Building a Model to Forecast a Song’s Popularity on Spotify

If you're an artist, producer, or music enthusiast, knowing how a song will perform ahead of time would be priceless. In this article, I explain the steps involved in building a predictive model that can forecast a track’s popularity on Spotify.

by John Hopkins

Collecting the Data

The quality of data is crucial for building a reliable model. The dataset was chosen from Kaggel (🎹 **Spotify Tracks Dataset | Kaggle**, it contains Spotify tracks over a range of 125 different genres initially obtained from the Spotify API. The data was released by Spotify on the 22nd of October 2022. I carefully reviewed and processed the data, ensuring that it was clean and consistent. The resulting data was 15 columns and 112,059 rows with 1,680,885 samples.

Strategies for Data Cleaning

> "Garbage in, garbage out" – the quality of a model depends on the quality of the data it uses. I applied automation and techniques such as machine learning-based methods, to ensure clean data and avoid bias in the results.

Data Cleaning Methods

No duplicates or missing data was found. The following features were deemed not relevant to building a song popularity model: 'Unnamed 0', 'track_id', 'artists', 'album_name', and 'track_name'. I then removed non-music tracks by filtering out tracks with a 'speechiness' score above 0.7.

The following features were label encoded and treated as categorical: 'mode', 'key', and 'time_signature'. As per Spotify's API documentation the 'time_signature' feature has a range from 3-7. I filtered out values less than 3 that were found. I dropped all entries of 'duration_ms' that were equal to 0.

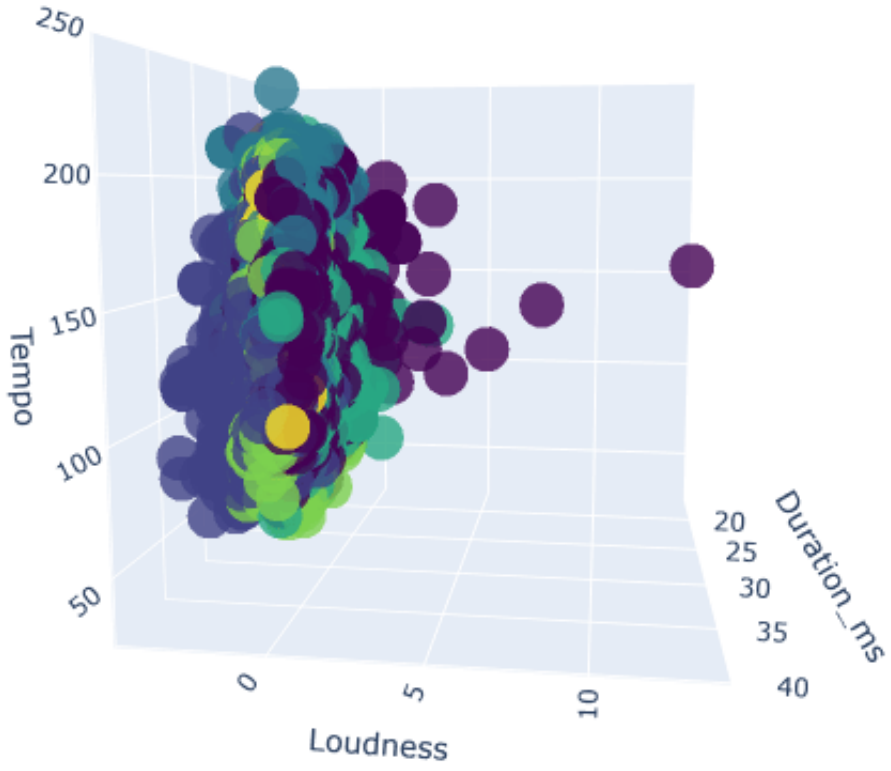
Outliers were determined using a z-score with a threshold of 3 and visualized with boxplots. The outliers for 'speechiness', 'instrumentalness', and 'time_signature' were not removed. This data is too important and can be easily misleading if treated. For example, treating 'speechiness' and 'instrumentalness' may remove rap songs on one end and classical on the other, both of which would have components of making a popular song. The features 'duration_ms', 'liveness', and 'loudness' were transformed using the Yeo-Johnson method and the remaining percentage of outliers was determined via the IQR method.

Cleaning Continued

The distribution of the numerical features was evaluated, and none were found to be normally distributed. Histograms and the Kolmogorov-Smirnov test were used. Several methods (log, sqrt, exp) of transformation were evaluated, however none produced any significant improvement.

K-means and PCA were used to try and determine any patterns or relationships in the data. The data was found to be overlapping and poorly separated into distinct clusters. The image below illustrates this point.

3D Scatter Plot of Clusters



Selecting the Right Features

Audio Features

Spotify provides audio features such as tempo, key, and loudness. I tried to leverage these features to determine how they affect a song's popularity.

Features Analysis

By analyzing the contents of the dataset, I was able to choose the relevant features and eliminate some that may bias the results. I also was able to choose the target variable and independent variables.

Analysis of Features

None of the continuous features had a strong correlation with the target variable. The features 'loudness' and 'energy' showed multicollinearity at a value of 0.77, therefore 'energy' was dropped, and loudness chosen because it's a feature that can be measured and not susceptible to human bias.

All of the remaining categorical features were unbalanced. The categorical features were label encoded and included in the model.

Categorical Features

The categorical features revealed some interesting insights.

Although the majority of songs in the data were not explicit, explicit songs had a higher mean popularity score.

Songs in minor had the same effect. The majority of songs were in major but minor had the edge in mean popularity score.

Songs in E slightly beat out the rest in mean popularity score.

The majority of songs were in 4/4 time and it also scored highest in mean popularity.

The top genre with a mean popularity score of 59 out of 100 was pop-film.

Selection and Evaluation of the Model

	Model	MAE	MSE	RMSE	R2	RMSLE	MAPE	TT (Sec)
rf	Random Forest Regressor	13.0008	303.6010	17.4240	0.3968	1.0608	0.8386	2.4700
et	Extra Trees Regressor	11.9644	311.6167	17.6525	0.3809	0.9657	0.7581	0.9900
xgboost	Extreme Gradient Boosting	17.2950	432.8782	20.8055	0.1400	1.4108	1.2954	1.0560
catboost	CatBoost Regressor	17.5633	443.9049	21.0689	0.1181	1.4308	1.3297	1.6420
knn	K Neighbors Regressor	16.7826	459.5787	21.4377	0.0870	1.2608	1.0692	0.7380
lightgbm	Light Gradient Boosting Machine	17.9977	460.7039	21.4639	0.0848	1.4597	1.3830	0.2900
gbr	Gradient Boosting Regressor	18.5170	484.9387	22.0212	0.0366	1.4888	1.4474	1.4300
ada	AdaBoost Regressor	18.7949	496.8372	22.2897	0.0130	1.5063	1.5010	0.4300
br	Bayesian Ridge	18.9430	501.1640	22.3866	0.0044	1.5009	1.4815	0.1540
ridge	Ridge Regression	18.9412	501.1839	22.3871	0.0043	1.5008	1.4817	0.5880
lar	Least Angle Regression	18.9412	501.1839	22.3871	0.0043	1.5008	1.4817	0.5700
lr	Linear Regression	18.9394	501.2078	22.3876	0.0043	1.5007	1.4817	0.6640
huber	Huber Regressor	18.9263	501.4774	22.3936	0.0038	1.5034	1.4928	0.1900
omp	Orthogonal Matching Pursuit	18.9688	502.1637	22.4089	0.0024	1.5018	1.4796	0.1460
en	Elastic Net	18.9895	502.3947	22.4141	0.0019	1.5024	1.4800	0.6000
lasso	Lasso Regression	19.0070	503.1660	22.4313	0.0004	1.5028	1.4792	0.5900
llar	Lasso Least Angle Regression	19.0070	503.1660	22.4313	0.0004	1.5028	1.4792	0.1520
dummy	Dummy Regressor	19.0129	503.4089	22.4367	-0.0001	1.5029	1.4791	0.1620
dt	Decision Tree Regressor	15.8694	548.4548	23.4189	-0.0896	1.3267	0.9234	0.2400
par	Passive Aggressive Regressor	22.7355	781.7740	27.9322	-0.5528	1.5971	1.6104	0.1640

The model with the best performance was the Random Forest Regressor with an R^2 of just 0.40. The models were evaluated based on MAE, MSE, RMSE, R^2, RMSLE and MAPE.

Challenges

1 **Data Availability**

The availability of data can limit the creation and refinement of the model.

2 **Model Interpretability**

Improving the interpretability of the model will make it easier to understand the results and identify areas of potential improvement.

3 **Computational Requirements**

Building a reliable model requires a significant amount of computational power and resources. Reducing the model's computational complexity in order to optimize its performance.

Conclusion

Even with all the data provided from a streaming service it was not possible to tap into and predict with reliable certainty what songs will become popular. However, it does feel satisfying to know that a great artform still remains a mystery and there's no secret recipe to success.