Spotify ML Project: Final Report

Abstract

This study utilizes song data from Spotify, obtained from Kaggle and originally sourced from the Spotify API, to predict the popularity of songs released between 2004 and 2019. By employing a Random Forest Model and utilizing predictors such as album release date, danceability, energy, key, loudness, mode, speechiness, acousticness, instrumentalness, liveness, valence, tempo, and duration, we aim to predict track popularity, which represents a rating of popularity made by an algorithm output. This algorithm is based on the number of plays and how recent the plays are. Our primary goals are to identify the characteristics that most influence song popularity and to evaluate how accurately we can predict future song popularity using these variables.

The data was split into training and testing sets, and 3-fold cross-validation was used to optimize model parameters. Feature engineering techniques were employed, including one-hot encoding for categorical variables and Principal Component Analysis (PCA) to reduce multicollinearity, although PCA was found to be unnecessary for this study. A Random Forest Model was fit with the optimal parameters: max_depth = 10, max_features = ‘sqrt’, min_samples_split = 3, min_samples_leaf = 1, and n_estimators = 100.

The model's performance was evaluated using R^2 and RMSE metrics on both the training and testing data sets, with the final model explaining 22.8% of popularity variance within the testing set. The most influential predictors in determining song popularity were: ‘year’ (year the song was released), ‘instrumentalness’ (likelihood of the song having no vocals), ‘duration_ms’ (length of song in seconds), ‘energy’ (perceived energy/intensity of a song), and ‘loudness’ (average loudness in dB). The findings of this study can provide valuable insights for music producers, artists, and industry stakeholders, helping them understand and predict a song's chance of success and what factors contribute to the popularity of a song.

Introduction

The music industry is a dynamic and multifaceted domain where understanding the factors that contribute to a song's popularity can significantly impact production decisions, marketing strategies, and the overall success of artists and producers. In such a competitive industry, it is difficult to know where to start or how to strategize when producing a song. With the development of digital music streaming platforms like Spotify, Apple Music, Amazon Music, and more, vast amounts of data have become available, offering new opportunities to analyze musical trends. In this study, we will attempt to explain some of the existing uncertainty and present music success in a data driven way. We will explore the determinants of track popularity using an extensive dataset sourced from the Spotify API, encompassing over 32,000 songs released between 2004 and 2019. Our primary objective is to predict the popularity of these tracks based on various musical and contextual attributes provided by the Spotify API. The dataset includes a range of variables such as album release date, danceability, energy, key, loudness, mode, speechiness, acousticness, instrumentalness, liveness, valence, tempo, and duration. Each song in the dataset is represented by these variables along with a popularity score, which serves as the response variable or prediction in our analysis. Note that the popularity score is a rating of popularity made by an algorithm output, which is based on the number of song plays, as well as how recent the plays are.

Our research seeks to answer two fundamental questions:

- **Which song characteristics are most useful in predicting the popularity of a song? In other words, what characteristics make a song popular?**
- **How well can we predict song popularity for future songs, given the variables provided by the Spotify API?**

To answer these questions, we will use supervised learning for regression by creating a Random Forest Model. We will also perform PCA and 3-fold cross-validation prior to fitting our Random Forest Model.

Before jumping into the model building process, it is first important to discuss feature engineering, which plays a crucial role in setting up an analysis. Since most machine learning algorithms can only handle numeric data, we had to change categorical variables within our data to numeric ones. To do this, we leveraged one-hot encoding for categorical variables like mode, key, year, and month. After one-hot-encoding our categorical variables, we decided to perform PCA, since some of these variables (such as `instrumentalness` and `speechiness`) may provide overlapping information. PCA transforms a set of correlated variables into a smaller number of uncorrelated variables, which is meant to reduce redundancy within the data and protect against overfitting. When performing this PCA, however, our analysis revealed that it was unnecessary, as the predictors were not highly correlated and using numerous components would ultimately complicate the interpretation.

After the feature engineering, we split the data into training and testing sets, with 80% of the data allocated for training and 20% for testing. This was done so that the model performance could be assessed on data never seen before by the model, which is especially important in the context of overfitting. After splitting the data, we conducted 3-fold cross-validation to optimize hyperparameters such as the number of trees, maximum tree depth, and minimum samples per leaf, ultimately selecting the best-performing parameters based on their R^2 values. This cross validation was done to ensure that our model used parameters that optimize performance. Note that the training set was used to perform this cross validation, as we wanted to keep the testing data away from the model, to ensure accurate assessment of overfitting. The CV yielded the highest R^2 value for max_depth = 10, max_features = 'sqrt', min_samples_split = 3, min_samples_leaf = 1, and n_estimators = 100.

Moving on to the actual model building, again, we performed supervised learning for regression, as our data is labeled and the response variable is quantitative. More specifically, we used a Random Forest Model for this analysis, because of its robustness in handling predictor multicollinearity and its capability to work with non-normally distributed response variables. Note that we originally decided to perform PCA before the Random Forest Model because Random Forest Models can oftentimes take a long time to run, especially when the data is highly dimensional. While our data is likely not large enough for this to actually affect our analysis, we figured it would be good practice to go ahead and reduce the dimensionality of the data prior to running the Random Forest Model. Random Forest Models are also able to provide insights on feature importance, which is especially informative in the context of determining song popularity, and it helps us answer one of our research questions. We did consider other numerical predictor models for this analysis, but ultimately we opted for Random Forest due to suitability for our specific data and interests.

The Random Forest Model was fitted for the training data, using the specific optimal parameters decided by our cross-validation. Running this model involved bootstrapping, which creates multiple subsets of the training data, enhancing the robustness of the model and limiting bias. While we simply had to fit the random forest model with the best predictors, this bootstrapping was happening under the hood. The performance of our model was evaluated through metrics such as R^2 and RMSE, providing insights into its predictive accuracy. Our results indicate that the model explains 22.8% of the variance in song popularity within the testing dataset, and 33.5% within the training set. The key predictors were found to be: the year of release, instrumentalness, duration, energy, and loudness. These findings offer valuable insights for music producers, artists, and industry stakeholders, highlighting the attributes that most influence a song's success.

Despite the challenges encountered, including data cleaning, multicollinearity, and potential overfitting, our study demonstrates that specific musical and contextual attributes do impact a song's popularity. By addressing these challenges and thoroughly evaluating the model's performance, we were able to create a model that predicts song popularity better than a random guess. While the training performance was slightly better than the testing performance, the

overfitting within this model is likely small, given that these two R^2 values are fairly close together. This analysis will hopefully help inform future musical trends and production decisions, ultimately aiding in the creation of more successful tracks. The comprehensive analysis provided here underscores the potential of machine learning models to transform the way we understand and predict musical success in the digital age.

Data

Summary

The data we will be utilizing in this project is Spotify data. This data was found in Kaggle, but was originally pulled from the Spotify API. Each row in the data is one song that was released between 2004 and 2019. In this data, the variable we will be trying to predict is popularity (track_popularity). We will use the following predictors to help predict popularity: album release date, danceability, energy, key, loudness, mode, speechiness, acousticness, instrumentalness, liveness, valence, tempo, and duration. Below is a data dictionary which describes the meaning of each of the variables we are focusing on:

Variable	**Variable Name in Data**	**Description**
Popularity *(response variable)*	track_popularity	Calculated by an algorithm and is based, in the most part, on the total number of plays the track has had and how recent those plays are.
Album Release Date	track_album_release_date	Provides the date an album was first released.
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
Energy	energy	A value between 0 and 1 that represents the perceived intensity and activity level of a song, essentially indicating how energetic or upbeat a track feels, with higher values signifying more energetic songs that are typically fast, loud, and noisy.
Key	key	The key that the song is in
Loudness	loudness	Provides the overall loudness of a track

		in decibels (dB)
Mode	mode	Indicates the modality (major or minor) of a track, the type of scale from which its melodic content is derived. Major is represented by 1 and minor is 0
Speechiness	speechiness	Detects the presence of spoken words in a track. The more exclusively speech-like the recording (e.g. talk show, audio book, poetry), the closer to 1.0 the attribute value.
Acousticness	acousticness	A confidence measure of how likely a track is to be acoustic, ranging from 0.0 to 1.0. A score of 1.0 indicates a high confidence that the track is acoustic.
Instrumentalness	instrumentalness	An audio feature that predicts if a song contains no vocals. The instrumentalness value is a number between 0.0 and 1.0, with values closer to 1.0 indicating a greater likelihood of no vocals.
Liveness	liveness	Detects the presence of an audience in the recording. Higher liveness values represent an increased probability that the track was performed live.
Valence	valence	A numerical value between 0 and 1 that indicates how positive or "happy" a song is perceived to be, with higher values signifying a more positive mood and lower values indicating a more negative mood
Tempo	tempo	The estimated overall tempo of a track in beats per minute (BPM)
Duration	duration_ms	A number that represents the length of the section in seconds

Usefulness of Data

This data provides several variables that will allow us to determine and predict the trends of music across a span of 15 years. This will allow us to analyze and observe how musical preferences can evolve over time such as through shifts in genre or production styles. Using a

machine learning model will allow us to forecast popularity trends in the future based on the variables above. We expect to build a model that will be able to analyze the numerical inputs of each variable and predict the popularity of the hypothetical song.

This type of analysis could be useful to music producers, artists, or anyone who is putting money and time into song production. It can let them know of the factors that may cause their song to be more popular and have more success, which can inform their artistic decisions.

Expected Challenges

During our data cleaning process, we did run into a few challenges that we were able to work through. Firstly, some of our categorical variables were labels with numbers. We wanted to make sure that we didn't misinterpret these variables to be quantitative variables down the line, so we decided to relabel these variables with the class they actually represent, to be easier to interpret. Additionally, certain variables were not useful for analysis, so we decided to remove them from the data. Lastly, cleaning the date released variable required several steps. First of all, the year variable had years ranging from 1960 all the way to 2019. Most songs were made past 2000, so we decided to limit this analysis to songs in the 2000s. Additionally, we split the date variable into a year and a month variable, because these will be more appropriate for analysis as they follow either a continuous, linear timeline or can be utilized as categorical dummy variables.

Since some of these variables are difficult to interpret literally (as they are based on models/probabilities themselves), we hypothesized that we may run into issues with interpretation of our model and predictor values. Our response variable is based on an algorithm as well, which is not ideal for interpretation as it is numerically more subjective. Several of our predictors are subjective measures as well, such as 'energy' representing the 'perceived' energy of a song. This is not a numerical calculation innately, so it is difficult to explain what it measures and if that is always an accurate measurement.

Additionally, because some of our variables are likely related (such as instrumentalness and speechiness), we predicted that we may have some issues with multicollinearity in our data. Multicollinearity occurs when variables in a model are giving redundant information in predicting the response variable. To address this issue, we will assess the between-predictor correlations and consider dimensionality reduction to ensure our final model predictors are unique from one another.

Analysis Methods

Introduction and Analysis Questions

As previously mentioned, our study is centered around Spotify data. More specifically, we are interested in understanding more about the popularity of Spotify songs. As a reminder, observations within this study are individual songs that are represented by various features provided by the Spotify API along with popularity scores. Each observation includes the variables: danceability, energy, key, loudness, mode, speechiness, acousticness, instrumentalness, liveness, valence, tempo and duration. Each song represents an observation with its respective feature values and a popularity score. With this being said, our primary goal in this analysis is to better understand these questions:

1. **Which song characteristics are most useful in predicting the popularity of a song? In other words, what characteristics make a song popular?**
2. **How well can we predict song popularity for future songs, given the variables provided by the Spotify API?**

To answer these questions, we will perform supervised learning for regression, as our data is labeled and the response variable is quantitative. To perform this regression analysis, we will fit a Random Forest Model. We will make predictions using a Random Forest Model, assess the accuracy of our predictions, and then analyze the feature importance of the model to see which variables are most important in predicting Popularity.

Model Selection and Considerations

The Random Forest Model was chosen because it performs well for predictors that may have some multicollinearity. As previously mentioned, some of the variables within our data are likely to give redundant information, as they are very similar. This idea was proven by a correlation matrix showing high correlations between certain predictors. Additionally, the Random Forest Model does not need the response variable to be normally distributed. Based on our EDA, we found that the Popularity variable is almost bimodal, as so many songs had a ranking between 0 and 10. For this reason, the normality assumption for linear regression may not be met and the Random Forest Model is more appropriate.

One consideration before fitting the Random Forest Model is the parameters that will be used. Some key parameters for a random forest model include: number of trees, maximum tree depth, and minimum samples per leaf. To find the optimal parameters for our model, we will use cross-validation to assess model performance on the training data via the R^2 value. We will test several values of each parameter based on common values of the parameter and what we hypothesize will work for our model. Note that this cross-validation will be done only on our training data, so that the testing data remains unseen by the model. This process will inform us of the parameter values to use in our final model that will best balance underfitting/overfitting with

accurate predictions. We will finally check for underfitting/overfitting of the model using the R^2 of the testing data to compare to that of the training data. A high R^2 on the training data with a low R^2 on the testing data will indicate overfitting, and we will reconsider our parameters if this is the case.

Before performing this analysis, we will split the data into a training set and a testing set. The training data will be 80% of the original data, while the testing data will be 20% of the data. This split was done so that the model performance could be assessed on data never seen before by the model, which is especially important in the context of overfitting. Oftentimes, machine learning models will include the noise in the training data in order to explain the variation in the response, instead of the true relationships and trends within the data. To protect against this, it is important to compare the performance of the model for the training data, versus the data never seen by the model before. We will then plug the training data into the Random Forest model, to train the model. During this process, a decision tree is built for each subset of the training data. Once all trees are built, predictions will be made by taking the average of all decision tree predictions. After the model has been trained, we will assess the accuracy of the model using data that has never been seen by the model (the testing data). This will be done so that we can analyze how well our model predicts popularity, as well as check for any overfitting/underfitting as previously described.

Feature Engineering

To prepare our dataset, we implemented several feature engineering techniques aimed at enhancing the predictive power and usability of the data. First, we handled categorical variables such as ‘mode,’ `key,’ `year,’ and `month.’ The `mode` variable, which has already been relabeled as "Major" or "Minor," was one-hot encoded to convert these categories into binary indicators, allowing the model to incorporate modality, the type of musical scale used in a piece which gives the music its particular tonal quality or “mood,” without introducing numerical bias. Similarly, the `key` variable, which has been mapped to note names, and the `month` and `year` variables underwent one-hot encoding, allowing each distinct category to contribute separately without introducing multicollinearity.

In addition, we aim to address potential multicollinearity among numeric predictors, which include features like `danceability,` `energy,` `loudness,` `speechiness,` `acousticness,` `instrumentalness,` `liveness,` `valence,` and `tempo.` Since some of these variables (such as `instrumentalness` and `speechiness`) may provide overlapping information, we can calculate the correlation between the variables to identify highly correlated pairs. For variables showing strong correlations, we considered applying Principal Component Analysis (PCA) to reduce redundancy. PCA transforms a set of correlated variables into a smaller number of uncorrelated variables. Each principal component is a linear combination of the original variables and is designed to capture as much of the data's variance as possible in fewer dimensions. For example, variables like `danceability`, `energy`, and `loudness` are often correlated in music data, so PCA

would combine these into new components that represent the core information without redundancy. This reduction in multicollinearity can make the model more robust by minimizing redundant information, while the reduced feature set improves computational efficiency. However, if principal components that explain a large majority of the variance of the model are not able to be found, PCA will not be necessary for our analysis and would instead just blur our interpretation of the results.

Assessing the Model

After fitting the model, we will analyze the accuracy of our model in order to interpret how well our model works for predicting the popularity of a song. To do this, we will first calculate the R^2 value, which will tell us what proportion of the variance of the independent variable is explained by our model. A high R^2 value indicates that the model is a good fit for prediction. An R^2 of 0.5 indicates that half of the variance is explained by the model, which is not extremely strong, but is typically considered decent. As this model is on Spotify data and is not high-stakes, this would be our goal for an R^2 value, although we hope for an R^2 value of closer to 0.8.

We will also calculate the R^2 value for the training set, which is the dataset that we initially trained our model on. Ideally, the R^2 value should be approximately equal for the training set and for the test set, because this means there is no overfitting or underfitting occurring. If the R^2 value is much higher on the training set than the test set, we know that our model is overfitting the data, which means we likely have too many predictors which make our model no longer generalizable.

To calculate another measure of accuracy, we will calculate the root mean squared error (RMSE) of the predictions we made on the test set. Since this value represents the error we made in our predictions, a higher RMSE indicates that our model is not a good fit as it currently stands.

In order to reach the highest possible R^2 value for our model, we will perform cross-validation to determine the optimal parameter values to set for the model. We will use the output from the CV to refit the Random Forest Model with its optimal parameters. However, 'max_depth' being intentionally set lower (as well as other parameter alterations) could help combat overfitting, if that is a problem we find after refitting the model.

To present our results from this model building process, we will examine our final model on the training set, the predictions on the test set, the R^2 values for each set, and the root mean squared error of the predictions on the test set. This will give us a picture of how successful our model is at predicting the popularity of songs, which can allow us to determine if this model would be useful to our stakeholders.

Possible Challenges

With this model, we anticipate running into challenges that stem from the fitting of the model. Random Forest models tend to have a problem with overfitting on complicated datasets. Another source of fitting problems could stem from hyperparameter sensitivity of the tree, like maximum depth and number of observations per branch. This will be assessed thoroughly with RMSE calculations and adjustments made to the model to maximize R^2. If our model does perform poorly due to parameter selection, this may help us learn more about how to tweak the parameters of a Random Forest Model to correctly account for overfitting/underfitting.

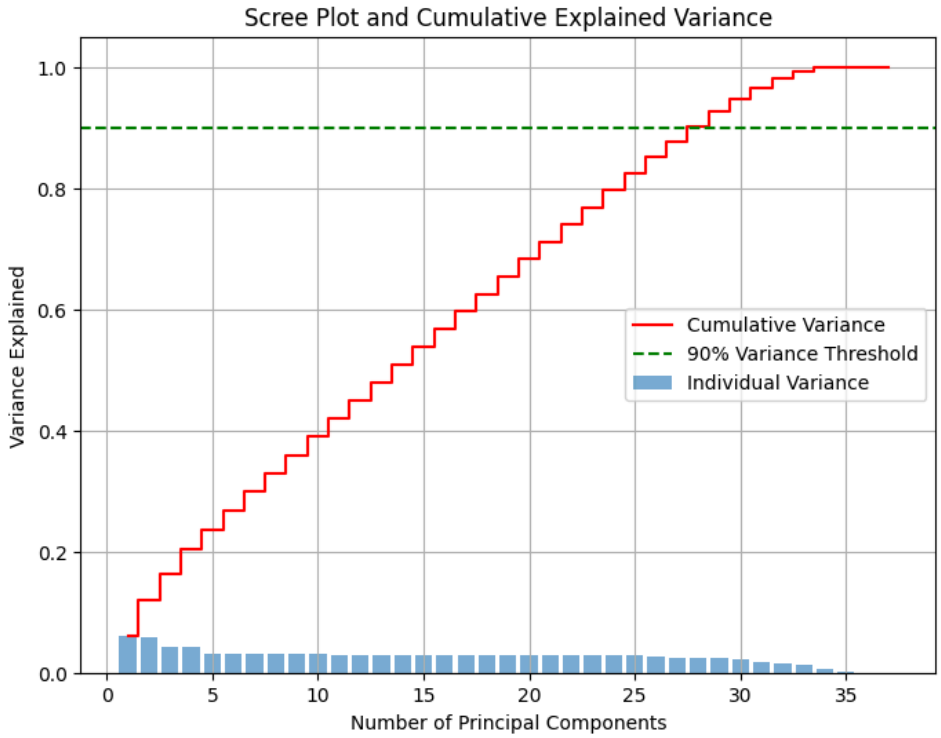
Another challenge we expect with this model is the feature importance score calculations. The feature importance poses a higher threat for data sets that tend to have variables or parameters with coinciding or codependent information. With the analysis for multicollinearity of the parameters, this challenge should be significantly reduced, as we will either use PCA to reduce this collinearity, or we will conclude that it isn't impactful enough to be problematic in our model fitting. There could also be a challenge with extrapolation of the model to apply to other song data.

Although we have properly cleaned and explored our data from the Spotify API, the data is still limited, as it only includes songs that are featured on Spotify released between 2004-2019. This could cause an imbalance in the data used to train and test the model, as some older songs (before 2004) and new songs may have parameters that are technically outside of the model's range. By ensuring a balanced train and test data split, this model will allow us to assess the importance of the parameters in determining the popularity of a song.

Results

Principal Component Analysis

In order to select the number of components to use for PCA, we created a scree plot to show the extent to which the variance is explained by each principal component.



The red line represents the cumulative variance achieved by using the amount of components listed on the x-axis in our model building. Since no components explain a majority of the variance, and the 90% variance threshold is not achieved until principal component 28, it is clear that PCA is not necessary for our dataset. The predictors are not correlated with one another enough to build principal components that would be more useful in our analysis than the predictors themselves. If we were to use 28 components, it would be extremely difficult to interpret. Therefore, we will move forward with our model building with the original dataset; we will not use PCA for this model.

Random Forest Model

In order to create our random forest model to predict song popularity, we first performed cross-validation for hyperparameter tuning to determine the optimal values for maximum depth of the trees and the number of trees to calculate in the random forest. We performed 3-fold cross-validation on the training set to test:

- max_depth = 3, 5, 10
- max_features = ‘sqrt’, ‘log2’

- min_samples_split = 3, 5, 10
- min_samples_leaf = 1, 2, 3
- n_estimators = 50, 100, 150

The CV yielded the highest R^2 value for max_depth = 10, max_features = 'sqrt', min_samples_split = 3, min_samples_leaf = 1, and n_estimators = 100. We refitted the Random Forest Model with these parameter values on the training set. Then, we predicted on the test set using this model to determine model accuracy and how well it works for prediction.

Training Statistics:

```
SSE (Train): 9447794.79
MSE (Train): 420.72
RMSE (Train): 20.51
R² (Train): 0.336
```

Testing Statistics:

```
SSE (Test): 2693435.91
MSE (Test): 479.69
RMSE (Test): 21.90
R² (Test): 0.228
```

The results from our Random Forest Model with the optimized parameters are above. The root mean squared error of this model for the test set was 21.90 and the R^2 value of this model for the test set was 0.228. An R^2 value of 0.228 means that about 22.8% of the variance in popularity of Spotify songs is explained by the predictors in our model. Although this isn't as high as would be ideal for prediction, the model is a better prediction than random guessing, so this does provide some information about factors that may make a song more popular. Additionally, we calculated the R^2 of the training set to be 0.336. Since this is similar to that of the test R^2 but still higher than it, we know that overfitting was not an issue.

Feature Importance

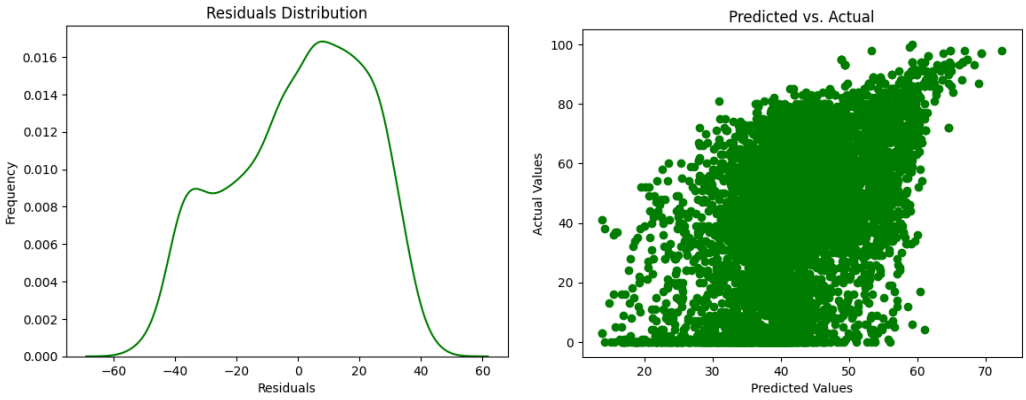
Next, we calculated the feature importance for each variable to see which of our predictors were most useful in predicting the popularity of a song.

	Feature	Importance
10	year	0.242562
5	instrumentalness	0.121628
9	duration_ms	0.085787
1	energy	0.079647
2	loudness	0.071333
4	acousticness	0.071176
8	tempo	0.056896
0	danceability	0.053132

```
3          speechiness    0.046777
7              valence    0.045560
```

The output (for the top 10 most important features) is above. The features that contributed the most to the outcome of the model equation were ‘year’, ‘instrumentalness’, ‘duration_ms’, ‘energy’, and ‘loudness’. So, to answer our previously mentioned research question: the song characteristics that are most useful in predicting the popularity of a song are the year it was made, the instrumentalness, the duration, the energy, and the loudness.

Model Error and Residuals



Looking at the plot to the left above, we can see that the distribution of the residuals is centered around 0. Additionally, note that there is a fairly weak trend between the predicted and actual values for song popularity, as shown in the scatterplot to the right. While we can see that there is a trend between these two values, there does seem to be a substantial dispersion, particularly for mid-range and low popularity scores.

There seems to be bias towards low popularity in the model. Many of the songs are clustered at low popularity levels with few songs accurately achieving a high popularity score. This makes sense because few songs become a “hit,” which would be the ones with extremely high popularity scores. It is interesting, however, that our model appears to be more accurate for predicting the songs that are extremely popular, but not the low-to-mid-range songs. Therefore, this model could be useful in situations in which a music producer or artist is working on a song that they hope to become extremely popular. The Random Forest Model, especially the features mentioned previously as the most important, will help to predict popularity of a Spotify song.

Conclusion

Findings

In this study, we explored the relationship between various song characteristics and their impact on Spotify song popularity, aiming to identify the most influential factors on song popularity and evaluate the predictive performance of a Random Forest Model. Our findings indicate that the year of release, instrumentalness, duration, energy, and loudness are the most significant predictors of a song's popularity, which answers our first research question.

Using the testing dataset, we found that our Random Forest Model explains around 22.8% of the variation in song popularity. This R^2 value is relatively low, especially given the complexity and power of Random Forest Models. This may indicate the intricate and complex nature of the music industry. As for our second research question, “How well can we predict song popularity for future songs, given the variables provided by the Spotify API?”, we would cautiously conclude: “not extremely well.” It appears that what drives a song's popularity may be partly random or strongly correlated with factors that are not quite captured in the available data. In general, song popularity can be somewhat random or based on luck, so although a model with better predictive power would be ideal, we think it would be difficult to achieve.

Although the model's predictive power is limited at around a 23% explained variance, it still performs better than random guessing and it highlights important trends within the music industry. These insights can offer value to stakeholders, such as music producers and artists, by highlighting the elements that contribute to a song's success. Future analysis could expand upon this baseline and determine what really sets a song up for likely success in such a competitive industry.

Limitations and Challenges

Despite its insights, there are still many limitations that present opportunities for better prediction. One notable limitation is the scope of the dataset, which included only Spotify songs released between 2004 and 2019. Expanding the dataset to include songs outside of this range, by including more current songs or those from other platforms, could improve the generalizability of the findings. Additionally, the use of the "popularity" variable, which is algorithmically derived, poses challenges in interpretation. Exploring alternative metrics such as the "number of hits" could provide more straightforward insights.

Throughout this study, we encountered several challenges pertaining to the training of our model. For instance, we began tuning the max depth parameter with “None” as an option. We quickly realized that allowing this option for maximum depth was leading to extreme overfitting within our model. We learned that ensuring optimal parameter selection is critical to avoid these issues in further studies. Additionally, Random Forest models, while powerful, can be a bit challenging to interpret. Exploring simpler models, such as linear regression, could offer greater

insight into the significance of individual predictors rather than grouping the predictors all together.

This leads us into the problem of interpreting our model. Random Forest models are known to provide high predictive accuracy but are often seen as "black boxes" because they do not easily reveal the process by which the predictions are made. To mitigate this, future studies could explore the use of simpler models, such as linear regression or decision trees, which offer more straightforward interpretations of the relationships between predictors and the response variable.

It's also worth considering the impact of external factors not captured in the current dataset. Elements such as marketing efforts, social media influence, cultural trends, and collaborations with popular artists can significantly affect a song's popularity. Incorporating such variables could potentially enhance the predictive power of the model. For example, data from social media platforms like Twitter or Instagram could provide real-time insights into trends and public reception of songs. The model encompasses songs up to 2019, but since the COVID-19 pandemic in 2020, there has been a surge in media consumption, as well as a drastic increase in social media influence. In the past four years, the quickly growing media platform, TikTok, has had an influential impact on the popularity of new music. TikTok has increased the number of hits on new and old songs alike when there are viral trends associated with them. This is not something that has necessarily been quantified in song data in previous research, however, the influence of social media should somehow be assessed and considered in a future application of this model.

Model Extensions

Future research could take several promising directions to enhance the model's predictive power and interpretability. For instance, transforming the regression problem into a classification task to predict whether a song becomes a "hit" could offer new insights. This approach would involve defining a threshold for what constitutes a "hit" and examining whether feature importance changes under this framework. We hypothesize that this would especially be useful considering that our model's predictive power was better for those that were a "hit," and fell short in the middle gray area of popularity. A classification model may work better for prediction considering these circumstances, and would still provide the same insight.

Another extension could be to develop models that focused on one particular artist, such as creating a model solely for Taylor Swift songs. Because fans' expectations and artists' unique styles may significantly impact which characteristics make a song popular, models like these could reveal how the predictors of popularity vary based on artists. Separating the models by artist could also increase each model's performance. Since it is very possible that what makes a song a "hit" changes drastically by who is singing it, in our model we may be "canceling out" the effects that make a song popular for a certain artist by including all artists in our analysis. Similarly, examining the predictors of popularity within specific genres such as R&B, rock, and

pop, could highlight genre-specific trends and factors, providing more tailored insights for artists and producers within those genres.

Lastly, investigating how the characteristics that make a song popular have evolved over time could offer valuable insights into changing musical styles and preferences. This could be done by splitting the data by year, making a model for each year, and analyzing the feature importance for each model. Analyzing these trends could reveal interesting shifts in what listeners value most in different eras.

Concluding Thoughts

In conclusion, while this study does provide valuable insights into the factors influencing Spotify song popularity, there are numerous opportunities for refinement and expansion. By addressing the identified limitations and exploring suggested extensions, future research can build on these findings to develop more accurate and interpretable models. This could ultimately aid stakeholders in making informed decisions in the dynamic music industry. The comprehensive analysis provided here underscores the potential of machine learning models to transform the way we understand and predict musical success in the digital media age. By continuously improving our analytical approaches and incorporating a broader range of data, we could gain deeper insights into the ever-evolving landscape of music popularity.

References

Arvidsson, J. (2023, November 1). *30000 Spotify Songs*. Kaggle. https://www.kaggle.com/datasets/joebeachcapital/30000-spotify-songs

Spotify Web API. Web API | Spotify for Developers. (n.d.). https://developer.spotify.com/documentation/web-api