Kevin Buchanan, F. Kevin Ryan, Lauren Sapone
ITAO 70310 - Machine Learning
Professor Martin Barron
24 November 2020

Spotify: Factors Determining Songs' Popularity

Introduction

In this analysis, we attempted to solve the problem of determining what characteristics of music affect the popularity of songs, so as to allow the prediction of popularity for future songs. To solve this problem, we extracted a Spotify dataset from Kaggle that tracked songs, measured out different characteristics of each song, and provided a "popularity" that was based on the song's recent popularity in the United States. Since this dataset has a number of factors we could use in our model, including a song's key, explicitness, and danceability, there is a large array of variables that could determine a song's "popularity". In order to accomplish our goal, we first created random forest and XGBoost models to get a baseline level of RMSE, which would serve as a measure of predictive accuracy in predicting the popularity of songs. From there, we tuned the XGBoost model in an attempt to lower the RMSE and increase the model's predictive accuracy. From this final XGBoost model, we extracted the variable importance for the explanatory variables in this dataset in relation to the response variable: "popularity." So, the "inputs" of our project are the song characteristic variables and the popularity variable, while the "output" is the variable importance of each of those variables.

Again, the goal of this project is to find out from this dataset what factors contribute to a song's popularity. This is significant because, as the music industry continues to grow over time with more artists trying to "make it big," it is important to determine if there are certain aspects of music that make songs more or less "popular." This motivated us to take on this project in this manner because our results could be helpful to aspiring and established artists alike that want to improve their spot in the charts, as well as to record labels that want to produce the most successful artists and tracks.

Related Work

There are several studies that have been done that are related to the project we completed. Our problem has been attempted before, but not using the same machine learning model that we are planning to use. The first study, "Musical trends and predictability of success in contemporary songs in and out of the top charts," discusses musical trends over time and predicts what acoustic factors make a song successful (Myra, Interiano, et al. 2018). This study is probably the most similar to our project out of all the reports we found - however, there are still a few key

differences. First, this study uses a random forest model to predict the success of songs, based on acoustic characteristics and then adding the "superstar" variable, which indicated whether or not that song's artist had appeared in the top charts recently. Our model improves upon this because we use XGBoost, which has stronger predictive power and is more resistant to overfitting than a random forest model. Secondly, the authors of this study predicted whether a song was successful or not (yes/no), whereas we formed a regression to discover why a song is popular (what factors contribute to the song's popularity). The next study, "Going Viral: Factors That Lead Videos to Become Internet Phenomena" looks at what factors make videos go viral - while this is examining videos instead of songs, some of the methods and factors might still be similar to what we discover (West 2011). Third, "Automatic Prediction of Hit Songs" uses cross-validation to predict what songs will be hits versus not (Dhanaraj, Ruth, and Logan 2005). And finally, a paper called "Automatically Determining the Popularity of a Song" states that its purpose is to "identify a connection, if such a connection exists, between the sequence of sounds and the lyrics of a melody and its popularity with the help of machine learning techniques" (Chiru, Costin, and Popescu 2017).

Data Description

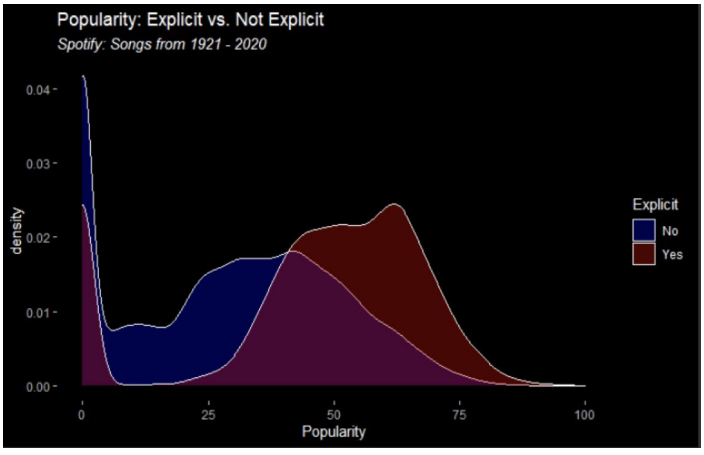
The dataset we chose to use is the Spotify Dataset 1921-2020, extracted from Kaggle. The dataset has over 160,000 songs that are ranked on a popularity scale of one to one hundred, based on U.S. opinion. The dataset has nineteen columns for various measures such as acousticness, danceability, duration, explicitness, etc.

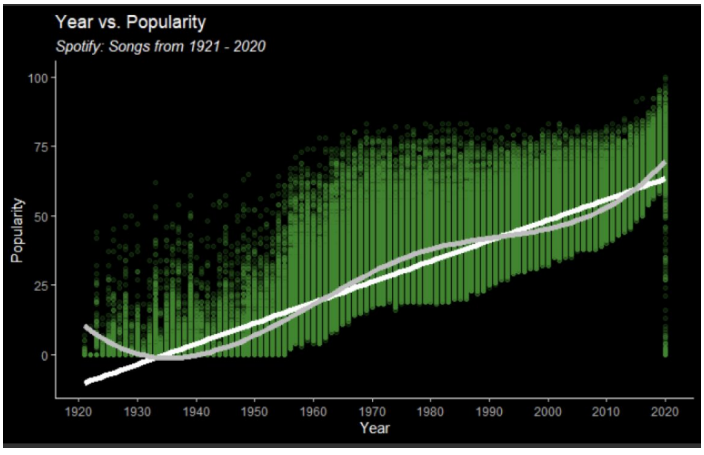
Our dataset was relatively easy to work with after downloading it from Kaggle, but there was some data pre-processing and cleaning that needed to be undertaken before we could begin our analysis. Luckily, there were no NA or missing values in the dataset, so we began with 169,909 complete observations across 19 variables. First, we explored the variables more closely and decided to take out the "Release date" and "ID" columns. "ID" contained an array of ordered random letters, symbols, and numbers that we felt was not useful for identifying a sample. "Release date" was repetitive of the "Year" column, but was also not symmetrical, as there were some columns showing just year, while others gave date, month, and year. Next, we ran some functions to remove all duplicates from the dataset, which dropped roughly 2,500 observations. From there, we converted all columns into their appropriate data type. Most were fine as numeric, but the "Key", "Explicit", and "Mode" variables needed to be converted to factors variables. From there, most of the cleaning was complete, we just reorganized the columns in the data set to be a bit more functional for reading (name and artists first, then the explanatory variables, and ending with the "popularity" response variable).

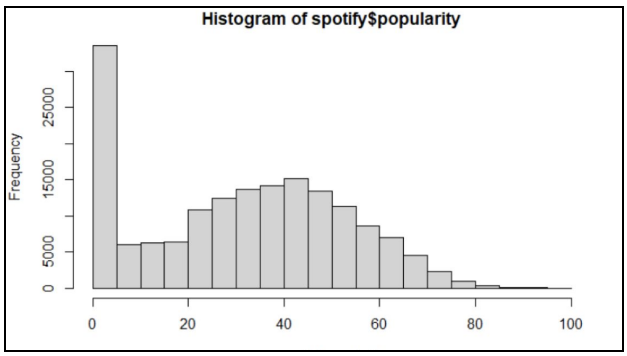
That was pretty much the extent of our data pre-processing and cleaning, so as a result, we were left with 167,369 unique observations across 17 variables. The full list of variables used and their descriptions are below, in alphabetical order:

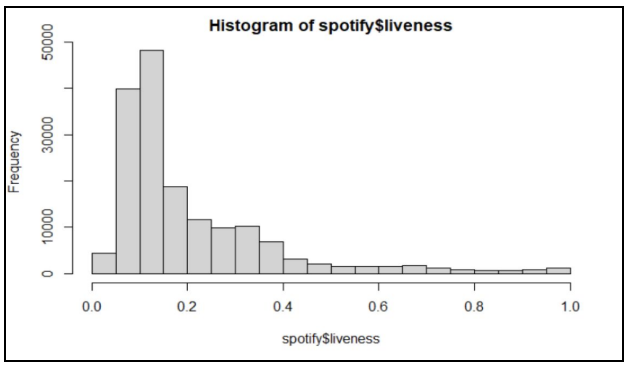
- Acousticness - The relative metric of the track being acoustic (range of 0 to 1)
- Artists - List of artists credited for production of the track (character)
- Danceability - relative measurement of track being danceable (range of 0 to 1)
- Duration _ms - duration of the song in milliseconds (continuous)
- Energy - the energy of the track (range of 0 to 1)
- Explicit - whether the track contains explicit content or not (binary: 0 or 1)
- Instrumentalness - Relative Ratio of the track being instrumental (range of 0 to 1)
- Key - primary key of the song (values on octave) - integers ranging from 0 to 11
    - on C as 0, C# as 1 and so on…
- Liveness - relative duration of the track sounding as a live performance (range of 0 to 1)
- Loudness - relative loudness of the tack in decibel (float from roughly -60 to 0)
- Mode - whether the track starts with a major chord progression or not (binary: 0 or 1)
- Name - title of the track (character)
- Popularity - popularity of the track lately in terms of the US perspective (range of 0 to 100)
- Spechiness - relative length of the track containing any kind of human voice (0 to 1)
- Tempo - tempo of the track in beats per minute [BPM] (float roughly 50 to 150)
- Valence - the positiveness of the track (0 to 1)
- Year - Year the song was released (range from 1921 to 2020)

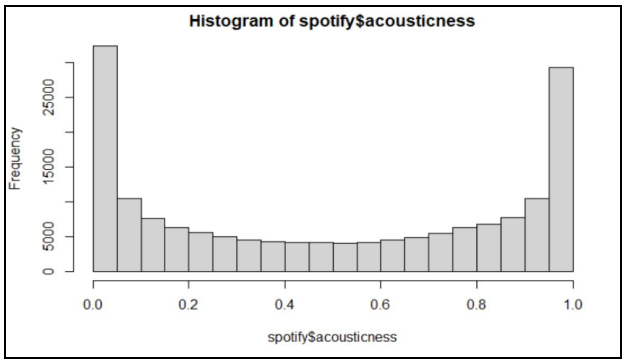
To learn more about our data, some visualizations were created to explore the distribution of values, see their relationship with popularity, etc. Below are some to show what was found.

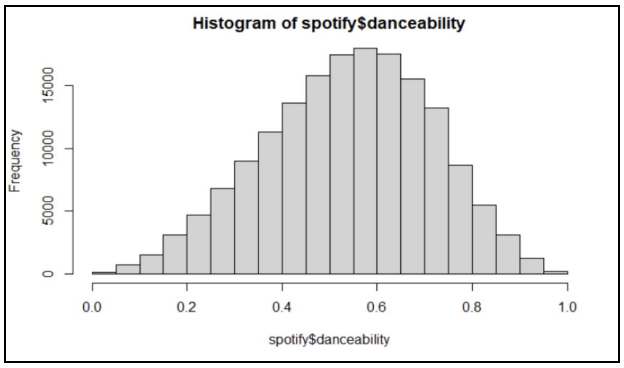












After that, we began our analysis. First, we split our data into training and test sets for our Random Forest Model. We used a Stratified Sampling technique to group the observations into training and test, grouping them by decade. In order to do that, we made a temporary "Decade" variable in our dataset, performed the sampling, and created the training and test sets without including the "Decade" variable into the sets. The actual training and test sets were split 80/20, meaning that the training set has 80% of our overall observations (133,896 rows) while the test set has 20% of our overall dataset (33,473 rows). Using those split sets, we were able to extract the same training and test sets for our XGBoost model in order to properly run our analyses.

Methods

We need to use ensemble models in order to predict popularity scores, tuning them to increase predictive accuracy and extract variable importance from them in order to understand which characteristics determine a song's popularity. To begin, we used a random forest bagging model and an XGBoost model to get a baseline comparison of RMSE (Root Mean Squared Error). After that, we tuned the XGBoost model to reduce the RMSE and get a final model. We evaluated our tuning using RMSE, RMSLE (Root Mean Squared Log Error) and MAE (Mean Absolute Error) in order to see which combinations of our parameters increased the predictive accuracy of our model, but most of our decisions weighed more to what minimized RMSE. Furthermore,

negative values in our RMSLE predictions caused null values that did not produce quantifiable results. Therefore, our tuning results were only evaluated using RMSE and MAE, with preference given to the results of the RMSE tuning.

From that final model, we obtained variable importance to determine what factors most influenced the popularity of a song, and plotted those values using SHAP in order to determine what levels of those characteristics attributed to higher or lower popularity ratings for songs. These models were appropriate to use for our problem because they were able to help us accurately predict what characteristics of a song contribute the most to its popularity, which is the ultimate goal of this project.

A random forest model produces a large amount of information about the data, works well with a large number of variables, can handle different types of variables, and is applicable to both regression and classification problems. This model is a good choice to use for our problem because the dataset does have a large number of variables of several different types - numeric, such as danceability and energy, and factor, such as key and explicitness. Furthermore, our problem is a regression problem, meaning that a random forest model is appropriate. However, when using a random forest model, we also have very little control over what the model does and tuning it. At best we can try different parameters and random seeds in the model to (hopefully) get better results/a lower RMSE. Since the random forest wasn't the focal point of our project, we only ran it once, so we didn't try different parameters, but we did use a random seed. Additionally, there is a chance of overfitting when using a random forest.

An XGBoost model has stronger predictive power and is more resilient to overfitting than a random forest model. It's also sensitive to outliers. These characteristics of the model are the reasons we decided to use it in addition to the random forest model, because it provided more accurate results. The limitations of an XGBoost model are that it's slow to train and hard to run in parallel since the trees build in sequence.

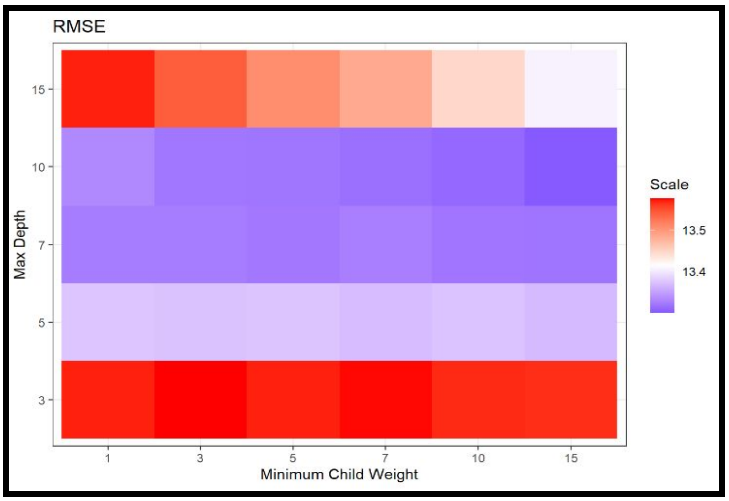
Results

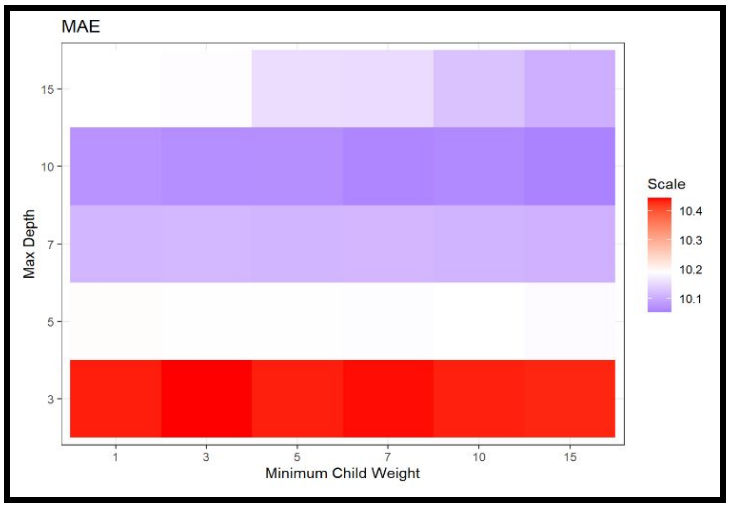
To begin, we used a linear regression model to get a baseline understanding of how our variables might contribute to a song's popularity score. The model was statistically significant with an Adjusted R-Squared of 78.47%. That is relatively good accuracy in explaining the variance, but in an attempt to increase its accuracy, we introduced an interaction term. It slightly increased the Adjusted R-Squared, but overall the general linear model is not sufficient with the goal we are trying to achieve in this analysis. From there, we incorporated ensemble models into our analysis, specifically random forest and XGBoost, to derive which variables/characteristics most contribute to a song's popularity rating. Running the random forest model gave us a baseline RMSE of 13.30 to compare to later on with our initial and final XGBoost models. After running

an initial XGBoost model, we found that the RMSE was a little higher than the random forest at 13.45. From there, our goal was to tune the XGBoost model to get as low of an RMSE and as high of an accuracy as possible.

We first tuned the XGBoost model with a learning rate of 0.1 and a high number of trees to get the optimal number of trees, which came out to be 432 trees. The hyper parameters we chose to tune were minimum child weight, maximum depth, gamma, column sample, subsample, and eta, evaluating each with the RMSE and MAE metrics outlined above. We implemented our tuning in several steps, visualizing our results to see which combinations minimized our metrics in the model.

We started tuning with minimum child weight and maximum depth. That optimal combination is maximum depth at 10 and minimum child weight being 15.

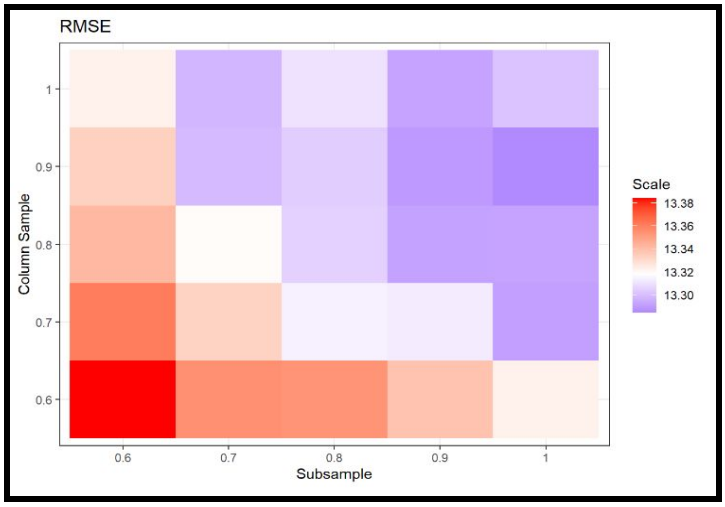


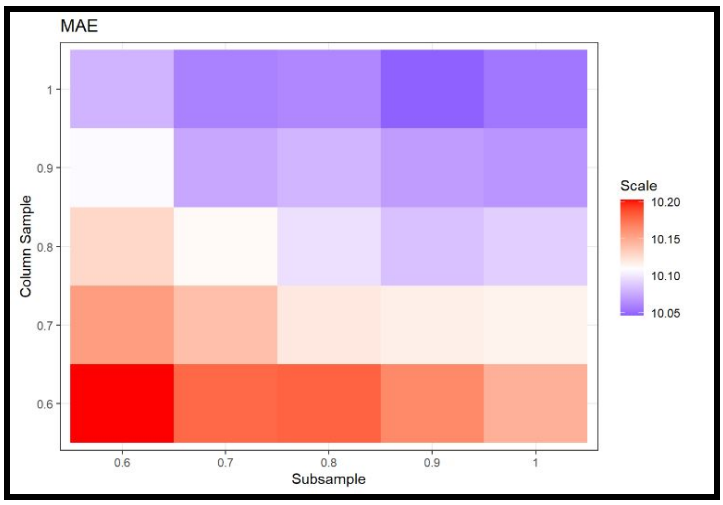


Then we tuned the gamma parameter, with the optimal gamma being 0.00.

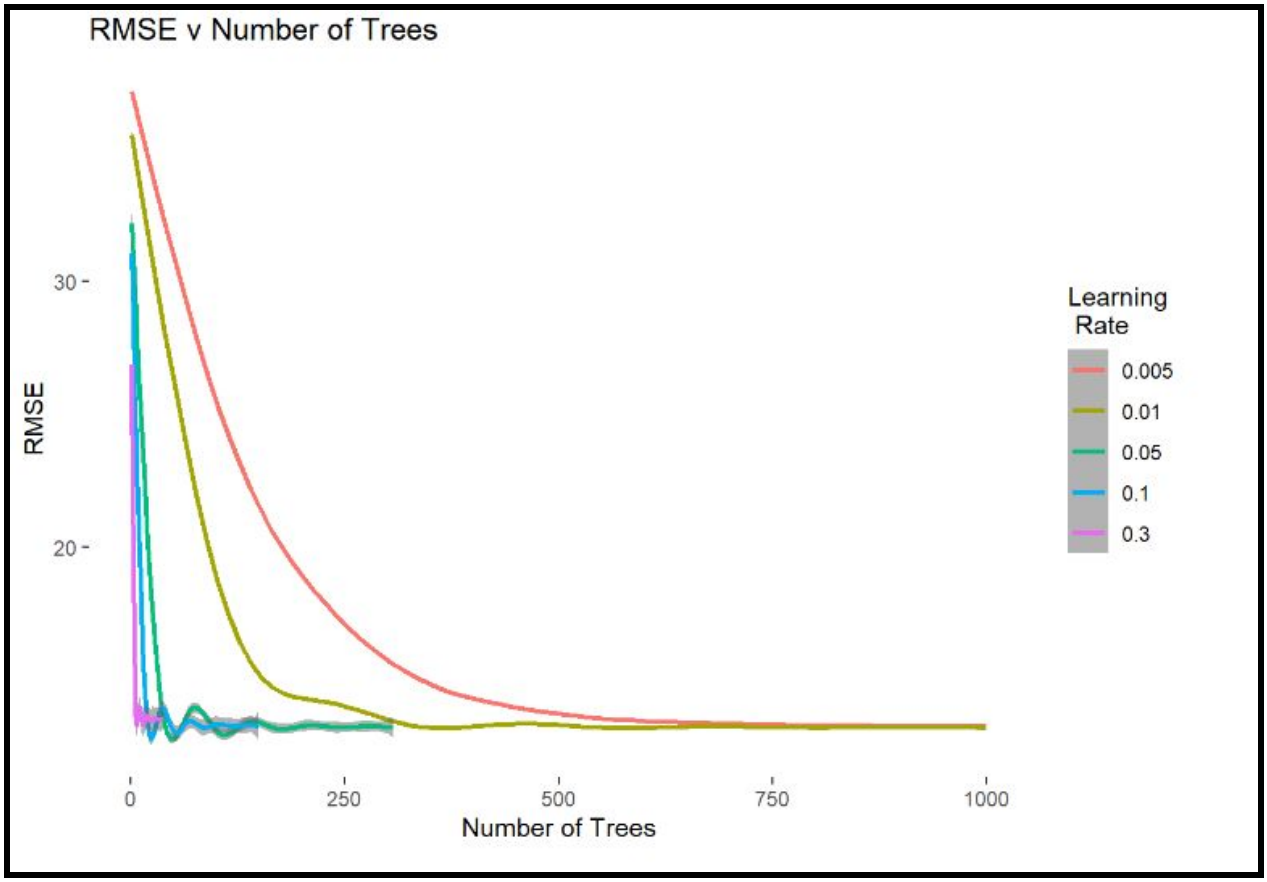
Gamma Values	RMSE	RMSLE	MAE
0.00	13.30161	NaN	10.05522
0.05	13.30726	NaN	10.05979
0.10	13.31345	NaN	10.06136
0.15	13.30326	NaN	10.05610
0.20	13.31798	NaN	10.07274

Next, we tuned column sample and subsample, and that optimal combination is column sample being 0.9 and subsample being 1.

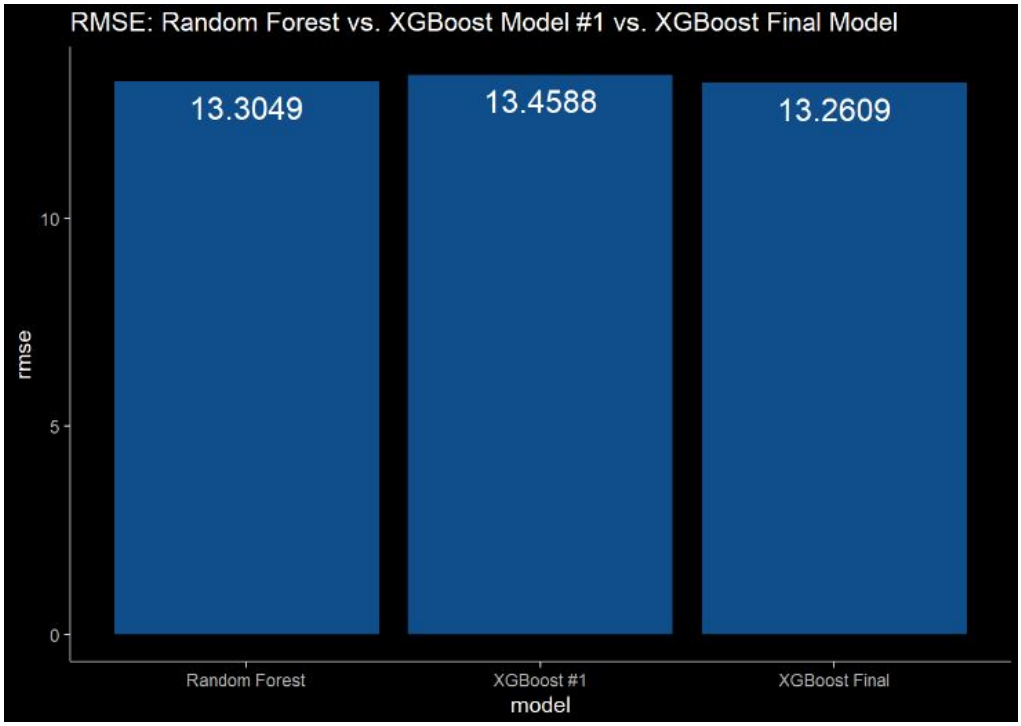




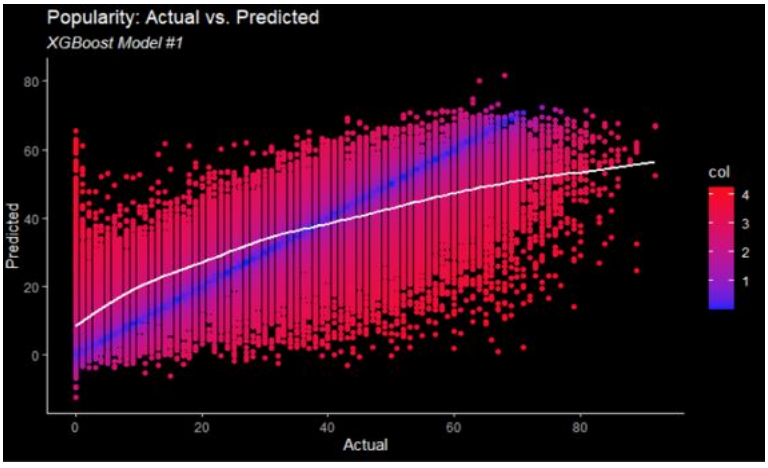
Finally, we tuned eta, getting an optimal learning rate of 0.05.

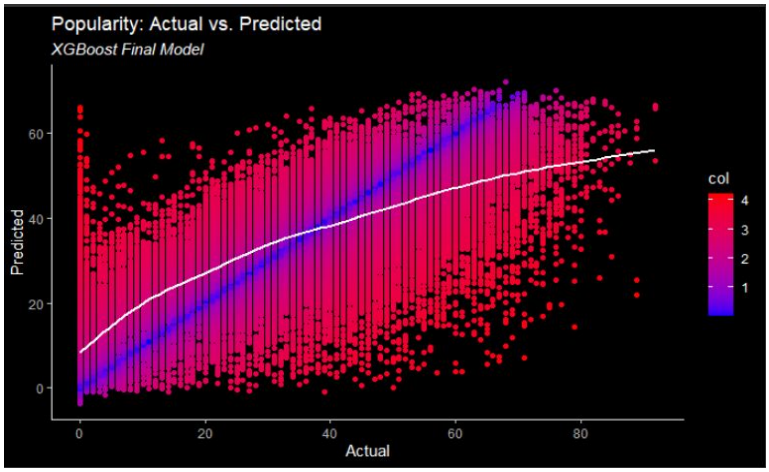


Following the tuning process, we reran a final XGBoost model with our optimal parameters to get a final RMSE. After tuning, our results showed us that the final XGBoost model had an RMSE of 13.26, which was about 0.2 lower than the original XGBoost model and slightly lower than the random forest model.

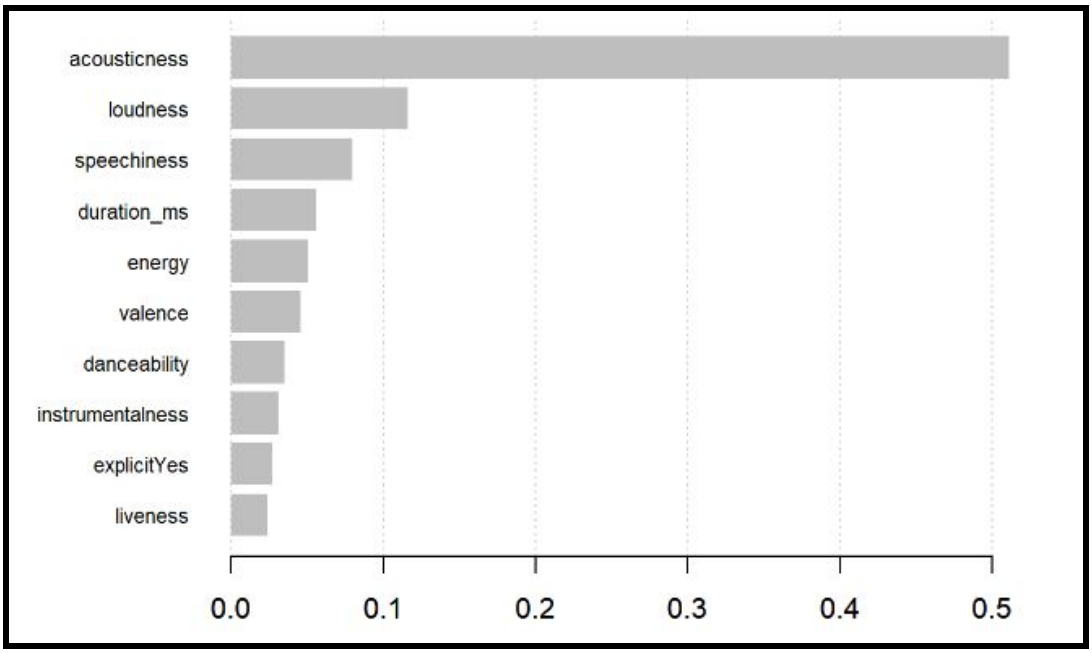


While that might not be a drastic decrease in RMSE, this tuning still led to increased predictive accuracy in our model. As seen in the below plots, the slope of our predictive model increased noticeably from our first XGBoost model to our final, with many more songs being accurately predicted in the test set with our final model, as evidenced by the larger size of the blue line in the from the first visualization to the second.

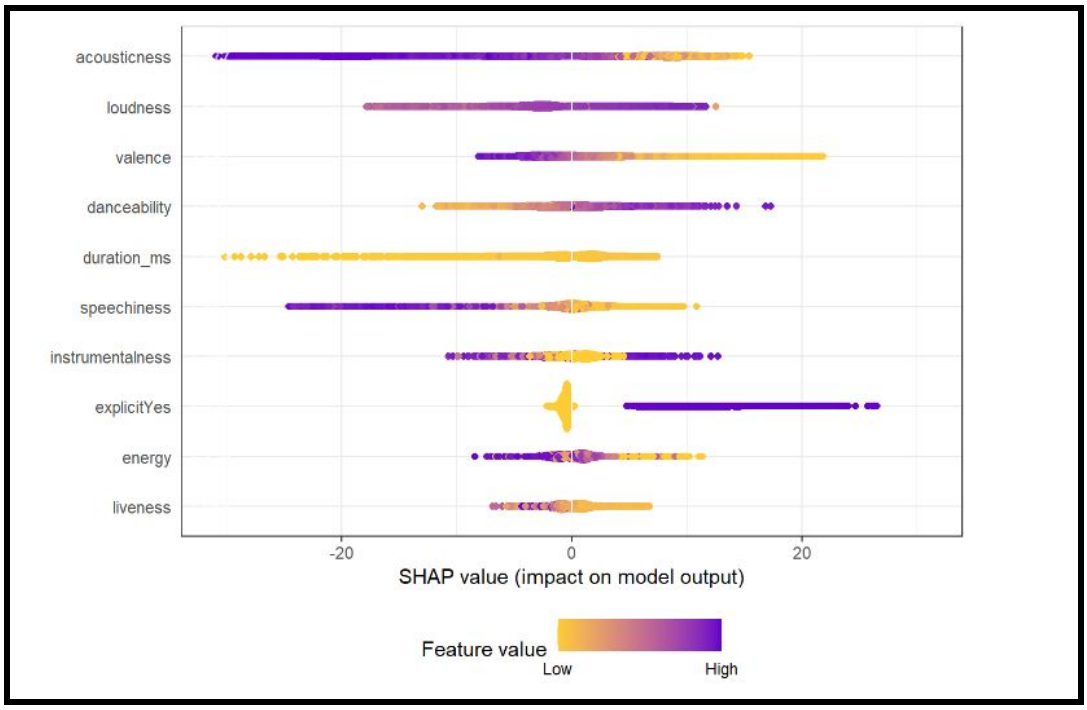




Using our final model, we plotted variable importance in determining popularity ratings for songs, which indicated that acousticness is the most important variable in determining popularity.



However, to see which amounts of accousticness and the other characteristics affect the popularity of a song, we plotted our variable importance using SHAP.



Although we did not implement this ourselves, theoretically, someone could take a new song that is not already in the dataset, plug the values of acousticness, danceability, etc, into our model, and get a predicted popularity of that song.

One challenge we experienced while completing this project was that in our initial attempt, the "year" variable completely dominated the variable importance plot. It had such a high variable importance compared to the other variables that you couldn't even tell what other variables had high importance. When we ran all of our models with the year variable included, there was a bigger drop in RMSE between the random forest and the final XGBoost model. We thought this model was going to be appropriate, because our RMSE was much lower - at around 9 - but then we plotted the variable importance, and realized that year was overshadowing all the other variables. So because of this, we decided to drop the "year" variable and re-run the model without it. The tradeoff was lower predictive accuracy/higher RMSE in our model, but this allowed us to get a better understanding of what other variables were actually contributing to a song's popularity, which resulted in the information explained above.

Discussion

The findings of this model show that acousticness is the most important variable in determining popularity of a song. The SHAP values explain how each variable is important in terms of affecting popularity. For example, the less acoustic a song is, the higher the popularity score of that song. The SHAP value plot also shows that a song with a higher "danceability" is more popular as well. This makes sense because the songs that play the most on the radio and are the most popular in our day-to-day lives are typically more dance style. In other words, these songs have more electronic beats, less guitar, and are faster-paced. One surprising insight gained from the SHAP value plot is that songs with lower levels of "speechiness" (relative length of the track containing any kind of human voice) actually tend to be more popular. We expected songs that were more "speechy" to be more popular, but maybe there is just a certain threshold of speechiness past which songs start to become less popular.

Overall, these results aren't too surprising when compared against current trends in the music industry. Electronic music is increasing in popularity every day, while music with fewer sounds of instruments in it is becoming the norm for the pop genre. Artists and music producers might find these results to be useful when creating new songs and making new music, in order to produce the most popular content for the airwaves. It seems like if you wanted to have the most popular song, you'd create one with low acousticness, higher loudness, lower speechiness, either a relatively high or relatively low level of instrumentalness, a higher level of explicitness, and a lower level of energy. However, something that is important to consider is the definitions of these variables - they don't all mean what you might expect them to just based on their name.

Conclusion & Future Work

In summary, we feel our project was very successful. We started with the Spotify dataset from Kaggle, and cleaned the dataset by removing 2 columns and deleting duplicates. Using Random Forest and XGBoost models, tuning the XGBoost model to increase predictive accuracy, we were able to obtain the variable importance of songs in the Spotify dataset to see which factors contributed most to a song's popularity level. We found that acousticness had the highest variable importance, and that lower levels of acousticness, higher levels of danceability, and lower levels of speechiness were all associated with higher popularity ratings. Artists, music producers, and record labels will all find these results useful in creating the most successful tracks that rise to the top of the charts.

If possible, our next step would be to re-score popularity, without having the artist name and year factor into the popularity rating. It would be really interesting to see if we could determine "relative popularity" and if the factors that affect a song's popularity change from year to year. We would also like to break the analysis down by genre, and see how the variable importance changes based on the genre of the song. Another way to use this dataset could be to see if we could figure out what values of each variable make up a specific genre of music. That way, musicians who want to break into a new genre or expand their audience could know precisely how to change their music to be more suitable to that new genre. Another step we might take given more time would be to allow the model to tune for more rounds in order to improve the modelling results.

Contribution

All three group members contributed to this report in different ways. While Kevin Ryan had a larger role in writing/running the code, Lauren and Kevin Buchanan focused more on the report and presentation. Of course, all three members contributed to the R-code, report, and presentation, there were just different focuses for everyone.

Bibliography

Ay, Yamac Eren. “Spotify Dataset 1921-2020, 160k+ Tracks.” Kaggle, 11 Oct. 2020, www.kaggle.com/yamaerenay/spotify-dataset-19212020-160k-tracks/tasks.

Chiru, Costin, and Oana-Georgiana Popescu. “Automatically Determining the Popularity of a Song.” SpringerLink, Springer, Cham, 3 July 2017, www.link.springer.com/chapter/10.1007/978-3-319-60837-2_33.

Dhanaraj, Ruth, and Beth Logan. "Automatic Prediction of Hit Songs." *ISMIR*. 2005.

Myra, Interiano, et al. “Musical Trends and Predictability of Success in Contemporary Songs in and out of the Top Charts.” Royal Society Open Science, 16 May 2018, www.royalsocietypublishing.org/doi/full/10.1098/rsos.171274.

West, Tyler. "Going viral: Factors that lead videos to become internet phenomena." *The Elon Journal of Undergraduate Research in Communications* 2.1 (2011): 76-84.