Spotify Genre Classification – Capstone Report

Introduction

Since its introduction in 2006, Spotify has become a become a massively successful disrupter in the music industry. With more than 217 Millions active users[1] and a music library consisting of 30 million songs[2], the company has become the front runner in the music streaming business.

Problem Statement – Can song traits predict the genre of the song?

Understanding the qualities that define a songs genre could:

- Help classify new songs that exhibit similar qualities to existing songs
- Shed insight on how user trends and preferences in music over time
- Potentially improve recommender systems by suggesting new songs with similar traits but in a genre not normally listened to by a user

Data Collection – A deep dive into the Spotify API

The data used for the project was primarily sourced from the Spotify API. The collection process can be summarized in two major steps:

a) Web scraping *https://spotifycharts.com/regional to pull the top 200 most popular songs from the past two years*

b) Using the list generated from step a), leverage the Spotify API service to pull additional data for the classification model

---

Step a) – Web Scraping the Spotify Charts

Data Collection:

Using the Python library requests, a Python script was utilized to scrape the daily stream numbers for the top 200 most popular songs (by daily stream count). A total of 850 csv files between the dates 01/01/2017 to 04/30/2019 were collected.

Error handling was added to capture any dates where the csv did not download correctly. The date of these files with errors were written to a specific file for review.

---

[1] https://www.theverge.com/2019/4/29/18522297/spotify-100-million-users-apple-music-podcasting-free-users-advertising-voice-speakers

[2] https://investors.spotify.com/home/default.aspx

Data Validation Issues:

Csv files for 4 dates were unable to be downloaded due to an error on Spotify's website. No data was provided for these 4 dates. Due to the top ranked songs not changing significantly day to day, the missing data was omitted as it is immaterial for the overall classification model to function properly.

Amalgamating the Data:

A short script amalgamated and cleaned the csv files. A helper function was used to read the csv files and combine into a Pandas dataframe.

Once summarized in one dataframe, the dataframe was cleaned to remove any headers from the csv files and some rudimentary data exploration was performed to find any potential inconsistencies dataset.

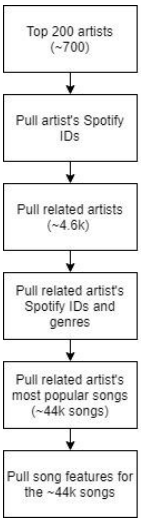
The final cleaned Pandas dataframe was written to a csv for analysis.

Step b) – Extracting Song Genre and Features from the Spotify API Service

Data Collection:

Using the cleaned csv from step a, a unique list of artists was extracted by loading the csv into a dataframe. A Pandas series of approximately ~700 unique artists was created from the Dataframe.

The module *Spotipy* (an open source library) was then used to make ~700 API calls to collect the unique Spotify artist id for each artist.



(ie. similar artists that did not make the top 200 charts)

Using the 700 unique Spotify ids, another API call was made to pull all related artists[3] to the ~700 unique artists.

The result was a new artist list totaling ~4.6k artists.

Using the ~4.6k artist list, another API call was made to pull ~4.6k unique Spotify artist ids as well as the artist's genre.

With the ~4.6k unique Spotify artist ids, an API call was made to pull the ~4.6k artists most popular songs. The result was a song list consisting of approximately 44k songs.

The final step was one final API call to pull the song features for each of the 44k songs.

[3] https://developer.spotify.com/documentation/web-api/reference/artists/get-related-artists/

Two datasets were created from the above process:

- A ~4.6k list of artists as well as their corresponding genres
- A ~44k list of the songs as well as their features (danceability, energy, key etc.)

Data Validation Issues:

During the API calls, there were instances where the API returned an error response. Many of these error responses were due to characters not recognized by normal utf-8 encoding (for example, songs with Chinese characters).

Due to the small amount of these instances relative to the overall dataset, all invalid songs or artists were removed from the final dataset.

Finalizing the Data for the Model:

In finalizing the data for the classification model, the last step included removing any artists without a genre on Spotifiy's system.

Two pickle files were created to house two dataframes which contained:

- Artist and artist's genre.
- A list of songs with song features.

For more information the song features collected, see the Appendix.

Explanatory Data Analysis and Statistical Methods

After the data collection process, rudimentary data exploration was performed on the dataset. The product of the EDA process was a series of questions and hypotheses surrounding the data.

Initial Hypotheses and general questions

1. What does the overall distribution of features look like?
2. Does featuring another artist on a track correlate to higher stream numbers?
3. What kind of growth in stream numbers has Spotify achieved over the past two years?
4. What does the distribution of stream numbers look like for the top 200 stream list?
5. When do Spotify users tend to stream the most? (day of week, month of year)
6. Who is the most popular artist/song in the last 2 years years?
7. What types of genres tend to be on the top 200 charts?
8. Is there any relationship between any of the features? If so, will the correlation between two features impede the classification model?
9. Will correlation between two features impede a ML model?

10. Is there a meaningful difference between subgenres in terms of features? Put another way, are there distinguishing features in the dataset difference between for example 'pop' music and 'dance pop' music?

---

1. What does the overall distribution of features look like across common genres?

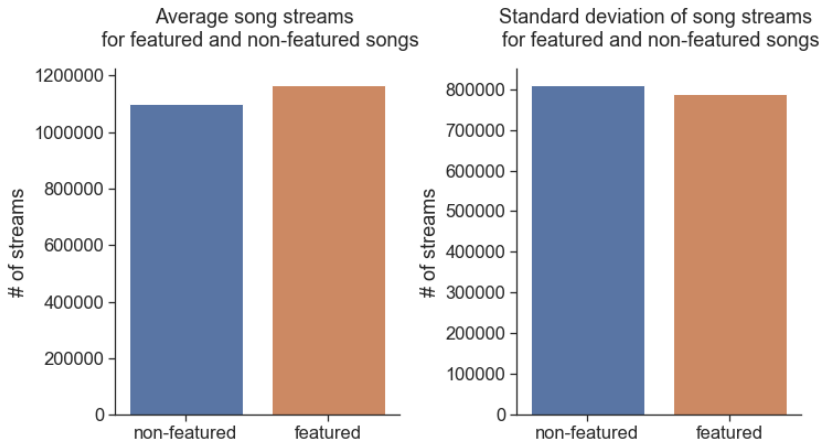
To determine the overall distribution of the features, violin plots for each feature and common genre were plotted. See appendix for details on the features and plots.

2. Does featuring another artist on a track correlate to higher stream numbers?

The basis of this hypothesis is to answer if there is any observed network effect across artists. Will fans of a certain artist have a propensity to listen to another artists song if it's featuring the artist they are more familiar with.

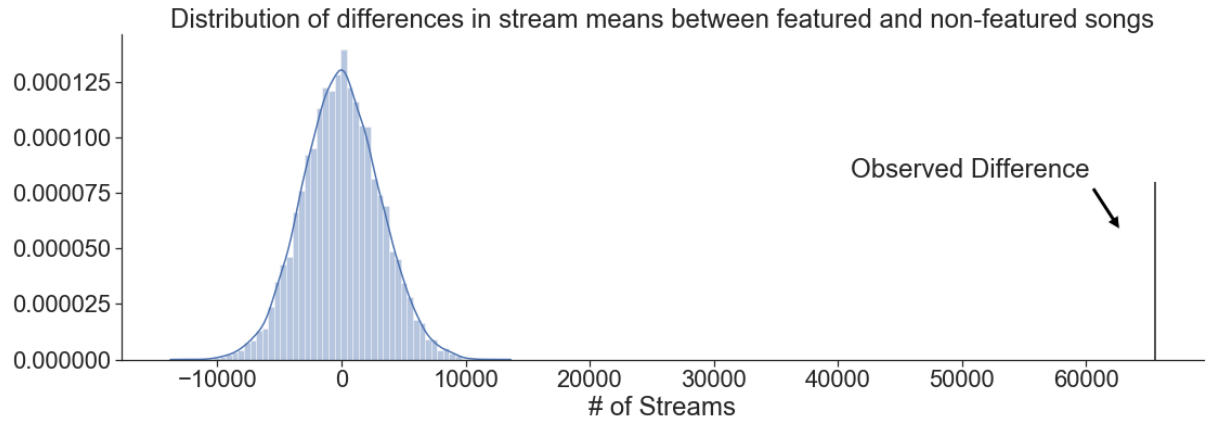
Leveraging regex, the keywords "feat" and "with" were extracted from the song name. A binary feature was then created to help with separating the two groups (1 for track featuring an artist, 0 for track not featuring anyone).

The result of mean streams by featured and non-featured stream numbers shows that featuring an artist **does** lead to higher stream numbers.



To determine if the observed difference in means is statistically significant, 10,000 bootstrap samples of adjusted means were taken from the two groups (featured/non-featured).

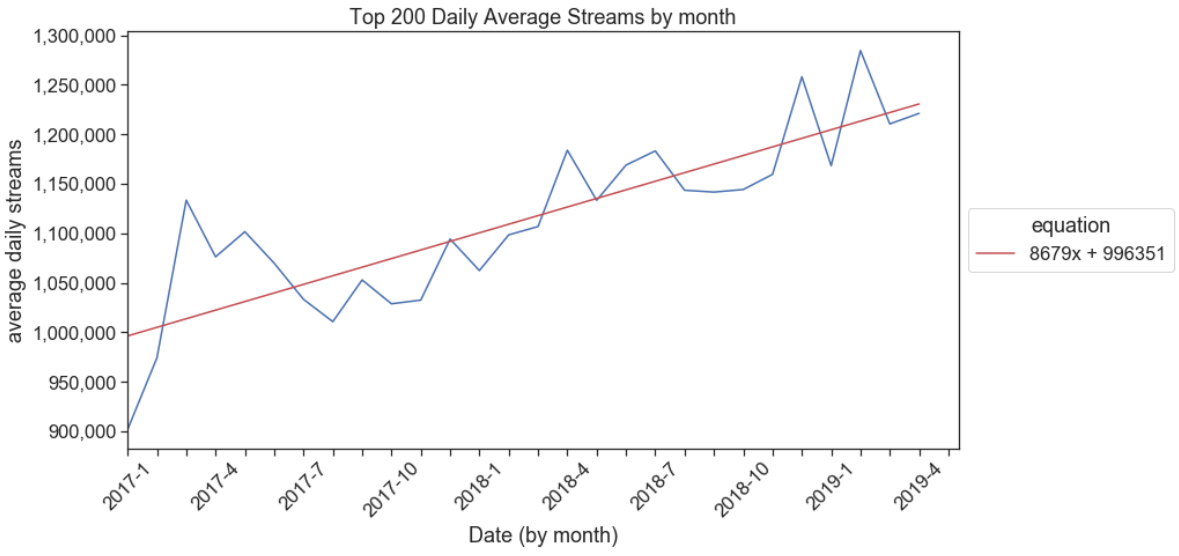
The resulting pvalue from the sampling was 0 suggesting there is a statistically meaningful difference in means between the two groups. The plot below shows the observed difference in means and the distribution of the bootstrap sampling.



3. What kind of growth in stream numbers has Spotify achieved over the past two years?

To determine growth rate in average daily streams among the top 200 most popular songs, the daily stream data was resampled by month using average as the aggregation. Over the course of 28 months, Spotify saw an average month over month growth rate of 0.87% in daily stream numbers among the top 200 most popular songs.

The result was the plot below:

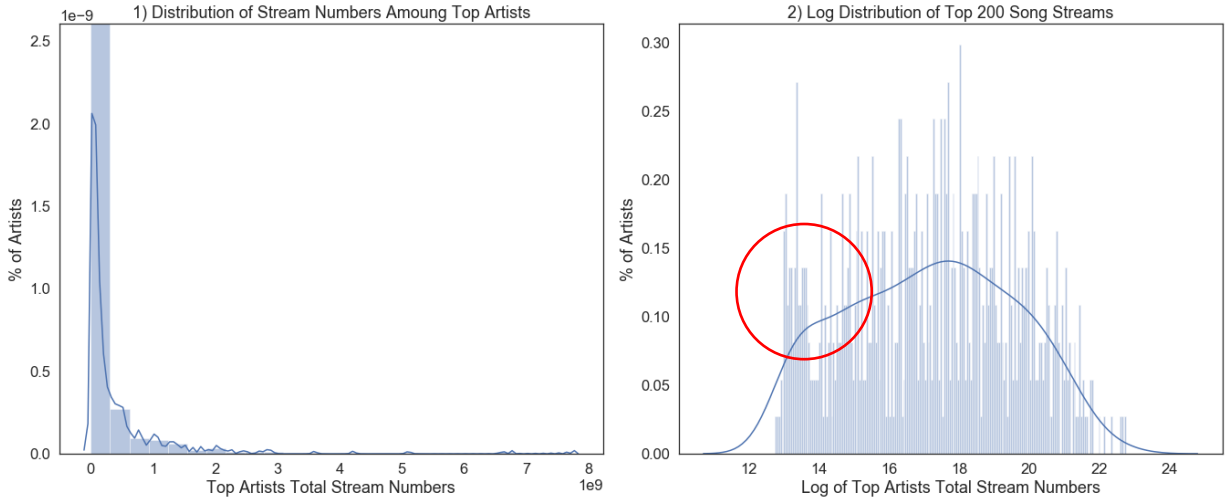


4. In terms of stream numbers, what is the stream distribution among top artists?

Plotting the distribution plots of stream numbers, we see an exponential relationship among the top artists and stream numbers.

Taking the log distribution of stream numbers, we observe a gaussian distribution suggesting that stream among the top 200 artist list is normally distributed.

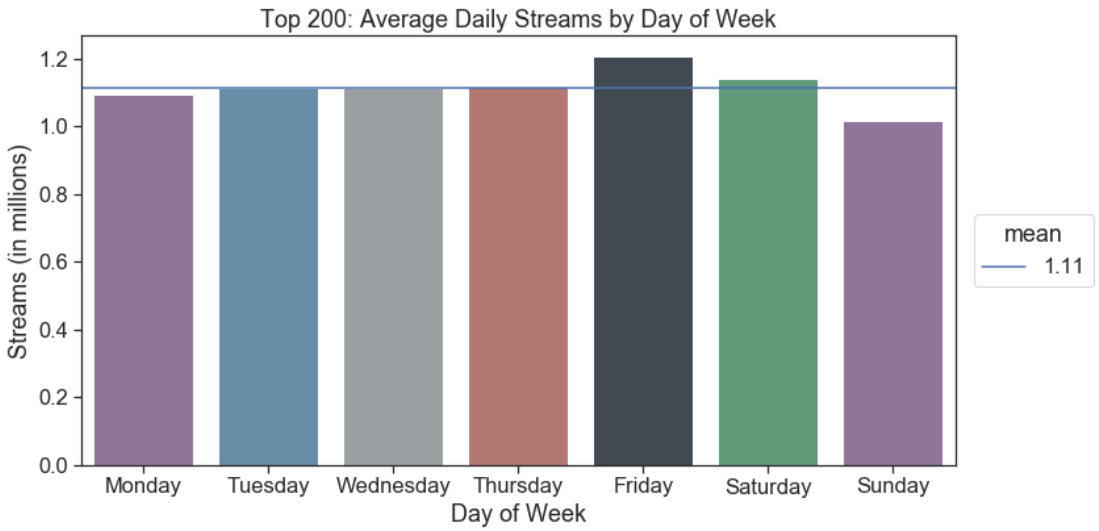
The wider left tail on the log distribution is expected given exponential distribution of streams. There are more artists with average stream numbers than there are superstars with extremely high stream numbers.



5. When do Spotify users tend to stream the most? (day of week, month of year)

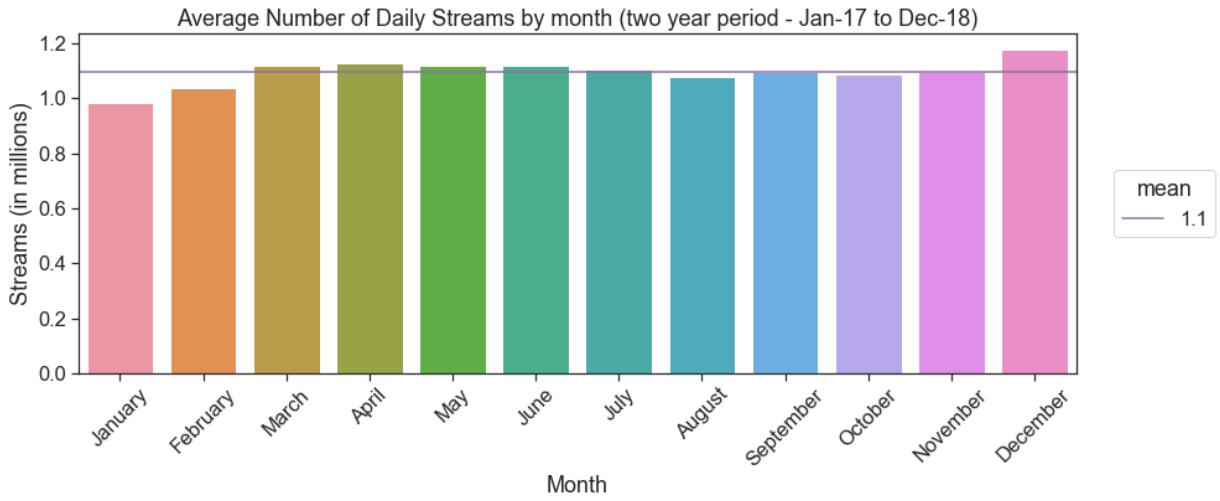
Resampling the stream data by 'day of week' shows the most popular day to stream music is Friday.

Surprisingly, we see that Sunday is the lowest day in terms of streaming numbers.



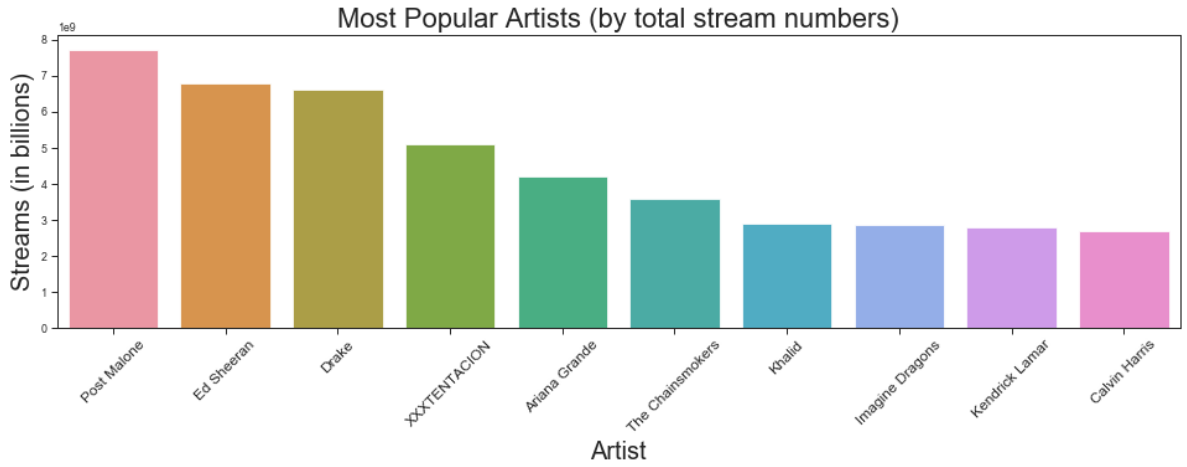
Resampling the stream data by 'month' shows the most popular month to stream music is December.

Once again, it is surprising to note that January is the lowest streamed month.



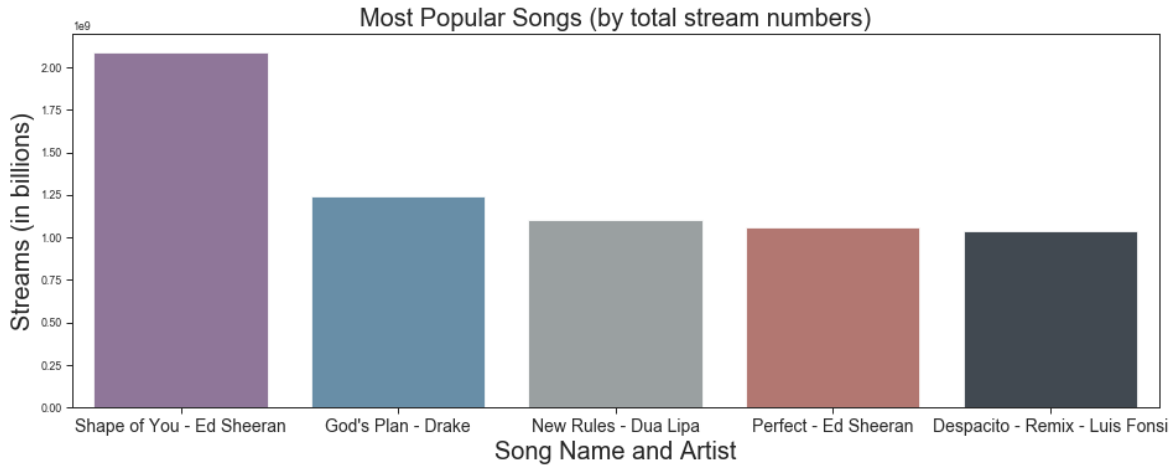
6. Who is the most popular artist in the last 2 years years?

Grouping the data by artist and summing stream values shows "Post Malone" is high streamed artist on Spotify.



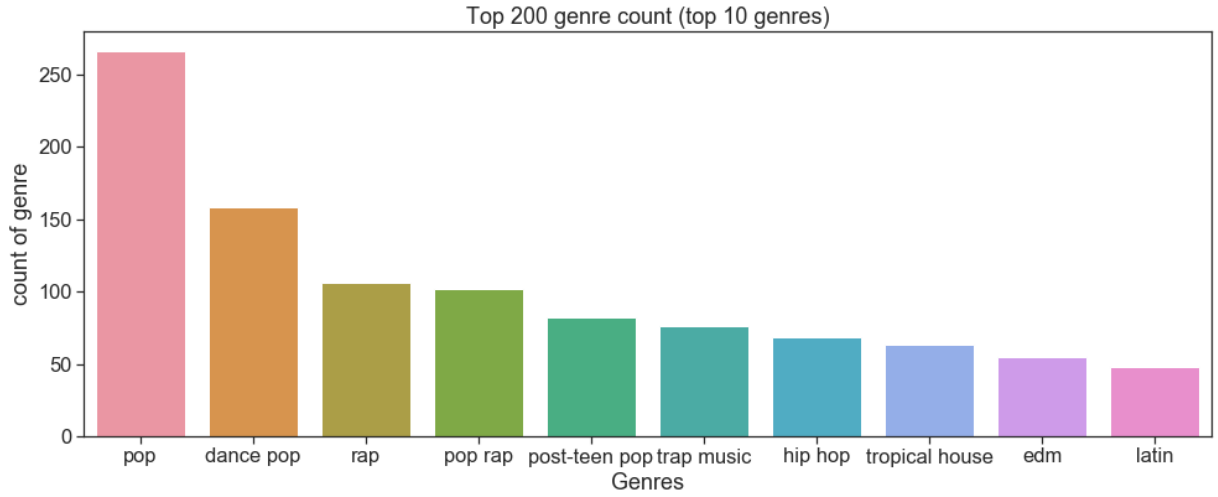
7. What was the most popular song in the last 2 years?

Once again, grouping the data by track name and aggregating on stream numbers shows that “Shape of You” by Ed Sheeran is the most popular song in the last two years.



8. What types of genres tend to be on the top 200 charts?

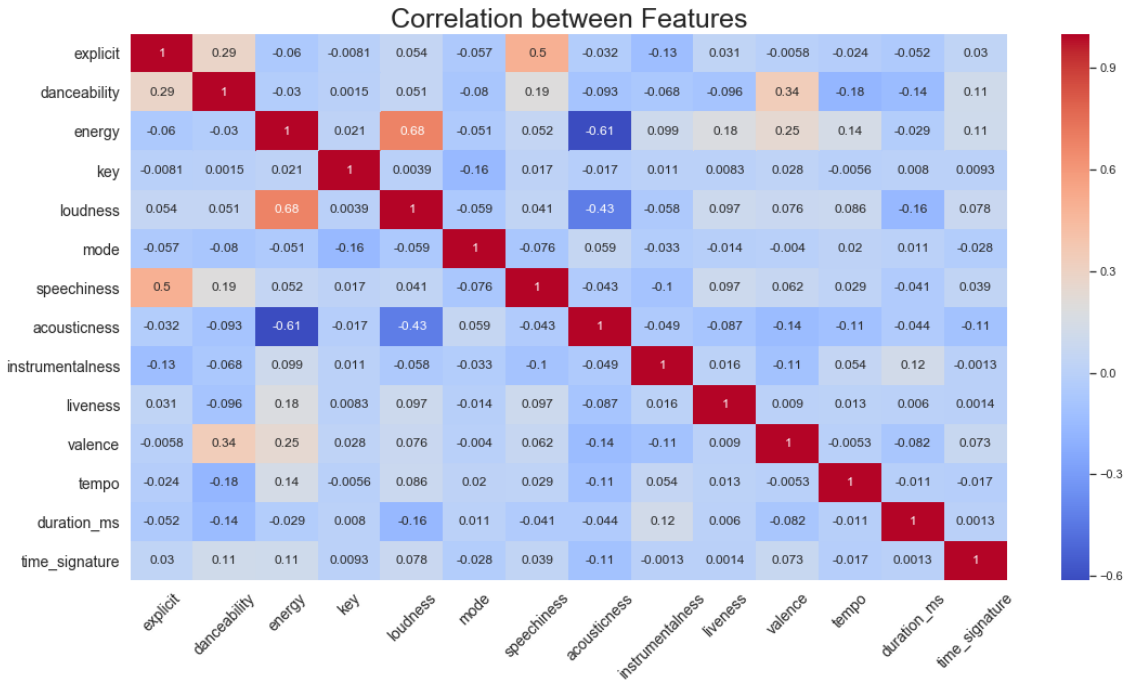
Aggregating the top 200 songs by genre count, we see that pop and dance pop are the two most popular genres.



9. Is there any relationship between any of the features? If so, will the correlation between two features impede the classification model?

To determine if there is correlation between the features[4] used for the model, a heatmap in seaborn was constructed. The output shows two correlstions of note:

- Acousticness and energy are negatively correlation with an r value of -0.61
- Loudness and energy are positively correlated with an r value of 0.68



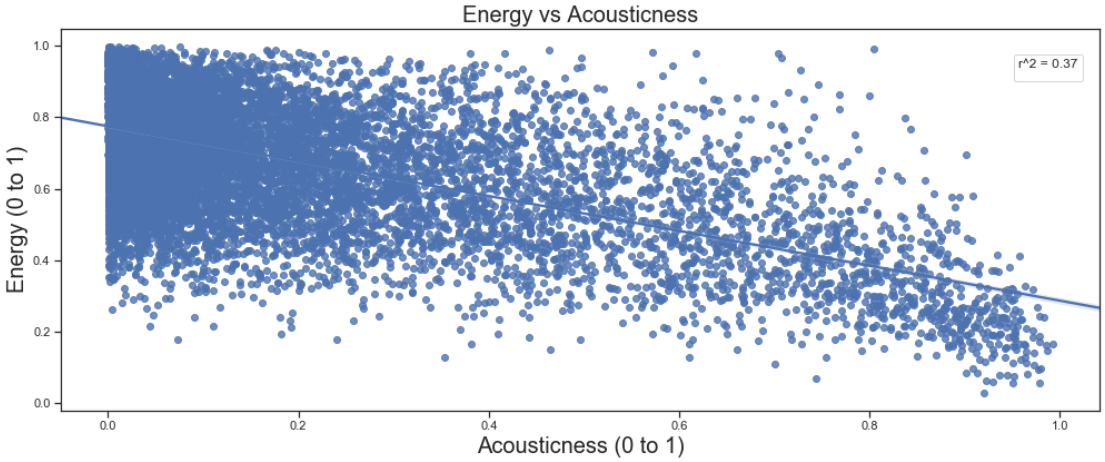
[4] See appendix for more details on the features for the classification model

10. Determining if correlation will impede the classification model

To determine if correlation will impede the classification model, a linear regression was constructed for both sets of features and the r^2 was evaluated.

Acousticness and energy

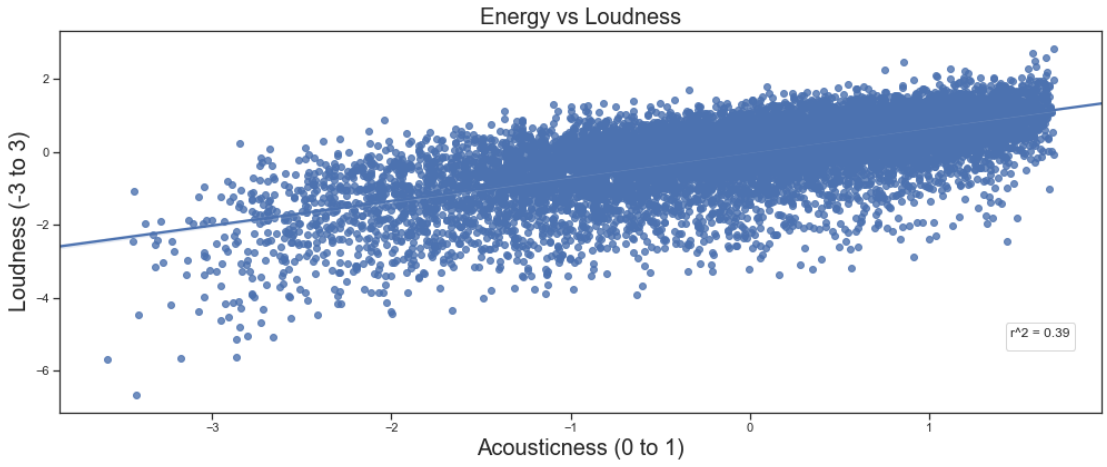
Acousticness and energy has an r^2 value of 0.37 which is not significant enough to omit one of the variable in the final model.



Loudness and energy

Important to note: Loudness in this dataset is measured in decibels from a baseline 0. Furthermore, decibels are usually measured as a logarithm (base 10) to encapsulate the wide range of numeric values that decibels can take on. For the purposes testing correlation, the exponent was not taken prior to the linear model. This does introduce a small amount of inaccuracy in the linear model, however, it is likely not significant enough to bias the end result. Finally, prior to creating the linear model, loudness was scaled using standard scaling.

Loudness and energy returns a similar value r^2 value of 0.39. Once again, this r^2 is not significant to omit a variable in the final model.



11. Is there a meaningful difference between subgenres in terms of features? Put another way, are there distinguishing features in the dataset between for example 'pop' music and 'dance pop' music?

To determine if the features of similar genres are distinguishable, the following procedures were performed:

a) Features between the genres were adjusted to have the same mean and a bootstrap sampling was performed to determine if there is a meaningful statistical difference between genres
b) A Mann–Whitney U test was performed to determine if there is a statistically significant difference between distributions

5 sets of subgenres and 13 features were compared. The results below show the average number of features that showed significance differences between the means and distributions.

Mean		Distribution	
Significant	Insignificant	Significant	Insignificant
5	8	8	4

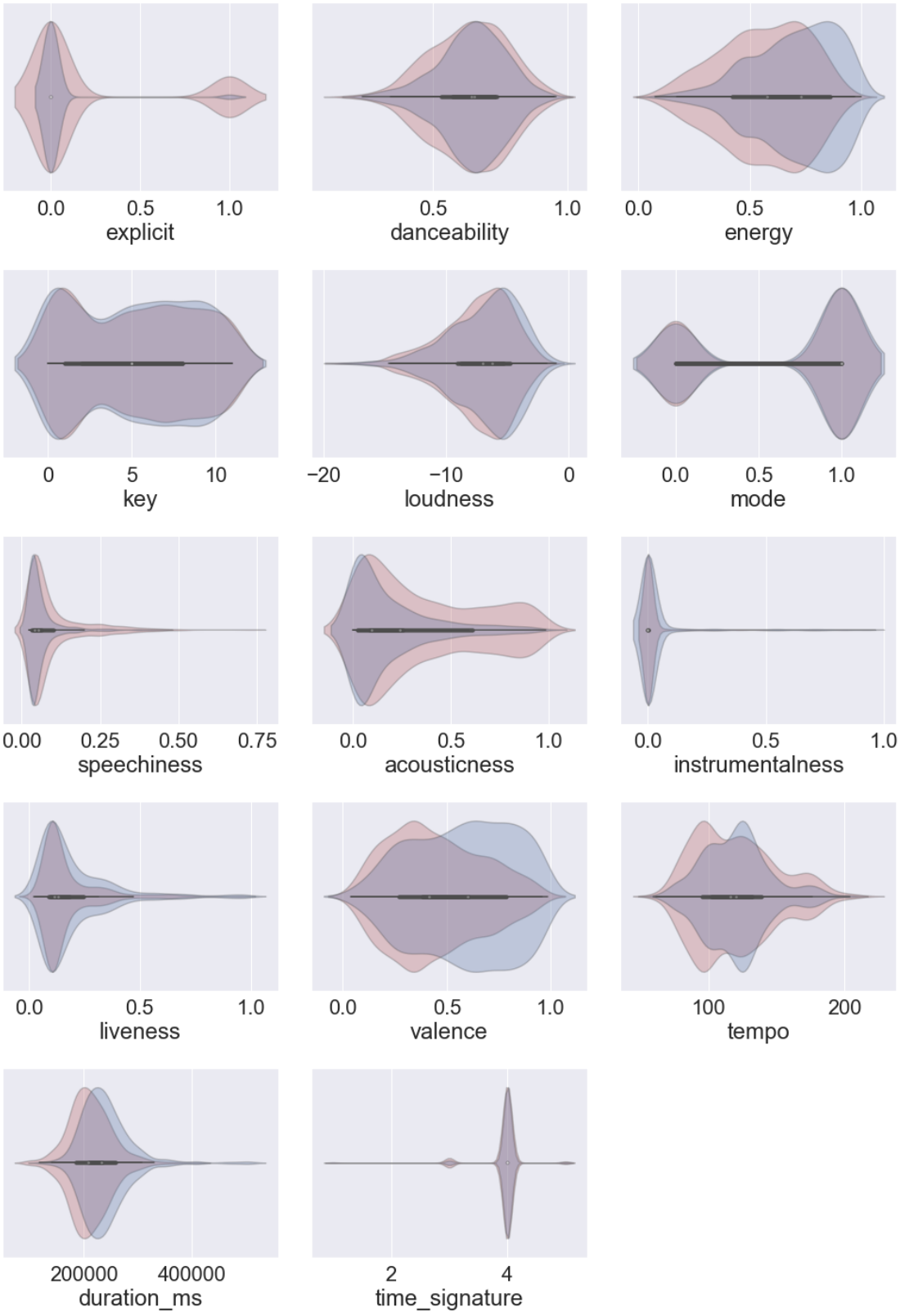
The results conclude that there are differences between subgenres, however, the degree of differences between the subgenres varies significantly.

For example, the following shows the difference in distributions between pop and dance pop:

Pop versus Dance Pop

Feature	Bootstrap μ pvalue	Mann-Whitney u test pvalue
explicit	0.0	0.0
danceability	0.9256	0.0684
energy	1.0	0.0
key	0.3157	0.2467
loudness	1.0	0.0
mode	0.7036	0.2733
speechiness	0.0	0.0
acousticness	0.0	0.0
instrumentalness	0.9899	0.0
liveness	0.9995	0.0317
valence	1.0	0.0
tempo	0.6991	0.0196
duration_ms	1.0	0.0
time_signature	0.9463	0.0221

The below plot displays the difference in mean values and distribution for each feature:



For the purposes of the classification model, the original dataset was condensed to include only very common genres (and similar subgenres).

Machine Learning

Summary

The decision to merge similar subgenres was done in order to improve overall accuracy of the model.

The logic behind the amalgamation of features was largely attributed to indistinguishableness of subgenres with only using the 14 features available for the model.

For example, the distinction between 'rock' and 'modern rock' is marginal when looking at features such as the tempo, valence, key etc.

However, when a person listens to two subgenres, they become distinguishable based on other subtleties. For example, the two songs below display the difference between 'rock' and 'modern rock'.

Rock - https://open.spotify.com/artist/0qEcf3SFlpRcb3lK3f2GZI

Modern Rock - https://open.spotify.com/artist/4OTFxPi5CtWyj1NThDe6z5

Using the 14 features, the model is able to correctly distinguish between genres 59% of the time (a random selection would be 25%)

Genres that contrast quite a bit (ex. rock and rap music) were more accurate whereas similar genres such as EDM and pop were more difficult for the model to select correctly.

Next steps – improvements to the model

To improve the model further, additional data is required. Introducing sound data into the model could possibly improve the accuracy.

Another idea would be introducing lyric data into the dataset. Lyrics vary widely depending on genre. Training the data on similar words or sentences might provide improvements in model accuracy.

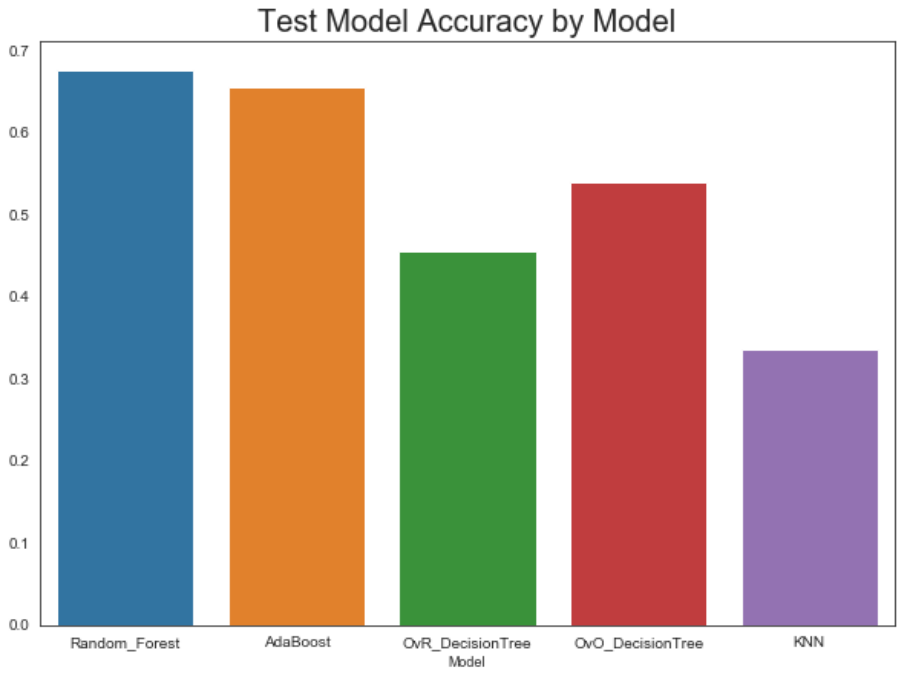
Model predictions

10 models were used to predict the song genre:

- Random Forest
- AdaBoost
- One Vs Rest (Naïve Bayes, Logistic Regression, Decision Tree)
- One Vs One (Naïve Bayes, Logistic Regression, Decision Tree)
- Support Vector Machine
- KNN

Each model was evaluated using out of the box parameters. 25% of the data was used as holdout to evaluate model performance.

Of the above models, a Random Forest performed the best in terms of test accuracy (67.6%). Based on the results of the out of the box models, it appears that this problem is better suited towards models that can predict non-linear variations in the data. As such, for model optimization, a random forest model was selected.



Improving Selected Model Performance

To improve the overall performance of the Random Forest model, the following steps were taken:

A. Scaling the data
B. Dropping unnecessary features
C. Grid Search – Hyperparameter tuning and cross validation

A) Scaling the data – Using Sklearn standard scaler, the dataset was scaled to a mean of 0 with a standard deviation of 1. Model performance increased marginally (~0.1%) after scaling the data. This marginal increase is expected due to most of the data already being scaled between 0 and 100 by Spotify.

B) Dropping unnecessary features – The feature "Mode" was dropped due to the it's feature importance being <~1%

C) Grid Search – Hyperparameter tuning was performed by using Sklearn grid search along with 4 fold cross validation and a holdout set of 25%.

The decision to perform both cross validation and use 25% of the training data as a hold out set was done due to the diminishing returns on increasing the training sample. The model converged using roughly ~50% of the training data. As such, it was unnecessary to train on such a large sample.

The following parameters were used in the parameter tuning.

bootstrap: True, False
max depth: 5,10,15,20,25,30
max features: 2,5,6,7,8,9,10,11
n estimators: 10,20,30,40,50,60,70,80,90,100

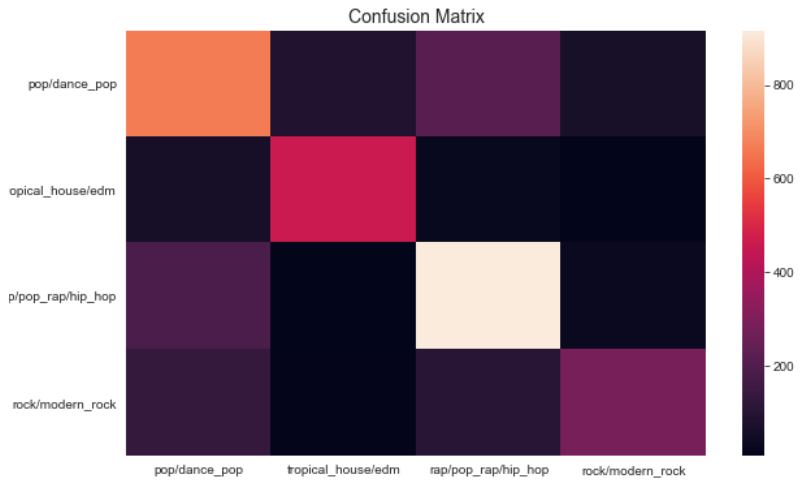
A total of 2,880 fits were performed over the 4 parameters. The optimal model is as follows:

bootstrap: True
max_depth: 25
max_features: 2
n_estimators: 90

The optimal model performed at 70% accuracy vs the accuracy of the out of the box performance of 67%

Analyzing Model Performance – Further Improvements

Hyperparameter tuning improved the model by 3%. To determine how to improve the model performance further, the confusion matrix for the 'best model' was plotted.



The genre the most difficult to predict is Pop & Hip-Hop. To improve model performance further, adding additional features could help differentiate between pop and rock.

One idea is to improve the model by adding lyric data. Pop & Rock music have very different lyrics. Adding lyrical data in the form of a sparse matrix could potentially improve the accuracy.

Appendix

Features Collected for the Classification Model – See the Spotify API for more details

- duration_ms - The duration of the track in milliseconds.
- Key - The estimated overall key of the track. Integers map to pitches using standard Pitch Class notation . E.g. 0 = C, 1 = C♯/D♭, 2 = D, and so on. If no key was detected, the value is -1.
- 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.
- Time_signature -An estimated overall time signature of a track. The time signature (meter) is a notational convention to specify how many beats are in each bar (or measure).
- Acousticness - A confidence measure from 0.0 to 1.0 of whether the track is acoustic. 1.0 represents high confidence the track is acoustic.
- 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. A value of 0.0 is least danceable and 1.0 is most danceable.
- Energy - Energy is a measure from 0.0 to 1.0 and represents a perceptual measure of intensity and activity. Typically, energetic tracks feel fast, loud, and noisy. For example, death metal has high energy, while a Bach prelude scores low on the scale. Perceptual features contributing to this attribute include dynamic range, perceived loudness, timbre, onset rate, and general entropy.
- Instrumentalness - Predicts whether a track contains no vocals. “Ooh” and “aah” sounds are treated as instrumental in this context. Rap or spoken word tracks are clearly “vocal”. The closer the instrumentalness value is to 1.0, the greater likelihood the track contains no vocal content. Values above 0.5 are intended to represent instrumental tracks, but confidence is higher as the value approaches 1.0.
- Liveness - Detects the presence of an audience in the recording. Higher liveness values represent an increased probability that the track was performed live. A value above 0.8 provides strong likelihood that the track is live.
- Loudness - The overall loudness of a track in decibels (dB). Loudness values are averaged across the entire track and are useful for comparing relative loudness of tracks. Loudness is the quality of a sound that is the primary psychological correlate of physical strength (amplitude). Values typical range between -60 and 0 db.

- 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. Values above 0.66 describe tracks that are probably made entirely of spoken words. Values between 0.33 and 0.66 describe tracks that may contain both music and speech, either in sections or layered, including such cases as rap music. Values below 0.33 most likely represent music and other non-speech-like tracks.

- Valence - A measure from 0.0 to 1.0 describing the musical positiveness conveyed by a track. Tracks with high valence sound more positive (e.g. happy, cheerful, euphoric), while tracks with low valence sound more negative (e.g. sad, depressed, angry).

- Tempo - The overall estimated tempo of a track in beats per minute (BPM). In musical terminology, tempo is the speed or pace of a given piece and derives directly from the average beat duration.