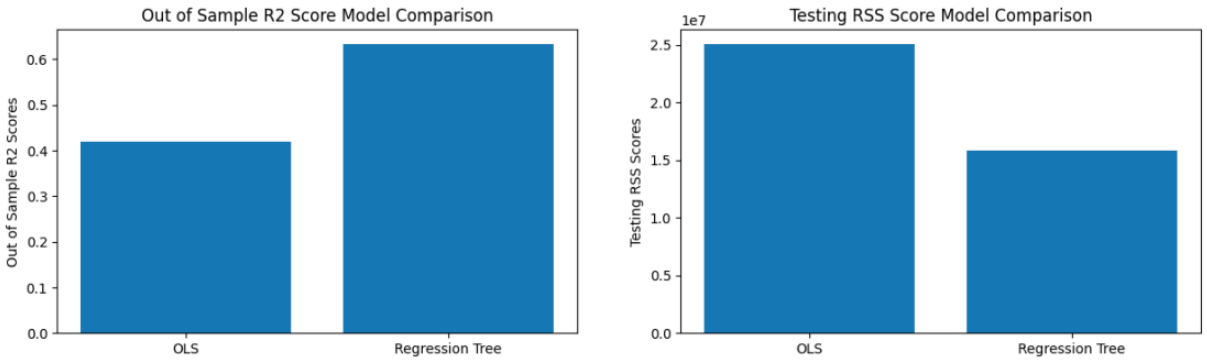

Figure 1: Comparing the regression models

	Baseline	Logistic Reg	Classification Tree
Accuracy	0.749	0.759	0.809
TPR	0.000	0.460	0.614
FPR	0.000	0.140	0.126

Figure 2: Comparing the classification models

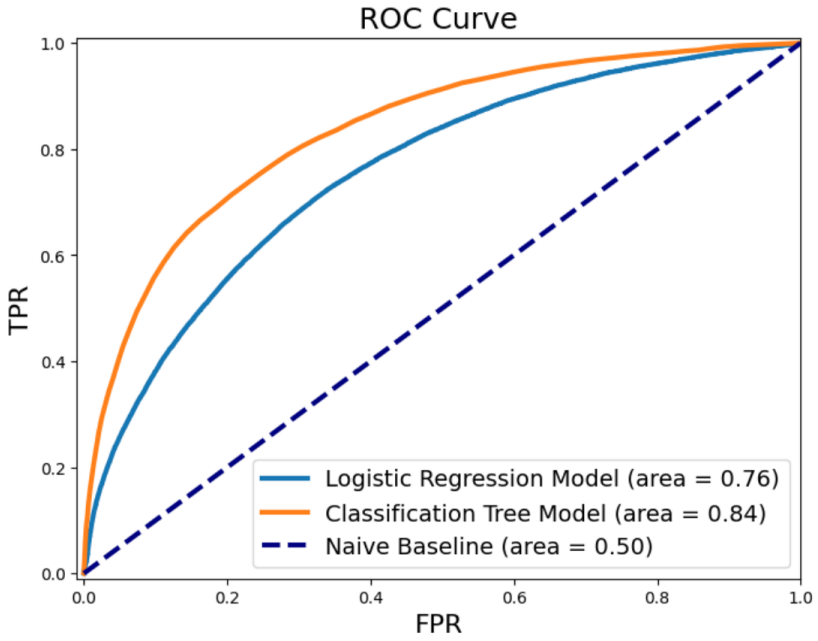

Figure 3: ROC Curve of the classification models